A Survey on Load Balancing Algorithms in Cloud Environment

M.Aruna
Assistant Professor (Sr.G)/CSE
Erode Sengunthar Engineering College, Thudupathi, Erode, India

D.Bhanu, Ph.D
Associate Professor
Sri Krishna College of Engineering & Technology
Kuniamuthur,Coimbatore, India.

R.Punithagowri
PG Scholar/CSE
Erode Sengunthar Engineering College, Thudupathi, Erode, India.

ABSTRACT
Cloud Computing is a pool of resources that can be shared among the users. At present, Cloud Computing is an emerging technology since it provides services at the user level. Users submit larger tasks to the cloud which performs the task by using the servers and the result is given back to the corresponding user. There are several issues or challenges in the Cloud Computing environment such as Availability, Security, and Resource Allocation etc., this paper concentrates on availability of nodes in the cloud. Balancing Load under the nodes will increase availability of nodes in Cloud. In order to enhance the performance of the entire cloud environment, efficient Load Balancing techniques are needed. Load Balancing(LB) algorithms distribute the load evenly across all the nodes in cloud. Load Balancing in Cloud Computing will increase the reliability and user satisfaction. This survey paper compares various Load Balancing algorithms in cloud.

Keywords
Green Computing, Virtualization, Migration, Load Balancer, Load Balancing.

1. INTRODUCTION
All the large business and small business companies are moving to cloud environment because of its scalability. The jobs arriving to the Cloud Environment are executed by the large data centers which have thousands of blade servers. Cloud Computing is a Service Oriented Architecture (SOA). It provides different types of services to the users. Users can get the services with no need to know their infrastructure. That is, users do not know where the service is originated and its infrastructure. Users need to pay only for what they used from cloud in the form of services. This is the simplicity of Cloud.

The best example of Cloud is the Electric Current Supply that we are using in our day-to-day life. The power is produced by large wind mills or somewhere else. Then the produced power is transformed to various areas by means of wires and transformers. Users consume these power and pay only for what they used. The power consumers do not know the details of where the power is produced and how it is transformed to their houses.

There are 4 different types of cloud environment. They are,

- Public Cloud(Free of Cost, anyone can access)
- Private Cloud(Pay for what you used, only for single organization people)
- Hybrid Cloud(Combined both public & private Clouds)
- Community Cloud(For Communication purpose)

Users can access the cloud resources in the form of services. There are 3 basic services provided by the Cloud Environment. They are,

- Platform as a Service (PaaS)
- Software as a Service (SaaS)
- Infrastructure as a Service (IaaS)

1.1 Virtualization
Cloud Computing is based on the Concept of virtualization technology. Virtualization means "something that is not real but gives all the facilities as a real one". It is the software implementation on the bare hardware so that the resources under the hardware can be utilized more effectively. Cloud Computing uses the virtualization technique to make use the cloud resources efficiently [3]. Two types of virtualization can be used in cloud environment.

1. Full Virtualization
2. Para Virtualization

1.1.1 Full Virtualization
In Full Virtualization [3], the installation of one computer is done on the other computer. It will result in a virtual machine that have all the facilities and softwares that are present in the actual machine.

1.1.2 Para Virtualization
In Para Virtualization [3], the hardware allows multiple operating systems to be run on a single machine. For this, VirtualBox tool is used. Here all the services are not fully available, rather than the services are provided in a partial manner.

1.1.3 Live Migration
Live Migration [4] is a virtualization technique that will increase the resource utilization more than the Full and Para Virtualization methods. In this method, the Virtual machine is migrated from one physical machine to other without halting the currently running programs.In traditional techniques, while performing migration, all the programs need to be shut down then only virtual machine can be migrated. But with Live Migration technique, all the currently running programs can continue their operation. Thereby increasing the resource utilization and throughput.

1.2 Green Computing
Green Computing [5] is based on the things that will not harm any natural environment and posseses less harm to the environment. Since all the small and large business vendors were moving to cloud environment, the usage of data center is increased. The power consumed by these data centers can also be increased. In order to achieve Green Computing, the power consumption is to be reduced. Power Consumption is directly

related to Carbon Emission. If the Carbon Emission is reduced then power consumption is also be reduced.

2. LOAD BALANCING

Load Balancing [1] is a technique to balance the load across cloud environment. It is the process of transfering load from heavily loaded nodes to low loaded nodes. As a result, no node should be heavily loaded. Thereby it will increase the availability of nodes.

