Load Distribute Across the Cloud Server: Challenges and Algorithms

Durga Patel
CSE Department
MANIT Bhopal
India

Veena Mishra
CSE Department
MANIT Bhopal
India

R. K. Pateriya
CSE Department
MANIT Bhopal
India

ABSTRACT

Cloud computing has become as an approved computing model to challenge processing the large volume of data which utilizes clusters of commodity computers. . Load balancing helps to utilize 100% of the resource provided by cloud service provider in a cloud scenario. This is a procedure which actually implements the concepts of task scheduling algorithm, nowadays the load imbalance virtual machine accelerate several networks, storage and energy consumption related irregularities. Due to the extensive operation of cloud computing resources, it became one of the prime goals of the cloud provider to utilize the computing resources efficiently and optimize the physical resource utilization. Numerous algorithms were recommended to suggest potent procedure and algorithms for allocating the user's demand to accessible cloud nodes. These all methodologies wish to gain complete performance of the cloud services and make the user more satisfying, authentic, sensible and well-organized services. Load balancing is implemented by various algorithms, few of these algorithms are promoted because of better energy efficiency, resource utilization and they are fault tolerant too. This paper explored different existing load balancing approaches in a cloud environment along with their anomalies.

Keywords

Cloud computing, Load balancing, virtual machine.

1. INTRODUCTION

Distributed computing along with Grid computing are the basic methods for a new virtualizes constitution of computing environment referred to as cloud computing, which in turn results to revoke the marketing and working costs and enabling on-demand computing [1].

Cloud computing is the technology which is gaining popularity as the number of technologies is growing. It is genuinely convenient for an entire enterprise to develop without holding resources and technologies, physically. This technology attracts a lot of investment in infrastructure which is not feasible for all the enterprise. To overcome these quandaries clouds provide accommodations on-demand substratum that can be accessed over the internet on pay-per-use [2].

The cloud computing is used by industries, companies, and the incorporate world because it provides flexible computing power, easy storage & retrieval of data files. For satisfactory levels of performance several optimization techniques and streamline operations are implemented. Consequently, it is significant to review specific areas in the cloud to ameliorate the computerized information utilization in addition to download performance for the users. One of the consequential issue associated with this field is dynamic load balancing or load scheduling. Load balancing algorithms were scrutinized strongly in distinct environments under Cloud scenario, several supplemental challenges pop up and these challenges are addressed with suitable solution. Once the effort and Request process is finished efficiently, tasks can be assigned efficiently to the cloud nodes. Whereas having the ability to abide by the numerous affecting constraints namely, heterogeneity and high communication delays [3].

Load Balancing algorithms are separated into two categories, static load balancing and dynamic load balancing. Static algorithms are largely opportune for integrated and stable environments and might engender excellent ends up in cloud environment. However, these methods are not versatile enough to reflect the dynamic changes to the attributes are made throughout the execution time. Dynamic algorithms are ideally versatile and consider different types of attributes for load and threshold calculation within the program, before and around run-time. These algorithms will adapt to changes and apply higher ends up to in collective and dynamic environments. As a result, some of these algorithms could become inefficient and cause more overhead resulting in an overall degradation of the performance. In this paper, we commented on some trending advanced load balancing algorithms developed particularly to suit the cloud computing environments. We give a comprehensive survey of these algorithms and analyzed their properties. In this paper, we associate these algorithms with each other according to their accompanying properties: the number of attributes, the overall network load, and time series [4].

2. LOAD BALANCING IN CLOUD COMPUTING

Nowadays, due to the enormous popularity of cloud, the organization is becoming aware of the power consumed by resources. Reducing capability consumption has been an essential specification for cloud environments not only to decrease operating cost, but further improve the system reliability [4].

Load Balancing divides the workload between two or more VMs(placed within hosts) so that maximum throughput is achieved without incorporating additional host keeping in mind cost as a factor, in return all users gain faster services. Load balancing one of the central issue in cloud scenario, which is attracting researchers to implement better approach. It is an approach in which the workload on the resources of a node affects respective resources on the distinct node in a network without disturbing the running task. On the other hand, The Load balancing technique distributes the excess dynamic local workload equally among all the nodes. The load balancing is one of the significant and existing conceptions in cloud computing issues and Proper load balancing leads to utilization of the available resources optimally, thereby minimizing the consumption of resource. Thus, the load needs to be distributed to all including the resources in a cloud-based construction, each resource permanently does the equal amount of responsibility. The load balancer decides which internet server should produce the request. The load balancer uses a distinct scheduling algorithm to demonstrate which server should manage and forward the request onto the operating server [4].

