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. 5, May 2015, pg.233 – 237

RESEARCH ARTICLE

Distributed and Dynamic Load Balancing in Cloud Data Center

Sweekriti Shetty[1], Sudheer Shetty[2]

[1]Department of Computer Science & Engineering, Sahyadri College of Engineering & Management, Mangaluru, Karnataka, India

[2]Associate Professor, Department of Computer Science & Engineering, Sahyadri College of Engineering & Management, Mangaluru, Karnataka, India

[1] shettysweekrithi86@gmail.com , [2] sudheer.cs@sahyadri.edu.in

***Abstract**— Cloud computing is one of the booming technologies in the present day. Number of users availing cloud services are increasing day by day, which in turn requires better performance of the cloud. Cloud Data Center is one of the major components of cloud computing. One of the major challenges of cloud data center is handling and servicing millions of requests from the users. Distributing the load fairly across all the available machines is called load balancing and the main aim of load balancing is to help in achieving high resource utilization and user satisfaction. Therefor, the load balancing algorithms used should be very efficient and should reach the goals o f the user. In this paper an algorithm is proposed which balances the load based on the priority of the virtual machines and state of the virtual machines used in the cloud.*

***Keywords**— Cloud Computing, Cloud Data Center, Virtual machines, Cloudsim, Cloud Analyst*

I. INTRODUCTION

Computing is being transformed into a model where users access services based on their requirements regardless of where they are hosted. Several computing paradigms have promised to deliver the above mentioned services and cloud computing is one among them [9]. Cloud computing is a technology that focuses on how we design computing systems, develop applications and utilize existing services for building software. This technology works on a pay-per-use basis i.e. you can pay only for the resources that you require and these resources are made available through the internet.

Several components like servers, data centers, virtual machines, etc., which are interconnected to form a cloud computing system. Virtualization is one of the concepts on which cloud computing is based on. Virtualization refers to the creation of a virtual version of something, whether it is hardware, software, storage or a network. It helps to reduce the overall cost that includes space, power etc.

Cloud Computing have various benefits and also a few drawbacks. One of the major challenges of cloud computing is Load Balancing. Handling and servicing millions of requests from the users arriving at the data center controller and distributing the load fairly across all the available machines are called load balancing [10]. The main aim of load balancing is to help in achieving high resource utilization and user satisfaction. There are two types of load balancing policies – static and dynamic [8]. In static algorithms, load is assigned to different machines by considering their processing capabilities. Few static algorithms used for balancing the load are Round Robin algorithm, Weighted Round Robin algorithm. In dynamic algorithms, load is assigned to different machines by considering the changes in every machine runtime conditions. Few dynamic algorithms used for balancing the load are Least Connection algorithm, Weighted Least Connection algorithm.

In this paper, we are proposing an algorithm called Modified Central Load Balancer, which allocates the incoming jobs to different virtual machines based on the priority and the state of the virtual machine. Priority is calculated based on the CPU speed and memory of the virtual machines. This algorithm improves resource utilization and serves the requests faster when compared to other existing algorithms.

II. RELATED WORK

Round Robin based load scheduling algorithms had limitation that once the virtual machine (VM) is allocated to a user application request its state is not maintained, which requires the execution of algorithm once again for the same request.Authors in [3] developed an improved round robin which maintains the state each time an application request is run by the server. This was done by using specific data structures called hash map to keep information about the VM allocated to a specific user application request and VM state list, which maintains the status of the VM (i.e.whether busy or free). Simulation results showed this algorithm increased the response time as compared to the ordinary Round Robin scheduler.

A software module called " load balancer" was used to allocate VM's to multiple user application requests, based on the availability of VM's, such that the allocation was optimal [4]. The load balancer will maintain a queue for the user application request and then suitably assign the appropriate VM. It also maintains the information regarding allotments to the VM's thereby knowing in advance which VM's are free. A variation of load balancer called as "active VM load balancer" was developed in [5]. In this approach the VM which is having less load will be assigned to the new application request. The "active VM load balancer" will send the ID of VM to the "data center controller", which sends requests to the VM having the ID received by it.

