Load Balancing in Cloud Data Center Using Modified Active Monitoring Load Balancer

Ankit Kumar
ME Student
Computer Science and Engineering Department
NITTTR Chandigarh, India
ankitmerwal@gmail.com

Mala Kalra
Assistant Professor
Computer Science and Engineering Department
NITTTR Chandigarh, India
malakalra2004@gmail.com

Abstract: **Cloud Computing is a hot topic of research for the researchers these days. With the rapid growth of Internet technology cloud computing have become main source of computing for small as well big IT companies. In the cloud computing milieu the cloud data centers and the users of the cloud-computing are globally situated, therefore it is a big challenge for cloud data centers to efficiently handle the requests which are coming from millions of users and service them in an efficient manner. Load balancing in this environment means equal distribution of workload across all the nodes. Load balancing provides a way of achieving the proper utilization of resources and better user satisfaction. Hence, use of an appropriate load balancing algorithm is necessary for selecting the virtual machines or servers. This paper focuses on the load balancing algorithm which distributes the incoming jobs among VMs optimally in cloud data centers. The proposed algorithm in this paper has been implemented using CloudAnalyst simulator and the performance of the proposed algorithm is compared with the three algorithms which are preexists on the basis of response time. The experiment carried out in the paper shows that the proposed algorithm performs better than the existing algorithms.**

Keywords: Virtual Machine, Load Balancing, CloudAnalyst, Cloud Data Center, Virtualization, User Base.

I. INTRODUCTION

Cloud computing became the new era of computing these days. Using the cloud computing environment users can pool various IT resources as per their requirements in an efficient manner. With its different services cloud computing is capable of providing an easy way to store and access the stored data and files from data centers which are situated at various geographical locations. The basic services of cloud computing are, Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS), these services provide a pay per use model to it's users. Cloud computing is also having different deployment models. The deployment models are categorized into three; Private Cloud, Public Cloud, Community Cloud and Hybrid Cloud.. Cloud computing possesses five unique characteristics, on-demand self service, elasticity, resource pooling etc.

Virtualization is having an important role in cloud computing since it gives the virtual version of IT resources to the users, these resources includes hardware, operating system and storage device or network resources. Virtualization creates an illusion of using the actual version of any IT resources. One of the big issues in cloud computing environment is, how to allocate the coming request to the appropriate virtual machines with minimum response time so that proper utilization of resources can be done.

The ability to allocate the coming load among the virtual machines in cloud computing environment so that the overall cloud performance can be increased. For this purpose, researchers have proposed several load balancing algorithms such Round Robin, Active Monitoring Load Balancer [1], Throttled load balancer [1] etc. and these algorithms are broadly used for executing the users requests within the minimum response time. Load balancing algorithms are categorized into two types that are static and dynamic algorithms. Static algorithms are used in an environment where dynamic changes are not made at run time e.g. Round Robin (RR) [1]. Dynamic load balancing algorithms are used where the changes are made at run time e.g. weighted least connection (WLC).

In this research work, we have presented Modified Active Monitoring Load Balancer algorithm for efficiently distributing the jobs among the available VMs. This paper also calculates and compares the load balancing algorithms with respect to the response time of VMs assigned jobs.

This paper is organized as follows: Section II focuses on the related work in load balancing, Section III, presents the proposed Modified Active Monitoring Load Balancer, Section IV, Experimental setup is discussed, Section V, presents the Result and Performance Analysis and Section VI, Concludes the work done in the paper and the Future Work.

II. RELATED WORK

Bhathiya et al. [1] proposed and discussed the new cloud simulating tool called CloudAnalyst. The simulator is used for analysis of the large scale applications which are running on cloud. It provides a GUI base simulation to the users so that easy understanding of the simulation environment can be done. They also gave a case study of how the CloudAnalyst can be used to model and evaluate a real world problem.

978-1-5090-0673-1/16/$31.00 ©2016 IEEE

Domanal et al. [3] have given an algorithm for load balancing which can distribute the requests among the VMs uniformly. The proposed algorithm by the authors focuses on two objectives first is response time and second is load distributions among the VMs.

S. Sharma et al. [5] presented various approaches of load balancing on the basis of different parameters. This research work also gives the direction to design a new algorithm on the basis of different parameter by analyzing the behavior of various existing algorithms. At the end of the research work author concluded that it is easy to understand the behavior of static algorithms as compare to dynamic load balancing algorithm.