3. LOAD BALANCING AND ITS NEED

The proper load balancing gives better resource utilization, thus minimizing the consumption of resources which further results in higher user satisfaction [2] [4]. There are primary goals of Load Balancing is given as below.

1) Improve system performance that makes algorithm cost efficient.
2) Shows effective result by decrease response time.
3) Reducing the energy consumption and decrease carbon emission that helps in green computing.
4) Gives the fault tolerant system by creating a backup.

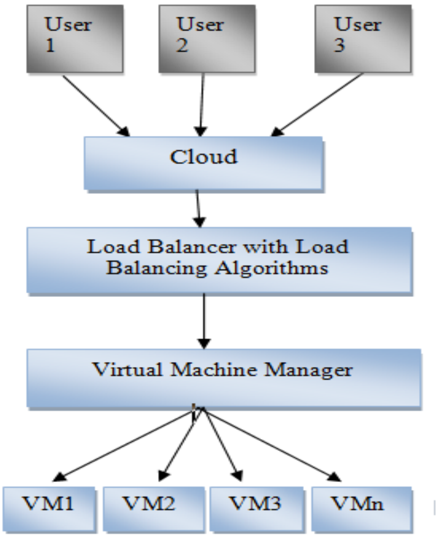

Fig I. Balancing the Load in Cloud Computing

4. ROLE OF FUTURE PREDICATION MODEL IN LOAD BALANCING

Several load balancing approaches have been proposed in the past few decades. All of these approaches use some threshold value to define the overloaded and underloded situations. When the load on the PM is below or above these threshold values some VM will be migrated. Since the load on the PM is changing very frequently, so migration may be triggered due to the temporary peak load. Hence, migrating VM based on these threshold values may increase the numbers of migration.

Prediction model can help the provider to avoid these un-necessary migrations. Before trigging the migration, first predict the load and if it is still overloaded or underloaded them trigger the migration. Hence, the use of prediction model can minimize the amount of migration and improve the system performance. [5]

5. CLASSIFICATION OF LOAD BALANCING ALGORITHMS

There are three categories of load balancing based on its initiation [6]. They are-

A. Sender-initiated: If the load balancing algorithm is initialized by the sender.

B. Receiver-initiated: If the load balancing algorithm is initiated by the receiver.

C. Symmetric: It is the combination of both sender initiated and receiver initiated.

According to current condition of system another two types of load balancing algorithm are there

A. Static Load Balancing:

In the static load balancing past information of resources, such as capacity, processing power, location etc. is required. Any change in load required at runtime is not possible. Though it is easy to implement, but not much efficient for various cloud environments, especially where it is not feasible to fix the requirements and resources. Static load balancing algorithms balance the load based on the information about the average regulation of the system [2].The capacity of the system is fixed at the starting of the execution time. It has an advantage of minimizing communication delays and reduced execution time [3] [4].

B. Dynamic Load Balancing:

Dynamic algorithms are skilled for the scenario to what place heterogeneous basic materials are present. In this, no prior development is required. The provisioning is done by the state-of-the-art arrangement of the system. The requirements may twist at the run-time by every user. Dynamic algorithms are difficult to implement with cent percent accuracy, as we don't know the scenario of runtime [2] [4]. For better implementation cloud should be configured in a planned way from scratch so that load at particular time frame is known in prior.

C. Centralized Load Balancing:

In the centralized scheme all the resource allocation and de-allocation process is done by the one central node called coordinator. The coordinator works as the administrator and stores all the information about the network and implemented the algorithm as per the requirement. [2]

C. Distributed Load Balancing

In these algorithms no-hit node is responsible for load distribution. There are a number of coordinators in different domains which are used to recognize the present network and responsible for load balancing. Each and every node in domain manages the local information base and pass it to the coordinator, which creates a global developer base to portray the status of the network. [2]

D. Hierarchical Load Balancing

In these algorithms cloud network is distributed in levels and each level participates in load balancing. These load balancing techniques consistently work in a master - slave mode. This campaign visualizes the cloud network in the compresses of the tree structure. Where the parent maintains the knowledge base of all the children and passes on to its own parent. Finally, from information gathered by all the parent nodes, the order is obligated. [3]

6. DISCUSSION AND COMPARISON BETWEEN DIFFERENT ALGORITHM

A. INS (Index Name Server):

It integrates de-duplication and access point assignment optimization. There are multiple parameters for assigning optimum point. It helps in getting the appropriate server for task completion under cloud[2][3].