A modified throttled algorithm for load balancing was developed by the authors in [6]. In this technique the best available VM is allocated to an application request based on the response time and processing time. Here "throttled VM load balancer" maintains the status of every VM. Upon an application request to the "data center controller", it asks the "throttled VM load balancer" to assign the perfect VM based on the application requirements.

A biologically inspired bee colony optimization technique is described in [1], to load balance distributed cloud hosting web services. In this technique, when the number of requests for web services varies, the allocation of web servers varies accordingly. Here, the servers are deemed as virtual servers which contains a service request queue. A measure called "profit" is calculated by the server which gives the service of the application request, based on performance metric like CPU time, etc. There is also "advert board" which indicates the idle servers whether any services are needed by them. The overall profit of the corresponding infrastructure is calculated based on different metrics. Here, the severs will play the role of bees and idle severs will be like the bees waiting to fetch the nectar.

Another biologically inspired load balancing technique is based on the modified ant colony optimization (ACO) technique [7]. ACO is based on the movement of ants using pheromone in a certain direction. This is related to cloud load balancing by considering software module as ants and the VM's as nodes. Initially, one node will be selected as the Regional Load Balancing Node (RLBN), which acts as the head node. Ants will originate from this RLBN and will be able to move in either direction towards the nodes, based on the load of nodes. In this approach a pheromone table is maintained, which maintains information about loads on the nodes. The main aim of ant is to find the least loaded node and assign application to it, thereby equally distributing the node in the network.

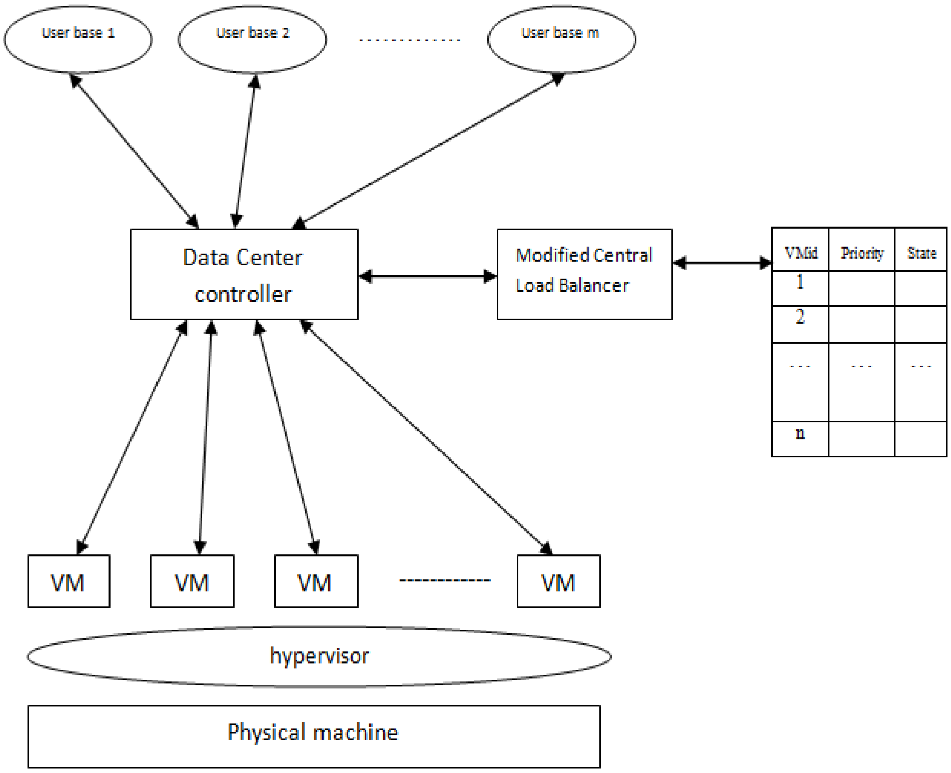
III. PROBLEM DEFINITION

Since load balancing is a major issue in Cloud Computing and utilization of resources is one of the important aspects of Cloud, an algorithm is proposed where the load balancing is done among all the virtual machines having different hardware configurations and load is distributed by considering these hardware configurations of the virtual machines and state of the virtual machines in the data center and all the resources will be utilized efficiently in comparison with other algorithms.

