Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 6, June 2015, pg.862 – 870

RESEARCH ARTICLE

A NOVEL LOAD BALANCING STRATEGY FOR EFFECTIVE UTILIZATION OF VIRTUAL MACHINES IN CLOUD

Asha C[1], Ms. Suriya Refai Begum[2], Dr. Prashanth C.S.R[3]

Student, M.Tech (Computer Science & Engineering), NHCE, Bangalore, India[1]

Sr. Assistant Professor, Computer Science & Engineering Department, NHCE, Bangalore, India[2]

Head of Department, Computer Science & Engineering Department, NHCE, Bangalore, India[3]

ashapreethy@gmail.com, suriyabegumvtu@gmail.com

Abstract - Cloud computing is one of the emerging high utility software penetrating in to our lives and dramatically changing the way people accesses the information. It helps to provide hardware, software as well as data in the form of collaborative services on demand from the clients. Beside the advantages of cloud computing Load balancing has become the major technical challenge that needs to be tackled. In order to resolve this issue load balancing algorithms designed should be very efficient in allocating the request and should avoid overloading or under loading of any virtual machine. In this paper we suggest a novel load balancing strategy that helps in the even distribution of load among all the virtual machines. Cloud Analyst is the simulation tool used for implementing proposed algorithm. Experimental results show that the proposed algorithm outperforms existing methods namely Active-VM load balancing algorithm and VM-Assign Load Balancer Algorithm. Simulation results prove that the proposed algorithm carves a bench mark in fare distribution of load among all the available virtual machines.

Keywords – "Load Balancing", "Cloud Analyst"," Virtual Machine", "Cloud Computing", "Datacentre Controller"

I. INTRODUCTION

Cloud computing is emerging as a wonderful computing paradigm due to its ability to provide computation as a service to end users. The services offered by Cloud are Infrastructure as a service, Platform as a service, Software as a service that are made available to the clients as a pay-as-u-go model. Cloud computing is coined as next generation architecture of IT enterprises due to its flexible and intuitive architecture and the services that are offered on demand from any part of the world. It has drawn lot of attention in academic as well as commercial spheres. Despite of the growing popularity of cloud computing, there are many crucial

problems that need to be tackled for the realization of this technology. Among all of them load balancing is of primary concern. So the main issue is scheduling of the incoming requests in an efficient manner among the virtual machines thus avoiding the under or over utilization of the virtual machines keeping the response time minimum. Many of traditional load balancing algorithms like Throttled, FCFS, Round Robin are used for execution of incoming client requests to virtual machines by maintaining minimum response time[1]. However various constraints like security, communication delays and failure to address underutilized resources resulted in unparticipating of the resources in execution of the client requests leading to imbalance in cloud platform. Therefore it is very important to focus on resource management. Dynamic resource management in cloud is given virtualization technology. It has paved a new way for improving power efficiency of datacentres by allowing the assignment of the multiple virtual machines to a single datacentre i.e. server[2].

In this paper we present a load balancing algorithm which evenly distributes the incoming request among all the available virtual machines. Here virtual machine is assigned with the new incoming request depending on its current load i.e Virtual machines that has least request is found and the new request is allocated depending on it. With this algorithm over utilization or underutilization of the virtual machines is overcome and later the algorithm is compared with existing active load balancing algorithm[3].

The rest of the paper is outlined as follows: The background and related work is discussed in section II and section III gives proposed load balancing algorithm and illustration. Section IV has experimental set up, Section V has simulation framework and simulation results and finally last section has conclusion.

II. BACKGROUND and RELATEDWORK

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 assigning the load among all the virtual machines 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[5] have developed Active monitoring load balancer algorithm Active monitoring load balancer algorithm which maintains the information about each and every VMs and the number of requests that are currently allocated to each VM along with the VM ID is that VM. When a request to allocate a new VM arrives, it identifies the least loaded VM among all the VMs and the least loaded VM is chosen for the request allocation. If there are more than one least loaded VM, then the first identified VM will be selected. Active VM Load Balancer returns the VM id to the Datacentre Controller. The datacentre Controller sends the request to the VM identified by that id. Datacentre Controller notifies the Active VM Load Balancer of the new allocation. Once the VM finishes the processing of the request then the datacentre receives the response. Later datacentre notifies the Active Monitoring load balancer for the VM de-allocation.

