International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 02 | Feb - 2022 ISSN: 2582-3930

Load Balancing and its Algorithms in Cloud Computing Environment: A Survey

Vedanth M
B.E
Department of ISE
Alva's Institute of Engineering and Technology, Mijar, Moodbidri

Pradeep Nayak
Assistant Professor
Department of ISE
Alva's Institute of Engineering and Technology, Mijar, Moodbidri

Ranjitha
B.E
Department of ISE
Alva's Institute of Engineering and Technology, Mijar, Moodbidri

Mahantesh G
B.E
Department of ISE
Alva's Institute of Engineering and Technology, Mijar, Moodbidri

Namitha D
B.E
Department of ISE
Alva's Institute of Engineering and Technology, Mijar, Moodbidri

Abstract – In this paper, the performance of cloud computing system is greatly influenced by the problem of load balancing. The complexity class of the load balancing problem with respect to the complexity class belongs to the NP-system complete which involves intensely huge search space with huge number of potential solutions and also to find the optimal solution, it takes longer time. Based on these circumstances, there is no methodology to solve the problem. A load balancer is like a "traffic cop" which directs the traffic to avoid a situation where few servers are overloaded while few are idle. Proper load balancing can ensure maximum throughput in minimum response time. This paper is a critical review of all the existing techniques of load balancing and comparison between them.

Keywords – cloud computing, load balancing, load balancing algorithms (static and dynamic), smart load balancing.

I. INTRODUCTION

In the field of network technology, the cloud computing technology is showing phenomenal growth due to the advancement of communication technology explosive use of Internet and solve large-scale problems. Every other person wants to use these services as it reduces the cost of hardware, provides 24/7 access from any corner of the world and also ensures data security.

A cloud consists of several computers or servers (nodes) connected to form a cluster. When too many random requests are generated by clients, it is apparent that some of these servers may get overloaded. This overloading of servers can deteriorate the performance of our cloud. Load imbalance may cause system bottleneck that is when workloads arrive too quickly for the processors to handle. The ineffectiveness brought about by the bottleneck often creates delays. These conditions are raising a high demand of load balancers or effective load balancing techniques. Effective load balancing results in minimizing the downtime, implementing fail-overs, enabling scalability, avoiding bottlenecks and over-provisioning.

The servers are legitimately assembled into clusters and the task of load balancing is disseminated among these clusters or groups. The load is allocated to the servers present in that cluster. Servers belonging to a cluster offer same services. Various load balancing approaches have been actualized for the cloud condition to provide productive load distribution.

This paper is requested as follows. Area II depicts what is load balancing and a portion of its estimating parameters. Area III depicts the classification of load balancing algorithms. In segment IV comparison among various load balancing calculation is given.

II. LOAD BALANCING

The total number of cloud users through Internet is growing at an alarming rate so the need of balancing the traffic on the enterprise applications in order to provide high performance and resources availability. Well, there are enormous numbers of load balancing algorithms existing in the cloud system, but almost all the algorithm face some issues or problem. The main objective of load balancing is to achieve an effective path for the requests among the web servers with a minimum response time. And the other challenge is that the firm faces are the high cost i.e managing of web servers where load balancer can be disseminated.

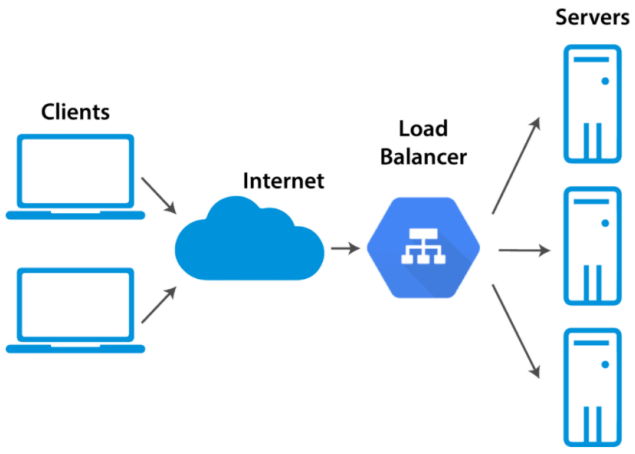

Fig1 – Load Balancing in Cloud Computing

Load balancing techniques can be evaluated based on some parameters Kanakala R. et al. (2015, 2017) in their work have mentioned some parameters like throughput, response time, performance, scalability, resource utilization and fault tolerance.

