Load Balancing for Improved Quality of Service in the Cloud

AMAL ZAOUCH
Mathématique informatique et traitement de l'information
Faculté des Sciences Ben M'SIK
CASABLANCA, MORROCO

FAOUZIA BENABBOU
Mathématique informatique et traitement de l'information
Faculté des Sciences Ben M'SIK
CASABLANCA, MORROCO

***Abstract*—Due to the advancement in technology and the growth of human society, it is necessary to work in an environment that reduces costs, resource-efficient, reduces man power and minimizes the use of space. This led to the emergence of cloud computing technology. Load balancing is one of the central issues in the cloud, it is the process of distributing the load and optimally balanced between different servers. ; Balanced load in the Cloud improves the performance of the QoS parameters such as resource utilization, response time, processing time, scalability, throughput, system stability and power consumption. Research in this area has led to the development of algorithms called load balancing algorithms. In this paper, we present the performance analysis of different load balancing algorithms based on different metrics such like response time, processing time, etc.... The main purpose of this article is to help us to propose a new algorithm by studying the behavior of the various existing algorithms.**

***Keywords*—Cloud Computing; Load Balancing; Cluster; Virtual Machines; Quality of Service**

I. Introduction

Current cloud computing environment serves in almost every field of our life. But while fulfilling lots and lots of user requests it faces few limitations to be overcome. Along with providing us facilities like virtualization, resource sharing, ubiquity, utility computing it ask us to focus on issues like security, authentication, fault tolerance, load balancing, and availability. The different types of services provided by cloud systems are software services (software as a service, SaaS) or physical services (platform as a service, PaaS) or hardware/infrastructure service (Infrastructure as a Service, IaaS). There are various advantages of cloud computing including virtual computing environment, on demand services, maximum resource utilization and easy to use services etc. But there are also some critical issues like security, privacy, load management and fault tolerance etc which needs to be addressed for better performance. As we know load is very unpredictable, even a spike will results overloaded nodes, which often lead to performance degradation and are vulnerable to failure. So Load balancing is one of the main challenges in Cloud computing which is required to distribute the dynamic workload across multiple nodes to ensure that no single node is overwhelmed.

In our paper we have carried out the study of eleven load balancing algorithms, various parameters are used to check the results. In this paper first Introduction is given then in II brief introduction to cloud and its components, III gives introduction load balancing, in section IV, we define the problematic of our work, V gives the study of related work and results with the help of table I and conclusion is given in VI.

II. Cloud and its Components

A Cloud consists of a number of datacenters, which are further divided into a number of nodes, and each node consists of a number of VMs. The requests are actually deployed on the VMs. Figure 1 gives the overview of generalized architecture of the cloud.

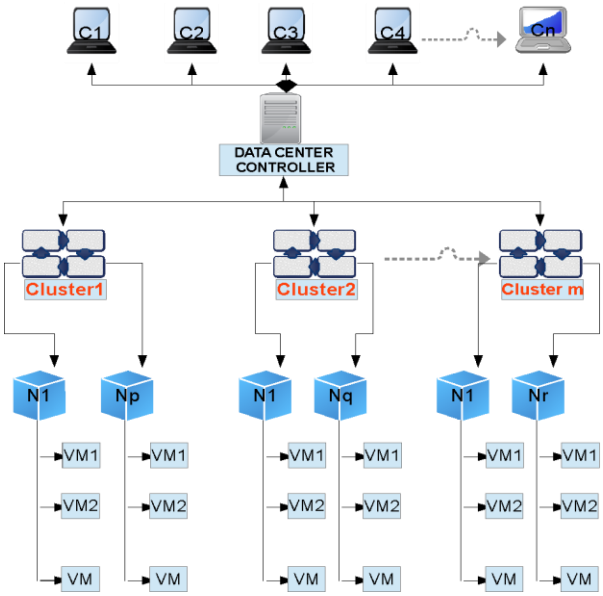

Fig. 1. Generalized architecture of a Cloud

$C_{i=1,2,\ldots,n}$: Users or brokers acting on their behalf submit service requests from anywhere in the world to the Data Center and Cloud to be processed.

