International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Study of Genetic Algorithms

Sukhveer Kaur[1], Vinay Bhardwaj[2]

[1]M-Tech Research Scholar, Shri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, India

[2]Assitant Professor, Department of CSE, Shri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, India

Abstract: ***Genetic Algorithms is a search heuristic which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. Many modifications have been suggested towards the improvement in Genetic Algorithms to meet QoS challenges through focusing on Average Throughput, Packet delivery ratio, Packet loss, energy and mechanism overheads. Various Optimization problems have been discussed. The aim of this paper is to facilitate future researches such that several proposed modifications in genetic algorithms can be probed quickly.***

Keywords: Genetic Algorithm (GA), selection, crossover, mutation and Quality of Services (QoS).

1. Introduction

Genetic algorithms are a part of evolutionary computing, which is a rapidly growing area of artificial intelligence. A genetic algorithm (GA) is a search heuristic that mimics the process of natural selection. This metaheuristic is routinely used to generate useful solutions to optimization and search problems [1].

They are a robust and flexible approach that can be applied to a wide range of learning and optimization problems. They are particularly suited to problems where traditional optimization techniques break down, either due to the irregular structure of the search space or because the search becomes computationally intractable [2]. Genetic algorithms generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover[1].

Genetic algorithms find application in bioinformatics, phylogenetics, computational science, engineering, economics, chemistry, manufacturing, mathematics, physics, pharmacometrics and other fields [3].

Genetic Algorithms replicate the survival of the fittest among individuals over consecutive generation for solving a problem. Each generation consists of a population of character strings that are similar to the chromosome that we see in our DNA. Each individual represents a point in a search space and a possible solution. The individuals in the population are then made to go through a process of evolution [9].

2. Implementation

In a genetic algorithm, a population of individuals to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties (its chromosomes) which can be mutated and altered; solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible.[13].

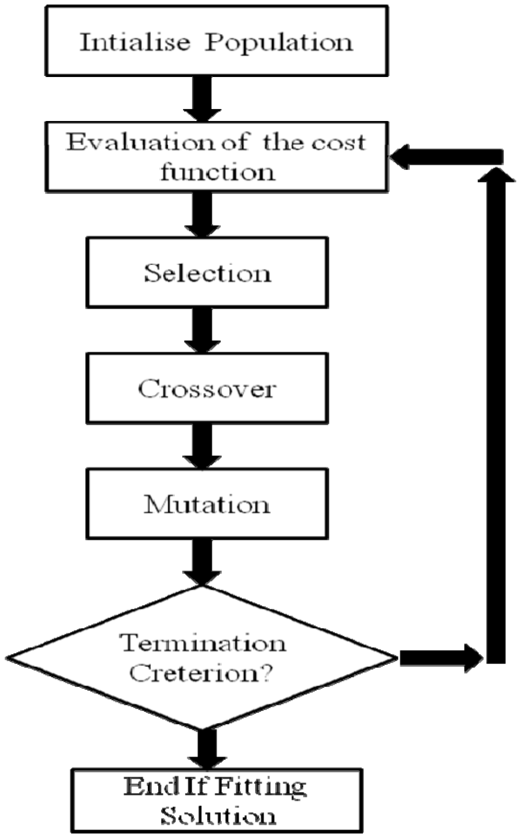

Figure 1: Genetic Algorithm

The evolution usually starts from a population of randomly generated individuals, and is an iterative process, with the population in each iteration called a generation. In each generation, the fitness of every individual in the population is evaluated; the fitness is usually the value of the objective function in the optimization problem being solved[13].The more fit individuals are randomly selected from the current population, and each individual's genome is modified to form a new generation. The new generation of candidate solutions is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population [14].

Licensed Under Creative Commons Attribution CC BY

A typical genetic algorithm requires [1]:

1. A genetic representation of the solution domain,
2. A fitness function to evaluate the solution domain.

A standard representation of each candidate solution is as an bits. The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size, which facilitates simple crossover operations [5].Variable length representations may also be used, but crossover implementation is more complex in this case. Tree-like representations are explored in genetic programming and graph-form representations are explored in evolutionary programming; a mix of both linear chromosomes and trees is explored in genetic expression programming [2].

Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators [4].

3. Genetic Operators

Genetic operators used in genetic algorithms are similar to those in the natural world. The Genetic algorithm evolves through these three operators [1]:

1. **Selection** which equates to survival of the fittest;
2. **Crossover** which represents mating between individuals;
3. **Mutation** which introduces random modifications.

