

Genetic Based Qos Task Scheduling In Cloud -Upgrade Genetic Algorithm

Ashima Mittal¹ and Dr. Pankaj Deep Kaur²

¹*Department of Computer Science and Engineering
Guru Nanak Dev University, Regional Campus
Jalandhar, India*

²*Department of Computer Science and Engineering
Guru Nanak Dev University, Regional Campus
Jalandhar, India*

¹*Er.ashimamittal2013@gmail.com,* ²*pankajdeepkaur@gmail.com*

Abstract

Cloud computing has been emerged as a new service model in computing world and giving lot of interest to the researchers to find its benefits. Task scheduling is a major conflict in cloud environment and genetic algorithms are one of the optimization techniques to solve that problem. Virtual machine's processing elements are important criteria to solve a scheduling problem. In proposed algorithm called upgraded Genetic algorithm, initial population is sorted according to the number of processing elements of each virtual machines. Proposed algorithm is compared with MGA in terms of cost and with MACO in terms of make span. Experiment results shows upgrade genetic algorithm gives better efficiency in term of cost and make span.

Keywords: *Task scheduling, Algorithms, Genetic algorithm, UGA, Make span, cost*

1. Introduction

With the increasing rate of internet usage and virtualization facility, cloud computing has used as new computing platform. On cloud platform, according to needs of users, services and guaranteed resources are provided. In cloud environment, term cloud represents the service provider which contains all the computing resources. Cloud can be considered as pool of computing resources which are available to users on pay basis.

Scheduling means schedule tasks to set of resources to satisfy user requirements and to meet certain level of performance. The main objective of scheduling involves minimizing costs, minimizing make span, minimize load and maximize the utilization of resources. There are two primary conditions on scheduling that are temporal and resource constraints. Random guided search, approximation and list scheduling are approaches including in scheduling. NP-hard scheduling problems are solved by algorithm that give nearby optimal schedule and have a high time complexity. For any upper limit on time complexity, the quality of schedule in general will also be limited [1]. Despite of high complexity of some algorithms like genetic algorithm (GA), Tabu search (TS), simulated Annealing (SA), they are adaptive in nature because of not making assumptions about the structure of problem. Cloud computing has many advantages: There is no need of initial installation of hardware and software resources, it helps in cost saving. It provides a flexibility and scalability of services and resources. Because of all the resources are managed by cloud service providers. So, Maintenance cost becomes less. Genetic Algorithm are most popular evolutionary computation technique [2]. GA is derived from the process of biological organism evolution and considered as artificial intelligence search meta-heuristic algorithm. Inspire of many scheduling algorithms, only genetic algorithm have a unique

combination of parallel population based search with stochastic selection of many individual, stochastic crossover and mutation [2].

2. Task Scheduling – Necessity of Cloud Computing

Scheduling aims to effectively schedule the tasks so as to proper utilization of resources and meet the QoS need of users. By considering user needs and virtual machine information, tasks are allocated to VM's. Application scalability and on demand resource provisioning are key factor of scheduling. Cloud provider has to take care of budgetary constraints and time needs of users while allocating tasks to VM's.

3. Related Work

Monir Abdullah et.al [3] proposed a improved genetic algorithm [IGA] by focusing on time consuming problem of genetic algorithm. Generating the initial solution is changed in proposed algorithm else remains same as Genetic algorithm. IGA algorithm performs better in term of cost and execution time.

Saeedet.al [4] proposed a modified version of genetic algorithm by combining it with fuzzy theory. Proposed algorithm assigns jobs by calculating resources available and length of job. Algorithm performs well in term of degree of imbalance (DI), cost, execution cost and execution time.

JianhuaGueta.al [5] proposed a new algorithm based on the variation rate of load and average load distance. The proposed work performs well in term of load imbalance and cost.

Ali haydarzadeganet.al [6] proposed a algorithm by considering homogeneous cloud environment and scheduling of independent tasks. Survival time of task is chosen as priority of tasks to run. Algorithm performs well for real time systems in term of execution time.

In [7] priority formula is used to select tasks for scheduling and resources needed. Algorithm proves well in term of monitory gains.

Congcong Xiong [8] et.al proposed a new algorithm by using G for the efficient use of resources in heterogeneous environment. The new approach improves quality of individuals by adopting rule-bound crossover and mutation. Proposed algorithm proves better to meet user's expectation in terms of cost, time and reliability.