Datacenter Controller, This component is used to control the various data center activities.

Cluster: is a set of Nodes.

Node: is a set of virtual machines

VM: A virtual machine (VM) is a software program or operating system that not only exhibits the behavior of a separate computer, but is also capable of performing tasks such as running applications and programs like a separate computer. A virtual machine, usually known as a guest is created within

another computing environment referred as a "host." Multiple virtual machines can exist within a single host at one time.

III. Problem Statement

As it s shown in the previous section, when a request arrive to the Datacenter Controller, it has to be allocated to one of the nodes composed the cluster, but the requests have to be distributed evenly and equally among the system to avoid workloads and degradation of system's performance; if we refer to the architecture of cloud shown in figure 1 we deduce that we need another components to balance the load overall the system called Load Balancer. The Load Balancer plays a very important role in the overall response time of the cloud. In Cloud Computing Scenario Load Balancing is composed of selecting Data Center for upcoming request and Virtual machine management at individual Data Center So, how can we guarantee a good quality of service though balancing the load in the cloud?

Our aim is to design a new Load Balancer to improve quality of service by optimizing load balancing in cloud computing.

IV. Load Balancing

Load Balancing is a computer networking method to distribute workload across multiple computers or a computer cluster, network links, central processing units, disk drives, or other resources, to achieve optimal resource utilization, maximize throughput, minimize response time, and avoid overload [1]. It is a mechanism that distributes the dynamic local workload evenly across all the nodes in the whole cloud to achieve a high user satisfaction and resource utilization, hence improving the overall performance and resource utility of the system. It also ensures that every computing resource is distributed efficiently and fair, prevents bottlenecks and fail-over.

A. Classification of Load Balancing Algorithms

Based on the current state of the system the algorithms can be classified as [2]:

Static Load Balancing: Static load balancing algorithms decide how to distribute the workload according to a prior knowledge of the problem and the system characteristics. Static load balancing algorithms are not pre-emptive and therefore each machine has at least one task assigned for itself. Its aims in minimizing the execution time of the task and limit communication overhead and delays. This algorithm has a drawback that the task is assigned to the processors or machines only after it is created and that task cannot be shifted during its execution to any other machine for balancing the load.

Dynamic Load Balancing: Dynamic algorithms use state information to make decisions during program execution. An important advantage of this approach is that its decision for balancing the load is based on the current state of the system which helps in improving the overall performance of the system by shifting the load dynamically.

B. Metrics For Load Balancing In Clouds

Various metrics considered in existing load balancing techniques in cloud computing are discussed below [15]:

Throughput is used to calculate the no. of tasks whose execution has been completed. It should be high to improve the performance of the system.

Overhead Associated determines the amount of overhead involved while implementing a load-balancing algorithm. It is composed of overhead due to movement of tasks, interprocessor and inter-process communication. This should be minimized so that a load balancing technique can work efficiently.

Fault Tolerance is the ability of an algorithm to perform uniform load balancing in spite of arbitrary node or link failure. The load balancing should be a good fault-tolerant technique.

Migration time is the time to migrate the jobs or resources from one node to other. It should be minimized in order to enhance the performance of the system.

Response Time is the amount of time taken to respond by a particular load balancing algorithm in a distributed system. This parameter should be minimized.

Resource Utilization is used to check the utilization of resources. It should be optimized for an efficient load balancing.

Scalability is the ability of an algorithm to perform load balancing for a system with any finite number of nodes. This metric should be improved.

Performance is used to check the efficiency of the system. This has to be improved at a reasonable cost, e.g., reduce task response time while keeping acceptable delays

V. Related Work

In cloud computing, load balancing is required to distribute the dynamic local workload evenly across all the nodes. It helps to achieve a high user satisfaction and resource utilization ratio by ensuring an efficient and fair allocation of every computing resource. Proper load balancing aids in minimizing resource consumption, implementing fail-over, enabling scalability, avoiding bottlenecks and over-provisioning etc. In this section, a systematic review of existing load balancing techniques is presented. This study concludes that all the existing techniques mainly focus on reducing associated overhead, service response time and improving performance etc. Various parameters are also identified, and these are used to compare the existing techniques.