1) Selection Operator

Selection is the stage of a genetic algorithm in which individuals are chosen from a population for later breeding (crossover).A generic selection procedure may be implemented as follows[2]:

1. The fitness function is evaluated for each individual, providing fitness values.
2. The population is sorted by descending fitness values.
3. Accumulated normalized fitness values are computed. The accumulated fitness of the last individual should be 1 .
4. A random number *R* between 0 and 1 is chosen.
5. The selected individual is the first one whose accumulated normalized value is greater than *R*.
6. Fitness may be determined by an objective function or by a subjective judgment.

2) Crossover Operator

Crossover is a genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next. It is similar to reproduction and biological crossover, upon which genetic algorithms are based. Cross over is a process of taking more than one parent solutions and producing a child solution from them [1].

3) Mutation Operator

Mutation is used to maintain genetic diversity from one generation of a population of genetic algorithm chromosomes to the next. It is similar to biological mutation. Mutation alters one or more gene values in a chromosome from its initial state. In mutation, the solution may change entirely from the previous solution. Hence Genetic Algorithm can come to better solution by using mutation [2]. With some low probability, a portion of the new individuals will have some of their bits flipped. Mutation induces a random walk through the search space. Mutation and selection (without crossover) create a parallel, noise-tolerant, hill-climbing algorithm [1].

4. Basic Genetic Algorithm

1. [Start] Generate random population of *n* chromosomes (suitable solutions for the problem)
2. [Fitness] Evaluate the fitness *f(x)* of each chromosome *x* in the population
3. [New population] Create a new population by repeating following steps until the new population is complete
   a) [Selection] Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected)
   b) [Crossover] With a crossover probability cross over the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.
   c) [Mutation] With a mutation probability mutate new offspring at each locus (position in chromosome).
   d) [Accepting] Place new offspring in a new population
4. [Replace] Use new generated population for a further run of algorithm
5. [Test] If the end condition is satisfied, stop, and return the best solution in current population
6. [Loop] Go to step 2

5. Genetic Algorithms Optimization Problems

1. Fuzzy Optimization Problems

The precise quantification of many system performance criteria and parameter and decision variables is not always possible, nor is it always necessary. When the values of variables cannot be precisely specified, they are said to be uncertain or fuzzy. If the values are uncertain, probability distributions may be used to quantify them. Alternatively, if they are best described by qualitative adjectives, such as dry or wet, hot or cold, clean or dirty, and high or low, fuzzy membership functions can be used to quantify them. Both probability distributions and fuzzy membership functions of these uncertain or qualitative variables can be included in quantitative optimization models [4].

2. Multi-objective Optimization Problems

Multi-objective optimization is an area of multiple criteria decision making, that is concerned with mathematical optimization problems involving more than one objective

Licensed Under Creative Commons Attribution CC BY

function to be optimized simultaneously [7]. Multi-objective optimization has been applied in many fields of science, including engineering, economics and logistics where optimal decisions need to be taken in the presence of trade-offs between two or more conflicting objectives[8].Minimizing cost while maximizing comfort while buying a car, and maximizing performance whilst minimizing fuel consumption and emission of pollutants of a vehicle are examples of multi-objective optimization problems involving two and three objectives, respectively[7].

3. Combinatorial Optimization Problems

Combinatorial Optimization is that which consists of finding an optimal object from a finite set of objects. In many such problems, exhaustive search is not feasible. It operates on the domain of those optimization problems, in which the set of feasible solutions is discrete or can be reduced to discrete, and in which the goal is to find the best solution. Some common problems involving combinatorial optimization are the traveling salesman problem ("TSP") and the minimum spanning tree problem ("MST")[9].

6. Related Work

Genetic Algorithms are used for energy efficient QoS multicast routing and for the various optimization problems. In Genetic Algorithm for energy efficient QoS [1], a source-based algorithm is proposed which takes into account energy consumption as well as end-to-end delay in route selection. The proposed algorithm applies crossover and mutation operations directly on trees, which simplifies the coding operation and omits the coding/decoding process. Heuristic mutation technique can improve the total energy consumption of a multicast tree. The proposed algorithm is more effective and efficient. In multi-objective genetic algorithm [8], it is used for multiple objectives .This algorithm shows the better results than other traditional algorithms used. Multi-objective Genetic algorithm has given the better results using the QoS parameters such as Constraints, delay, traffic from adjacent nodes, number of hops and to provide adaptive route in manets.

The genetic algorithm [4], has investigated the approach for solving fuzzy nonlinear programming problem with fuzzy coefficients and fuzzy constraints. This shows that the genetic algorithm is a good candidate tool for solving fuzzy nonlinear optimization problems without requiring mathematical reductions or transformations. The computation needs not to use the extension principle nor the interval arithmetic and α-cuts. The empirical results show that the proposed approach can obtain very good solutions within the given bound for each fuzzy coefficient that accomplishing flexible nonlinear programming.

