VOL. 11, NO. 17, SEPTEMBER 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

A GENETIC ALGORITHM BASED LOAD BALANCING TECHNIQUE (GALBT) FOR APPLICATION PROCESSING IN CLOUD

J. Janet[1], G. Sreelatha[2] and A. B. Manju[2]
[1]CSE and IT Sri Venkateswara College of Engineering and Technology, Thiruppachur, Tiruvallur, India
[2]Department of Computer Science Engineering, Sri Venkateswara College of Engineering and Technology, Thiruppachur, Tiruvallur, India
E-Mail: janet.fredrick@gmail.com

ABSTRACT

The applications of Cloud computing has penetrated all services recently. Enormous data processing in cloud is done by distributing the data among the virtual machines. The virtual machines load capacities are dynamically changing based on the request sent by the client. Genetic Algorithm based Load Balancing Technique (GALBT) has been proposed to equally distribute the load among the virtual machines and for rapid processing. Virtual machine scheduler (Vm scheduler) has been designed to estimate the time and resource requirements of the task processing. Based on estimation, the task is forwarded to virtual machines for processing. This project is developed using cloud simulator and results have been compared with round robin and throttled algorithms to show our strategy performs better for application processing.

Keywords: genetic algorithm, load balancing, application processing, Vm scheduler.

1. INTRODUCTION

In legacy systems, resources needed are to be brought and then they make use of it, whereas in cloud environment cloud resources are touting to provide users with all resources at low rates. People pay only for what they use. So, many legacy systems are moving to a cloud computing environment. Cloud users expect fast accessing, processing, etc from the cloud service providers, whereas internal working process of cloud environment is hided from the users. Internally cloud service providers have to manage with the available resources to satisfy all the users. In general not all the resources of the cloud are used evenly. Few are overloaded while few are idle. To efficiently make use of available resources many load balancing policies are used. In cloud environment the number of users at any time is not constant. Hence load balancing should also be planned accordingly.

Load balancing makes the resources to be available to all the users and making the user's task without waiting for very long time. There are many load balancing algorithms in existence they are round robin, first come first serve, priority scheduling etc. Each and every scheduling criterion is best suited for a particular situation. Here we have proposed Genetic Algorithm Based Load Balancing Technique (GALBT). Genetic Algorithm is a natural process by which the evolution takes place. In nature fittest parents produce offsprings. In machine learning type of genetic algorithm there exist randomly selected solutions from which the new powerful results are created from the current. In common there are three types of operations existing in genetic algorithm process they are selection, crossover and mutation.

In selection process the parents are selected for the reproduction. Parents' selection is based on the chromosomes. The fittest parents get selected; they undergo further process of crossover and mutation operations. In crossover the parents' chromosomes are recombined to produce the new individual, whereas in mutation the parents' chromosomes are altered. Based on crossover and mutation the new individuals are produced. In this paper only the concept of selection and learning technique of Genetic algorithm has been used for task processing.

2. LITERATURE SURVEY

There are many related works developed by using many different strategies. The scheduling criterion uses many conditions few authors focus on multi-objective scheduling, few on performance of scheduling and few are based on nature inspired approaches like honey bee approach, ant colony approach and genetic algorithm approaches.

There are many genetic algorithms based approaches for load balancing policies; in Genetic approach to dynamic load balancing [1] has targeted the load balancing problem in task migration. When task migration request from the users are overloaded into the system, that creates interprocess communication until the task migration is initiated. To overcome this problem GA based Load Balancing has been proposed in which the learning mechanism will learn about the unnecessary request. The request sent to each processor is considered has the population and the fitness is calculated based on the effectiveness of migration.

Aiming to improve the efficiency of load distribution among the public cloud, different strategies have been introduced to different situations. Gaochoa Xu and Junji pang have applied game theory of load balancing in load balancing model based on cloud partitioning for public clouds [2]. In this paper load balancing is initiated right after the cloud partitioning is done. As the job arrives in the cloud, the main controller will forward the job to particular partition based on the load in them, then the jobs are assigned to the nodes based on the strategy.

In Genetic Algorithm based Data Replica Placement Strategy [3], Tripartite Graph based model have been derived based on Genetic algorithm. This Explains how the big data applications like scientific data processing is processed in a distributed cloud environment.