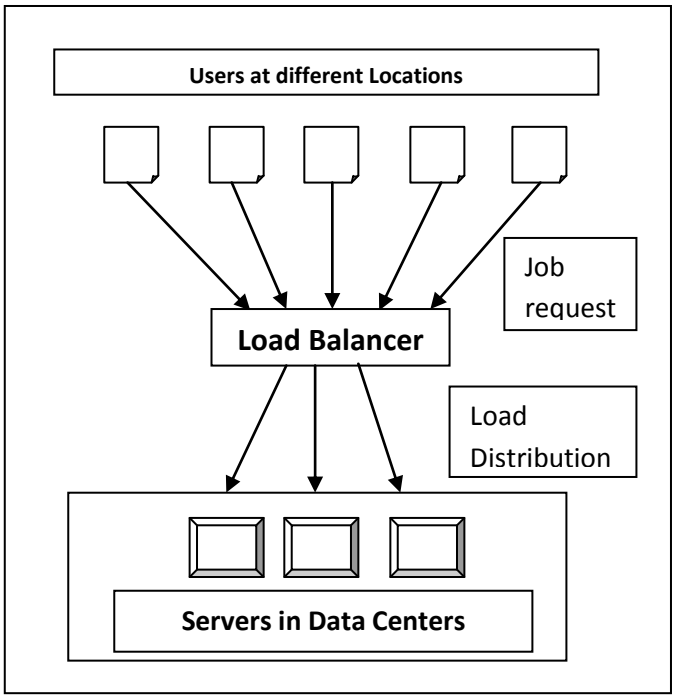

Fig 1: Load Balancing Technique

Figure 1 shows the major works of Load Balancing Technique. The Load Balancer may be any software or hardware which receives jobs from different users in different locations. The received loads are distributed evenly across all the servers in Data Center.

2.1 Need for Load Balancing

If all the jobs are arrived to the single node, then its queue size is increased and it becomes overloaded. There is a need to balance the load across several nodes, so that every node is in running state but not in overloaded state. The goals are as follows[2]:

- To increase the availability
- To increase the user satisfaction
- To improve the reource utilization ratio
- To minimize the waiting time of job in queue as well as to reduce job execution time
- To improve the overall performance of Cloud environment

2.2 Basic Types of Load Balancing Algorithms

- Depending on the initiator of the algorithm, Load Balancing algorithms can be categorized into three types [3]:

Sender Initiated

Sender identifies that the nodes are overwhelmed so that the sender initiates the execution of Load Balancing algorithm.

Receiver Initiated

The requirement of Load balancing situation can be identified by the receiver/server in cloud and that server initiates the execution of Load Balancing algorithm.

Symmetric

It is the combination of both the sender initiated and receiver initiated types. It takes advantages both types.

- Based on the current state of the system, load balancing algorithms can be divided into two types:

Static Schemes

The current status of the node is not taken into account [6]. All the nodes and their properties are predefined. Based on this prior knowledge, the algorithm works. Since it does not use current system status information, it is less complex and it is easy to implement.

Dynamic Schemes

This type of algorithm is based on the current system information [6]. The algorithm works according to the changes in the state of nodes. Dynamic schemes are expensive one and are very complex to implement but it balances the load in effective manner.

Status Table

Status table [1] is a data structure to maintain the current status of all the nodes in the cloud environment. This information can be used by some of the dynamic scheme algorithms to allocate jobs to the nodes that are not heavily loaded.

3. LOAD BALANCING ALGORITHMS

Several Load Balancing algorithms were proposed. Some of those algorithms are discussed here.

3.1 Dynamic Round Robin Algorithm

It is an extension to the Round Robin algorithm[7]. Normally, each physical machine has number of virtual machines. This algorithm mainly works on reducing the power consumption of physical machine. For that, it uses two rules:

1. The virtual machine in one physical machine has finished its execution but all other virtual machines are still running then if any new virtual machine is arrived means the corresponding physical machine does not accept the new virtual machine. Such physical machines are called "retiring" state physical machines (i.e.), when all the other virtual machines has finished their execution then we can shut down that physical machine.

2. 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 machine. After the successful migration, we can shut down the physical machine.

This algorithm works to save more power by shut down the physical machines. Hence the cost for power consumption is low. But it does not scale up for large data centers.

3.2 Hybrid Algorithm

It is the combination of both Dynamic Round Robin and First - Fit algorithms [1]. It is used to reduce the power consumption by physical machines. The First - Fit algorithm is applied during rush hours to increase resource utilization of physical machines and Dynamic Round Robin algorithm is

applied during non-rush hours to consolidate virtual machines and shut down the physical machines. This algorithm is used for improving Resource allocation and Power Consumption. This algorithm does not scale up for large data centers.

