© 2015 IJSRSET | Volume 1 | Issue 6 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

A Comparative Survey on Various Load Balancing Techniques in Cloud Computing

Patel Yagnesh R., Dr. Chirag Thaker

Department of Information Technology, Shantilal Shah Engineering College, Bhavnagar, Gujarat, India

ABSTRACT

Cloud computing is an emerging technology adopted by both industry and academia pro viding a flexible and efficient way to store and retrieve the data files. Today's cloud comprises too many hardware and software resources. Load balancing is the crucial parameter required for efficient operation of various components in cloud computing environment. Clients request specific resources from the cloud. A efficient load balancing strategy must needed to be employed to map clients request to available resources in such way that lead to minimum load on the system and provide the resources at rapid rate. In this paper various load balancing algorithms and how this algorithms help in solving the issues of load distribution among various resources / virtual machine are discussed.

Keywords: Cloud computing; Load balancing; Load balancing algorithms; Virtual machine; Resource allocation.

I. INTRODUCTION

Cloud computing is a distributed computing environment which includes large set of virtualized computing resources, various development platforms, infrastructures and useful software are delivered as a service to customers on pay as per use basis usually over the internet [1]. According to the official NIST definition, "cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [2]. It provides computing as a service not as a product. The three basic service layer of the cloud computing are: infrastructure, platform and applications that are popularly referred to as IaaS (infrastructure as a Service), PaaS (platform as a service), SaaS (Software as a Service) that helps organizations, academia and businesses to store and retrieve data files efficiently.

In cloud computing, load balancing is one of the major techniques that have a dramatic impact on resource availability. The term availability, was always a main concern in cloud computing. Fundamentally, availability explains the ubiquitousness of the network information in case of resource scaling. Load balancing could be illustrated, as proper strategy for task scheduling that will lead to balanced load distribution in cloud networks. It is an important key to improve the network performance [3]. Proper load balancing algorithm must needed to be employed to ensure that each user request is allocated to virtual machine / resources in efficient manner and in such way that it incur minimum load on virtual machines and resources.

This survey paper presents the various load balancing algorithms and with their work to fulfill users need by ensuring that it lead to minimum load on the cloud system.

The paper is organized as follows: In section II, includes the related works about load balancing. Section III presents the various load balancing algorithms and finally Section IV derives to the conclusion.

II. METHODS AND MATERIAL

A. Load Balancing

Load balancing could be illustrated, as proper strategy for task scheduling that will lead to balanced load distribution in cloud networks. Load balancing algorithms can be broadly classified into two types: Static algorithms and Dynamic algorithms. In Static Scheduling the assignment of tasks to processors is done before program execution begins i.e. in compile time. Scheduling decision is based on information about task execution times, processing resources etc. which is assumed to be known at compile time [4]. Static scheduling methods are non-pre-emptive. The goal of static scheduling methods is to minimize the overall execution time. These algorithms cannot adapt to load changes during run-time [6].

Dynamic scheduling (often referred to as dynamic load balancing) is based on the redistribution of processes among the processors during execution time. This redistribution is performed by transferring tasks from the heavily loaded processors to the lightly loaded processors with the aim of improving the performance of the application. It is particularly useful when the requirement of process is not known a priori and the primary goal of the system is to maximize the utilization of resources [4]. The major drawback of dynamic load balancing scheme is the run-time overhead due to the transfer of load information among processors, decision-making for the selection of processes and processors for job transfers and the communication delays associated with the task relocation itself.

Reviewing the literature, different load balancing algorithms have been proposed to utilize the available resources. Efficient load balancing algorithm should be robust, and simple enough to be compatible with variety types of applications. The following points are defining a standard framework to design an effective load balancing algorithm [5]:

- ***Complexity*:** The algorithm should not be too complex as complexity will add more overhead on the system.
- ***Scalability*:** The algorithm should be scalable enough to manage all the existing services, if the network scaled up/down.
- ***Fault tolerance*:** The algorithm should be able to manage the load, even if any failure occurs in the network.
- ***Performance and make span*:** The load balancing should be able to optimize the response time to enhance performance.

