Genetic Based Scheduling In Grid Systems: A Survey

Joshua Samuel Raj [1] , Riya Mary Thomas [2]

1 Assistant Professor, School of Computer Science and Technology, Karunya University, India
2 Post-Graduate Student, Department of Computer Science and Engineering, Karunya University, India

Abstract- **Genetic algorithm techniques are used in optimization for grid computing as they get their inspirations from evolutionary idea of natural evolution. Moreover genetic operators can be tailored for the problem at hand by exploiting procedures that have been fully applied to problems that have resisted solutions to common techniques and thereby making them beneficial. The grid scheduling optimization problem is modeled as a population of candidate solutions and the genetic algorithm is applied for getting the fittest candidates. This paper presents a broad overview on the formalization of works contributed by genetic algorithm to the field of grid scheduling.**

Keywords- **Flowtime, Genetic Algorithm, Grid computing, Grid scheduling, Heuristics**

I. INTRODUCTION

Grid computing combines computers from multiple administrative domains to achieve a common goal to solve a particular task. Computing, most basically stated, is distributed computing taken to the next evolutionary level. They are geographically distributed and interconnected through heterogeneous networks [1]. There are some characteristics of Grid like dynamicity, independence etc which make it complex for resource scheduling in a Grid environment. Due to the complexity of the grid system, the meta-heuristics approaches are proposed for scheduling the jobs in the grid systems. One of these approaches is genetic algorithm [2].

It is an optimization method of searching based on evolutionary process. It is also known as a population based approach where individuals symbolize possible solutions, which are successively evaluated, selected, crossover, mutated and replaced [3]. The population is evaluated and the best solutions are selected to reproduce and mate to form the next generation. Under a number of generations, good behavior, dominates the population, resulting in an increase in the quality of the solutions. Genetic Algorithm can be simplified because it does not need restore process, forcing and decoder process.

II. GENETIC ALGORITHM

Genetic Algorithm (GA) is a meta-heuristic search algorithm based on the principles of evolution and natural genetics. GAs combines the utilization of past results with the examination of new areas of the search space. This algorithm provides effective, efficient techniques for optimization and machines learning applications [4]. The efficiency of the GA depends upon an appropriate combination of exploration and exploitation. It is widely used in business, scientific and engineering circles. Three operators to complete this are: selection, crossover, mutation. Selection according to fitness is a source of utilization, and crossover and random mutations promote exploration. Fig.1 shows the flow of genetic algorithm operations.

A. Selection: Selection operators are used to select the individuals to which the crossover operators will be applied. They choose a chromosome from the present generation for the addition in the next generation.

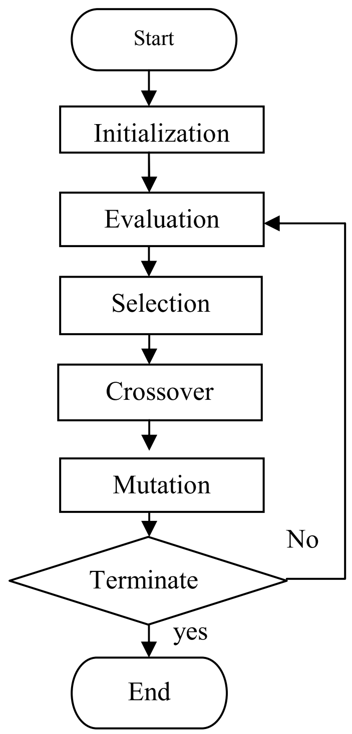

Fig.1 Flow of Genetic Algorithm

978-1-4673-2907-1/13/$31.00 ©2013 IEEE

B. Crossover: The crossover operators are the most important component of any evolutionary like algorithm. Selecting the individuals from the parental generation and interchanging their *genes*, the new individuals are obtained. To obtain the individuals of better quality that will provide for the next generation and enable the search to explore new regions of solution space.

C. Mutation: Mutation adds new information in a random way to the genetic search process and eventually helps to avoid getting trapped at local optima. It is an operator that introduces the diversity in the population tends to become homogenous due to the repeated use of reproduction and crossover. Fig.2. shows the genetic operators and their types.