Throttled load balancer [4] this algorithm ensures only a pre-defined number of Internet Cloudlets are allocated to a single VM at any given time. If more request groups are present than the number of available VM's at a data center, some of the requests will have to be queued until the next VM

becomes available.

Active VM Load Balancer [5][6] maintains information about each VM and the number of requests currently allocated to the VMs. When a request for the allocation of a new VM arrives, the balancer identifies the least loaded VM. If there are more than one, the first identified is selected. The balancer returns the VM id to the Data Centre Controller and the Data Centre Controller sends the request to the VM identified by that id. Data Center Controller notifies the balancer of the new allocation for table updation. When VM finishes processing the request, Data Center controller notifies the balancer for VM deallocation.

Meenakshi Sharma et al. [6] proposed a new Efficient Virtual Machine Load Balancing Algorithm. The proposed algorithm finds the expected response time of each resource (VM). When a request from the data center controller arrives, algorithm sends the ID of virtual machine having minimum response time to the data center controller for allocation to the new request. The algorithm updates the allocation table, increasing the allocation count for that VM. When VM finishes processing of request, data center controller notifies algorithm for VM deallocation. The experimental result compares proposed VM load balancing algorithm with the Throttled Load Balancer and Active VM Load Balancer. The efficient selection of a VM increases the overall performance of the cloud environment and also decreases the average response time and cost compare to Throttled Load Balancer and Active VM Load Balancer.

Jasmin James et al. [7] proposed Weighted Active Monitoring Load Balancing (WALB) Algorithm which has an improvement over the Active VM Load Balancer. This algorithm creates VM's of different processing power and allocates weighted count according to the computing power of the VM. WALB maintains index table of VM's, associated weighted count and number of request currently allocated to each VM. When a request to allocate a VM arrives from the Data Center Controller, this algorithm identifies the least loaded and most powerful VM according to the weight assigned and returns its VM id to the Data Center Controller. The Data Center Controller sends a request to the identified VM and notifies the algorithm of allocation. The algorithm increases the count by one for that VM. When VM finishes processing, algorithm decreases the count of that VM by one. The experimental result shows that the proposed algorithm achieves better performance factors such as response time and processing time, but the algorithm does not consider process duration for each individual request.

Mintu M Ladani et. al. [8] proposed a new virtual machine load balancing algorithm Modified Weighted Active Monitoring Load Balancing Algorithm". This algorithm creates VM's of different processing power and allocates weighted count according to the computing power of the VM. It maintains index table of VM's, associated weighted count and number of request currently allocated to VM. When a request to allocate VM arrives from the Data Center Controller, this algorithm identifies VM with least load, least process duration and most powerful VM according to the weight assigned and returns its VM id to the Data Center Controller. Data Center Controller sends a request to the identified VM and notifies the algorithm of allocation. The algorithm increases the count by one for that VM. When VM finishes processing, algorithm decreases the count of that VM by one. Modified Weighted Active Monitoring Load Balancing algorithm balances the load between the available VMs and considers most important factor process duration to achieve better performance parameters such as response time and processing time.

Vaidehi. M et. al. [9] Enhanced Load Balancing to Avoid Deadlock proposed a technique to avoid deadlock among virtual machines while processing a request by migrating the virtual machine. The cloud manager in the data center maintains a data structure containing VM ID, job ID, and VM status. The VM status represents percentage of resource utilization. Cloud manager distributes the load as per the data structure and also analysis VM status routinely. If any VM is overloaded, which causes deadlock, then one or two jobs are migrated to a VM which is underutilized by tracking the data structure. If there are more than one available VM, then assignment is based on least hop time. On completion of the execution, the cloud manager automatically updates the data structure. The proposed algorithm yields less response time by VM migration from overloaded VM to underutilized VM by considering hop time to avoid deadlock without interacting with the data center controller in updating the data structure. This increases the number of jobs to be serviced by cloud provider, thereby improves working performance as well as business performance of the cloud.