3.3 Equally Spread Current Execution (ESCE) Algorithm

Cloud Manager estimates the size of each job in the queue and looking for the availability of resources for that job[8]. If all the resources are available for that job then immediately the job scheduler allocates that job to the resource. This algorithm is for improving response time and processing time of a job. But it is not fault tolerant and it has the problem of single point of failure.

3.4 Central Load Balancing Policy for Virtual Machine

This algorithm balances load evenly across the distributed systems and cloud computing environments [9]. This policy improves the overall performance of the system but does not consider the systems that are fault-tolerant.

3.5 Enhanced Equally Distributed Load Balancing Algorithm

This algorithm [9] works to distribute the load evenly across all the nodes in cloud environment. It handles the requests with priorities. It is a distributed algorithm by which the load can be distributed not only in a balanced manner but also it allocates the load systematically by checking the counter variable of each data center. After checking, it transfers the load accordingly (i.e.) minimum value of the counter variable will be chosen and the request is handled easily and takes less time and gives maximum throughput. For each arrival of job, the counter variable is increased by 1 and for dispatching a job, the counter variable is decreased by 1. The algorithm is executed by a single central server node. In this method, the central server node gets overhead by receiving more jobs. The algorithm allocates job based on the counter variable and it does not consider the weight of the job.

3.6 Decentralized Content aware Load Balancing Algorithm

It is based on Workload and Client Aware Policy (WCAP) [10]. It uses a Unique and Special Property (USP) to specify the unique and special property of the requests as well as computing nodes. USP helps the scheduler to decide the best suitable node for processing the requests. This strategy is implemented in a decentralized manner with low overhead. By using the content information to narrow down the search, this technique improves the searching performance of the system. It also helps in reducing the idle time of the computing nodes hence improving their utilization.

3.7 Join-Idle-Queue

This algorithm provides large scale load balancing with distributed dispatchers by, first load balancing the idle processors across dispatchers for the availability of idle processors at each dispatcher and then, assigning jobs to the processors to reduce average queue length at each processor[10]. By removing the load balancing work from the critical path of request processing, it effectively reduces the system load, incurs no communication overhead at job arrivals and does not increase actual response time.

3.8 Honeybee Foraging Behavior Algorithm

It is a decentralized honeybee-based load balancing technique that is a nature-inspired algorithm for self-organization [10]. It achieves global load balancing through local server actions. Performance of the system is enhanced with increased system diversity but throughput is not increased with an increase in system size. It is best suited for the conditions where the diverse population of service types is required.

3.9 Min-Min Algorithm

Each job has the execution time and completion time[7]. The cloud manager identifies both the execution time & completion time of each unscheduled job in queue. The job which has minimum completion time is identified and then assigns the job to the processor that has the capability to complete the job within its specified completion time. But larger task has to be waited for long period of time in the queue.

3.10 Max-Min Algorithm

It works as the Min-Min algorithm. But it gives more priority to the larger tasks[11]. The jobs that have large execution time or large completion time are executed first. The problem is that smaller jobs have to be waiting for long time.

3.11 RASA Algorithm

It is the combination of Min-Min and Max-Min Algorithms [11]. The algorithm builds a matrix C where C_{ij} represents the completion time of the task T_i on the resource R_j. If the number of available resources is odd, the Min-Min Algorithm is applied to assign the first task, otherwise the Max-Min algorithm is applied. The remaining tasks are assigned to their appropriate resources by one of the two strategies alternatively. Alternative exchange of Min-Min and Max-Min strategies results in consecutive execution of a small and a large task on different resources and hereby, the waiting time of the small tasks in Max-Min algorithm and the waiting time of the large tasks in Min-Min algorithm are ignored.

3.12 Improved Max-Min Algorithm

It is an extension to the Max-Min Algorithm [12]. The Max-Min algorithm selects the task with the maximum completion time and assigns it to the resource on which achieve minimum execution time. The basic idea of an improved version of Max-Min algorithm assign task with maximum execution time to resource produces minimum complete time rather than original Max-Min assign task with maximum completion time to resource with minimum execution time. It uses the advantages of Max-Min and also covers its disadvantages.

3.13 2-Phase Load Balancing Algorithm

It is the combination of OLB (Opportunistic Load Balancing) and LBMM (Load Balance Min-Min) Scheduling algorithms to utilize better executing efficiency and maintain the load balancing of the system [7]. OLB scheduling algorithm keeps every node in working state to achieve the goal of load balance and LBMM scheduling algorithm is utilized to minimize the execution of time of each task on the node thereby minimizing the overall completion time. This algorithm works to enhance the utilization of resources and enhances the work efficiency.

