A Genetic Algorithm inspired task scheduling in Cloud Computing

Mohit Agarwal
Dept. of Physics & Computer Science
Dayalbagh Educational Institute
Agra, India
rs.mohitag@gmail.com

Dr. Gur Mauj Saran Srivastava
Associate Professor, Dept. of Physics & Comp. Science
Dayalbagh Educational Institute
Agra, India
gurmaujsaran@gmail.com

***Abstract*— In the current scenario, Cloud computing carved itself as an emerging technology which enables the organization to utilize hardware, software and applications without any upfront cost over the internet. The challenge before the cloud service provider is, how efficiently and effectively the underlying computing resources like virtual machines, network, storage units, and bandwidth etc. should be managed so that no computing device is in under-utilization or over-utilization state in a dynamic environment. A good task scheduling technique is always required for the dynamic allocation of the task to avoid such a situation. Through this paper we are going to present the Genetic Algorithm based task scheduling technique, which will distribute the load effectively among the virtual machine so that the overall response time (QoS) should be minimal. A comparison of this Genetic Algorithm based task scheduling technique is performed on CloudSim simulator which shows that, this will outperform the existing techniques like Greedy based, First – Come first – Serve (FCFS) techniques.**

Keywords*— Cloud Computing, Cloud Service provider, Genetic Algorithm, Virtual Machine, QoS.

INTRODUCTION

Cloud computing emerges out as an fastest growing technology in the IT world which focuses on providing the IT services and computing resources round the clock and across the globe to its users over the internet [1]. The underlying cloud computing infrastructure and services are generally owned and maintained by the third party, known as cloud service provider. The major advantage of cloud computing over the existing IT technologies are – On-demand self-service, broad network services, rapid elasticity, resource pooling, and measured service. Growth of cloud service may leads to the slowdown in throughput, utilization of computing resources and which ultimately reduce the efficiency of the cloud system. One of the drawback associated with cloud computing is that it works in dynamic environment and proper task scheduling mechanism should be deployed to ensure that no computing node is under or over utilized otherwise this will ultimately affect the overall response time or throughput. Scheduling basically performed to map the tasks or cloudlets to the specific Virtual Machines (VM's) to increase the resources utilization. An efficient task scheduling mechanism is always required not only to maintain the throughput, but also helps the service provider to maintain the conditions of SLA's (Service Level Agreement) by providing the good QoS (Quality of Service).

In this paper, we proposed a Genetic Algorithm based task scheduling in cloud computing environment which proves very efficient in multiobjective optimization problems and also compared the results with the well-established strategies in the field. The rest of the paper is organized as follows. Section 2, proposes the Genetic Algorithm introduction, Section 3 shows the related work, Section 4 our proposed approach and Section 5 presents the results and analysis followed by conclusion.

GENETIC ALGORITHM

Genetic Algorithms are inspired by the theory of evolution, according to which fittest species/solution will survive and allowed to reproduction while the unfit or weak will be discarded or extinct [2]. Genetic algorithms are renowned to for their stability and efficiency for finding the global optimum solution in a very complex and large sample space where most of the other algorithms fails. They are also beneficial in finding the optimum solution for both single objective as well as multiobjective problems [3]. That's why we prefer them in task scheduling in cloud computing environment.

In GA, every individual solution is known as chromosome and the collection of chromosome is said to be population. Basic terminologies in GA are as follows:

A. *Initial Population:* - GA primarily works on the set of chromosomes which may be the string of bits (0 or 1) or real numbers.

B. *Crossover:*- This is responsible for the generation of new population on the basis of fitness value. Fitness of each individual or chromosome is evaluated and on the basis of which the fittest ones are allowed to reproduce. This involves the exchange of the genes in the chromosomes which are allowed to reproduce. Crossover may be uniform, single-point or two-point crossover. It is governed by the crossover probability value.

ISBN: 978-1-5090-1666-2/16/$31.00 ©2016 IEEE

C. *Mutation:-* This helps in generation of new population. Mutation probability normally keeps very low.

RELATED WORK

Task mapping in a cloud computing environment is the NP-complete problem. Cloud computing is relatively the new field and lot of scope is available to work on performance enhancement by using the effective task scheduling algorithm. Lot of work on task scheduling has been done and in this section we are going to discuss some relevant work. In [4] author has proposed a first-fit strategy, which is used by some cloud computing system such as Eucalyptus [5], In such methods, the problem of starvation is nearly solved, and the makespan is reduced for the jobs. But, this method not support the optimal usage of the resources as requests will execute on each resource. In [6], Armstrong et. al. proposes Minimum Execution Time (MET) approach to assign order to every job in arbitrary manner to the computing nodes on which they are expected to be executed faster, regardless of the load on that node. Scheduling techniques like Round Robin, Greedy and FCFS for load balancing also exist. In [7] Yang Xu et. al. proposes an intelligent method for load balancing which involves data distribution to improve cloud computing performance. Few soft computing techniques like ACO [8] are also reported. In [9] the author proposed an improved generic algorithm (IGA) as scheduling algorithm which is used for efficient utilization of the computing resources in system. In [10] the author proposed genetic algorithm based task scheduling in Hadoop MapReduce.