2.1 Tripartite graph model

In this approach task is denoted as T, control flow as C, dataset as D and then the scientific application is denoted as P=<T,C,D>. At each datanode task is processed. In tripartite graph model control flow is the sequence at which the task is executed. In this graph model task scheduling is aimed at minimizing total transmission time of the dataset.

Genetic Algorithm Based Data Replica Placement Strategy

In these two strategies has been considered

1. Data Replica Scheduling Strategy determines which replica can be used and2. Data Replica Distribution Strategy determines where the replica should be stored.

Before task processing the dataset needed by each task has to be transmitted to that datanode. The dataset needed by each task and the location of each dataset should be known for dataset transmission. So that the minimum transmission match can be found and transmitted to the datanode. To calculate the minimum transmission match calculate the list of data replicas results that has small total data transmission time across the data nodes then from the list of datasets pick one with minimum transmission match.

A Multi-objective optimization using Ant colony algorithm by Liyun Zuo creates a scheduling method based on multiple objective based on resource-cost model [26]. A paper by Y. Zhu on research for the Virtual machine oriented cloud resource scheduling algorithm proposes few methods to overcome the resource management issues in data centers that ensures quality service from the cloud service providers[27].An Ant colony based scheduling [28] uses live migration of virtual machines. Ants monitors the virtual machine load, based on the monitoring it finds the optimal migrations of virtual machines. .Combination of genetic algorithm and ant colony optimization method is used [29] to shorten the energy cost and processing time. A standalone software program has been designed to effective resource utilization and load balancing in Agent based dynamic load balancing [30].

2.3 Problem statement

The process of task scheduling considers many conditions which act as an agreement between the users and cloud service providers to promise the quality of service. Users want their task to be processed with minimum time and cost whereas service providers have to ensure the quality of service provided to the users. Users task get segmented into a set of subtask and processing of each subtask requires resources. Based on the available resources and time, cloud providers have to plan for task processing. The issue of load balancing has been considered to ensure the evenly distribution of task among the virtual machines.

2.4 System model

In this paper, the system framework model is shown in Figure-1. Here GABLT is used to evenly distribute the load among the virtual machines. Vm scheduler is the central part of the design will schedule the task to the virtual machines.

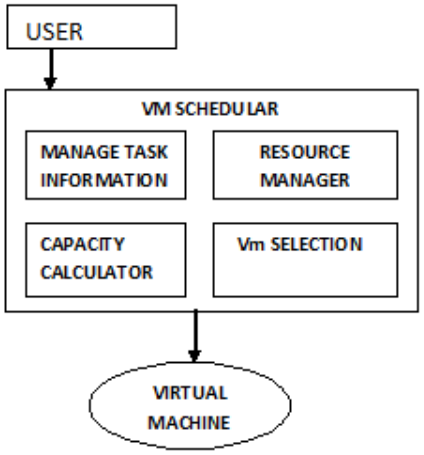

Figure-1. System framework.

Users send their task to the cloud service providers, which will be subdivided into many subtask to get processed. Vm scheduler has four subparts Task information manager, Resource manager, Capacity calculator, Vm selector. Two main parameters required for processing the task are CPU time and the memory usage. Few tasks consume more of CPU time whereas few require huge memory. Based on the two parameters are task nature is learnt by the Vm scheduler.

Users task carry information regarding the QoS and their own preferences for the services. This allows users to decide their budget cost and the target time to complete the task processing. Task send by the users will carry these two things, budget cost and target time to complete task processing.

Vm scheduler will decide to schedule the task to virtual machines based on collective information gathered from the users as well as by nature of task. Task information manager will manage the information of task like budget cost, target time, CPU utilization and memory usage of a particular task.

Resource manager will periodically monitor the virtual resources which are currently being used and idle. Capacity calculator will gather information about the currently available virtual machines. Based on the collective information gathered by Vm scheduler, task will be processed to virtual machines.