3.14 Power Aware Load Balancing (PALB) Algorithm

This algorithm is implemented in cluster controller [13]. There is a presence of Job Scheduler whose work is to simulate requests from users for virtual machine instances. The cluster controller maintains the utilization state of each

active compute node and makes decisions on where to instantiate new virtual machines. This algorithm is mainly designed to reduce the power consumption. This algorithm is used to power off the physical machines that are in idle state rather than entering into low power state. But it does not scale up for large cloud data centers.

4. COMPARISON OF LOAD BALANCING ALGORITHMS

Table 1 shows the comparison of LB algorithms which were discussed above.

Table 1. Comparison of LB Algorithms

Algorithm	Description	Advantages
Dynamic Round Robin Algorithm	1. Uses two rules to save the power consumption 2. Works for consolidation of VM	Reduce the power consumption
Hybrid Algorithm	1. Combination of Dynamic Round Robin and First-Fit Algorithm 2. Applied in non-rush hours and rush hours	1. Improved Resource Utilization 2. Reduced Power Consumption
ESCE Algorithm	Estimate the size of job and look for availability of resources	Improved response time and processing time
Central Load Balancing policy for VM	Balances the load evenly	Improves overall performance
Enhanced Equally Distributed Load Balancing Algorithm	Based on the counter variable, the job is allocated by Central Server	1. Computing Resource is distributed efficiently and fairly 2. Reduces request to response ratio
Decentralized Content Aware Load Balancing Algorithm	1. Uses Unique and Special Property(USP) of nodes 2. Uses content information to narrow down the search	1. Improves the searching performance hence increasing overall performance 2. Reduces idle time of nodes
Join-Idle Queue Algorithm	1. Assigns idle processors to dispatchers for the availability of idle processors 2. Then assigns jobs to processors to reduce average queue length	1. Reduces system load 2. Less communication overhead
Honeybee Foraging Behavior	Achieves global load balancing through local server actions	Improved scalability
Min-Min Algorithm	1. Estimates minimum execution time and minimum Completion time 2. Jobs having minimum completion time is executed first	Smaller tasks are executed quickly
Max-Min Algorithm	1. Same as Min-Min 2. Gives more priority to larger tasks than smaller one	Larger tasks are executed quickly and efficiently
RASA Algorithm	Combination of both Min-Min and Max-Min Algorithms	1. Efficient resource allocation 2. Minimum execution time
Improved Max-Min Algorithm	1. Improved version of Max-Min Algorithm 2. Assigns task with minimum execution time	Scheduling jobs effectively
2-Phase Load Balancing Algorithm	1. Uses OLB to keep each node busy 2. Uses LBMM to achieve minimum execution time of each job	1. Efficient utilization of resources 2. Enhances work efficiency
PALB Algorithm	1. Implemented in Cluster Controller 2. Use Job Scheduler to simulate requests from users for virtual machine instances	Physical Machines that are in idle state are move to power off state to conserve energy

5. PROPOSED SYSTEM

Existing Load balancing algorithms have some drawbacks in improving overall performance of the cloud environment. Still there is a problem of overloading nodes in the Cloud environment. It is very difficult to manage entire cloud environment. Hence the proposed idea is to divide the entire cloud environment into several partitions based on its geographical locations [1]. Now the Load balancing algorithm can be applied only to the partitions, not to the entire cloud.

Fig 2 shows the cloud environment after partitioning is done. The load balancing algorithm is applied to each partition in order to avoid overloading of nodes.

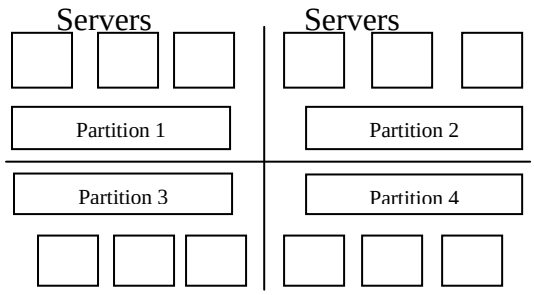

Fig 2: Partitioned Cloud Environment

For balancing load in cloud partition model [1] there are two important components needed:

- Load Balancer
- Main Controller

Load Balancer is associated with each partition whose work is to maintain the state information and is to be updated in periodic intervals. Whenever the controller receiving a job it has to communicate with each partition to collect state information. Then the job is allocated to the partition if it is in idle or normal state. After assigning job to the partition, the balancer has to update the status information of each node in that partition. Based on this information, the job is allocated to the nodes.