In Round Robin Load Balancer [10] [11], Data Center Controller assigns first request to a virtual machine, picked randomly from the group. Subsequently, it assigns requests to the virtual machines in circular order. Once request assigned to a virtual machine, then the virtual machine is moved to the end of the list. The advantage of Round Robin algorithm is that it does not require inter-process communication. Since the running time of any process is not known prior to execution, there is a possibility that some nodes may get heavily loaded.

Weighted Round Robin algorithm [10] is a modified version of Round Robin Load Balancer. This algorithm assigns a relative weight to all the virtual machines. If one VM is capable of handling twice as much load as the other, then the VM gets a weight of 2. In such cases, Data Center Controller will assign two requests to a VM with weight 2 against one request assigned to a VM with weight 1.

Argha Roy et. al. [12] proposed Dynamic Load Balancer to avoid fault tolerance in cloud computing. Dynamic load balancer is used as an intermediate node between clients and cloud which monitors the load of each virtual machine in the cloud pool. When the users send the request to the dynamic load balancer, it gathers the processor utilization and memory utilization of each active server. If the processor utilization and memory utilization is less than 80%, the dynamic load balancer instantiates a new virtual machine on that server. Now, the request is assigned to this newly created VM. Otherwise, the algorithm Instantiates a new VM on the next server with the lowest processor and memory utilization. The algorithm also checks fault occurrence of a server. If any fault occurs, then the VMs will be shifted to another server whose processor and

memory utilization is less than 80%. The proposed dynamic load balancer algorithm achieves high scalability, dynamic load balancing, fault tolerance and low overhead.

Round Robin with Server Affinity [13]: A VM Load Balancing Algorithm for Cloud Based Infrastructure, the limitation of the available Virtual Machine. , that the Round Robin Algorithm does not save the state of the previous allocation of a VM to a request from given Userbase, while the same state is saved in the proposed algorithm. The Round Robin with server affinity VM load balancer maintains two data structures, which are as listed below

1) Hash map: This store the entry for the last VM allocated to a request from a given Userbase.

2) VM state list: this stores the allocation status (i.e., Busy/Available) of each VM.

In the proposed algorithm, when a request is received from the Userbase, if an entry for the given Userbase exists in the hash map and if that particular VM is available, there is no need to run the Round Robin VM load balancing algorithm, which will save a significant amount of time.

Load Balancing in Cloud Computing Using Modified Throttled Algorithm [14], this algorithm focuses mainly on how incoming jobs are assigned to the available virtual machines intelligently. Modified throttled algorithm 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; the Modified Throttled Load Balancer maintains an index table of VMs and the state of the VM (BUSY/AVAILABLE). At the start all VM's are available. 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. When compared to existing Round-Robin and Throttled algorithms, the response time for proposed algorithm has improved considerably.

Table I illustrates a comparison between the reviewed algorithms in terms of the challenges discussed in Section IV. for example, for "Efficient VM Load Balancer, it considers the expected response time by modifying Throttled LB. Active Monitoring Load Balancer, does not consider the hardware characteristics of server for processing the requests, while weighted active monitoring LB (WALB) identifies the least loaded and most powerful VM. Basis on the same algorithm i.e. (ALB), "modified active monitoring LB" has been proposed and it adds factor "process duration" to assign the job to the VM. As for Enhanced Load Balancing Algorithm using Efficient Cloud Management System, we can see that migration of VMs contributes also on a good balancing of load, in this approach, assignment of jobs is basis on the least hope time i.e. the request is handled by the VM with a minimum time taken to become available after migration. Thus the deadlock avoidance enhances the number of jobs to be serviced by cloud service. In Round Robin LB, request are assigned in circular manner, so it does not consider heterogeneity of resources that there is possibility that some nodes may get heavily loaded while others are overloaded, that is why a new algorithm "weighted round robin" come to solve this problem by giving a weight for each VM, which influence the assignment of jobs, hence, load balancing will be improved and response time too. To avoid Fault Tolerance, Dynamic Load Balancer: Improve efficiency in Cloud computing algorithm monitors the load of each VM, if the VM is overloaded it instantiates a new VM to handle the request.Another issue in the current load balancing algorithm is that they don't save the previous state of allocation of a virtual machine to a request from a user, Round Robin with server affinity algorithm is a technique that raise this issue by maintaining two data structures hash map and VM state list, we can deduce that this parameter can be applied to another algorithm. Modified throttled concerns with the fact that how incoming jobs are assigned to the available virtual machines effectively and efficiently. This algorithm works on the grounds of throttled algorithm by maintaining an index table of virtual machines and their states. In this modified algorithm an attempt is made to improve the response time and achieve efficient usage of available virtual machines. In all of the proposed algorithms, response time has improved.