- Response time (RT): It is the time required by the system to respond a task. In other words, it is the sum of transmission time, waiting time, and service time. Thus, the system performance is inversely proportional to the response time. The optimal response time results in a better make span value.
- Associated Cost (AC): This cost depends on the percentage of resource utilization. For example, the services offered by EC2 can reduce the entire cost up to 49 % while the resource is fully utilized ([8]). The cloud user tries to depreciate the cost of resource provisioning by degrading the on-demand resource cost and over-subscribed resource cost of over provisioning and under-provisioning [9].
- Energy Consumption (EC): The energy consumption of a cloud system is the amount of energy absorbed by all ICT devices connected in the system [35]. Three kinds of devices to calculate the energy consumption are personal terminals (desktop, laptop, handsets, etc.), networking nodes (routers, switches, hubs, etc.), local servers (application servers).

There are four different solutions to conserve energy, and those are the use of energy-efficient hardware, application of energy-aware scheduling technique, power-minimization in the server cluster, and power-minimization in wired and wireless networks [36]. The estimation of energy consumption of the system is presented in equation (4) based two state virtual

A good load balancing algorithm aims to:

1. To maximize throughput.
2. To reduce response time.
3. To optimize resource utilization.
4. To avoid overload of any single resource.

III. CLASSIFICATION OF LOAD BALANCING ALGORITHMS

There are a few Load Balancing algorithms which are extensively characterized into two classifications dependent on framework load:-

1. Static Load Balancing
2. Dynamic Load Balancing

Static Load Balancing: It is an approach where load balancing is achieved by providing prior information about the system. The performance of the node is determined at the beginning of execution. Nodes calculate the work allotted to them and submit the result to the remote node. Then based on performance workload is distributed in start without considering the current load. Static load balancing methods are non-preemptive i.e when the load is assigned to one node it cannot be moved to another node. This fundamental bit of leeway of this methodology is that it requires less correspondence thus diminishes the execution time the principle disadvantage of this methodology is that it doesn't think about the present condition of the system while settling on allocation choices.

Some static load balancing algorithms are round robin, min-min, max-min and randomized load balancing algorithm.

A. ROUND ROBIN ALGORITHM

It is a load balancing technique where a DNS server rotates which out of several servers IP addresses is to be used. It has a list of IP addresses and provides a different IP address for each successive request, returning to the first one after the last has been provided. The main advantage of using round robin load balancing is its simple implementation. But it does not always result in the most efficient traffic distribution, because many round robin load balancers assume that all servers are the same.

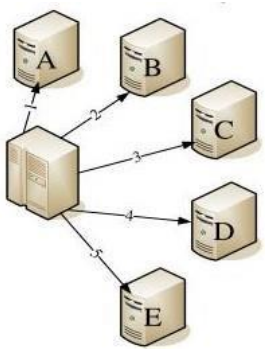

Fig2 – Round Robin Load Balancing Technique

B. MIN-MIN ALGORITHM

The cloud computing user identifies both the execution and completion time of the unassigned tasks waiting in a queue. This algorithm of Min-Min comes under the static load balancing algorithm as the parameters related to the job are known in advance. In this type of algorithm, the cloud user first focuses with the jobs having minimum execution time by assigning them to the existing processors according to the capability of completing the job in specified completion time. The job requests having longer or maximum execution time has to wait for its turn for the unspecific period of time. Until all the tasks are assigned in the processor, the assigned tasks are updated in the system and the task is then removed from the waiting queue as it has been executed. This algorithm performs betters when the numbers of jobs having small execution time is more than the jobs having large execution time. The vital demerit of this algorithm is that it can lead to cloud disaster.

C. MAX-MIN ALGORITHM

The working of the max-min algorithm [11] is similar to the min-min algorithm the only difference is that in max-min algorithm priority is given to the task with highest completion time (HCT). Once the lengthy tasks are completed, the task with minimum execution time is assigned to the processor. This algorithm also maintains time sequences of the task engaged and keep informed the execution time periodically to the load balancers as well as processor

D. RANDOMIZED ALGORITHM

It is another approach to load balancing in this a list of server IPs is delivered to the client, and then the client can randomly select the IP from the list. The client-side random load balancing tends to provide better distribution of load as compared to round robin.