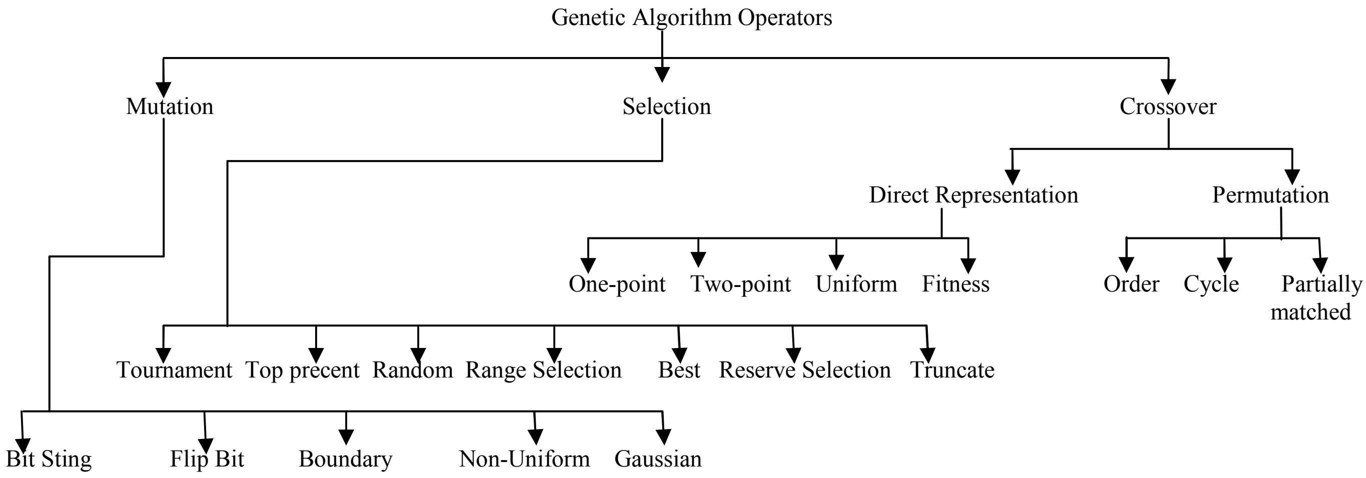

Fig 2.Genetic algorithm operators

III.COMPARATIVE STUDY

The comparative study includes the various variants of Genetic Algorithm are taken.

A) Panmictic Genetic Algorithm: A genetic algorithm without any structure is called Panmictic GA. Panmictic GAs creates a population, calculate it in parallel, and use the results to generate the next population. Panmictic GAs exploits the selection mechanisms that use large-scale information about the whole population to perform a global selection. In comparative selection the population's total fitness is used to perform selection [5]. Panmictic GAs applies the crossover operator to pairs of individuals randomly taken from individuals selected from the whole population.

B) Parallel Genetic Algorithm: The parallel genetic algorithms (PGA) are extensions of the Panmictic GA. The main feature of PGA is their ability to facilitate speciation, a process by which different subpopulation evolve in various directions simultaneously. The three types of Parallel Ga's are Global Single-Population Master-Slave GAs, Single-Population Fine-Grained or Cellular GAs, and Multiple-Population GA's [6].

In master-slave GA there is only single panmictic population. Evaluation of individuals is distributed by scheduling fractions of the population among the processing slave nodes. This kind of model has the advantage for ease of implementation and does not alter the search behavior. Fine-is higher than the average fitness of population, the probabilities decrease, whereas, the crossover and mutation grained PGA contains only a single population and are spatially structured. The computing systems consist of a large number of processing elements and are connected in a specific high-speed topology. For example the population of individuals in a cellular PGA is organized as a two-dimensional grid. The selection and mating in this GA are restricted to small groups. In Multi-population GA several subpopulations exchange individuals infrequently.

C) Hierarchical Genetic Algorithm: Hierarchical GA is a new evolutionary method which gives good computation complexity in solving global optimization problems by using different length genotypes [7]. HGS enables a parallel search in the optimization field by many small populations. The creation of these populations is governed by the dependent genetic process with low complexity. The HGS for independent job scheduling on computational grids for which both Flowtime and makespan parameters are simultaneously optimized. HGS performed for instances representing features of reliable and semi consistent computing environments.

D) Adaptive Genetic Algorithm: In Adaptive Genetic Algorithm (AGA) the probabilities of crossover and mutation vary with the fitness, separated by two aspects. First one is when the individual fitness value of the population tends to be reliable or in local optimum, the crossover and mutation probabilities increase [8]. On the converse, the probabilities decrease. Second one is when the fitness of current individual probabilities increase. The two adaptive strategies can guarantee the population diversity and convergence, but, the

978-1-4673-2907-1/13/$31.00 ©2013 IEEE

above strategies also have some defects such as high computing complexity.

E) Hybrid Genetic Algorithm: Hybrid genetic algorithms works by incorporating a quick and capable problem specific search procedure. They review the different mechanisms of utilizing local search information within genetic search and the various techniques to achieve a balance between exploration and exploitation. This genetic algorithm procedure takes the following steps: A population of candidate solutions (for the optimization task to be solved) is initialized [9]. New solutions are created by applying genetic operators. The fitness of the resulting solutions is evaluated and suitable selection strategy is applied to determine which solutions will be maintained into the next generation. Then the procedure is iterated.