IV. PROPOSED SCHEME

In the proposed algorithm, all the requests from the users from all around the world arrives at the Data Center Controller, which is one of the major components of Cloud. Data Center Controller forwards the request to the Modified Central Load Balancer algorithm to assign the request to the available virtual machines. This algorithm handles a table which contains the id of the virtual machine, priority of the virtual machine and the state of the virtual machine. The algorithm will search the table to find the virtual machine with highest priority and which is available at that moment. If found, the algorithm will reply back to the Data Center Controller with the id of that machine (Vmid) and the Data Center Controller will assign the request to that machine, else, it will wait for the Virtual Machine to be free and once free that machine will be assigned to the request.

The overall architecture of the proposed algorithm is as shown in Fig.1.



VMid	Priority	State
1		
2		
...	...	...
n		

Fig.1: Architecture of Modified Central Load Balancer

The proposed algorithm - Modified Central Load Balancer (MCLB) interacts with all users , Data Center Controller and VMs as shown in the Fig.2. The proposed algorithm can be described as follows:

1. All the user requests are received by the Data Center Controller.
2. On receiving the request, the Data Center Controller processes the request to find the Virtual machine to which it can allocate the job.
3. Data Center Controller (DCC) has the proposed algorithm. Modified Central Load Balancer handles a table that contains the id of the virtual machine (VMid), the priority of the virtual machine and state (AVAILABLE/BUSY) of the virtual machine.
4. Modified Central Load Balancer searches the table to find the Virtual Machine with highest priority and which is

available.
If found:

a) The id of the virtual machine is returned to the Data Center Controller.
b) Data Center Controller assigns the job to the virtual machine with that VMid.
c) Table maintained by the Modified Central Load Balancer will also be updated about the new allocation

If not found:

a) The Data Center Controller will wait for a signal from the Virtual Machine which is sent once the Virtual Machine is free. These signals are obtained from the observer pattern used by VMs.
b) On receiving the signal, the job will be assigned to that Virtual Machine.

5. Virtual Machine will process the request and reply back with the response cloudlet to the Data Center controller.
6. The Data Center Controller will reply back to the user who has sent the request also it will notify the Modified Central Load Balancer about the de-allocation of the Virtual Machine and the table will be updated.
7. If there are any requests waiting to be served, continue from step 3
8. Else, continue from step 1.

