Volume 5, Issue 4, 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Review of Load Balancing in Cloud Computing

Gunpriya Makkar, Pankaj Deep Kaur
Computer Science and Engineering
Guru Nanak Dev University,
Jalandhar, India

Abstract*— Clouds are highly configured infrastructures that deliver platform and software as service, which helps users to make subscription for their requirements. Cloud computing is expanding globally, because of its simple service oriented model. The numbers of users using the cloud are increasing day by day. The cloud is based on data centers generally that are very powerful in handling such large number of users accessing data. The reliability of clouds depends on how the loads are being handled; clouds must be featured with some _mechanism to overcome such problems. Load Balancing is a mechanism of reassigning the total load to the individual nodes to achieve effective resource utilization and to improve the response time of the job, and also to remove a condition in which some of the nodes of the system are over loaded while the others are under loaded. Load balancing mechanism in cloud computing helps clouds to increase their capability which results in powerful, reliable and efficient clouds. This paper is a brief discussion on load balancing on cloud model.

Keywords*— Cloud Computing, Load Balancing, Virtualization, Honey Bee Foraging Algorithm, Round Robin, Throttling Load Balancing

I. INTRODUCTION

Cloud computing is one of the fastest implementing technology in the decade. Many companies are trying to implement and introduce clouds, due to its simple and flexible architecture. These result in the increasing number of users reaching cloud. Although clouds are bifurcated in public private and hybrid models but still problem of reliability may arise in these clouds. Cloud computing has been adopted by organization which includes, social networking websites, online application design by Google app managers and by Google doc which are some of the important implementation and a step ahead in cloud computing. This all suggests that cloud computing will change the way we interact with the resources via Internet. Cloud models use virtualization technology; this technology helps in slicing a single data centre or high power server to act as multiple machines. It depends on the hardware configuration of the data centre or server in how may virtual machine they can be divided. Load balancing is the pre requirements for increasing the cloud performance and for completely utilizing the resources.

Load balancing is one of the major issues related to cloud computing. The load may be memory, CPU capacity, network load or delay load. It is always required that work load must be shared among the various nodes of the distributed system so as to improve the resource utilization and also for better performance of the computing system. This can aid to avoid the situation where some of the nodes are either overloaded or under loaded in the network. Load balancing can be either centralized or decentralized. Load Balancing algorithms are used for implementing. Today cloud computing is a set of several data centers which are sliced into virtual servers and located at different geographical location for providing services to clients. The objective of paper is to review load balancing in virtual servers for higher performance rate.

The existing system does have these polices of load balancing , but still the efficiency of these algorithms are studied and presented to find the best suited algorithm for load balancing of virtual servers. Load balancing works in the manner to decide which virtual machine is in steady state while which virtual machine will go on hold. Load balancing helps in reducing the bandwidth usage which results in decreasing the cost of machine and maximizing the services offered by the service providers.

The arrival of load can cause some server to be overloaded while other server may be idle or under loaded. Equally distributing the load improves the performance of the cloud by transferring load from the overloaded server. Efficient scheduling and efficient resource allocation is a characteristic of cloud model based on which the system's performance is calculated. These characteristics have an effect on cost optimization, which can be further achieved by improving the response time and processing time.

II. GOALS OF LOAD BALANCING

In order to balance the requests of the resources it is important to recognize a few major goals[1] of load balancing algorithms:

- *Cost effectiveness*: the primary aim is to achieve an overall improvement in performance of the system at a affordable cost.
- *Scalability and flexibility*: the distributed system in which the algorithm is implemented may change in size or topology. So the algorithm must be scalable and flexible enough to allow such changes to be handled easily.

- *Priority*: prioritization of the resources or jobs need to be done on before hand through the algorithm itself for better service to the important or high prioritized jobs in spite of equal service provision for all the jobs regardless of their origin.

III. METRICS TO BE CONSIDERED

There are some qualitative metrics that need to be improved for better load balancing in cloud computing [2][3].

- *Throughput***:** It is the total number of tasks that have completed execution for a given scale of time. It is required to have high through put for better performance of the system.
- *Associated Overhead*: It describes the amount of overhead during the implementation of the load balancing algorithm. It is a composition of movement of tasks, inter process communication and inter processor. For load balancing technique to work properly, minimum overhead should be there.
- *Fault tolerant*: We can define it as the ability to perform load balancing by the appropriate algorithm without arbitrary link or node failure. Every load balancing algorithm should have good fault tolerance approach.
- *Response time*: In Distributed system, it is the time taken by a particular load balancing technique to respond. This time should be minimized for better performance.
- *Resource Utilization*: It is the parameter which gives the information within which extant the resource is utilized. For efficient load balancing in system, optimum resource should be utilized.
- *Scalability*: It is the ability of load balancing algorithm for a system with any finite number of processor and machines. This parameter can be improved for better system performance.
- *Performance*: It is the overall efficiency of the system. If all the parameters are improved then the overall system performance can be improved.