Shridhar G. Domanal and G. Ram Mohana Reddy [6] have developed VM-Assign load balancer. VM-Assign load balancer algorithm maintains the information about each VM and the number of requests that being currently allocated to VM. When a request for the allocation of a new VM arrives, then it first identifies VM which is least loaded and checks whether the chosen least loaded VM is being used in last iterations for processing of the requests or not. If the VM is not utilized in last iteration then that particular VM is chosen for the request allocation. The id of that VM is returned to the datacentre Controller by VM-Assign load balancer. Later on the datacentre Controller sends the request to the VM which is identified by that id. Datacentre Controller notifies the VM-Assign load Balancer of the new allocation. Once the VM finishes the processing of the request, then the datacentre receives the response. Later it is the work of the datacentre to notify the VM-assign load balancer for the VM de-allocation.

Proposed system named as Enhanced VM State Load Balancing Algorithm employs a method for selecting the VM for processing the request sent by the client. Here the Load Balancer maintains an index table for each VM on the server. The index table contains information about each and every VM. It contains the information of number of requests that are active on each VM along with the VM Id of the VM. In the proposed technique, every request from user bases arrives at Datacentre Controller. Datacentre Controller queries the Enhanced VM State Load Balancer for allocation of requests. Enhanced VM State Load Balancer maintains an index table that consists of id and the states of virtual machines.

In the first phase the Load balancer parses the index table to find out the state of the virtual machine. It checks whether the virtual machine is Available or Busy. If the virtual machine state is marked as Available in the index table, then it moves on to next phase. In the second phase it checks for the least loaded VM among all the virtual machines marked as Available. In the third phase it checks whether the chosen least loaded VM is used in the last iterations for processing of the request or not. If the VM is has not been used in last iteration then that VM is chosen for the request allocation.

Finally Enhanced VM State Load Balancer returns the VM id of that VM to the datacentre Controller, later on the datacentre Controller sends the request to the VM that is identified by that id. Datacentre Controller reports the Enhanced VM State load Balancer of the new allocation. Once the VM completes the processing of the request, datacentre receives the response. Later datacentre alerts the Enhanced State VM load balancer for the VM de-allocation.

Algorithm: Enhanced VM State Load Balancer

Input: No of incoming jobs X1, X2 . . . . Xn, Available VM y1, y2⋯ . . . . . yn
Output: All incoming jobs X1, X2 . . . . Xn are allocated least loaded virtual machine among the available y1, y2⋯⋯⋯ yn
1: Initially all the VM's have 0 allocations.
2: Enhanced 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. Index table is parsed.
4: Check whether the VM is busy or available.
If YES: Go to step 3 to find next VM
If NO: Go to Step 5
5. Check whether the VM is allocated in the last iterations.
Case I: if allocated in last iterations
If YES: Go to step 3 to find next VM
If NO: Go to Step 6
6: Table is parsed and least loaded VM is selected for execution. If found Least loaded VM is chosen for request processing.
7: Enhanced VM State load balancer returns the VM id to the data center.
8: Request is assigned to the VM. Data center notifies the Enhanced VM State load balancer about the allocation.
9: Enhanced VM State load balancer updates the requests hold by each VM.
10: When the VM finishes the processing the request, data center receives the response.
11: data center notifies the Enhanced VM State load balancer for the VM de-allocation and Enhanced VM State load balancer updates the table.
12: Repeat from step 2 for the next request.

