Optimal Load Balancing in Cloud Computing By Efficient Utilization of Virtual Machines

Shridhar G.Domanal and G. Ram Mohana Reddy
Department of Information Technology
National Institute of Technology Karnataka
Surathkal, Mangalore, India
{shridhar.domanal,profgrmreddy}@gmail.com

Abstract— Load balancing is the major concern in the cloud computing environment. Cloud comprises of many hardware and software resources and managing these will play an important role in executing a client's request. Now a day's clients from different parts of the world are demanding for the various services in a rapid rate. In this present situation the load balancing algorithms built should be very efficient in allocating the request and also ensuring the usage of the resources in an intelligent way so that under-utilization of the resources will not occur in the cloud environment. In the present work, a novel VM-assign load balance algorithm is proposed which allocates the incoming requests to the all available virtual machines in an efficient manner. Further, the performance is analyzed using Cloudsim simulator and compared with existing Active-VM load balance algorithm. Simulation results demonstrate that the proposed algorithm distributes the load on all available virtual machines without under/over utilization.

Keywords - Load balancing, Resource Utilization, Cloud Computing, Virtual machine, CloudAnalyst.

I. INTRODUCTION

Distributed computing leads to a new technology called Cloud computing used by both academia and industry to store and retrieve the files and necessary documents. The main issue is to schedule the incoming requests in an efficient way with minimum response time and at the same time resources should not underutilized. Many algorithms like FCFS, Round Robin, Active-VM monitoring and Throttled are used for executing clients request with a minimum response time and also assigning the requests to the virtual machines [1]. But the constraints such as high communication delays, underutilization of the resources are not addressed clearly and efficiently, which leads to many of the resources does not participate in executing the requests and hence leads to imbalance of cloud system.

Load balancing is very much essential because every virtual machine in the cloud system does the same amount of work throughout, therefore increasing the throughput and minimizing the response time. We can balance the load of a machine by dynamically shifting the workload local to the machine to remote nodes or machines which are less utilized. This maximizes the user satisfaction, minimizing response time, increasing resource utilization, reducing the number of job rejections and raising the performance ratio of the system

Management of the dynamic resources in cloud platform can be efficiently given by virtualization technology. It provides a new way to improve the power efficiency of the datacenters i.e., (server) consolidation, which enables the assignment of multiple virtual machines (VMs) to a single physical server [2]. Consequently, some of the servers can be turned off or put into sleep state, thereby, lowering power consumption of the cloud computing system.

In this paper, we present a novel VM-assign algorithm which allocates incoming jobs to available virtual machines. Here the virtual machine assigned depending on its load i.e. VM with least request is found and then new request is allotted. With this algorithm underutilization of the virtual machine is improved significantly and later it is compared with existing Active-VM algorithm.

The rest of the paper is outlined as follows: The background and related work is discussed in Section II, in Section III proposed algorithm is given, Section IV gives details about experimental setup, Section V gives the results and analysis; finally the conclusion is given in Section VI.

II. BACKGROUND AND RELATED WORK

In this section, we briefly summarize the load balancing algorithms used in the cloud computing environment. The main focus is on the efficient utilization of the virtual machines and balancing the virtual machines with the incoming request. Load balancing is defined as a process of making effective resource utilization by reassigning the total load to the individual nodes of the collective system and thereby minimizing under or over utilization of the available resources or virtual machines.

Hemant S. Mahalle, Parag R. Kaveri and Vinay Chavan [3] have developed Active monitoring load balancer algorithm which maintains information about each VMs and the number of requests currently allocated to which VM. When a request to allocate a new VM arrives, it identifies the least loaded VM. If there are more than one, the first identified is selected. Active VM Load Balancer returns the VM id to the Data Center Controller the data Center Controller sends the request to the VM identified by that id. Data Center Controller notifies the Active VM Load Balancer of the new allocation.

978-1-4799-3635-9/14/$31.00 ©2014 IEEE

Shridhar G. Domanal and G. Ram Mohana Reddy [4] have developed Modified Throttled algorithm which maintains an index table of virtual machines and also the state of VMs similar to the Throttled algorithm[5]. There has been an attempt made to improve the response time and achieve efficient usage of available virtual machines. Proposed algorithm employs a method for selecting a VM for processing client's request where, VM at first index is initially selected depending upon the state of the VM. If the VM is available, it is assigned with the request and id of VM is returned to Data Center, else -1 is returned. When the next request arrives, the VM at index next to already assigned VM is chosen depending on the state of VM and follows the above step, unlikely of the Throttled algorithm, where the index table is parsed from the first index every time the Data Center queries Load Balancer for allocation of VM.