4. Evolution of Basic Genetic Algorithm

A genetic algorithm is optimization algorithm based on the evolution of natural genetics. GA is used for the scheduling problems with either independent or interdependent tasks. Choosing the right representation of gene is the first step of genetic algorithm. Pear and vector are two representations of genes.

4.1. Encoding and Initialization [9]

The central factor in the success of genetic algorithm is the way in which candidate solution is encoded. There are basically 3 types of encoding techniques- Bit string encoding, integer encoding, floating point encoding. In initialization, it is a set of all the individuals used to generate a optimal solution using genetic algorithm. Each individual represents the solution in population. Every individual represents chromosome information. By using specific criteria, some individuals are selected and used to generate next generation.

4.2. Fitness Function

For each chromosome, fitness value is generated by fitness function. The suitability of that chromosome in solving task scheduling problem is indicated by fitness value. In general task scheduling problems, fitness function depends only on execution time of tasks but in cloud task scheduling problem it is different due to the SLA agreement between user and cloud provider. Fitness function in genetic algorithm is defined by user's satisfaction and provider's profit.

4.3. Selection [10]

Selection scheme plays important role in genetic algorithm; it selects the best chromosome according to the fitness function and improves the chances of survival of fittest individual.

There are numerous selection method used in genetic algorithm-

1. Rank selection
2. Roulette wheel method
3. Boltzmann selection
4. Elitism
5. Tournament selection
6. Steady state selection

4.4. Crossover [11]

Crossover represents reproduction by selecting some individuals and cross part of them. Crossover is done to produce a better individual by taking the best characteristics from both individuals that are used in crossover. Generally the crossover probability is between 0.5 and 1.0. Types of crossover are-

1. Single point crossover
2. Two point crossover
3. Uniform crossover
4. Arithmetic crossover
5. Heuristic crossover

4.5. Mutation [12]

Mutation operation is done to expand the search space by changing one or more bits of individuals randomly and produce different offspring. Types of mutation are-

1. Dynamic mutation genetic algorithm (DMGA) [13]
2. Schema mutation genetic algorithm(SMGA)[14]
3. Basic mutation operator
4. Non-uniform mutation
5. Adaptive mutation
6. Compound mutation genetic algorithm(BCM-GA)[15]
7. Clustered based adaptive mutation(CSAM) [16]
8. Hyper mutation based genetic algorithm(HMGA) [17]

4.6. Replacement

It is a last stage of any generation breeding cycle. In replacement technique, two parents are breed to produce two new individuals but not all the individuals can return to population, it depends on replacement technique that which individual will be replaced and which will be return back. Generational updates and steady state updates are two methods of maintaining population. Other methods are-

1. Random replacement
2. Weak parent replacement
3. Both parents replacement

5. Problem Formulation

Cloud model is three tier architecture the first tier is concerned with clients who have J VM's requests to run their application, second tier deals with broker that is responsible to provide business services, third tier deals with service provider that gives N VM's instances for rent, the main job of brokers is to find the best combination of instances and schedule the VM requests to those instances. By taking care of cost, make span and performance index requests are scheduled on VM instances.

The problem of scheduling means to schedule J client requests on N combination of VM instances available by taking care of cost and execution time at the same time.

Our proposed model focused on the task scheduling problem using an optimizing algorithm i.e. Upgrade genetic algorithm. The proposed algorithm considers the processors available on each VM into account. Main objective function of the proposed scheme is to calculate processor cost and makespan and to reduce them as minimum as possible.

6. Proposed Approach

Upgrade genetic algorithm (UGA)

Begin main

[Initialize] Generate an initial population of individuals by applying topological sorting on the chromosomes according to the processing elements of VM. Sorted list of chromosomes is used further.

[Fitness] Fitness value of every chromosome is calculated in given population.

While termination conditions are not meant, repeat the following steps

[Selection] Select two parent individual from the population according to fitness value.

[Crossover] A crossover probability is used to generate new offspring by combining two parents.

[Mutation] change the bit position of new child by using mutation probability.

[Accepting] Now the new generated offspring becomes the part of next generation population.

[Replace] New generation replace the old generation and used as current generation.

End main

Step 1: Initialization- we have generated a sorted list of chromosomes rather than random initialization.

Chromosome list is sorted according to the processing elements(disk, processors, IO) of VM's. First generation s allocated on a VM having maximum processing elements.