OUR PROPOSED TECHNIQUE

Task scheduling

Task scheduling may be defined as the process of mapping of the task or jobs to the virtual machines (VM's) i.e. computing resources for their processing. It is well known fact that the task scheduling is a NP-complete problem [11]. So a well efficient algorithm is always required for task scheduling. As we know jobs or task need computing resources for their execution and there is also a possibility that the computing resource might not be available at the current location. Task migration is required to transfer these tasks to the computing resource rich location for faster execution.

In the experiment and analysis section, we have taken the 12 tasks (T0, T1, T2 ….T11) which are supposed to be executed on any of the given 5 VM's (VM0, Vm1…VM4). For example, chromosome {3, 3, 0, 4, 3, 3, 0, 3, 0, 3, 3 2} represent VM allocation to tasks in order. So there can be 5^{12} ways to allocate them. Our task is to allocate the tasks in easy and efficient manner so that the execution time of all tasks should be minimal time by using the GA based approach by considering all the factors which can affect the execution time.

Objective function

The primary objective for the task scheduling in cloud computing environment is to minimize the execution time of tasks by placing them on suitable VM's, so that maximum time taken by the task is minimum of the resulting solution. To achieve the same, we formulate the given below objective function:

$$FT_i = \sum_{j=0}^{m} E_{ji} + C_{ji} \qquad (1)$$

Where: FT_i = finish time for ith VM

i = sequence number of the VM.

j = Sequence number of the cloudlet.

E_{ji} = time required by jth cloudlet when process by ith VM

C_{ji} = (task Isize + task Osize)/Bandwidth

EXPERIMENT AND ANALYSIS

Experiment Enviornment and Parameter Setting

We have compared our GA based task scheduling strategy with the greedy-algorithm, FCFS (First-Come-First-Serve) scheduling strategy and find that the GA based strategy has more superior performance in comparison to others. In this paper we have used the CloudSim simulator [12] designed by the University of Melbourne, Australia to test our result. To test our GA based approach and other strategies; Table 1 is presenting the GA parameters,Table 2 is used to show the configuration of twelve tasks, Table 3 is used to represent the characteristics of five VM's while the Table 7 represents sample population.

Result Analysis

From table 4 we can see that in FCFS algorithm tasks are assigned to the virtual machines in order of occurrence, the allocation policy is simple but it is not efficient as it taking the huge time i.e. 343.52 time units for the execution all tasks while the greedy based strategy is taking 101.79 time units and GA- Based approach is taking the 100.55 time units. So it is easy to conclude that GA based approach for task scheduling is much faster and efficient in comparison to the other two.

TABLE 1: GA PARAMETER SETTING

Population Size	20
Maximum Iteration	150
Cross-Over Type	Single Point
Cross-Over Probability	0.9
Mutation probability	0.1
Mutation Operator	Bit flip
Selection Operator	Tournament

TABLE 2: TASK CONFIGURATION.

TaskID	Task Length	Task ISize	Task Osize
0	1000	1000	1000
1	1000	1000	900
2	2000	500	1000
3	2000	700	900
4	5000	1000	1200
5	5000	1200	1000
6	5500	1050	1050
7	5500	1300	800
8	10100	1200	900
9	10100	1500	1700
10	100000	1800	1800
11	100000	2000	2000

TABLE 4: TASK SCHEDULING GA- BASED APPROACH

TaskID	VMID	Execution Time	Start Time	Finish Time
0	1	2.67	.1	2.77
1	1	2.63	2.77	5.40
2	4	1.3	.1	1.4
3	0	6.99	.1	7.09
4	2	5.27	.1	5.37
5	2	6.47	5.37	12.84
6	0	18.75	7.09	25.84
7	4	3.17	1.4	4.57
8	0	33.75	25.84	59.59
9	4	5.64	4.57	10.21
10	3	100.45	.1	**100.55**
11	4	50.8	10.21	61.01

TABLE 7: SOME SAMPLE POPULATION

3	3	0	4	3	3	0	3	0	3	3	2
4	2	3	4	1	0	3	1	2	4	1	1
1	3	4	0	4	1	0	2	2	4	2	2
3	3	1	4	4	4	2	4	3	2	0	3

TABLE 3: VM'S CONFIGURATION

VMID	MIPS	Bandwidth
0	1000	1000
1	1000	1000
2	2000	500
3	2000	700
4	5000	1000

TABLE 5: TASK SCHEDULING FCFS- BASED APPROACH

TaskID	VMID	Execution Time	Start Time	Finish Time
0	2	1.45	.1	1.55
1	3	1.24	.1	1.34
2	0	6.97	.1	7.07
3	0	6.99	7.07	14.06
4	4	2.94	.1	3.04
5	1	10.73	.1	10.83
6	1	11.7	10.83	22.53
7	4	3.17	3.04	6.21
8	4	5.42	6.21	11.63
9	1	21.07	22.53	43.7
10	3	100.45	1.34	**101.79**
11	4	50.8	11.63	62.43