B.Wickremasinghe, R.N.Calheiros and Rajkumar Buyya have developed Throttled algorithm which is completely based on virtual machine. Here the client first requests the load balancer to check the right virtual machine which access that load easily and perform the operations which is given by the client [6]. In this algorithm the client first requests the load balancer to find a suitable Virtual Machine to perform the required operation

In the present work we are considering Active-VM load balancer and proposed VM-assign algorithm for comparison. Our main focus is to distribute the load efficiently on the available virtual machines and ensuring that under or over utilization of the resources/ virtual machines will not occur in the cloud system.

Algorithm: VM-Assign Load Balancer

Input: No of incoming jobs $x_1, x_2 \ldots\ldots x_n$
Available VM $y_1, y_2 \ldots\ldots\ldots y_n$

Output: All incoming jobs $x_1, x_2 \ldots\ldots x_n$ are allocated least loaded virtual machine among the available $y_1, y_2 \ldots\ldots\ldots y_n$

1: Initially all the VM's have 0 allocations.
2: VM-assign load balancer maintains the index / assign table of VMs which has no.of requests currently allocated to each VM.
3: When requests arrive at the data center it passes to the load balancer.
4: Index table is parsed and least loaded VM is selected for execution.
Case I: if found
a. Check whether the chosen least loaded VM is used immediately in the last iteration.
if YES
goto step ***4*** to find next least VM
if NO
Least loaded VM is chosen
5: VM-assign load balancer returns the VM id to the data center.
6: Request is assigned to the VM. Data center notifies the VM-assign load balancer about the allocation.
7: VM-assign load balancer updates the requests hold by each VM.
8: When the VM finishes the processing the request, data center receives the response.
9: data center notifies the VM-assign load balancer for the VM de-allocation and VM-assign load balancer updates the table.
10: Repeat from step 2 for the next request.

III. PROPOSED VM-ASSIGN LOAD BALANCE ALGORITHM

This algorithm focuses mainly on finding out the least loaded virtual machine and how incoming jobs are allocated intelligently. The basic methodology of the proposed VM-assign algorithm is given in the following Fig. 1.

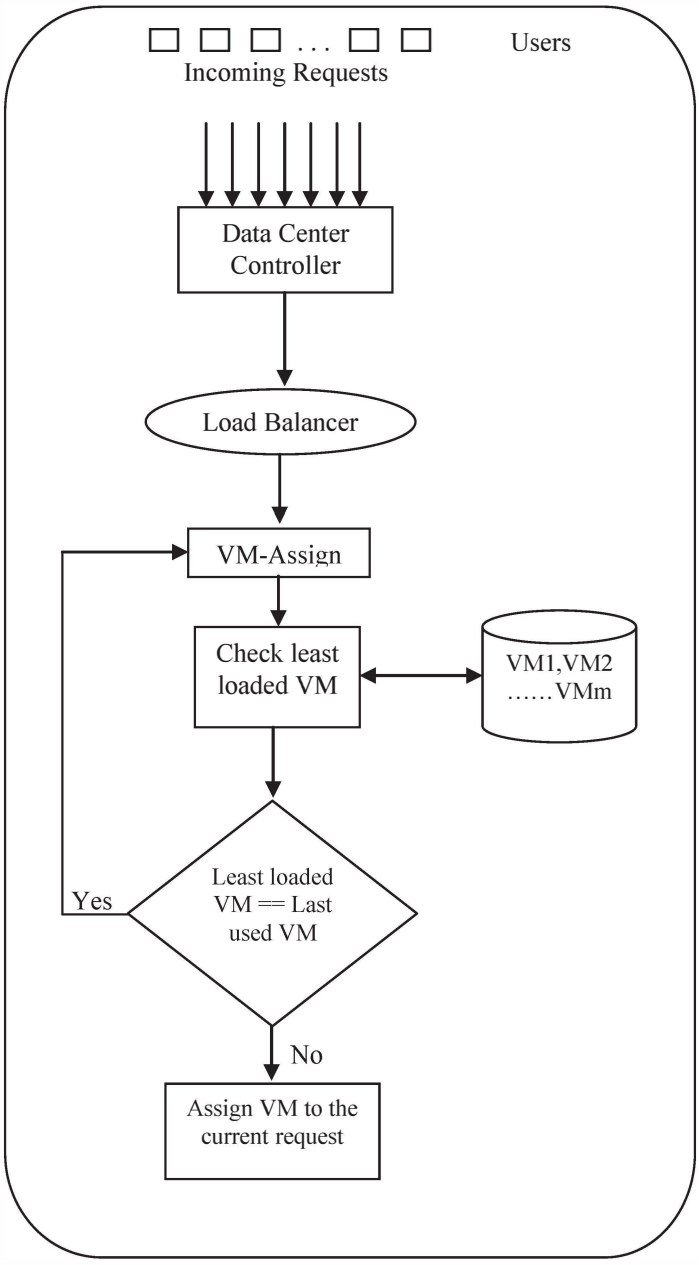

Fig.1: Flow of the Proposed VM-assign Algorithm

The Fig.1. Shows the complete flow of the proposed algorithm. There are "n" users are present who request for services in the cloud. The data center which receives the incoming requests and pass to the load balancer. There are "y" virtual machines for processing the requests.