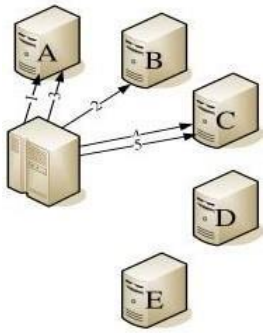

Fig3 – Randomized Load Balancing Technique

Dynamic Load Balancing: These algorithms watch changes on the system outstanding task at hand and redistribute the work as needs be. Dynamic algorithms can have three distinctive controlling structures: concentrated, distributed, or semi-distributed. In brought together load circulation, a solitary node (focal hub) is liable for all load dispersion in the system. In appropriated the duty is similarly isolated among all nodes. Though in a semi-distributed the system is portioned into groups where each group is centralized.

This algorithm comprises of three strategies: transfer strategy, location strategy and information strategy.

a. Transfer strategy - chooses which tasks are qualified for move to different nodes for handling.
b. Location strategy – proposes a remote node to execute a transferred task.
c. Information strategy - is the data place for the load balancing algorithm. It provides area and transfer strategies to each node.

Some dynamic load balancing algorithms are ESCEA, throttled, biased random, token rating, honey bee foraging behaviour and ANT COLONY load balancing algorithm.

A. ESCE (Equally Spread Current Execution) ALGORITHM

ESCE is also called as active VM (Virtual Machine) load balancing algorithm. A.A. jaiswal, et al. (2014) and surbhi kapoor, et al. (2015) have demonstrated that this algorithm equally distributes the workload on each VM. It maintains a list of virtual machines and the number of requests already allotted to that virtual machine. When a new client request is received DCC (Data Centre Controller) asks ESCE load balancer for next VM allocation. It will then scan the list from

starting to get a least loaded VM. If there is more than one virtual machine present then the first one is chosen and its id is sent to DCC. DCC allots the client request to the selected VM and the list is updated by an increasing number of allocation counts of that particular VM. The main disadvantage of ESCE load balancing is its high computational overhead.

B. THROTTLED ALGORITHM

This load balancing algorithm is generally appropriate for virtual machines. In this load balancer keeps up the list of entire virtual machines in the system and their states whether they are accessible or occupied (busy). On getting a customer demand, it checks the ordering table. If the VM is accessible, at that point the task is appointed to that machine. If the fitting VM isn't discovered, at that point it restores a worth 1 to DCC and places the request in the line. After finishing the processing of the designated demand, reaction cloudlets are sent to DCC which consequently send a notice for de-allotment. The ordering table is refreshed after every assignment and de-allotment of the resource. This algorithm has better resource use yet checks the whole list from the earliest starting point.

C. BIASED RANDOM ALGORITHM

Biased random algorithm adjusts the load through random sampling of framework area. In this virtual chart of the framework is made. In a directed graph, every node signifies a vertex and each in-degree signifies free resources of that node. The load is distributed by the load balancer to the node which has at any rate one in-degree. The in-level of the node is increased and decremented when the undertaking is finished and when the assignment is designated separately. This is done through the procedure of random sampling. This algorithm is appropriate for huge networks.

D. HONEY BEE FORAGING BEHAVIOUR ALGORITHM

This algorithm is based on the behaviour of real honey bees in finding their food sources. There are three groups of bees: employed bees, onlookers and scouts. Every employed bee goes to a food source to determine a neighbouring source, then evaluates its nectar amount and dances in the hive. Each onlooker chooses one of the sources depending on the dances of the employed bees and then goes to that source.

E. ANT COLONY ALGORITHM

We have gone through the work of A.A. jaiswal et al. (2014) and Sarmila G.Punetha, et al. (2015) to get that Ant Colony Algorithm is a nature inspired algorithm, this algorithm depends on the conduct of genuine ants. In this, the ant picks the shortest path in search of its food. At the point when the request is initiated ant starts its movement. Ant will ceaselessly check whether the node is overloaded or under loaded. In the event that an ant finds an over-burden node, it will turn around. Also, if an ant finds any under loaded node, it will continue. Along these lines, ant's conduct is utilized to gather data from various nodes. The disadvantage of this algorithm is that delay can be caused by moving forward and backwards.