III. PROPOSED ALGORITHM

ENHANCED VM STATE LOAD BALANCER ALGORITHM

This algorithm mainly focuses on the allocation of the incoming requests to the virtual machines in an intelligent way so that over utilization or underutilization situations do not arise. The basic methodology of the proposed algorithm is depicted in the following Figure 1. Figure 1 Show the clear cut idea of the flow of the proposed algorithm. There are 'n' users existing in the user base request for the services in the cloud. Datacentre of the cloud receives the requests and pass the requests on to the load balancer. There exist 'y' virtual machines for processing up of the requests. The proposed Algorithm maintains an index table which has all the information about the virtual machines. Here a small attempt is made for the effective utilization of the virtual machines with this algorithm

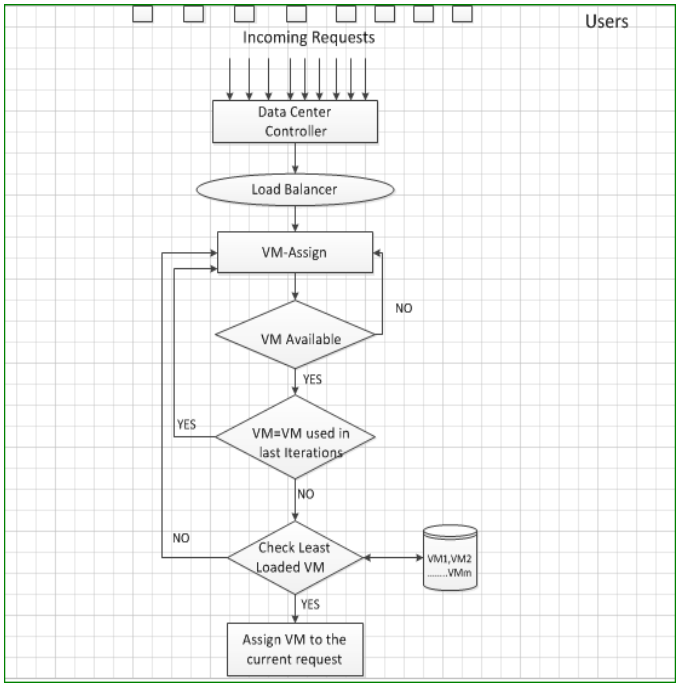

Figure 1: Flow of the Enhanced VM State Load Balancer Algorithm

This algorithm supports the better utilization of the virtual machines by even distribution of the incoming request as opposed to the exiting load balancer techniques.

IV. EXPERIMENTAL SETUP

The experiment is carried out in a simulator. For conducting the experiment cloud sim based cloud analyst simulator is being used. This simulation tool has been developed at the Melbourne University[9]. Cloud analyst simulator gives an experience of real time scenario where the experiment is carried out by considering the six geographical locations. It is just an abstraction of the real world internet, implementing only those features that are important for the simulation. Based on particular application the users from specific region can be easily identified. For example face book users from Asia and Africa etc. Communities of users as well as datacentres are that are supporting the social networks are identified mainly by their location. The simulator is flexible as it provides various entities like user bases and datacentres that belong to one of these regions. The Cloud Analyst is being built on top of the Cloudsim tool kit, by extending all of the Cloud Sim functionalities along with the introduction of some special concepts that model Internet and various other Internet Application behaviors. The snapshot of cloud analyst architecture is given below in figure 2.

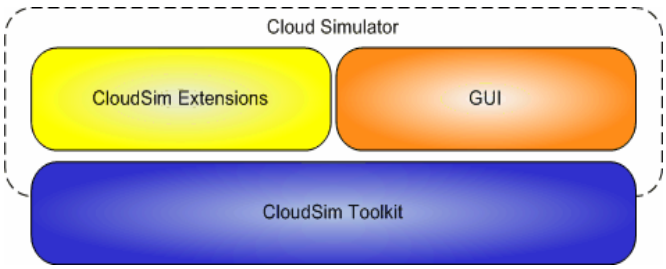

Figure 2: Architecture of Cloud Analyst

Various hypothetical applications like Twitter, Face book and other social networking applications are used for conducting the experiment. Six different geographical locations are considered and single time zone is taken in to consideration for all user locations. From each continent one hundredth of the total users are considered and it is assumed that only 5% of the total number of users are online during peak hours and during the off peak hours users get reduce to only one tenth of the off peak hours. Cloudsim facilitates modelling, simulation and various other experimentation programmatically on cloud. The cloud analyst is equipped with the functionalities of the Cloudsim and performs GUI based simulation.
A snapshot of the cloud analyst GUI simulation tool kit is shown in the Figure 3.