B. Literature Survey On Load Balancing Algorithms

The cloud comprises of many geographically distributed datacenters, each consisting of hundreds of servers. When new user request is arrive at the Data Center Controller, it passes it to the VMLoadBalancer. VMLoadBalancer determine appropriate virtual machine and assign it to user request. VMLoadBalancer employs one of the following load balancing algorithm to map user request to virtual machine.

Round Robin: The data center controller assigns the requests to a list of VMs on a circular manner. The first request is allocated to a randomly picked VM from the group and then the subsequent requests are assigned in a circular order. Once the VM is assigned a request, it is moved to the end of the list. Though the work load distributions between processors are equal but the job processing time for different processes are not same. So at any point of time some nodes may be heavily loaded and others remain idle [7].

Weighted Round Robin: It is the modified version of Round Robin in which a weight is assigned to each VM. If one VM is capable of handling twice as much load as the other, the powerful server gets a weight of 2. In such cases, the Data Center Controller will assign two requests to the powerful VM for each request assigned to a weaker one. The major issue in this allocation is same as that of Round Robin that is it also does not consider the advanced load balancing requirements such as processing times for each individual requests [8].

Dynamic Round Robin*:* This algorithm mainly works for reducing the power consumption of physical machine. This algorithm uses two rules as follows:

I) If a virtual machine has finished its execution and there are other virtual machines hosted on the same physical machine, *t*his physical machine will accept no more new virtual machine.Such physical machines are called to be in "retiring" state, i.e. when rest of the virtual machines finishes their execution, and then this physical machine can shut down.

II) The second rule says that if a physical machine is in retiring state for a long time then instead of waiting, all the running virtual machines are migrated to other physical machines. After the successful migration, we can shut down the physical machine. This waiting time threshold is called “retirement threshold” [9].

Equally Spread Current Execution (ESCE) Algorithm: In this method the load balancer continuously scans the job queue and the list of virtual machines. If there is a VM available that can handle the request then the VM is allocated to that request. If there is an overloaded VM that needs to be freed of the load, then the load balancer distributes some of its tasks to the VM having least load to make every VM equally loaded [9]. The balancer tries to improve the response time and processing time of a job by selecting it whenever there is a match. But it is not fault tolerant and has the problem of single point of failure.

Throttled Load Balancing Algorithm: Throttled algorithm 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 [10]. In this algorithm the client first requests the load balancer to find a suitable Virtual Machine to perform the required operation.

Modified Throttled Load Balancing Algorithm: 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. 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 [11]. 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.

Active Monitoring Load Balancer (AMLB) Algorithm: Active VM Load Balancer 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 [10]. 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.

Weighted Active Monitoring Load Balancing Algorithm: Jasmin James et. al proposed this method [10] which is a combination of Weighted Round Robin and Active Monitoring Load Balancing Algorithm. In this algorithm different weights are assigned to VMs depending on the available processing power. Among the least loaded VMs the tasks are assigned to the most powerful one according to their weights. In this way it removes the shortcomings of Active Monitoring Load Balancing Algorithm by not only considering the load but also the processing power of available VMs.

VM Assign Load Balancing Algorithm: It is a modified version of Active Monitoring Load Balancing algorithm. The first allocation of VM is similar to previous algorithm. Then if next request comes it checks the VM table, if the VM is available and it is not used in the previous assignment then it is assigned and id of VM is returned to Data Center, else we find the next least loaded VM and it continues, unlikely of the Active load balancer, where the least loaded VM is chosen but it will not check for the previous assignments [12]. According to Shridhar G. Domanal et. al this algorithm will utilize all the VMs completely and properly unlike the previous one where few VMs will be overloaded with many requests and rest will remain underutilized. But it is not clearly mentioned in the paper that how it happens. This algorithm will not use the VM if it is already allocated in the last round. But there is no logic behind it. Because it may still be the least loaded VM having good processing speed. So more tasks can be assigned to it. Finding the next least loaded VM will distribute the tasks evenly only when there are multiple VMs which are equally loaded or the next least loaded VM has a high processing speed compare to the previous one. But the algorithm only considers the load and if the VMs are equally loaded then the task can be assigned to any of them irrespective of the fact that whether the VM is used in the last iteration or not. Since allocation of a task change the state of VM so in the previous algorithm least loaded VM will be found automatically and even task distribution will take place [12].