VI. Conclusion & Future Work

Cloud Computing provides everything to the user as a service which includes application as a service, platform as a service and infrastructure as a service. One of the major issues in cloud computing is load balancing. Load balancing is required to distribute the load evenly among all servers in the cloud to maximize the resource utilization, increases throughput, to provide good response time, to reduce energy consumption. Our research about the reviewed algorithms shows that the current design of throttled has better results compared with the other algorithms especially if it is working with response time algorithm. There many metrics that govern the load balancing in a virtualized data centers, the threshold algorithm guarantees most of them except some which are as follows (Sharma S. et.al, 2008)[16]: Overload Rejection: If Load Balancing is not promising additional overload rejection measures are needed. When the overload situation ends then first the overload rejection measures are stopped. After a short guard period Load Balancing is also closed down. Fault Tolerant: This parameter gives that algorithm is able to bear twisted faults or not. It enables an algorithm to continue operating properly in the event of some failure. If the performance of algorithm decreases, the decrease is relational to the seriousness of the failure, even a small failure can cause total failure in load balancing. Process Migration: Process migration parameter provides when does a system decide to migrate a process? It decides whether to create it locally or create it on a remote processing element. The algorithm is capable to decide that it should make changes of load distribution during execution of process or not.

Therefore, as our future work, we are planning to improve throttled to make it more suitable for cloud environment and more efficient in terms of Process Migration.

TABLE I. SYNTHESIS TABLE OF EXISTING LOAD BALANCING ALGORITHMS

Techniques	Metaphors	Conclusion
Throttled load balancer	This algorithm ensures only a pre-defined number of Internet Cloudlets are allocated to a single VM at any given time.	**Response time** improved But other parameters are not taken into account such as: weight of VM, processing time, etc …
MODIFIED THROTTLED ALGORITHM	Focuses mainly on how incoming jobs are assigned to the available virtual machines. load nearly distributed uniformly among VMs.	**Resource Utilization** **Response time** has improved
Efficient Virtual Machine Load Balancing Algorithm.	The proposed algorithm finds the expected response time of each resource (VM).	Increases **performance** of the cloud environment Decreases **response time** and **cost**
Active VM Load Balancer	maintains information about each VM and the number of requests currently allocated to the VMs	Does not consider the hardware capacity of VMs.
Weighted Active Monitoring Load Balancing (WALB) Algorithm	Allocates weighted count according to the computing power of the VM. But the algorithm does not consider **process duration** for each individual request	Increase **response time** and **processing time**.
Modified Weighted Active Monitoring Load Balancing Algorithm	This algorithm identifies VM with least load, least **process duration** and most powerful VM according to the weight assigned But it considers **process duration**.	Increase **response time** and **processing time**. Hence they tried best to consider the most affecting factor (process duration) in performance increase
Enhanced Load Balancing to Avoid Deadlock	Propose a technique to avoid deadlock among virtual machines while processing a request by migrating the virtual machine.	Improves : **Migration time** **Performance** **Response time**
Round Robin Load Balancer	Data Center Controller assigns first request to a virtual machine, picked randomly from the group. it assigns requests to the rest of VMs in circular order.	There is a possibility that some nodes may get heavily loaded while others are overloaded. Decrease **Resource Utilization**
Weighted Round Robin algorithm	This algorithm assigns a relative weight to all the virtual machines.	Improvement of **resource utilization**
Dynamic Load Balancing: Improve efficiency in Cloud Computing	When the users send the request to the dynamic load balancer, it gathers the processor utilization and memory utilization of each active server	**Fault tolerance** high **scalability** low **overhead**
Round Robin with Server Affinity: A VM Load Balancing Algorithm for Cloud Based Infrastructure	The limitation of Round Robin Algorithm is that it does not save the state of the previous allocation of a VM to a request **while** the same state is saved in the proposed algorithm.	Improved **Response time** Data center **processing time**

REFERENCES

[1] P.Mathur, “Cloud Computing: new challenge to the entire computer industry”, 1stInternational conference on parallel, distributed and grid computing, 2010, pp978-1- 4244-767

[2] Ms. Ms. Parin. V. Patel, Mr. Hitesh. D. Patel, Asst. Prof. Pinal. J. Patel, "A Survey On Load Balancing In Cloud Computing", International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 9, November- 2012 ISSN: 2278-0181.