Figure 3: Main screen of cloud analyst GUI

Six user bases as well as the peak and non-peak users are given in the Table 1. We have considered different users from the month of June 2014. The same data is investigated with three different scheduling algorithms and the response time of each algorithm is used for result analysis. Each datacentre has the capability to host the virtual machines that are needed to execute the application. Each virtual machine has the storage capacity of 100GB, 4GB RAM and each and every machine has 4 CPU's and 10k MIPS power.

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

V. RESULTS AND ANALYSIS

Results of the experiment that are conducted on the cloud analyst simulator are analyzed w.r.t to the effective utilization of the virtual machines by avoiding over or underutilization conditions that may arise. The proposed Enhanced VM Load Balancer manages to distribute the incoming requests to all the virtual machines based on the state, no. of active requests on the VM and by checking whether the VM was used in the previous iteration or not. But in case of the existing load balancing algorithms namely Active Monitoring and VM-Assign load balancer, the state of the VM will not be checked before assigning the request to the VM. This mainly results in the imbalance of the load among the virtual machines. So instead of employing the existing algorithms if we use proposed algorithm then the VM is utilized completely and properly. The algorithm is initially tested with five virtual machines and then with 25 virtual machines. In both the cases our proposed algorithm outperforms the existing algorithms by balancing the load on the available VM's. The following Table 2 gives clear idea on how many times each VM has been used effectively.

TABLE II

VMS USAGE WITH 5 VMS

VM ID	**Active VM Load Balancer Algorithm**	**VM-Assign Load Balancer Algorithm**	**Enhanced VM State Load Balancer Algorithm**
VM0	1174	258	254
VM1	78	252	254
VM2	10	253	254
VM3	4	251	254
VM4	2	254	252

From Table 2 we can confirm that the proposed model works well in ensuring even distribution of work load. In the done work, we can infer that the Enhanced VM State load balancer distributes the incoming requests to all VM's in an intelligent way compared to the existing algorithms. Here we conducted the experiment with 5 VM's. In case of the Active Monitoring Load Balancer the VM0 is over utilized but the VM4 is never utilized for processing the request and also other VM's are underutilized.

When we consider the VM-Assign Load Balancer we encounter over or underutilization conditions of the VM's. So we prefer Enhanced VM Load Balancer algorithm for better performance.

We have repeated the same experiment by considering 25, 50 and 100 VM's. The Table 3 mentioned below indicates the usage of 25 VM's.

TABLE III

VMS USAGE WITH 25 VMS

VM ID	**Active VM Load Balancer Algorithm**	**VM-Assign Load Balancer Algorithm**	**Enhanced VM State Load Balancer Algorithm**
VM0	1107	75	53
VM1	126	51	53
VM2	9	51	53
VM3	4	51	53
VM4	2	51	53
VM5	2	51	52

VM6	2	51	52
VM7	2	51	52
VM8	2	51	51
VM9	1	50	51
VM10	3	52	51
VM11	2	51	51
VM12	1	50	51
VM13	1	50	51
VM14	4	52	51
VM15	2	50	51
VM16	3	51	51
VM17	1	49	51
VM18	2	50	51
VM19	2	50	51
VM20	2	50	51
VM21	2	50	51
VM22	2	50	51
VM23	2	50	51
VM24	2	50	51

From TABLE II and TABLE III we can conclude that existing load balancer algorithms are rising up the situation of over utilization or underutilization of the virtual machines whereas the proposed load balancing algorithm results in better resource utilization. Further the results for 50 and 100 virtual machines exhibit the same result pattern.

The graphs mentioned below shows the results obtained for the experiment conducted with 5 VM's.