Weighted Least Connection : The WLC algorithm [13] assigns tasks to the node based on the number of connections that exist for that node. This is done based on a comparison of the SUM of connections of each node in the cloud and then the task is assigned to the node with least number of connections. However, WLC does not take into consideration the capabilities of each node such as processing speed, storage capacity and bandwidth.

Opportunistic Load Balancing: OLB is a static load balancing algorithm whose goal is to keep each node in the cloud busy. OLB does not consider the current load on each node. It attempts to dispatch the selected job to a randomly selected available VM. OLB does not consider the execution time of the task in that node. This may cause the task to be processed in a slower manner increasing the makespan [14] and will cause some bottlenecks since requests might be pending waiting for nodes to be free.

Central Load Balancer: It is an updated version of Throttled algorithm. It maintains a table containing the state of each VM along with their priority. The priority is calculated based on the CPU speed and capacity of memory [15]. The VM assignment policy is similar to that of Throttled load balancing algorithm except that in this algorithm the VM with highest priority will get the first preference. If VM is busy then the VM with next highest priority is checked and the process continues until a new VM is found or the whole table is searched. The algorithm efficiently balances load in a heterogeneous environment but it suffers from bottleneck as all the requests will come to central load balancer. Moreover the algorithm is based on the priority of VMs which is calculated in a static way and is not updated during job allocation.

Table 1. Presents the comparative analysis of above maintained load balancing algorithm with their pros and cons.

TABLE 1. Various Load Balancing Algorithms With Their Pros And Cons

Sr. No	Algorithm	Description	Pros	Cons
1	Round Robin [7]	First request is allocated to a randomly picked VM. Subsequent requests are assigned in circular manner.	Even distribution of load	Processing time of request is not considered.
2	Weighted Round Robin [8]	Based on processing power weight is assigned to VM. More request are assigned to powerful VM.	Better resource utilization.	Processing time of request is not considered.
3	Dynamic Round Robin [9]	Reduce power consumption by shutting down VM which have completed their task.	Low power consumption	Low scalability.
4	Equally Spread Current Execution (ESCE) [9]	Request is assigned to any available VM that can handle it. If there is an overloaded VM then the balancer distributes some of the tasks to some idle VM to balance the load.	Improved processing and response time	Not fault tolerant because of single point of failure.

5	Throttled load balancing [10]	Request is accepted if available VM is found in the table otherwise -1 is returned and the request is queued.	TLB tries to distribute the load evenly among the VMs.	Does not consider the current
6	Modified Throttled load balancing [10]	Unlike Throttled here the index table is searched from the next to already assigned VM.	Gives better response time.	State of index table may change during next allocation.
7	AMLB Algorithm [11]	The number of requests currently allocated to each VM is maintained. Request is allocated to the least loaded VM.	Consider both load and availability of VM.	Processing power of VM is not considered.
8	Weighted AMLB [10]	Weights are assigned to VMs depending on their available processing power. Task is assigned to the least loaded and most powerful VM.	Considers both the load as well as the processing power of available VMs.	Assigning weight increases the complexity of the algorithm.
9	VM – Assign Load Balancing Algorithm [12]	Similar to the above algorithm but unlike it VM is assigned if it is available and not used in the previous assignment.	Utilizes all the VMs completely and properly.	It is not clearly mentioned in the paper that how it happens.
10	Weighted Least Connection [13]	Assigns tasks to the node having least number of connections.	Balances load efficiently.	Processing speed and storage capacity are not considered.
11	OLB [14]	OLB is Static load balancing algorithm which attempts to dispatch the selected job to the available randomly chosen VM.	Keeps each node in the cloud busy.	Execution time of the task is not considered.
12	Central Load Balancer [15]	Suitable for VM allocation is like Throttled but based on priority which calculated using CPU speed and memory capacity of VM.	Heterogeneous environment.	Priority is fixed and bottleneck problem.