Step 2: Fitness value

Fitness value is calculated according to fitness function.

Step 3: selection

In proposed algorithm, three selection types are used that are tournament, roulette wheel and elitism.

Step 4: Crossover

Crossover operator is most important part of genetic algorithm. Proposed scheme focus on one point, two point and uniform crossover.

Step 5: Mutation

There are several mutation functions but we use mutation type swap.

Step 6- Evaluation

Evaluation is done on two main objective functions that is makespan and processor cost. The selected approach will give better efficiency in term of makespan and cost.

Step 7- Termination

Genetic algorithm is terminated after user defined specification. We generated 100 evaluations to get better results.

7. Implementation and Results

A good scheduling algorithm leads to better efficiency of system Algorithm is run on Intel i3 processor with 500 Gb HDD, eclipse with java version 1.6.Cloudsim 3.0 simulator has been used to check the performance of upgrade genetic algorithm.

Our algorithm is compared with MGA [18] in terms of cost in the same cloudsim environment and with MACO [19] in terms of makespan. Upgraded genetic algorithm (UGA) proves better in term of makespan and cost as shown in graph below. Speed of processor is taken in MIPS (million instructions per second).

Table 1. Values Used in Implementation

Parameters	Values
Number of generations	100
Number of iterations	150
Number of population size	100
Crossover type	One point
Mutation type	Swap
Termination condition	Number of iterations
Selection type	Elitism, tournament and roulette wheel.
Mutation probability	0.8
Crossover probability	0.9

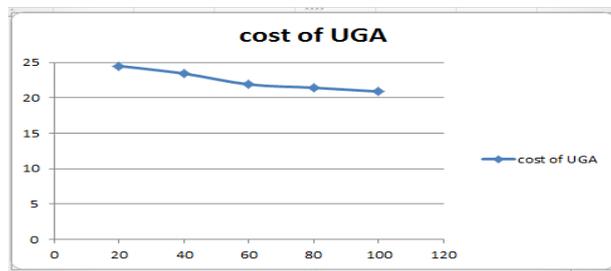


Figure 1. Graph between Cost and Number of Processors (UGA)

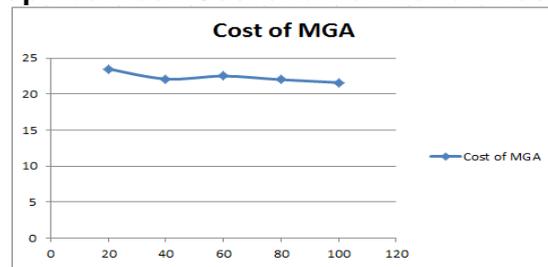


Figure 2. Graph between Cost and Number of Generations (MGA) [18]

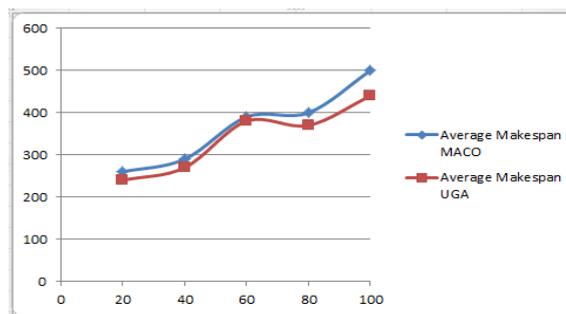


Figure 3. Graph between Makespan of UGA and MACO [19]

8. Conclusion and Future Work

Clouds enable the users to use utility services. In this research we have proposed an upgrade genetic algorithm by changing the initial step of standard genetic algorithm. By sorting an initial population, give us valuable results. If we consider the problem on a large scale such as in the cloud computing environment, with the advantage of the Genetic Algorithm approach; this technique is able to give better results. Cost and makespan are two parameters on which fitness is calculated. UGA is compared with MGA in terms of cost and with MACO in terms of makespan. Our proposed work gives better performance in terms of cost and makespan. In future, we will be further enhancing our algorithm by changing the crossover and mutation probabilities types to check the results. Testing will also be performed in more real time environment like amazon, eucalyptus etc.

Acknowledgements

I would like to thank my guide and my friends to help me throughout the paper.