B. WLC (weighted least connection):

WLC assigns task to all the nodes based on the number of connections built to particular node. This is carried out by comparing the SUM of connections of every node in the cloud with the before mentioned the Lord has assigned to the node by the whole of the least number of connections [2][3]. WLC doesn’t demand the capabilities of each node a well known as processing assist, bandwidth and storage power.

C. ESWLC (Exponential Smooth Forecast based on Weighted Least Connection):

Adding time series and trials to WLC leads to improved WLC, named as ESWLC. That is ESWLC assigns an indisputable responsibility to a node after having the list of tasks assigned to that node after visualizing the node abilities. ESWLC makes the decision based on the past inputs of the node's CPU capacity, memory, the number of connections and currently used disk space. Then ESWLC predicts the suitable node for job allocation based on exponential smoothing [2] [3].

D. DDFTP (dual direction downloading algorithm from FTP servers):

The algorithm proposes a dual direction downloading algorithm from FTP servers. DDFTP works by partitioning a file into m/2 partitions, with original file size equal to m. Then, each server node start processing assigned task based on an actual pattern [3].

E. MapReduce:

MapReduce has some goals for cloud environment that are - commodity nodes (cheap, but unreliable),commodity network (low bandwidth) and automatic fault-tolerance

F. OLB (Opportunistic Load Balancing algorithm):

OLB gives static load balancing environment that has the direction of protection every host in the cloud. LBMM enhanced version of OLB by joining a layered architecture to the algorithm.

G. LBMM (Load Balancing Min-Min):

LBMM consist a three layer load balancing framework. It also uses the functions of OLB. (OLB). [2] [3] [7]

H. Min-min Algorithm:

The traditional Min-Min algorithm is a very simple and feassible for implementation, that gives a better balancing of load. The biggest obstacle of min-min: [2] [4]

- It does not concern the workload of each resource.
- It is a load imbalance.

I. LBIMM (Improved Load Balancing Min-Min):

An algorithm is confirmed at the bottom of Min-Min algorithm in the term to minimize the makespan and revive the resource employment (LBIMM). [2] [4]

J. PA-LBIMM (Priority-aware-LBIMM):

User-priority aware is instructed PA-LBIMM so as to execute the expected guarantees that VIP customers can enjoy better function than the other ordinary users. [7]

K. Round-Robin algorithm:

This algorithm uses the production of time quantum or slices. Here the time

Is distributed into separate slices and each node is if a particular presage quantum or time interval and in this quantum, the node will perform its operations. [1] [2]

L. Throttle Load Balancing Algorithm:

Throttled algorithm is overall based on a virtual machine. In this algorithm, the load balancer maintains a register, table of virtual machines as well as their states (Available or Busy). [4]

M. Ant Colony Optimization (ACO):

In this approach, incoming ants update all the entries in the pheromone table of the hub. For example, when ants travelling from source to destination will update tall the entry in the pheromone table. If an ant is at a fine point when there is no pheromone, it makes a random decision. Here tasks are placed to server with non-preemptive scheduling so waiting time period of high priority task are increased.

N. Modified ACO:

This algorithm used for minimizes the make span. Modification means the basic pheromone updating formula. It results better utilization of resource, but overhead is higher than the ant colony optimization [8].

O. Bees Algorithm (BA):

In this approach, algorithm similar to neighborhood search is combined with random search and used for both combinatorial and functional optimization**.** In Bees Algorithm, scout bees are moving randomly from source to destination for searching the food. When they find the food resource with quality threshold the scout bees return and dance. Dancing is a signal for colony communication and it contains information related to food sources like direction from the hive, quality and quantity [9].

P. Honey Bees:

This algorithm implements the behavior of honey bee that follows a pattern of finding reap food. Bees are categorized into two classes namely, scout bees and forager bees. The scout bees find the food sources, and then they come back to inform, by employing a dance referred to as vibration dance. The aim of dance provides the concept of the standard and/or amount of food, additionally its distance from the source(beehive).