III. CONCLUSION

In this paper, we surveyed multiple algorithms for load balancing in cloud computing and describe how each algorithm helps in managing load on virtual machines. Load balancing algorithms not only provides balanced distribution of load among various resources but also provide greater levels of fault tolerance and high scalability. This review paper also present the pros and cons associated with load balancing algorithms with their functioning. The algorithms described in this paper are not only useful to balance the load but also help in efficient utilization of resources, increase overall throughput and decrease the response time. All these will reduce the operational cost and will attract more users towards cloud computing. The challenges of these load balancing algorithms are addressed so that more efficient load balancing techniques can be developed in future.

IV. REFERENCES

[1] Saeed Parsa and Reza Entezari- maleki, "RASA: A New Task Scheduling Algorithm in Grid Environment". World Applied Sciences Journal 7 (Special Issue of Computer & IT): 152-160, 2009 ISSN 1818.4952© IDOSI Publications, 2009J.

[2] L. Badger, T Grance, R. P. Comer and J. Voas, DRAFT cloud computing synopsis and recommendations, Recommendations of National Institute of Standards and Technology (NIST), May-2012.

[3] Shahrzad Aslanzadeh, Zenon Chaczko and Christopher Chiu, "Cloud Computing—Effect of Evolutionary Algorithm on Load Balancing", Springer International Publishing Switzerland 2015.

[4] X. Evers , "A Literature Study on Scheduling in Distributed Systems ",1992.

[5] Galante, G., de Bona, L.C.E.: A survey on cloud computing elasticity. In: IEEE Fifth International

Conference on Utility and Cloud Computing (UCC), pp. 263-270 (2012).

[6] Klaithem Al Nuaimi, Nader Mohamed, Mariam Al Nuaimi and Jameela Al-Jaroodi , "A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms ", IEEE Second Symposium on Network Cloud Computing and Applications , 2012, pp. 137-142.

[7] Amandeep Kaur Sidhu, Supriya Kinger , "Analysis of Load Balancing Techniques in Cloud Computing ", International Journal of Computers & Technology , Volume 4 No. 2, March-April, 2013, pp. 737-741.

[8] Qi Zhang, Lu Cheng, Raouf Boutaba; Cloud computing: sate-of-art and research challenges; Published online: 20th April 2010, Copyright : The Brazillian Computer Society 2010.

[9] M.Aruna , D. Bhanu, R.Punithagowri , "A Survey on Load Balancing Algorithms in Cloud Environment ", International Journal of Computer Applications, Volume 82 – No 16, November 2013 , pp. 39-43.

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

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

[12] Shridhar G. Domanal and G. Ram Mohana Reddy, "Optimal Load Balancing in Cloud Computing by efficient utilization of virtual machines", IEEE, International conference. CCEM 2014. In press.

[13] Lee, R. and B. Jeng, "Load-balancing tactics in cloud", International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), October 2011, IEEE, pp. 447-454.

[14] Shu-Ching Wang, Kuo-Qin Yan, Wen-Pin Liao, Shun-Sheng Wang, "Towards a Load Balancing in a Three-level Cloud Computing Network", 3rd International Conference on Computer Science and Information Technology (ICCSIT), Vol. 1, IEEE, 2010, pp. 108-113.

[15] Gulshan Soni, Mala Kalra, "A Novel Approach for Load Balancing in Cloud Data Center", International Advance Computing Conference (IACC) IEEE , 2014, pp.807-812.