IV. COMPARISON AMONG DIFFERENT LOAD BALANCING ALGORITHMS

Algorithm	**Advantage & Disadvantage**
Round Robin Algorithm	Simple, fast response, no starvation but not scalable.
Min-Min Algorithm	Simple and fast for smaller tasks but cause starvation of larger tasks.
Max-Min Algorithm	Simple in execution but cause starvation of smaller tasks.
ESCE Algorithm	Equal load balancing, maximize throughput but not fault tolerant.
Throttled Algorithm	Good performance, better resource utilization but scans the entire list from the beginning.
Biased Random Algorithm	Suitable for large networks but performance degrades with an increase in diversity.
Honeybee Foraging Behaviour Algorithm	Performance increases by increasing system size, throughput will not increase with the increase in resources.
Ant Colony Algorithm	Decentralized, has network overhead, good resource utilization but causes delay.

V. CONCLUSION

In this paper, we have studied about load balancing and how different static and dynamic load balancing algorithms work in equally distributing the load. These classic load balancing algorithms play an essential role but there is a need to evolve. So it would be better if load balancers are designed smart enough (smart load balancers) to recognize technical characteristics like response time, execution time, size of data and load on each resource. Load balancers can be made to learn through their experiences to recognize an incoming query and respond accordingly. Based on user requirement they can select the appropriate algorithm which gives the best result as per their needs.

VI. REFRENCES

[1] SHANTI SWAROOP MOHARANA , RAJADEEPAN D. RAMESH & DIGAMBER POWAR , 2013, "ANALYSIS OF LOAD BALANCERS IN CLOUD COMPUTING", *International Journal of Computer Science and Engineering (IJCSE)Volume 2, Issue 2.*

[2] Simar Preet Singh , Anju Sharma and Rajesh Kumar , 2016, "Analysis of Load Balancing Algorithms using Cloud Analyst", *International Journal of Grid and Distributed Computing Volume 9(No.9).*

[3] Soumya Ray and Ajanta De Sarkar, October-2012, " "EXECUTION ANALYSIS OF LOAD BALANCING ALGORITHMS IN CLOUD COMPUTING ENVIRONMENT ", *International Journal on Cloud Computing: Services and Architecture ,Volume-2(No.5).*

[4] Chitranshi Gautam, Deekshit Singh and Arpit Sharma 2020, "ANALYSIS OF LOAD BALANCING ALGORITHMS IN CLOUD COMPUTING", *International Journal of Engineering Applied Sciences and Technology, Volume-4,Issue 10.*

[5] Kanakala R.; Reddy V.K.; Karthik K., 2015, "Performance analysis of load balancing techniques in cloud computing environment", *International Journal of Computer Sciences and Engineering Vol.-5(1), Jan 2017, E-ISSN: 2347-2693.*

[6] Aslam S., Shah M.A., 2015, "Load Balancing Algorithms" in Software Engineering Conference (NSEC).

[7] Kumar R. and Prashar T., 2015, 'Performance Analysis of Load Balancing Algorithms in cloud computing', *International journal of computer Applications, Vol.120, No.7, pp 19-27.*

[8] Kapoor S., Dr. Chetna Dabas, 2015, "Cluster Based Load Balancing in Cloud Computing", *Eighth International Conference on Contemporary Computing (IC3).*

[9] Sarmila, G.Punetha,, Dr.N.Gnanambigai, Dr.P.Dinadayalan, 2015, "Survey on Fault Tolerant – Load Balancing Algorithms in Cloud Computing", *IEEE Sponsored 2nd International Conference On Electronics And Communication System (ICECS), Pages-1715-1720.*

*[10]*Jaiswal A.A., Dr. Sanjeev Jain, 2014, " An Approach towards the Dynamic Load Management Techniques in Cloud Computing Environment", *International Conference on Power, Automation and Communication (INPAC).*

*[11]*Santhosh, B., & Manjaiah, D. (2014),"An improved task scheduling algorithm based on max-min for cloud computing",*International conference on Advances in computer & communications Engineering(ACCE-2014),vol 2,issue2,may2014.*

*[12]*Shahbaz Afzal and G Kavita, 2019, "Load balancing in cloud computing – A hierarchical taxonomical classification", *Journal of Cloud Computing:Advances, Systems and Applications,volume-8.*