Main controller receives all the jobs that arrive from the cloud. Whenever the main controller receives the job it has to decide, which partition to receive the job. Each partition has the state information associated with it. It may be in idle state, normal state or heavily loaded state.

The state of the node in particular partition is set by considering several parameters of that node. The parameter may be static or dynamic.

- Static parameters include number of CPUs, memory size and speed of the processor or CPU.
- Dynamic parameters include CPU utilization ratio and memory utilization ratio.

Load of the node is calculated by these parameters. The algorithm is to be designed for the nodes that are idle or normal and it has to update the status information of each node periodically.

Load Balancer in each partition maintains the status table. The status table contains information about the load of all the nodes in that partition. Based on the information in the status table, Load Balancer decides the status of the partition. This table is updated by a Load Balancer periodically. There is a chance for inconsistent of the data in the status table. For better efficiency, balancer maintains the two status table and each of which are associated set by the "Flag". The Flag is set by either "Read" or Write". One table is set by read whereas another one is set by write. The information in table which is set by "read" is always used by the algorithm. "write" denotes that the table information is being updated and it may not be correct. Once the information is updated in the "write" table then the flag is set to "read".

6. FUTURE WORK

The future work is to develop two algorithms, one for the partitions of Cloud environment that are in idle state and another one for the partitions that are in normal states. Switching mechanism is needed for applying these two alternative algorithms. If the partition is in idle state one simple algorithm is to be used and later the same partition can became normal state and alternative algorithm is to be used. The algorithm designed for normal state partitions should be more efficient so that it avoids the partition becoming overloaded.

7. CONCLUSION

Though there are several issues in cloud environment, it has been widely adopted by many organizations and industries. Researchers are doing many works to resolve those issues. For Load Balancing issue, the solution is to develop suitable algorithms that balance the load across the partitioned cloud environment. Two developed algorithms should work accordingly as the partition status changes. It reduces the server overhead, increase throughput, increase performance, reduce server power consumption and also distribute the load across nodes.

8. REFERNECES

[1] Gaochao Xu, Junjie Pang and Xiaodong Fu, "A Load Balancing Model Based on Cloud Partitioning for the Public Load", IEEE Transactions on Cloud Computing, 2013.

[2] Zenon Chaczko, Venkatesh Mahadevan, Shahrzad Aslanzadeh and Christopher Mcdermid, "Availability and Load Balancing in Cloud Computing", International Conference on Computer and Software modeling IPCSI, 2011.

[3] Ratan Mishra and Anant Jaiswal, "Ant Colony Optimization: A solution of Load Balancing in Cloud", International Journal of Web & Semantic Technology (IJWesT), April 2012.

[4] Liang Liu, Hao Wang, Xue Liu, Xing Jin, WenBo He, QingBo Wang, Ying Chen, "Green Cloud: A New Architecture for Green Data Center", ACM Journal,2009.

[5] V.Srimathi, D.Hemalatha, R.Balachander, "Green Cloud Environmental Infrastructure" , International Journal of Engineering And Computer Science, December 2012.

[6] Venubabu Kunamneni, "Dynamic Load Balancing for the cloud", International Journal of Computer Science and Electrical Engineering, 2012.

[7] Karanpreet Kaur, Ashima Narang, Kuldeep Kaur, "Load Balancing Techniques of Cloud Computing", International Journal of Mathematics and Computer Research, April 2013.

[8] Dr.Hemant S.Mahalle, Prof. Parag R. Kaveri, Dr. Vinay Chavan, "Load Balancing on Cloud Data Centers", International Journal of Advanced Research in Computer Science and Software Engineering, January 2013.

[9] Shreyas Mulay, Sanjay Jain, "Enhanced Equally Distributed Load Balancing Algorithm for Cloud Computing" International Journal of Research in Engineering and Technology, June 2013.

[10] Nidhi Jain Kansal, Inderveer Chana, "Cloud Load Balancing Techniques: A Step Towards Green Computing", International Journal of Computer Science, January 2012.

[11] S.Mohana Priya, B.Subramani, "A New Approach for Load Balancing in Cloud Computing", International Journal of Engineering and Computer Science, May 2013.

[12] O.M.Elzeki, M.Z.Reshad, M.A.Elsoud, "Improved Max-Min Algorithm in Cloud Computing", International Journal of Computer Applications, July 2012.

[13] Jeffrey M. Galloway, Karl L.Smith, Susan S. Vrbsky, "Power Aware Load Balancing for Cloud Computing", World Congress on Engineering and Computer Science, 2011.