3. GENETIC ALGORITHM BASED LOAD BALANCING TECHNIQUE

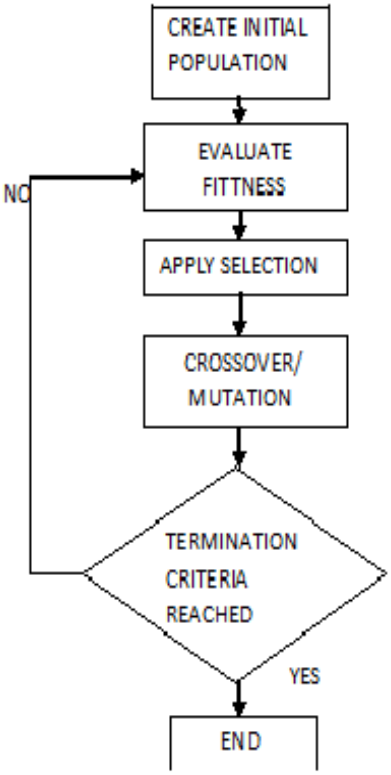

Figure-2. Genetic algorithm flowchart.

In general genetic algorithm has three operations like selection, crossover and mutation. In this paper virtual machines selection is based on genetic algorithm. Selection of virtual machines will be done based on few conditions they are 1.Virtual machines should have enough resources to process the task. Otherwise task has to wait for resources and it wills extent the time to process the task. When it exceeds the user's targeted time then task processing should be stopped. 2. The task nature will predict its actual resource requirements and its total time to get processed. Based on this information the cost of task processing will be predicted and it should not exceed the budget cost of user, if it exceeds task processing should be stopped.

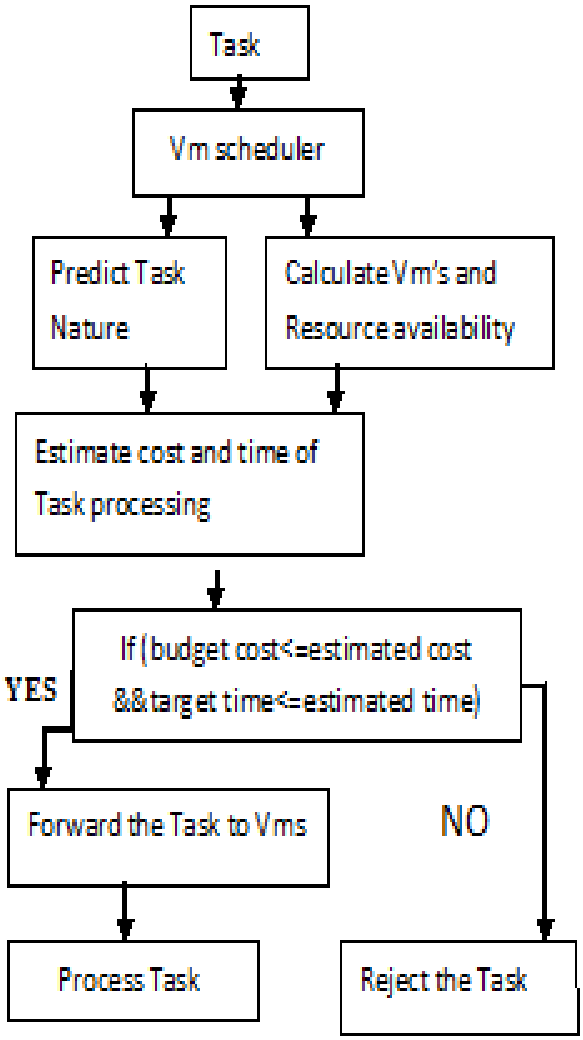

Figure-3. Flowchart of task processing.

4. ALGORITHM: CALCULATE LOAD CAPACITY OF VIRTUAL MACHINE AND RESOURCE AVAILABILITY

1: Function Capacity (DC,H,Vms,C)
2: If H=Available Vms in H_1, H_2……in DC
3: {
4: Divide the task into equal parts based on available Vms in host
5:}
6: End if
7: Vm_c=(Number of available Vms)*
(Number of shifts)*
(Utilization)*(Efficiency)
// calculate capacity of Vms
8: End
9: Function Resources(H,Vms,R)
10: Cal R=List of all available Resources.
11: If(R>=0)
12: Allocate R to Vm_c
13: Else

14: Goto Capacities
15: Cal=Vm,R.
16: End if
17: End

5. NOTATIONS USED

This section describes the symbols used in the algorithm

DC - Datacenter
H - Host
Vms - Virtual Machines
Vm_c - Virtual Machine capacities
R - Resources

5.1 Simulation and Results

There are two kinds of experiments one is real time experiments and the other one is simulation experiments.