IV. RELATED WORK

There are various techniques to balance the load of cloud computing. Some of which are discussed in this paper:

A. *Honey Bee Foraging Algorithm***:** This whole algorithm[3], is based on the process of honeybees finding the food and alarming others to go and eat the food. First forager bees go and find their food. After coming back to their respective beehive, they dance. After seeing the strength of their dance, the scout bees follow the forager bees and get the food. The more energetic the dance is, the more food available is. So this whole process is mapped to overloaded or under loaded virtual servers. The server processes the requests of the clients which is similar to the food of the bees. As the server gets heavy or is overloaded, the bees search for another location i.e. client is moved to any other virtual server. In this way, this whole technique works.

B. *Honey Bee Foraging Algorithm***:** In this algorithm[4], the processes are divided between all processors. Each process is assigned to the processor in a round robin order. Though the work load distributions between processors are equal but the job processing time for the different processes are not same. So at any point of time some nodes may be heavily loaded and others remain idle. This algorithm is mostly used in web servers where Http requests are of similar nature and distributed equally.

C. *Active Clustering Algorithm*: Active Clustering[5] works on the principle of grouping same kind of nodes together and then working on these groups. The process involved is: A node begins the process and selects another node from its neighbors called the matchmaker node that satisfies the criteria that it should be of a different kind than the first one. The matchmaker node then establishes a connection between one of its neighbor which is of the same kind as the initial node. The matchmaker node then detaches the connection formed between the initial node and itself. The above set of operations is followed repeatedly.

D. *Biased Random Sampling Algorithm:* This algorithm[3] is based on the construction of the virtual graph having connectivity between the all nodes of the system where each node of the graph is corresponding to the node computer of the cloud system. Edges b/w nodes are of two types as Incoming edge and outgoing edge that is used to consider the load of particular system and also allotment the resources of the node. It is scalable technique to balance the load of the cloud system. It is also reliable and effective load balancing approach that is mainly developed to balance the load of distributed system.

E. *Compare and Balance Algorithm*: This algorithm[6] uses the concept of compare and balance to reach an equilibrium condition and manage unbalanced system's load. On the basis of probability (no. of virtual machine running on the current host and whole cloud system), current host randomly select a host and compare their load. If load of current host is more than the selected host, it transfers extra load to that node. Each host of the system performs the same procedure.

F. *Throttled Load balancing Algorithm:* The Throttled algorithm[7] will find the node for assigning the new task. The job manager will maintain a list of node details using index list; with that it assigns the job to a specific node. If that node is ready to accept the job means it will accept and process else it will wait for the other node that is requesting for processing.

G. *Min-Min Algorithm***:** This algorithm[8] begins with a set of all unassigned tasks. Firstly, minimum completion time for all tasks is found out. Then out of these minimum times the minimum value is selected which is the minimum time among all the tasks. Then according to that minimum time, the task is scheduled on the machine. Then the execution time for all other tasks is updated by adding the execution time of the assigned task to the other task's execution times and assigned task is removed from the list of the tasks that are to be allotted to the machines. Then again the same procedure is followed until all the tasks are assigned on the resources. But the main drawback of this approach is that it can lead to starvation.

H. *Connection mechanism*: This Load balancing algorithm [9] is based on the least number of connection mechanisms which is a part of dynamic scheduling algorithm. It needs to count the number of connections for each server to estimate the load. The load balancer will record the number of connections for each server. The number of connection increases by one when a new connection is dispatched to it while it decreases the number by one when connection finishes or timeout happens.

I. *Equally Spread Active Execution Load Algorithm:* The architecture model (Fig. 1) for which the proposed algorithm[10] is implemented. The jobs are submitted to the cloud computing system by the clients. The submitted jobs are queued in the stack when they arrive to the cloud. The cloud manager estimates the size of the job and checks whether there is any available virtual machine and also checks the capacity of that virtual machine. Once the size of the job and the available resource that is the virtual machine, size match the job scheduler instantly allocates the resource to the job in queue. There is no issue of fixing the time slots for scheduling the jobs in some periodic way as it is in the round robin scheduling algorithm. The benefit of the ESCE algorithm is that there is an enhancement in processing time and the response time. The submitted jobs are equally spread, the load of the computing system is balanced and no virtual machines are under-utilized. Due to this advantage, there is a reduction in the cost and the data transfer rate of the virtual machine.