Table I. Merits and De-merits of Load Balancing Algorithms

Algorithm	**Environment**	**Merits**	**De-merits**
Round Robin Algorithm [1] [2]	Dynamic	• It is based on random sampling. • Used to reduce cost and time	• It is a Lesser resource utilization algorithm.
Min -Min algorithm [2]	Dynamic	• Simple, well organized algorithm so that begin an ameliorate lineup that reduce the total execution time of tasks.	• This algorithm did not consider the workload of an individual asset.
WLC (Weighted Least Connection) [2] [3]	Dynamic	• It is place on comparison of the SUM of connection in each node	• It has less processing speed, storage capacity, bandwidth.
ESWLC (Exponential Smooth Forecast Based On Weighted Least Connection) [2] [3]	Dynamic	• Improved version of WLC by taking into account the time series& trials.	• Take more processing time than WLC.
DDFTP (Dual Direction Downloading Algorithm From FTP Servers) [3]	Dynamic	• Fast and efficient technique. • Reduce network communication and network overhead.	• Full replication of data files.
MapReduce [3]	Static	• Give less amount of overhead for the reduce tasks	• Increase processing time • Reduce tasks capacity is not taken into consideration
INS (Index Name Server) [2] [3]	Dynamic	• Integrate de-duplication & access point selection optimization. • Proved to handle some sort of dynamic load balancing	• Some parameter which is involvedcalculating optimum selection point.
Throttle Load Balancing Algorithm [4]	Dynamic	• Throttled algorithm is completely based on the virtual machine. • The load balancer maintains an index, table of virtual machines as well as their states (Available or Busy).	• Need higher maintains capability for virtual machines
LBIMM (Load Balancing Improved Min- Min Algorithm) [2] [4]	Dynamic	• Reduce the makespan and increase the resource utilization.	• Load Imbalance.
PA-LBIMM (User-Priority Aware Load Balancing Algorithm) [2] [4]	Dynamic	• Capable of decreasing completion time of tasks, improving load balance of resources and gain better overall performance than Min-Min algorithm.	• Sometimes it is difficult to group the tasks with the highest priority requirement and lower priority requirement.
Ant Colony Optimization [6]	Dynamic	• Ensures availability, gives efficient resource utilization, • Increases the number of requests and processed by cloud and decreases the time needed to process multiple requests. • Maximize the throughput because of optimal resource	• If the data center architecture is more complex than this algorithm is also complex. • Fault tolerance is not good. • Overhead increases.

		utilization.	
Particle Swarm Optimization [7]	Dynamic	• Increases the Quality Of Service (QOS). • Minimize the span time.	• Space not optimized.
CLBDM (Central Load Balancing Decision Model) [7]	Static	• It Solves all issues of the Round Robin Algorithm. • Reduces the human administrator by automated task forwarding.	• Inherits Round Robin issues • Single point of failure (if CLBDM fails, then the whole task fails) • The threshold might not be used for all cases.
Improved Particle Swarm Optimization [7] [8]	Dynamic	• Space is optimized. • Minimize the span time	• Fault tolerance is not good.
Neighbor Awareness Random Sampling [9]	Dynamic	• It improves the probability of assigning jobs to lightly loaded nodes.	• Performance degrades as population diversity increases.
Modified ACO framework [10]	Dynamic	• Better utilization of their source.	• Fault tolerance is not good. • Overhead high
BA (Bees Algorithm) [10]	Dynamic	• Helpful for choosing the optimal solution.	• The number of tunable parameters used. However, it is possible to set the parameter values by conducting trial values
Honey Bee [10]	Dynamic	• Maximum throughput. • Reduced Waiting Time.	• Throughputdoes not increase with system size.

Then forager bees go behind the Scout Bees to the placement that they found food and so begin to reap it. They came back to the Beehive and do a tremble for different bees within the hive signifying what quantity of food is left. This concept is used and local server action is recorded for analysis, to provide better system performance and resource utilization [8].

Q. Particle Swarm Optimization (PSO):

Particle swarm optimization is a technique which depends on the development and knowledge of swarms. PSO applies the idea of social collaboration for critical thinking. It is exceptionally strong, stochastic and improved strategy

R. Improved particle Swarm Optimization (IPSO):

IPSO mechanism takes the characteristics of complex networks into consideration to establish a corresponding resource-task allocation model. This technique is proposed to gain resource load balancing optimization more than the PSO technique and improve the resource utilization in the cloud network.