In Genetic algorithms for modeling and optimization [6], is intended as an introduction to GAs aimed at immunologists and mathematicians interested in immunology. It describes show to construct a GA and the main strands of GA theory before speculatively identifying possible applications of GAs to the study of immunology. An illustrative example of using a GA for a medical optimal control problem is provided. The paper also includes a brief account of the related area of artificial immune systems.

7. Conclusions

In this paper, we presented the modifications in Genetic Algorithms for optimization problems to meet QoS routing challenges through focusing on different factors such as Average Throughput, Packet delivery ratio, Packet loss and control overhead. Optimization problems of genetic algorithm such as Fuzzy optimization problems, multi-objective optimization problems and combinatorial optimization problems are explained. In this paper, we also presented the genetic algorithm and its operators in detail to facilitate future researches such that several proposed modifications using genetic algorithms can be done quickly and easily.

References

[1] Ting Lu and Jie Zhu, "Genetic Algorithm for Energy-Efficient QoS Multicast Routing" IEEE Communications letters, Volume 17, Issue 1, January 2013.
[2] Sheng-Yuan Tseng, Yueh-Min Huang, Chang-Chun Lin, "Genetic algorithm for delay- and degree-constrained multimedia broadcasting on overlay networks", Volume 29, Issue 17, 8 November 2006.
[3] R.Simpson, C.Dandy and J.Murphy, "Genetic Algorithms compared to another techniques for pipe optimization", JWRPM, ASCE, Volume 8, Issue 1, 1994.
[4] N. Ravi Shankar, G. Ananda Rao, J. Madhu Latha and V. Sireesha, "Solving a Fuzzy Nonlinear Optimization Problem by Genetic Algorithm", Int. J. Contemp. Math. Sciences, Voume 5, Issue 16, 2010.
[5] A. Rangel-Merino, J. L. López-Bonilla, R. Linares y Miranda , "Optimization Method based on Genetic Algorithms" , Apeiron, Volume 12, Issue 4, October 2005.
[6] John McCall, "Genetic algorithms for modeling and optimization", Journal of Computational and Applied Mathematics Science Direct, Volume 184 Issue 1 , 2005.
[7] Deb, Agrawal, Pratap, and Meyarivan , "A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II ", Internatinal journal of computer Applications and technology,Volume 1, Issue 1 , April 2008.
[8] Dr.Ketan Kotecha, Sonal Popat,"MultiObjective Genetic Algorithm based adaptive QoS Routing in MANET", IEEE Communications, Volume 19, Issue 2, 2007.
[9] Brian Aldiss , Ian Watson,Steven Speilberg ,"Artificial Intelligence: AI", International journal of computer science and Applications,Volume 4,Issue 1,June 2001.
[10] John McCarthy, **"**Defending AI research: a collection of essays and reviews", Stanford University Conference, Volume 5 ,Issue 3, November 2007.

Licensed Under Creative Commons Attribution CC BY

[11] John McCarthy,"Artificial intelligence, Logic and Formalizing Common Sense", Artificial Intelligence, Volume 5, Issue 3 , July 2001.
[12] Yi-Dong Shen, KewenWang, Thomas Eiter, "FLP answer set semantics without circular justifications for general logic programs", Artificial Intelligence, Volume 213, Issue 13, June 2014.
[13] S. Y. Tseng, Y. M. Huang, and C. C. Lin, "Genetic algorithm for delay and degree-constrained multimedia broadcasting on overlay networks," Computer Commun., Volume 29, Issue 17, 2006.
[14] F. Xiang, L. Junzhou, W. Jieyi, and G. Guanqun, "QoS routing based on genetic algorithm," Computer Commun., Volume 22, Issue 15-16, September 1999.
[15] Z. Michaelewicz, "Genetic Algorithms + Data structures = Evolution Programs", third ed., Springer, Berlin, Volume 7, Issue 3, 1999.
[16] M.D. Vose, "The Simple Genetic Algorithm", MIT Press, Cambridge, MA, Volume 17, Issue 3, 1999.

Author Profile

Sukhveer Kaur is Student of M. Tech in the department of Computer Science and engineering at Shri Guru Granth Sahib World University (SGGSWU), Fatehgarh Sahib, Punjab. She has done B.Tech from Baba Banda Singh Bahadur Engineering College (BBSBEC), Fatehgarh Sahib under Punjab Technical University, Jalandhar, India

Vinay Bhardwaj is Working as an Assistant Professor in Sri Guru Granth Sahib World University, had completed M. Tech (Software System) from Guru Nanak Dev University, Amritsar and B.Tech in Computer Science from L.L.R.I.E.T Moga. His Research Area is Wireless Sensor Network. He had published a number of papers in International Journals.

Licensed Under Creative Commons Attribution CC BY