References

- [1] Y. -C. Lee and A. Zomava, "A Novel state transition Method for metaheuristic-Based scheduling in heterogeneous computing systems", *IEEE transactions on parallel and distributed systems*, vol. 19, no.9, (2008) September, pp. 1215-1223, DOI 10.1109/TPD.2007.70815
- [2] M. Mitchell, "An Introduction to Genetic Algorithms", MIT Press, (1996).
- [3] M. Abdullah and M. Othman, "An improved genetic algorithm for job scheduling in cloud computing environment", 2nd world conference on innovation and computer science, (2012).
- [4] S. Javanmardi, M. Shojafar, D. Amendola, N. Cordeschi, H. Liu and A. Abraham, "Hybrid job scheduling algorithm for cloud computing environment", SpringerVerlag Berlin Heidelberg (2014).
- [5] J. Gu, J. Hu, T. Zhou and G. Sun, "A New Resource Scheduling Strategy Based on Genetic Algorithm in cloud computing", *journal of computers*, vol. 7, no.1, (2012) January.
- [6] A. Heydarzadegan, Y. Nemati, M. I. Jamnezhad and M. Moradi, "Offering a New Approach to optimal scheduling of tasks in the cloud using chromosome partitioning", *International Research Journal of Applied and Basic Sciences* (2014).
- [7] H. A. Ravani, H. A. Bheda and V. J. Patel, "Genetic algorithm based resource scheduling technique in cloud computing", *international journal of advanced research in computer science and management studies* vol. 1, issue 7, (2013) December, ISSN: 2321-7782.
- [8] C. Xiong, L. Feng and L. C. Congcong, "A New Task Scheduling Algorithm Based On Improved Genetic Algorithm In cloud Computing Environment", *Advances in Information Science and Services Sciences(AISS)* vol. 5, no. 3.
- [9] A. P. Engelbrecht, "Computational Intelligence An Introduction", Second Edition.
- [10] A. L. Haupt and S. E. Haupt, "Practical Genetic Algorithms", John Wiley & Sons, (1998).
- [11] H. Cheng and S. Yang, "Hyper-mutation Based Genetic Algorithms for Dynamic Multicast Routing Problem in Mobile Ad Hoc Networks", *Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2012 IEEE 11th International Conference on. IEEE, (2012).
- [12] K. Jebari and M. Madiafi, "Selection Methods for Genetic Algorithms", *Int. J. Emerg. Sci.*, vol. 3, no. 4, (2013) December, pp. 333-344, ISSN: 2222-4254 © IJES.
- [13] M. D. Vose, "The Simple Genetic Algorithm", The MIT Press, (1999).
- [14] S. Deutsch, "The case for large-size mutations," *IEEE Transactions on Biomedical Engineering*, vol. 48, Issue: 1, (2001), pp. 124-127.
- [15] T. -P. Hong and H. -S. Wang, "A dynamic mutation genetic algorithm", *IEEE International Conference on* vol. 3, (1996).
- [16] F. Li and T. Zhang, "Study on Genetic Algorithm Based on Schema Mutation and Its Performance Analysis", *Electronic Commerce and Security, 2009. ISECS'09. Second International Symposium on*. vol. 1, (2009).
- [17] F. Li, T. Zhang and C. Jin, "A kind of genetic algorithm based on compound mutation strategy and performance study", *Wireless Communications, Networking and Mobile Computing, 2009. WiCom'09. 5th International Conference on*. IEEE, (2009).
- [18] G. E. Junwei, Y. Yongsheng, "Research of cloud computing task scheduling algorithm based on improved genetic algorithm", *Proceedings of the 2nd International Conference on computer Science and Electronics engineering ICCSEE* (2013).
- [19] M. A. Tawfeek, A. El-Sisi, A. E. keshk, F. A. Torkey, "An Ant colony Algorithm for cloud Task scheduling", *International Workshop on Cloud Computing and Information Security CCIS* (2013).

Authors



Ashima Mittal, has completed her M.tech from Guru Nanak Dev University regional campus, Jalandhar. She completed her B.tech from Bhai Gurdas Institute of Engineering and Technology, Sangrur. Her area of interest is cloud computing, Algorithms.



Dr. Pankaj deep kaur, is working as an Assistant Professor in the Department of Computer Science and Engineering, Guru Nanak Dev University, RC, Jalandhar, India. She completed her Ph.D. in Resource Scheduling in Cloud Computing from Thapar University, Patiala (2014) and has over ten years of teaching and research experience. Her research interests include Cloud Computing and Big data.