Real time experiments though give us a real world conditions and a real world environment, the cost of experimenting the project is not always affordable. Hence many have adapted simulation environment to experiment their projects.

5.2 Benefits of simulation

a) Cost effective

b) Able to run experiments numerous number of times.

c) Easy to model and experiment

d) Easy to compare results with other experiments

This project has been modeled using Cloud Analyst tool. This section explains in detail about the results of the project. Here Genetic Algorithm Based Load Balancing Technique has been compared with Round Robin, Throttled and Equally spread current execution load.

In order to verify that GALBT performs better than existing three algorithms various experiments has been carried out and the result obtained is clearly mentioned with the help of table and graph.

A. Average overall response time

Average response time has been calculated for Round Robin, Equally Spread the current execution load, Throttled and GALBT.

Table-1. Average Overall Response Time.

Techniques	**AORT**(ms)
Round Robin	126.58
Equally Spread current Execution Load	125.97
Throttled	111.98
GALBT	110.55

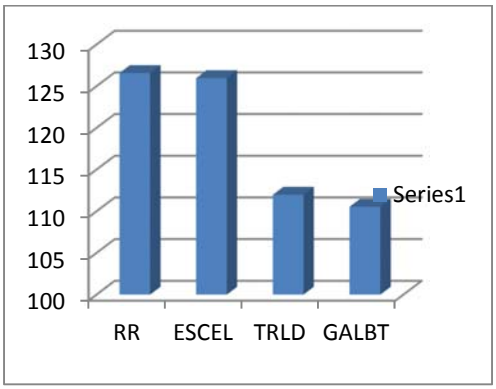

Figure-4. Average overall response time.

Comparing the Average Overall Response of GALBT with all three existing algorithms GABLT's response time is lesser. From Figure-4, it is clear that GABLT performs better.

B. Data center processing time (DCPT)

Data Center Processing Time has been calculated for Round Robin, Equally Spread the current execution load, Throttled and GALBT

Table-2. Datacenter processing time.

Techniques	**DCPT**(ms)
Round Robin	13.77
Equally Spread current Execution Load	13.23
Throttled	1.82
GALBT	0.73

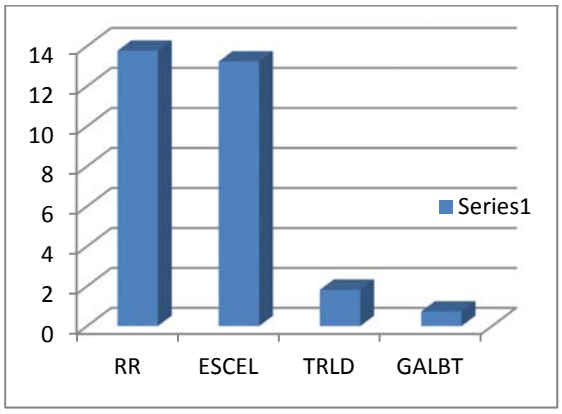

Figure-5. Datacenter processing time.

Comparing the Data Center Processing Time of GALBT with all three existing algorithms GABLT's response time is lesser. From Figure-5, it is clear that GABLT performs better.

C. Total virtual machine cost (TVMC)

Total Virtual Machine Cost has been calculated for Round Robin, Equally Spread the current execution load, Throttled and GALBT.

Table-3. Total virtual machine cost.

Techniques	TVMC
Round Robin	50.41
Equally Spread current Execution Load	50.41
Throttled	50.41
GALBT	49.21

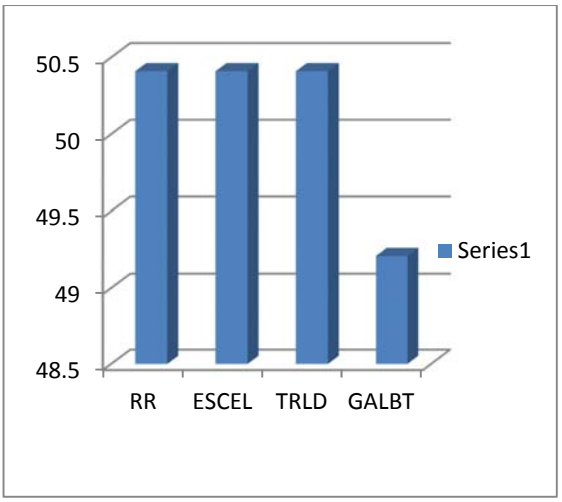

Figure-6. Total virtual machine cost.