M. Rahman et al. [8] have given the concept of Load Balancer as a Service (LBaaS) in Cloud Computing as their related work shows that researchers have yet not focused on this problem domain. Therefore initially they have focused on load balancing problem, importance of load balancing, expected characteristics in cloud computing and finally on the Load Balancer as a Service in cloud computing. This service model (LBaaS) was highly adopted by market player.

Bhadani et al. [9] presented modified throttled load balancing algorithm which is used to handle the load in a distributed environment of cloud computing.

Wang et al. [10] presented a combination of two scheduling algorithms, one is Opportunistic Load Balancing and second is Load Balance Min Min that can maintain load balancing of system and also utilize better executing efficiency of the coming jobs.

Xiao et al. [12] suggested a new approach of overloading avoidance and green computing, by presenting a system that can improve the number of servers in green computing by using the virtualization technology. Therefore, the virtualization technology does the dynamic allocation based on the application demands of the data center resources.

Sundaram et al. [15] evaluated two well known dynamic load balancing algorithms namely, shortest queue routing (SQR) and shortest expected delay routing (SEDR). They used a hybrid performance model that combines general stochastic Petri nets and product form queuing networks.

III. PROPOSED MODIFIED ACTIVE MONITORING LOAD BALANCER

The main purpose of the proposed algorithm "Modified Active Monitoring Load Balancer" is to assign the upcoming request to the available VMs wisely. The proposed technique is based on Active Monitoring Load Balancer which assigns the incoming request to the least loaded virtual machine without checking the memory utilization.

The Modified Active Monitoring Load Balancer maintains a table of virtual machines along with state of virtual machines. In the proposed work a new method of distributing the load among the virtual machine has been introduced through which a better load balancing and proper utilization of resources in a cloud computing environment can be achieved.

In this proposed algorithm whenever the request arrives from the user bases at the DCC, then the DCC asks the Modified Active Monitoring Load balancers to assign the upcoming request among the VMs. Modified Active Monitoring Load Balancers contains a table that have the different parameters of VMs i.e. ids, load, state etc. The Modified Active Monitoring Load Balancer scans the table and find out the least loaded VM whose state is available, if there are more than one than it finds the memory utilization of VMs and select the highest priority VM, then it returns VMid to DCC.

Finally the DCC assigns the request to the virtual machine on the basis of VMid which is given by the Modified Active Monitoring Load Balancer algorithm.

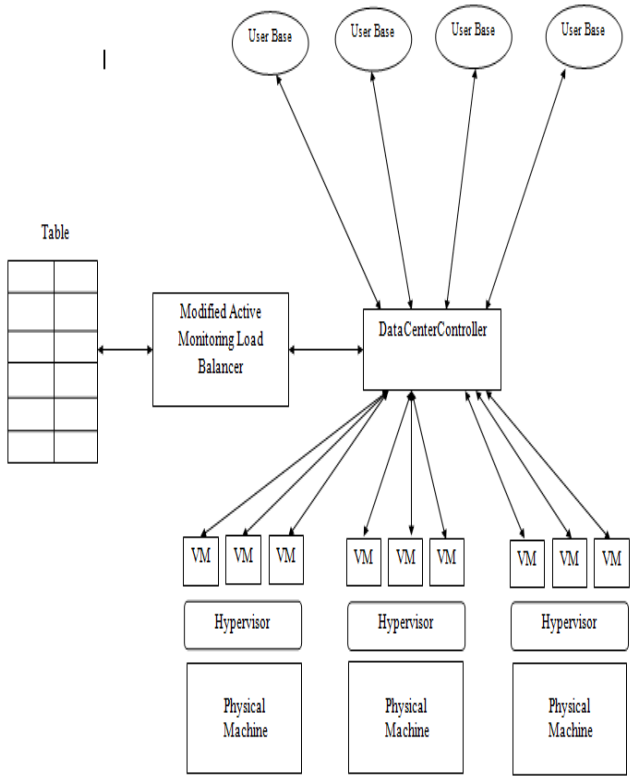

Figure 3.1 Modified Active Monitoring Load Balancer

Figure 3.1 depicted the proposed algorithm. The proposed algorithm is described as below:

1. The "Modified Active Monitoring Load Balancer" contains a table of virtual machine id (VMid), state (BUSY/AVAILABLE) and the number of requests currently allocated to the VMs. Initially all VMs are in available state.