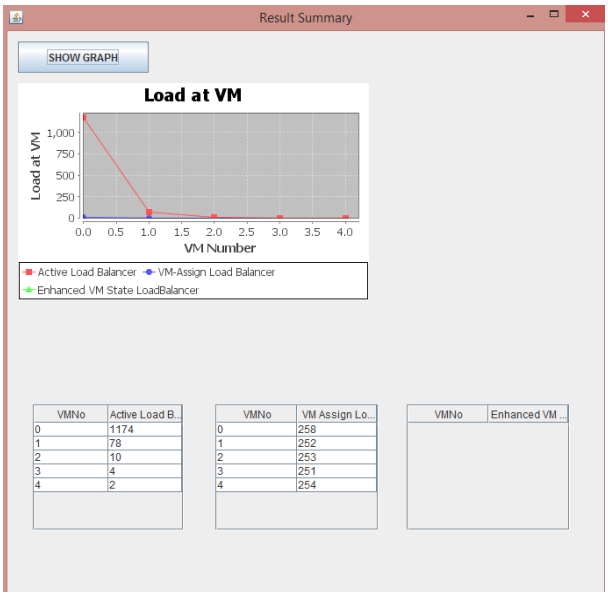

Figure 4: Comparison graph showing the utilization of VM in Active Monitoring and VM-Assign Load Balancer

Here the Virtual machines are over utilized in some cases and in some other cases the virtual machines are underutilized degrading the overall performance of the cloud system. To overcome this problem new algorithm named Enhanced VM state Load Balancer is proposed. The following graph shows the comparison among VM assign load balancer and proposed algorithm.

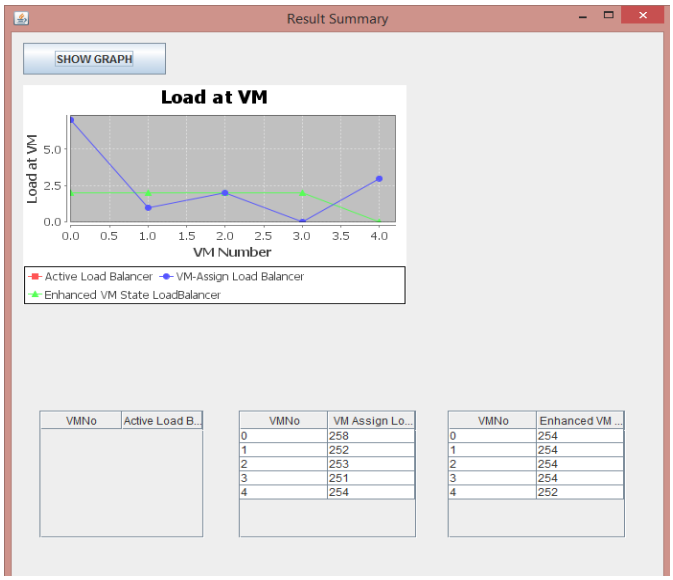

Figure 5: Comparison graph showing the utilization of VM in VM-Assign Load Balancer and Enhanced VM State Load Balancer

VI. CONCLUSION and FUTURE WORK

In this paper we have introduced an efficient Load Balancing algorithm mainly designed to manage the load on the VM of the data center by considering the current status of all the virtual machines and the state of the virtual machines. It has been established that the model works well in ensuring an even distribution of the workload. In the done work, it has been assumed that all of the incoming requests are considered to be independent to each other.

As a future scope the algorithm can be improvised by considering some more dynamic situations of the incoming requests and check how the algorithm performs if we combine both dynamic and static loads.

REFERENCES

[1] Suriya Begum, Dr. Prashanth C.S.R, "*Review of Load Balancing in Cloud Computing"*, in IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No 2, January 2013.

[2] Suriya Begum, Dr. Prashanth C.S.R "*Investigational Study of 7 Effective Schemes of Load Balancing in Cloud Computing"*, in IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 6, No 1, November 2013.

[3] Klaithem Al Nuaimi, Nader Mohamed, Mariam Al Nuaimi and Jameela AI-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.

[4] 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.

[5] 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.

[6] Shridhar G. Domanal and G. Ram Mohana Reddy,*" Optimal Load Balancing in Cloud Computing by Efficient Utilization of Virtual Machines"* in IEEE, International conference. 2014.

[7] Suriya Begum, Dr. Prashanth C.S.R *"Mathematical Modelling of Joint Routing and Scheduling for an Effective Load Balancing in Cloud"* in International Journal of Computer Applications (0975 – 8887) Volume 104 – No.4, October 2014.

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

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

[10] 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.

[11] 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 (lJCSlTS), ISSN: 2249-9555 Vol. 3, No.1, February 2013.

[12] 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.

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

[14] 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.