Comparing the Total Virtual Machine Cost with all three existing algorithms GABLT's response time is lesser. From Figure-6, it is clear that GABLT performs better.

6. CONCLUSIONS

A Genetic Algorithm based Load Balancing Technique was proposed for load balancing among the virtual machines. This model explains the relationship between the resources and processing cost. The main aim is to balance the load among the virtual machines by using the knowledge about the nature of task, available resources and available virtual machines. In addition to all above, users are allowed to choose their threshold conditions for the services. Our Experimental results show that the proposed GALBT performs far better than the existing algorithms.

ACKNOWLEDGEMENTS

This work was carried out with the aid of DSTgrant under "Young Scientist" scheme sanctioned to the primary author Dr.J. Janet.

REFERENCES

[1] M.munetomo, y. Takai, Y. sato. 1994. Genetic Approach to Dynamic Load Balancing. IEEE world congress on computational intelligence, proceedings of the first IEEE conference on.

[2] Gaochao Xu, Junjie Pang, and Xiaodong Fu. 2013. Load Balancing Model Based on Cloud Partitioning For Public Clouds. Tsinghua science and technology I SSN l 1007-0214 l 04/12 l. 18(12): 34-39.

[3] Lizhen Cui, Junhua Zhang, Lingxi Yue, Yuliang Shi, Hui Li, Dong Yuan. 2015. Genetic Algorithm Based Data Replica Placement Strategy for Scientific Applications in Clouds. in IEEE Transactions on services computing, DOI 10.1109/TSC.

[4] R. Buyya and S. Venugopal. 2004. The Gridbus Toolkit for Service Oriented Grid and Utility Computing: An Overview and Status Report. in IEEE International Workshop on Grid Economics and Business Models. pp. 19-66.

[5] S. Venugopal, R. Buyya, and K. Ramamohanarao. 2006. A Taxonomy of Data Grids for Distributed Data Sharing, Management, and Processing. ACM Computing Surveys. 38: 1-53.

[6] O. Tatebe, Y. Morita, S. Matsuoka, N. Soda, and S. Sekiguchi. 2002. Grid Datafarm Architecture for Petascale Data Intensive Computing. in 2nd IEEE/ACM International Symposium on Cluster Computing and the (CCGrid2002), Berlin, Germany. pp. 102-110.

[7] A. Chervenak, E. Deelman, M. Livny, M.-H. Su, R. Schuler, S. Bharathi, G. Mehta and K. Vahi. 2007. Data Placement for Scientific Applications in Distributed Environments. in 8th Grid Computing Conference. pp. 267-274.

[8] D. Yuan, Y. Yang, X. Liu, and J. Chen. 2010. A Data Placement Strategy in Scientific Cloud Workflows. Future Generation Computer Systems. 26: 1200-1214.

[9] T. Lim and H. Haron. 2013. Performance comparison of genetic algorithm, differential evolution and particle swarm optimization towards benchmark functions. Proc. de IEEE conference on open systems (icos), Kunching.

[10] D. Yuan, Y. Yang, X. Liu, W. Li, L. Cui, M. Xu, J. Chen. 2013. A Highly Practical Approach towards

Achieving Minimum Datasets Storage Cost in the Cloud. IEEE Transactions on Parallel and Distributed Systems. 24(6): 1234-1244.

[11] Z. Chaczko, V. Mahadevan, S. Aslanzadeh andC. Mcdermid. 2011. Availability and load balancing in cloud computing, presented at the 2011 International Conference on Computer and Software Modeling, Singapore.

[12] M. Randles, D. Lamb, and A. Taleb-Bendiab. 2010. Acomparative study into distributed load balancing algorithms for cloud computing, in Proc. IEEE 24th International Conference on Advanced Information Networking and Applications, Perth, Australia. pp. 551-556.