2. The DCC (Data Center Controller) receives a new request.
3. The DCC then queries the Modified Active Monitoring Load Balancer to allocate the upcoming request.
4. Modified Active Monitoring Load Balancer parses the table from the top and finds least loaded VM whose state is available.
5. If there are more than one
   a. The Modified Active Monitoring Load Balancer checks memory utilization of each VM and finds highest priority virtual machine.
   b. The Modified Active Monitoring Load Balancer returns the VMid of highest priority VM to Data Center Controller.
   c. DCC notifies the Modified Active Monitoring Load Balancer of new allocation.
   d. Modified Active Monitoring Load Balancer updates the information of the table accordingly.
6. Return VMid to Data Center Controller.

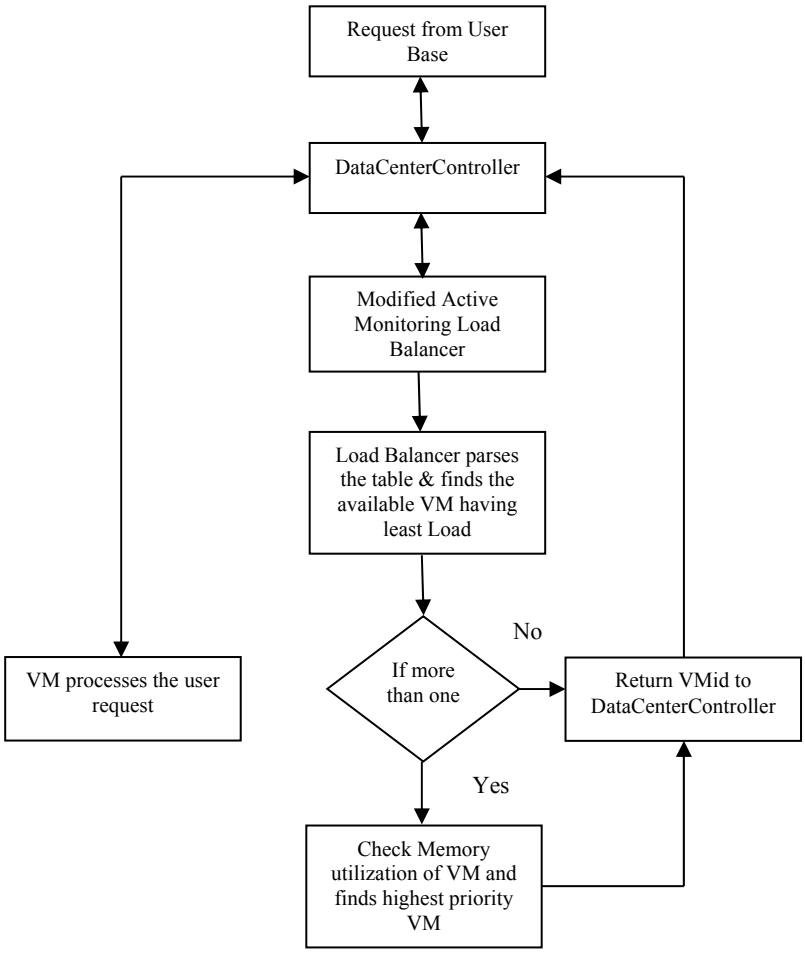

Figure 3.2 Flow Chart of Proposed Algorithm

7. DCC sends the request to the Chosen VM
8. After finishing the request by the VM, the DCC receives the response cloudlet and notifies the Modified Active Monitoring Load Balancer for the de-allocation of the VM.
9. Whenever the request of the VM is processed the allocation table of VM is decremented by one by the Modified Active Monitoring Load Balancer.
10. If there are more requests, go to step 3 else continue from 2.

IV. EXPERIMENTAL SETUP

The algorithm which is proposed in this research work has been implemented and integrated in CloudAnalyst simulator [1]. The CloudAnalyst simulator is a CloudSim [1] based tool for simulating large scaled applications in a cloud environment. CloudAnalyst has extended CloudSim functionality having the features of original framework and also visualization capability.

For our experimental purpose we have taken an assumption of a system which is 1/10 of the scale of Twitter (25/07/2015).