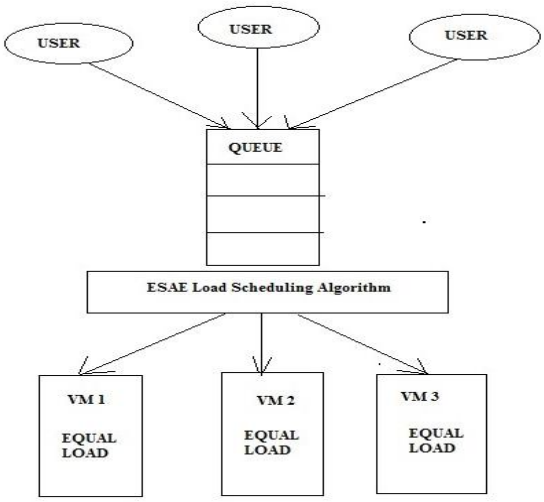

Fig. 1: Equally spread Active execution load to the cloud system

V. CONCLUSION

In this paper, various load balancing algorithms in the Cloud Computing environment have been surveyed. We discussed major issues which must be taken into consideration while designing any load balancing algorithm. We have discussed the already proposed algorithms by various researchers. It is observed with thorough study that, load balancing algorithm works on the principle on which situation workload is assigned, during compile time or run time. Depending on the compile time or run time it may be static or dynamic. Static algorithms are more stable than dynamic algorithm and it is easy to predict the behavior of static algorithm also. Dynamic algorithms are really works better in case of distributed environments. The main aim of load balancing is to satisfy the customer requirement by dynamically distributing the load among the nodes and to make the most of resource utilization by reassigning the load to individual nodes. This process ensures that all the resources are evenly distributed. This increases the performance of the system.

ACKNOWLEDGMENT

History of all great works is to witness that no great work was ever done without either the active or passive support of a person's surroundings and one's close quarters. This research paper is made possible through the help and support from everyone, including: parents, teachers, family, friends, and in essence, all sentient beings. Especially, I would like to dedicate my acknowledgment of gratitude toward the following significant advisors and contributors: First and foremost, I would like to thank my guide, Ms. Pankaj Deep Kaur for her support and encouragement. She kindly read my paper and offered invaluable detailed advices on organization and the theme of the paper. Finally, I sincerely thank to my parents, family, and friends, who provide the advice and support. The product of this research paper would not be possible without all of them.

REFERENCES

[1] ZENON CHACZKO, VENKATESH MAHADEVAN, SHAHRZAD ASLANZADEH, CHRISTOPHERMCDERMID(2011) "AVAILABILITY AND LOAD BALANCING IN CLOUD COMPUTING "INTERNATIONAL CONFERENCE ON COMPUTER AND SOFTWARE MODELING IPCSIT VOL.14 IACSIT PRESS, SINGAPORE 2011

[2] Foster, I., Y. Zhao, I. Raicu and S. Lu, "Cloud Computing and Grid Computing 360-degree compared," in proc. Grid Computing Environment Workshop.

[3] Buyya R., R. Ranjan and RN. Calheiros, "InterCloud: Utility oriented federation of cloud computing environments for scaling of application services," in proc. 10th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP), Busan, South Korea, 2010.

[4] Martin Randles, Enas Odat, David Lamb, Osama Abu- Rahmeh and A. Taleb-Bendiab, "A Comparative Experiment in Distributed Load Balancing", 2009 Second International Conference on Developments in eSystems Engineering.

[5] Ram Prasad Padhy , P Goutam Prasad Rao, "LOAD BALANCING IN CLOUD COMPUTING SYSTEMS"

[6] Yi Zhao, Wenlong Huang, 2009 "Adaptive Distributed Load Balancing Algorithm based on Live Migration of Virtual Machines in Cloud" Fifth International Joint Conference on INC, IMS and IDC.

[7] P.Jamuna,R.Anand, "Optimized Cloud Partitioning Technique to Simplify Load Balancing"

[8] T. Kokilavani J.J. College of Engineering & Technology and Research Scholar, Bharathiar University, Tamil Nadu, India" Load Balanced Min-Min Algorithm for Static Meta-Task Scheduling in Grid Computing" International Journal of Computer Applications.

[9] P.Warstein, H.Situ and Z.Huang (2010), "Load balancing in a cluster computer" In proceeding of the seventh International Conference on Parallel and Distributed Computing, Applications and Technologies, IEEE

[10] Jaspreet kaur / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 Vol. 2, Issue 3, May-Jun 2012, "Comparison of load balancing algorithms in a Cloud"