[13] K. Nishant, P. Sharma, V. Krishna, C. Gupta, K. P. Singh, N. Nitin and R. Rastogi. 2012. Load balancing of nodes in cloud using ant colony optimization, In: Proc. 14th International Conference on Computer Modelling and Simulation (UKSim), Cambridgeshire, United Kingdom. pp. 28-30.

[14] B. Adler. 2012. Load balancing in the cloud: Tools, tips and techniques, http://www.rightscale. com/info center/white-papers/Load-Balancing-in-the-Cloud.pdf.

[15] S. Penmatsa and A. T. Chronopoulos. 2011. Game-theoretic static load balancing for distributed systems. Journal of Parallel and Distributed Computing. 71(4): 537-555.

[16] S. Aote and M. U. Kharat. 2009. A game-theoretic model for dynamic load balancing in distributed systems, in Proc. The International Conference on Advances in Computing, Communication and Control (ICAC3 '09), New York, USA. pp. 235-238.

[17] M. D. Kidwell, S. Forrest. 1993. Using genetic algorithms to schedule distributed tasks on a bus-based systems. Proceedings of the Fifth International Conference on Genetic Algorithms. pp. 368-374.

[18] Liang B., Yanli H., Songyang L., Weiming Z. 2010. Task Scheduling with Load Balancing using MultipleAnt Colonies Optimization in Grid Computing. in 2010 Sixth International Conference on Natural Computation (ICNC 2010), DIO: 10.1109/ICNC.2010.5582599, pp. 2715-2719.

[19] R. X. T and X. F. Z. 2010. A Load Balancing Strategy Based on the Combination of Static and Dynamic, in Database Technology and Applications. 2nd International Workshop.

[20] Li K., Xu G., Zhao G., Dong Y. and Wang D. 2011. Cloud task scheduling based on load balancing ant colony optimization. 2011 Sixth Annual ChinaGrid Conference 978-0-7695-4472-4/11 $26.00 © 2011 IEEE. DOI 10.1109/ChinaGrid.2011.17.

[21] Livny M.; Melman M. 2011. Load Balancing in Homogeneous Broadcast Distributed Systems. Proceedings of the ACM Computer Network: Performance Symposium. pp. 47-55.

[22] Brototi M, K. Dasgupta, P. Dutta. 2012. Load Balancing in Cloud Computing using Stochastic Hill Climbing- A Soft Computing Approach. In: proc. 2nd International Conference on Computer, Communication, Control and Information Technology (C3IT)- 2012.

[23] Jaspreet Kaur. 2012. Comparison of load balancing algorithm in a Cloud. International Journal of Engineering Research and Applications (IJERA). 2(3): 1169- 1173.

[24] N. Sran and N. Kaur. 2013. Comparative Analysis of Existing Load Balancing Techniques in Cloud Computing. International Journal of Engineering Science Invention. 2(1).

[25] Dr. Hemant S. Mahalle Prof. Parag R. Kaveri Dr. Vinay Chavan. 2013. Load Balancing On Cloud Data Centres. International Journal of Advanced Research in Computer Science and Software Engineering ISSN: 2277 128X.2013.

[26] Leishu, Shoubin, Dong, Chunsheng Zhu, Takahiro Hara. 2015. A Multi-Objective Optimization Scheduling Method Based on the Ant Colony Algorithm in Cloud Computing. IEEE Access ISSN: 2169-3536.

[27] Y. Zhu; H. Liang. 2013. Research for the virtual machine-oriented cloud resource scheduling algorithm", Information Management, Innovation Management and Industrial Engineering (ICIII), 6th International Conference in the year. Vol. 1.

[28] W. T. Wen; C. D. Wang; D. S. Wu; Y. Y. Xie. 2015. An ACO-based Scheduling Strategy on Load Balancing in Cloud Computing Environment. Frontier of Computer Science and Technology (FCST), Ninth International Conference in the year.

[29] J. Zhao; H. Qiu. 2013. Genetic algorithm and ant colony algorithm based Energy-Efficient Task Scheduling. Information Science and Technology (ICIST), International Conference in the year.

[30] J. Grover; S. Katiyar. 2013. Agent based dynamic load balancing in cloud computing. Human computer Interactions (ICHCI), International conference in the year.