VM-assign load balancer algorithm maintains an index/ assign table of virtual machines and also the load of VMs. There has been attempt made to efficient usage of available virtual machines depending on its load. Proposed algorithm employs a method for selecting a VM for processing client's request. It checks for least loaded VM. Initially all VM are free so it follows Round Robin. Then if next request comes then it checks for VM table, if the VM is available and it is not used in the previous assignment then, it is assigned with the request and id of VM is returned to Data Center, else we find the next least loaded VM and it continues and follows the above step, unlikely of the Active load balancer, where the least loaded VM is chosen but it will not check for the previous assignments.

IV. EXPERIMENTAL SETUP

Experiment is carried out in the simulator. For the experimentation Cloudsim based CloudAnalyst simulator has been used [6]. As cloud infrastructure is distributed in nature requests will be coming from all geographical locations and should be handled intelligently.

CloudAnalyst simulator gives the real time scenario with six different geographical locations. i.e depending on the specific application no of users from particular locations can be identified, e.g Twitter users from Asia, Africa etc. The simulator is very flexible and it provides, virtual machines, data centers, band width and many more for experimentation [7]. A snapshot of the CloudAnalyst architecture is shown in Fig. 2 [5].

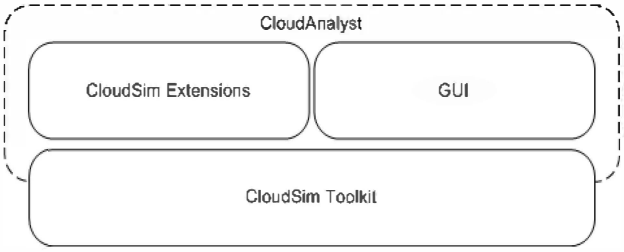

Fig.2: Architecture of CloudAnalyst simulator

Hypothetical applications like Facebook users, Twitter users, Internet users are considered for experimentation. Six different geographical locations (six different continents of the world) are considered [3]. A single time zone is considered for all user locations. For simplicity one hundredth of the total users from each continent is considered and it is assumed that only 5% of total users are online during peak hours and in off peak hours, users are one tenth of the peak hours.

For experimentation internet users at six different continents are considered i.e. six user bases and peak and non peak users are given in the Table 1. We have considered internet users at different continents from the month of June 2012. The same data is experimented with three different scheduling algorithms and response time of each algorithm is also considered for the result analyses.

TABLE I. Simulation Configuration

User Base	Region	Simultaneous online Users during peak Hours	Simultaneous online users during off peak hours
North America	0	135000	13500
South America	1	125000	12500
Europe	2	255000	25500
Asia	3	535000	53500
Africa	4	30000	3000
Oceania	5	10000	1000

Each Data Center has a capacity to host a no of virtual machines which are needed for particular application. Machines have 100 GB of storage space, 4 GB of RAM, each machine has 4 CPU and a power of 10k MIPS.

V. RESULTS AND ANALYSIS

Results are analyzed w.r.t efficient utilization of the virtual machines by avoiding the under or over loading conditions. And then comparing with the existing active VM load balancer.

A. Load balancing of VMs

As mentioned in section III the proposed algorithm will not allow the VM which was allocated in its previous step. But this is not the same case with Active load balancing algorithm in which it assigns the least loaded VM depending on the current load. So with this in Active load balancer algorithm few VMs are overloaded with many requests and remaining VMs will handle only few requests with this under utilization of the VMs takes place. This results in imbalance of the load on the VMs. But if we use the proposed VM-assign load balance algorithm all the VMs are utilized completely and properly. Our algorithm proves that there is no under utilization of the resources in the cloud. The algorithm is tested for initially with five VM and later 25 VM. In both cases our proposed algorithm balances the load on all available VMs in an efficient way. Hence we can say that our algorithm will overcome the under/over utilization of resources usage problem. The following Table 2 gives the information about how many times each VM has been used efficiently.