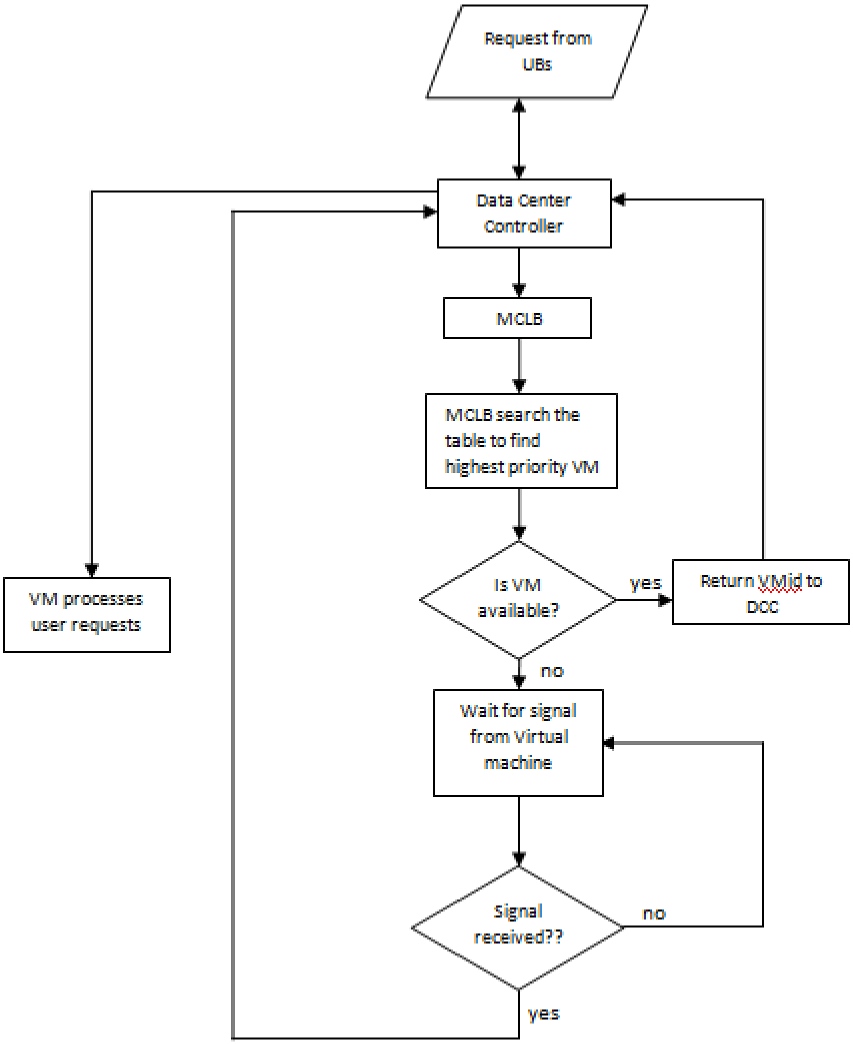

Request from UBs
Data Center Controller
MCLB
MCLB search the table to find highest priority VM
Is VM available?
yes
Return VMid to DCC
no
VM processes user requests
Wait for signal from Virtual machine
Signal received??
no
yes

Fig. 2: Flow Chart of the Proposed Scheme

V. CONCLUSION AND FUTURE WORK

In the proposed algorithm, we are trying to balance the load among all the available virtual machines and the overloading and under loading of virtual machines are avoided. Allocation of jobs is done by considering the priority and the state of the virtual machine which helps in the fair allocation of the jobs and efficient user utilization. Simulation results will also show that, proposed algorithm – Modified Central Load Balancer is more efficient compared to Round Robin Algorithm, Throttled Algorithm and Equally Spread Current Execution load Algorithm.

The proposed algorithm can be further improved by including other factors of the virtual machines to calculate the priority of that Virtual Machine. Also a load balancing algorithm can be developed by considering the processor utilization and memory utilization. This will distribute the load more efficiently.

REFERENCES

[1] Nakrani, Sunil, and Craig Tovey. "On honey bees and dynamic server allocation in internet hosting centers." Adaptive Behavior 12.3-4 (2004): 223-240.

[2] Radojevic, Branko, and Mario Zagar. "Analysis of issues with load balancing algorithms in hosted (cloud) environments." MIPRO, 2011 Proceedings of the 34th International Convention. IEEE, 2011.

[3] Mahajan, Komal, Ansuyia Makroo, and Deepak Dahiya. "Round Robin with server affinity: a VM load balancing algorithm for cloud based infrastructure."*Journal of information processing systems* 9.3 (2013): 379-394.

[4] Liang, Po-Huei, and Jiann-Min Yang. "Evaluation of Cloud Hybrid Load Balancer (CHLB)." *International Journal of E-Business Development* (2013).

[5] Moharana, Shanti Swaroop, Rajadeepan D. Ramesh, and Digamber Powar. "Analysis of load balancers in cloud computing." *International Journal of Computer Science and Engineering* 2.2 (2013): 101-108.

[6] Domanal, Shridhar G., and G. Ram Mohana Reddy. "Load Balancing in Cloud Computingusing Modified Throttled Algorithm." *Cloud Computing in Emerging Markets (CCEM), 2013 IEEE International Conference on*. IEEE, 2013.

[7] Nishant, Kumar, et al. "Load balancing of nodes in cloud using ant colony optimization." *Computer Modelling and Simulation (UKSim), 2012 UKSim 14th International Conference on*. IEEE, 2012.

[8] Kokilavani, T., and Dr DI George Amalarethinam. "Load balanced min-min algorithm for static meta-task scheduling in grid computing." *International Journal of Computer Applications* 20.2 (2011): 43-49.

[9] Soni, Gulshan, and Mala Kalra. "A novel approach for load balancing in cloud data center." *Advance Computing Conference (IACC), 2014 IEEE International*. IEEE, 2014.

[10] Randles, Martin, David Lamb, and A. Taleb-Bendiab. "A comparative study into distributed load balancing algorithms for cloud computing." *Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference on*. IEEE, 2010.