Simulation Duration: 60 min

User Base Table:
For the implementation purpose we defined user bases in 6 regions globally with the following parameters:

The Deployment Configuration of Application
Service Broker Policy: Closest Data Center

The other parameters which are used for implementation are given in the following table:

Table 4.1 6 Region represented by 6 user bases

Name	Region	Request per User per hr	Data Size per request (bytes)	Peak hours start (GMT)	Peak hours Ens (GMT)	Avg Peak Users	Avg off Peak Users
UB1	0	80	200	14	16	500000	50000
UB2	1	80	200	16	18	200000	20000
UB3	2	80	200	18	20	400000	40000
UB4	3	80	200	2	2	180000	18000
UB5	4	80	200	21	23	80000	8000
UB6	5	80	200	8	10	90000	9000

Table 4.2 Parameter Used

Parameters	Value Used

VM Image Size	1000
VM Memory	1024 Mb
VM Bandwidth	1000
Data Center- Architecture	X86
Data Center- OS	Linux
Data Center- Number of Machines	5
Data Center- VMM	Xen
Data Center- Memory per Machine	2048 Mb
Data Center- Storage per Machine	100000000 Mb
Data Center- Processor Speed	10000 MIPS
Data Center- Number of Processor per Machine	4
Data Center- Available BW per Machine	1000000
Data Center- VM Policy	Time Shared
User Grouping Factor	1000
Request Grouping Factor	100
Executable Instruction Length	250

All other parameters like Latency Matrix (in millisecond) and Bandwidth Matrix values (in Mbps) tables are taken same as provided in CloudAnalyst simulator [13].

V. Result and Performance Analysis

In our experiment for evaluating the performance of the proposed algorithm we have considered two test cases. In the first case the load is kept constant and we have increased the number of virtual machines and in the second case the numbers of virtual machines are kept constant and load is increased according to data size per request.

Case 1: Load Constant and Number of VMs varied from 5 to 30

Figure 4.1 shows the achieved results. After performing the experiment the proposed algorithm "Modified Active Monitoring Load Balancer" gives better response time as compared to the Round-Robin, Active Monitoring and Throttled Algorithms.

The graphical representation of comparison of four algorithms is given below:

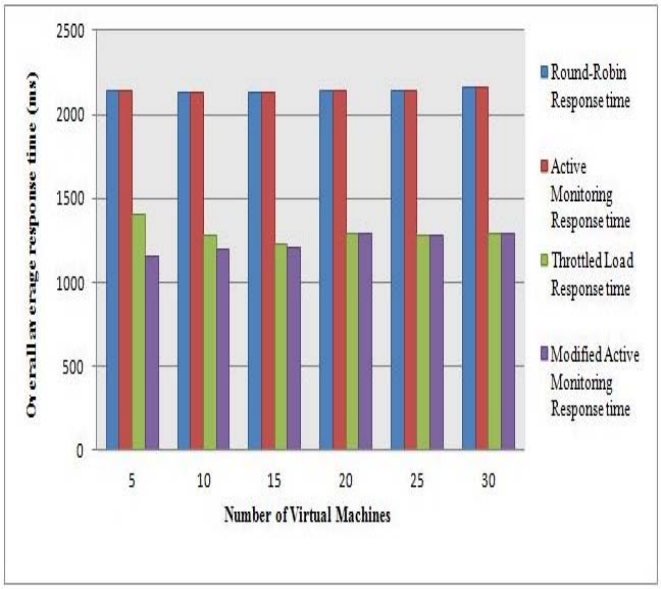

Figure 4.1 Comparison of load balancing algorithms when load is constant

Case 2: Number of VMs is constant and load is increased on the basis of data size per request

While considering the second case that is keeping the number of virtual machines constant and increasing the load of data per size, our proposed algorithm performs better than the rest three algorithms on the basis of response time.

Figure 4.2 shows the achieved results.

The graphical representation of comparison as given below:

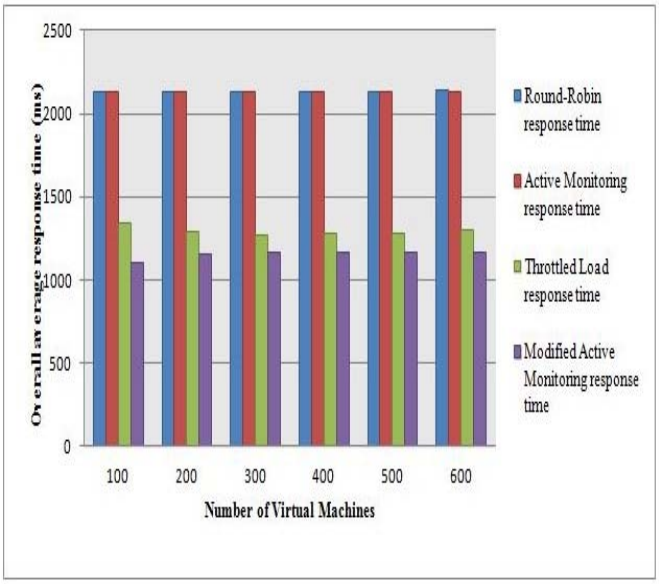

Figure 4.2 Comparison of load balancing algorithms when no. of VMs is constant

VI. Conclusion And Future Work

The cloud data center may suffer from two situations i.e. over loading of virtual machines and under loading of virtual machines, if the proper load distribution among the virtual machines is not done. These two situations occur in a cloud data center due to unequal distribution of load in cloud environment. In proposed algorithm, we tried to avoid these two situations by distributing the load among the virtual machines in an appropriate manner based on their priority, state and memory utilization. With the help of the proposed algorithm the load of the user requests can be distributed among the virtual machines efficiently. However, the proposed algorithm is efficient for user requests load distribution among the virtual machines but it does not consider the virtual machines reliability and energy awareness therefore the future work focuses on load distribution of users request along with VM Reliability and VM energy awareness.

References

[1]. W. Bhathiya, "CloudAnalyst a CloudSim-based tool for modelling and analysis of large scale cloud computing environments", MEDC Project, Cloud Computing and Distributed Systems Laboratory, University of Melbourne, Australia, pp. 1-44, June 2009.

[2]. Mahajan, K., & Dahiya, D., "A Cloud Based Deployment Framework For Load Balancing Policies" IEEE seventh International Conference on Contemporary Computing, pp. 565-570, August 2014.

[3]. Domanal, S. G., & Reddy, G. R. M., "Load Balancing in Cloud Computing using Modified Throttled Algorithm" IEEE international conference on Cloud Computing in Emerging Market, pp. 1-5, October 2013.

[4]. Mondal, B., Dasgupta, K., & Dutta, P., "Load Balancing In Cloud Computing Using Stochastic Hill Climbing-A Soft Computing Approach" Elsevier, Procedia Technology, pp.783-789, 2012.

[5]. Sharma, S., Singh, S., & Sharma, M. "Performance Analysis Of Load Balancing Algorithms" World Academy of Science, Engineering and Technology, 38, pp. 269-272, 2008.

[6]. Zhang, S., Zhang, S., Chen, X., & Huo, X., "Cloud Computing Research And Development Trend" IEEE Second International Conference on Future Networks, pp. 93-97, January 2010.

[7]. Mithra, P. B., & Shameem, P. M., "A Novel Load Balancing Model For Overloaded Cloud Partition" International Journal of Research in Engineering and Technology, ISSN: 2321-7308, vol. 3, Issue 7, pp. 412-417, May 2014.

[8]. Rahman, M., Iqbal, S., & Gao, J., "Load Balancer as a Service in Cloud Computing", IEEE 8th International Symposium on Service Oriented System Engineering, pp. 204-211, April 2014.

[9]. Bhadani, A., & Chaudhary, S., "Performance Evaluation of Web Servers Using Central Load Balancing Policy Over Virtual Machines on Cloud", ACM, Bangalore Conference, January 2010.

[10]. Wang, S. C., Yan, K. Q., Liao, W. P., & Wang, S. S., "Towards a Load Balancing in a Three-level Cloud Computing Network", IEEE 3rd International Conference on Computer Science and Information Technology, vol. 1, pp. 108-113, July 2010.

[11]. Liu, H., Liu, S., Meng, X., Yang, C., & Zhang, Y., "LBVS: A Load Balancing Strategy For Virtual Storage", IEEE Conference on Service Sciences, pp. 257-262, May 2010.

[12]. Xiao, Z., Song, W., & Chen, Q., "Dynamic Resource Allocation Using Virtual Machines For Cloud Computing Environment", IEEE Transaction on Parallel and Distributed Systems, vol. 24, no.6, pp. 1107-1117, 2013.

[13]. Nuaimi, K. A., Mohamed, N., Nuaimi, M. A., & Al-Jaroodi, J., "A Survey Of Load Balancing In Cloud Computing: Challenges And Algorithms" IEEE second symposium on Network Cloud Computing, pp. 137-142, December 2012.

[14]. Domanal, S. G., & Reddy, G. R. M., "Load Balancing in Cloud Computing using Modified Throttled Algorithm", IEEE International Conference on Cloud Computing in Emerging Markets, pp.1-5, October 2013.

[15]. Sundaram, C. R. M., & Narahari, Y., "Analysis Of Dynamic Load Balancing Strategies Using A Combination Of Stochastic Petri Nets And Queueing Networks" Springer Conference, Berlin Heidelberg, pp. 397-414, 1993.

[16]. Hsiao, H. C., Chung, H. Y., Shen, H., & Chao, Y. C., "Load Rebalancing for Distributed File Systems in Clouds" , IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 5, pp. 951-962, 2013.