TABLE 2. VMs Usage with 5 VMs

SI. No	Active Load Balancer	VM-assign Load Balancer
VM0	1178	258

VM1	78	252
VM2	10	253
VM3	4	251
VM4	2	254

From the Table 2 we can observe that using VM-assign load balancer distributes the incoming requests to all VM's in an intelligent way compared to Active load balancer. In the proposed algorithm we can see, utilization of the resources is neither underutilized nor over utilized but where as in Active load balancer VM 0 is over utilized and VM 4 is never utilized and other VM's are underutilized.

We have repeated the experiment with the 25, 50 and 100 virtual machines. The following Table 3 gives the information about the usage of the 25 virtual machines.

TABLE 3. VMs usage with 25 VMs

SI. No	**Active Load Balancer**	**VM-assign Load Balancer**
VM0	18609	3693
VM1	20078	4376
VM2	6569	4330
VM3	5845	4312
VM4	4973	4350
VM5	4568	4318
VM6	4164	3818
VM7	4907	4588
VM8	4337	4565
VM9	3886	4610
VM10	3267	4615
VM11	3321	4582
VM12	2966	4065
VM13	3869	4890
VM14	2988	4863
VM15	2948	4828
VM16	2332	4852
VM17	2425	4971
VM18	2047	5053
VM19	2467	4440
VM20	2249	5098
VM21	2564	5278
VM22	1883	5309
VM23	1984	5472
VM24	1648	5618

From Table 2 and 3 we can analyze that Active-VM load balancer algorithm is over utilizing the initial VM's and under utilizing the later ones. But our proposed algorithm distributes the incoming requests to all VM's intelligent way and hence all the resources are used efficiently. We have also verified response time of both algorithms and the values are same in both algorithms. Further, the results for 50 and 100 virtual machines follow the same pattern of allocation as shown in Tables 2 and 3.

VI. CONCLUSION

In this paper, an efficient algorithm is designed which manages the load at the server by considering the current status of the all available VMs for assigning the incoming requests intelligently. The VM-assign load balancer mainly focuses on the efficient utilization of the resources/VMs. We proved that our proposed algorithm optimally distributes the load and hence under / over utilization (VMs) situations will not arise. When compared to existing Active-VM load balance algorithm, the load was not properly distributed on the VMs and from Table 3 we can say that VM 0 is used almost ten times more than the VM 24. It proves that initial VMs are over utilized and later VMs are underutilized. Our proposed algorithm solves the problem of inefficient utilization of the VMs / resources compared to existing algorithm.

As a future scope the proposed algorithm can still be improved by taking some more dynamic situations of the incoming requests and how the algorithm responses if we mix both static and dynamic loads.

REFERENCES

[1] Klaithem Al Nuaimi, Nader Mohamed, Mariam Al Nuaimi and Jameela Al-Jaroodi, "A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms", in Second Symposium on Network Cloud Computing and Applications, 978-0-7695-4943-9/12, IEEE 2012.

[2] Zhen Xiao, Weijia Song and Qi Chen, "Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment" IEEE transactions on parallel and distributed systems, vol. 24, no. 6, june 2013.

[3]Hemant S. Mahalle, Parag R. Kaveri and Vinay Chavan, "Load Balancing On Cloud Data Centres" in International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, issue 1, January 2013.

[4] Shridhar G. Domanal and G. Ram Mohana Reddy," Load Balancing in Cloud Computing Using Modified Throttled Algorithm" IEEE, International conference. CCEM 2013. In press.

[5] Brototi Mondal, Kousik Dasgupta and Paramartha Dutta, "Load balancing in cloud computing using stochastic hill climbing-a soft computing approach" in Procedia Technology 4 (2012) 783 – 789, ELSEVIER C3IT-2012.

[6] B.Wickremasinghe, R.N.Calheiros,R.Buyya, Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing in: Proceedings of the 24th International Conference on Advanced Information Networking and Applications (AINA 2010), Perth, Australia,, 2010.

[7] Mr.Manan D. Shah, "Allocation Of Virtual Machines In Cloud Computing Using Load Balancing Algorithm" in International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555 Vol. 3, No.1, February 2013.

[8] Ms.Nitika, Ms.Shaveta, Mr. Gaurav Raj, "Comparative Analysis of Load Balancing Algorithms in Cloud Computing" in International Journal of Advanced Research in Computer Engineering & Technology Volume 1, Issue 3, May 2012.

[9] M. Armbrust et al., "Above the Clouds: A Berkeley View of CloudComputing," technical report, Univ. of California, Berkeley, Feb. 2009.

[10] Hung-Chang Hsiao, Hsueh-Yi Chung, Haiying Shen, and Yu-Chang Chao, "Load Rebalancing for Distributed File Systems in Clouds" IEEE transactions on parallel and distributed systems, vol. 24, no. 5, may 2013.