F) Distributed Genetic Algorithm: Distributed Genetic Algorithm (DGA) is executed on the Grid, the important issues and problems are Scalability, Dynamic Changes and Heterogeneity. The results of the DGA are not tempted very much by the dynamic reduction of the number of resources [10]. This algorithm is also suitable for parallel execution because of the comparatively coarse grained communication. In this algorithm the total population is divided into sub population and the independent search will be performed in each sub population. These sub population is called as an island. After some generations, some individuals are exchanged along with islands. This operation is called migration. Moreover crossover rate and mutation rate are other parameters in DGA: the interval of migration, the number of migrates, and the migration topology. These parameters are defined by the user. DGA has the scalability for searching the solutions with respect to the number of the resources and the results of the DGA are not influenced by the dynamic reduction of the number of resources.

G) Cellular Genetic Algorithm: Cellular Genetic Algorithm is a regular GA where the individuals are placed in a known geographical distribution. Operators are useful locally on a set made of each individual and the surrounding neighbors, thus promoting intra neighborhood exploitation and inter-neighborhood exploration of the search space. The overlapped small neighborhoods of cellular GAs help in exploring the search space because of the induced slow distribution of solutions through the population providing a kind of exploration, where exploitation takes place inside each neighborhood by genetic operations [11]. The cellular genetic algorithm is used in each island of Parallel cellular GA (PEGA). The individuals can only cooperate with their near neighbors and the parents are therefore chosen from the neighborhood of the current individual with a given principle.

H) Hierarchical Parallel Genetic Algorithm: GE-HPGA using Grid computing offers a probable framework for providing considerable speed-up to evolutionary design optimization in science and engineering. It works in two levels. First one is the upper level multiple-population coarse-grained islands and Second one is the lower level global single population master-slaves. In the lower level the processing load is divided into several slaves, under the coordination of a master which is responsible for initializing the population, executing the selection procedure, applying the genetic operators, and distributing individuals to slaves [12].

I) Reliability with Genetic Algorithm: Genetic algorithm to compute the reliability of a stochastic- flow network in which each curve or node has several capacities. The algorithm is based on generating all lower boundary points for the given demand and then the system reliability can be calculated in terms of such points. The system reliability of a flow network is the probability that the maximum flow of the network is not less than the given demand [13]. For an application in public-resource computing environments, it provides reliable scheduling based on resource reliability evaluation. It is becoming gradually more essential. Existing reputation models are used for reliability evaluation ignoring the time influence.

J) Energy-aware with Genetic Algorithm: Genetic Algorithms (GAs) with exclusive and struggle replacement Mechanisms as energy-aware schedulers. The main purpose of these grid schedulers are to efficiently and optimally allocate tasks originated by applications to a set of available resources and one should consider a series of requirements including energy efficiency [14]. Energy-efficient scheduling in CGs becomes thus a complex effort due to the multi-constraints, different optimization criteria and different priorities of the resource owners. Energy efficiency has been introduced by the large scales of activity computing environments and data centers. Due to the value of power and energy consumption in modern-day and future computing and communication systems different techniques and latest technologies have been investigated and developed.

K) Multiple Qos Genetic Algorithm: Genetic algorithm is used to solve optimization problems by imitating the genetic process of genetic organisms. First, the population that is a set of chromosomes is generated and it undergoes many genetic processes. By using selection, crossover and mutation operations, GA is able to develop the population to generate an optimal solution [15]. The evolutionary GA using Multiple Qos satisfaction for scheduling independent tasks on heterogeneous grid resources in offline mode is proposed. The main focus of this work is to minimize the makespan and user's required Qos is satisfied when selecting the resources. The overall completion time of the given tasks and user's required Qos is satisfied when selecting the resources.

978-1-4673-2907-1/13/$31.00 ©2013 IEEE

TABLE 1
THE COMPARISON OF VARIOUS GENETIC ALGORITHMS

Algorithm	**Exploitation**	**Objective**	**Dynamic/Static**
Panmictic Genetic Algorithm[5]	Explicit	Multi-Objective	Dynamic
Parallel Genetic Algorithm[6]	Explicit and Immediate	Multi-Objective	Dynamic
Hierarchical Genetic Algorithm[7]	Classical	Bi-Objective	Dynamic
Adaptive Genetic Algorithm[8]	Average Division	Multi-Objective	Static
Hybrid Genetic Algorithm[9]	Optimal and Over	Multi-Objective	Dynamic
Distributed Genetic Algorithm[10]	Maximum Level	Multi-Objective	Dynamic
Cellular Genetic Algorithm[11]	Intra Neighborhood	Multi-Objective	Dynamic
Hierarchical Parallel Genetic Algorithm[12]	Optimal	Multi-Objective	Dynamic
Reliability with Genetic Algorithm[13]	Optimal	Multi-Objective	Static
Energy-Aware with Genetic Algorithm[14]	Low effectiveness	Bi-Objective	Dynamic
Qos Genetic Algorithm[15]	Random	Multi-Objective	Static