Table I:It demonstrates an examination among all the described algorithm. Furthermore, this examination demonstrates the positives and negative side of every algorithm and the earth in which they are implemented

7. CHALLENGES IN CLOUD COMPUTING, LOAD BALANCING

Here we contend the challenges to be focused when attempting to present an optimal consolidation to the presence of load balancing in the Cloud Computing environment.[4][2]

*1. Spatial Distribution of the Cloud Nodes :*In the cloud computing there is a need to design a load balancing algorithm for spatially distributed nodes which tolerate high delays. [4]

2. Algorithm Complexity: It is necessary that all the load balancing algorithm has to be less complex in term of operations and implementations. So, the complexity of all balancing algorithm is less which leads more positive response.

3. Failure point: Cloud computing use, distributed or centralized algorithms for load balancing. But in Centralized algorithm, the many issues is the one controller for the whole system. So, in such cases distributed algorithm provides a better approach and overcome the point of failure issue. But they are much more complex and needs more coordination.

4. Storage/ Replication: Full replication algorithms demand higher costs for more storage. However, partial replication algorithms could save parts of the message sets in each node (with an actual level of overlap) based on each node's capability such as processing capability and capacity. This could supervise to better utilization.

8. CONCLUSION

In the cloud computing environment adjustment of the node in network is the biggest announcement. There are a number of algorithms introduced to balancing the load over the network. Distinct scheduling and vigorous provisioning are playing a significant part in allocating the assets to assignments in the cloud system. These scheduling algorithms deal with the dynamic provincial task at hand similarly over every one of the hubs to accomplish higher resource utilization and user assistance proportion by ensuring that each registering asset is isolated conveniently. Cost and time are the key challenges for every IT engineer to develop products that can enhance the service performance in the cloud. This paper discusses the several existing load balancing algorithms in distinct networking environments. These algorithms are analyzed on particular load balancing parameters and strategies. The analysis is done to obtain the most of the algorithm have a focus on proper resource utilization, balance load, reduce cost and increase response time.

9. REFERENCES

[1] Divya Chaudhary, "Analytical Study of Load Scheduling Algorithms in Cloud Computing ", proceeding of 2014 International Conference on Parallel, Distributed and Grid Computing, pp7-12, 2014.

[2] Garima Rastogi, "Analytical Literature Survey on Existing Load Balancing Schemes in Cloud Computing", IEEE, pp1506-1510, 2015.

[3] Klaithem Al Nuaimi," A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms ", NCCA, 2012, International Symposium on 2012, IEEE, pp137-142, 2012.

[4] Hamid Shoja, "A Comparative Survey On Load Balancing Algorithms In Cloud Computing ", ICCCNT, IEEE, pp1-5,2014.

[5] Yingchi Mao, "Adaptive Load Balancing Algorithm Based on Prediction Model in Cloud Computing", proceeding of the second international conference on innovative computing cloud computing, pp165, 2015.

[6] G.Punetha Sharmila, "Survey Onfault Tolerant –Load Balancing Algorithmsin Cloud Computing ", Proceeding of 2nd International Conference On Electronics And Communication System (Icecs 2015), IEEE,pp1715-1720,2015.

[7] Hank Chen, Professor Frank Wang," User-Priority Guided Min-Min Scheduling Algorithm For Load Balancing in Cloud Computing ", IEEE, pp1-8,2013.

[8] Ekta Gupta, Vidya Deshpande," A Technique Based on Ant Colony Optimization for Load Balancing in Cloud Data Center ", 014 13th International Conference on Information Technology, IEEE, pp12-17,2014.

[9] G. Shobana," Nature Inspired Preemptive Task Scheduling for Load Balancing in Cloud Datacenter ", Proceedings of ICICES 2014, IEEE, pp1-6,2014.

[10] Shahrzad Aslanzadeh, Zenon Chaczko, "Load Balancing Optimization in Cloud Computing: Applying Endocrine-Particle Swarm Optimization", IEEE, pp165-169,2015.

[11] KaiPan, JiaqiChen, "Load Balancing in Cloud Computing Environment Based on An Improved Particle Swarm Optimization", IEEE, pp595-598,2015.

[12] Ariharan v," Neighbor-Aware Random Sampling (NARS) algorithm for load balancing in Cloud computing", IEEE, pp1-5,2015.