TABLE 6: TASK SCHEDULING GREEDY- BASED APPROACH

TaskID	VMID	Execution Time	Start Time	Finish Time
0	0	3.73	0.1	3.83
1	1	2.63	0.1	2.73
2	2	2.65	0.1	2.75
3	3	2.2	0.1	2.3
4	4	2.94	0.1	3.04
5	0	17.11	2.3	19.41
6	1	11.7	2.73	14.43
7	2	7.09	2.75	9.84
8	3	10.26	3.04	13.3
9	4	5.64	3.83	9.47
10	0	334.05	9.47	**343.52**
11	1	201.33	9.84	211.17

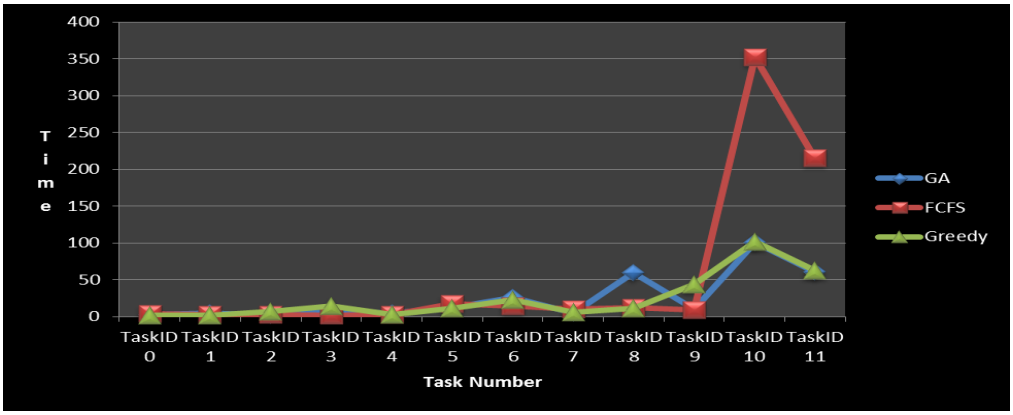

Fig. 1. Execution time of each cloudlet

CONCLUSION

Through this paper, we have proposed the Genetic algorithm based task scheduling mechanism which clearly outperforms the other performance of the other two algorithms in cloud computing environment. GA based approach allocated the given tasks in such way that the maximum time required by the task is minimal. The proposed algorithm can be further applied on the other QoS for resource scheduling and there is a scope for the variants of GA also.

REFERENCES

[1] M.D. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis, A. Vakali, "Cloud computing: distributed Internet computing for IT and scientific research." IEEE Internet Computing, 13(5), 2009, pp. 1013.

[2] A. Konak, D. W. Coit, and A. E. Smith, "Multi-objective optimization using genetic algorithms: a tutorial," Reliability Engineering & System Safety, vol. 91, no. 9, pp. 992–1007, 2006.

[3] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," IEEE Transactions on Evolutionary Computation, vol. 6, no. 2,pp. 182 197, 2002.

[4] R. P. Brent, "Efficient implementation of the first-fit strategy for dynamic storage allocation", ACM Transactions on Programming Languages and Systems (TOPLAS), vol. 11, (1989), pp. 388-403.

[5] D. Nurmi, R. Wolski, C. Grzegorczyk, G. Obertelli, S. Soman, L. Youseff, et al., "The eucalyptus opensource cloud-computing system", in Cluster Computing and the Grid, 2009. CCGRID'09. 9th IEEE/ACM International Symposium on, (2009), pp. 124-131.

[6] T. R. Armstrong, D. Hensgen, "The relative performance of various mapping algorithms is independent of sizable variances in runtime predictions", in Proc. of 7th IEEE Heterogeneous Computing Workshop (HCW 98), pp. 79-87,1998.

[7] Yang Xu, Lei Wu, Liying Guo, Zheng Chen,Lai Yang, Zhongzhi Shi, "An Intelligent Load Balancing Algorithm Towards Efficient Cloud Computing", in Proc. of AI for Data Center Management and Cloud Computing: Papers, from the 2011 AAAI Workshop (WS-11-08), pp. 27–32, 2008.

[8] Ratan Mishra and Anant Jaiswal, "Ant colony Optimization: A Solution of Load balancing in Cloud",in International Journal of Web & Semantic Technology (IJWesT), Vol.3, No.2, pp. 33–50, 2012.

[9] P. Kumar, A. Verma, Scheduling using improved genetic algorithm in cloud computing for independent tasks, in Proceedings of the International Conference on Advances in Computing, Communications and Informatics, Chennai, India, 2012, 137-142

[10] Y. Ge and G.Wei, "GA-based task scheduler for the cloud computing systems," in Proceedings of the International Conference on Web Information Systems and Mining (WISM '10), vol. 2, pp.181–186, IEEE, October 2010.

[11] J. D. Ullman, "Np-complete scheduling problems." J.Comput. Syst. Sci, 10(3), 1975.

[12] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," Software—Practice and Experience, vol. 41, no. 1, pp. 23–50, 2011.