IV.CONCLUSION

In this paper we have addressed some of the variants of genetic algorithms and their characteristics. Achieving high quality planning of jobs into machines is crucial for grid systems due to their highly heterogeneous and dynamic nature. From our study we observe that the Hierarchic Genetic Strategies (HGS) is an efficient method for scheduling on computational grids. According to this framework, HGS can be applied in solving continuous global optimization problems with multi-modal and weakly convex objective functions. The Hierarchical GA which gives good computation complexity in solving global optimization problems by using different length genotypes. A survey has been made regarding the characteristics of different genetic algorithms. The comparisons show that hierarchical genetic algorithm can overcome many of the detracting features in the other variants of algorithms discussed.

REFERENCES

[1] Ajith Abraham, Hongbo Liu, Crina Grosan, Fatos Xhafa, Nature Inspired Meta-heuristics for Grid Scheduling: Single and Multi-objective Optimization Approaches,□Centre for Quantifiable Quality of Service in Communication Systems,

[2] R. Buyya, Economic-based distributed resource management and scheduling for grid computing, Ph.D. Thesis, Monash University, Australia, 2002.

[3] Javier Carretero, Fatos Xhafa, Genetic Algorithm Based Schedulers for Grid Computing Systems, International Journal of Innovative Computing, Information and Control Volume 3, Number 6, December 2007.

[4] Albert Y. Zomaya ,Observations on Using Genetic Algorithms for Dynamic Load-Balancing, IEEE Transactions On Parallel And Distributed Systems, Vol. 12, No. 9, September 2001.

[5] Travis Desell, Boleslaw Szymanski, and Carlos Varela, The Effects of Heterogeneity on Asynchronous Panmictic Genetic Search Department of Computer Science, Rensselaer Polytechnic Institute, Troy NY 12180, USA,.

[6] Erick Cantu-Paz, A Survey of Parallel Genetic Algorithms, Department of Computer Science and Illinois Genetic Algorithms Laboratory University of Illinois at Urbana-Champaign.

[7] J.Kołodziej, Fatos Xhafa, Hierarchic Genetic Scheduler of Independent Jobs in Computational Grid Environment, supported by 10/I/GW=2008 ATH Grant.

[8] Yang Gao, Hongqiang Rong, Joshua Zhexue Huang, Adaptive grid job scheduling with genetic algorithms, Future Generation Computer Systems 21 (2005) 151–161.

[9] Byung Joo Park, Hyung Rim Choi, Hyun Soo Kim, A hybrid genetic algorithm for the job shop scheduling problems, Computers & Industrial Engineering 45 (2003) 597–613

[10] Yusuke Tanimura, Tomoyuki Hiroyasu, Mitsunori Miki, Discussion on Searching Capability of Distributed Genetic Algorithm on The Grid, supported by a grant to RCAST at Doshisha University from the Ministry of Education, Science.

[11] B. Dorronsoro, Daniel Arias ,Francisco Luna, Antonio J. Nebro, Enrique Alb , A Grid-Based Hybrid Cellular Genetic Algorithm For Very Large Scale Instances Of The Cvrp, supported by the Spanish Ministry of Education and Science, and by European FEDER under contract TIN2005-08818-C04-01.

[12] Dudy Lima Yew-Soon Ong, Y.Jin, Bernhard Sendhoff Bu-Sung Lee, Efficient Hierarchical Parallel Genetic Algorithms Using Grid Computing.

[13] Younes, M. R. Hassan, A Genetic Algorithm for Reliability Evaluation of a Stochastic- Flow Network with Node Failure, International Journal of Computer Science and Security (IJCSS), Volume (4): Issue (6).

[14] Joanna Kołodziej, Joanna Kołodziej, Fatos Xhafa, Genetic Algorithms for Energy-aware Scheduling in Computational Grid.

[15] P. Devaki, M. L. Valarmathi,Job Scheduling using Genetic Algorithm with Qos Satisfaction in Grid, European Journal of Scientific Research ISSN 1450-216X Vol.74 No.2 (2012), pp. 272-285,© Euro Journals Publishing, Inc. 2012.

Joshua Samuel Raj, working towards the Ph.D degree in grid scheduling received his B.E degree from PETEC under Anna University and M.E degree from Jeya College of Engineering under Anna University. His areas of interest include Grid Computing, Mobile Adhoc Networking, Multicasting and so forth.

Riya Mary Thomas, pursuing her M.Tech in Computer Science and Engineering from the department of Computer Science in Karunya University received her B.E in 2011 from Nandha College of Engineering under Anna University.