[3] Kansal, N. J., & Chana, I. (2012). Existing load balancing techniques in cloud computing: a systematic review. Journal of Information Systems and Communication, 3(1), 87-91.

[4] Makroo, A., & Dahiya, D. An efficient VM load balancer for Cloud. Applied Mathematics, Computational Science and Engineering 2014

[5] Meenakshi Sharma, Pankaj Sharma, “Performance Evaluation of Adaptive Virtual Machine Load Balancing Algorithm”,International Journal of Advanced Computer Science and Applications, pp 86-88, Volume 3, Issue2, ISSN: 2156-5570, 2012.

[6] Meenakshi Sharma, Pankaj Sharma, Sandeep Sharma, “Efficient Load Balancing Algorithm in VM Cloud Environment”, International Journal of Computer Science and Technology, pp 439-441, Vol 3, Issue 1, ISSN: 0976-8491[online], ISSN: 2229-433[print], Jan-March 2012

[7] Jasmin James, Bhupendra Verma, "Efficient VM Load Balancing Algorithm for a Cloud Computing Environment", International Journal on Computer Science & Engineering, pp 1658-1663, Volume. 4, ISSN: 0975-3397, September 2012.

[8] Mintu M. Ladani, Vinit Kumar Gupta, "A Framework for Performance Analysis of Computing Clouds", International Journal of Innovative Technology and Exploring Engineering (IJITEE), pp 245-247, Volume 2, Issue 6, ISSN: 2278-3075, May 2013.

[9] Vaidehi. M, Rashmi. K. S, Suma. V, ”Enhanced Load Balancing to Avoid Deadlock in Cloud”, International Journal of Computer Applications on Advanced Computing and Communication Technologies for HPC Applications, pp 31-35, June 2012.

[10] Shanti Swaroop Moharana, Rajadeepan D. Ramesh, Digamber Powar, “Analysis of Load Balancers in Cloud Computing “, International Journal of Computer Science and Engineering (IJCSE), Volume 2, Issue 2, ISSN 2278-9960, pp 101-108, May 2013.

[11] Namrata Swarnkar, Atesh Kumar Singh, Shankar “A Survey of Load Balancing Technique in Cloud Computing”, International Journal of Engineering Research & Technology, pp 800-804, Vol 2, Issue 8, August 2013.

[12] Argha Roy, Diptam Dutta, “Dynamic Load Balancing: Improve efficiency in Cloud Computing”, International Journal of Emerging

Research in Management Technology, pp 78-82, Vol 2, Issue 4, ISSN:2278-9359, April 2013.

[13] Komal Mahajan, Ansuyia Makroo and Deepak Dahiya Round Robin with Server Affinity: A VM Load Balancing Algorithm for Cloud Based Infrastructure http://dx.doi.org/10.3745/JIPS.2013.9.3.379 J Inf Process Syst, Vol.9, No.3, September 2013

[14] Domanal, S. G., & Reddy, G. R. M. (2013, October). Load Balancing in Cloud Computingusing Modified Throttled Algorithm. In Cloud Computing in Emerging Markets (CCEM), 2013 IEEE International Conference on (pp. 1-5). IEEE.

[15] Shahapure, N. H., & Jayarekha, P. LOAD BALANCING IN CLOUD COMPUTING: ASurvey. International Journal of Advances in Engineering & Technology, Jan. 2014.

[16] Sharma, S., Singh, S., & Sharma, M. (2008). Performance analysis of load balancing algorithms. World Academy of Science, Engineering and Technology, 38, 269-272.