IJRSET 2015 SPL Volume 2, Issue 10 Pages: 33-37

ISSN 2394-739X

International Journal for Research in Science Engineering and Technology

A Survey of efficient load balancing algorithms in cloud environment

[1] J. Mr. Srinivasan, [2] Dr. Suresh, [3] C. Gnanadhas
[1] Research Scholar, [1, 2, 3]Department of CSE,
[1] Bharathiar University, [2, 3] Vivekanandha College of Engineering for Women
[1, 2, 3] Coimbatore. Tiruchengode

Abstract:-

Current information technology playing main role is cloud computing ,which is sharing data and provide many resources to users, the cloud computing stores the data and distributed resources in the open environment. In open environment the data storage increases quickly like cloud computing having more challenges, the one of the main challenge is efficient load balancing. So in this paper we made survey report about efficient load balancing algorithms in cloud environment.

Keywords: - efficient load balancing,distributed, resources, environment, cloud computing.

1. INTRODUCTION

Load balancing in cloud computing systems is really a challenge now. Always a distributed solution is required, because it is not always practically feasible or cost efficient to maintain one or more ideal services just as to fulfill the required demands. Jobs can't be assigned to appropriate servers and clients individually for efficient load balancing as cloud is a very complex structure and components are present throughout a wide spread area[7]. An ideal load balancing algorithm should avoid overloading or underloading of any specific node. But in case of a cloud computing environment the selection of load balancing algorithm is not easy, because it involves additional constraints like security, reliability, throughput etc. So, the main goal of a load balancing algorithms is time of job by distributing the total load of system. The algorithm must also ensure that it is not overloading any specific node. As the number of servers grows, the risk of a failure increases and such failures must be handled carefully. The ability to maintain unaffected service during any number of simultaneous failures is termed as high availability [2]. Load balancing is also provided by few operating systems. Microsoft's Network Load balancing (NLB) a software-based solution that allows you to effortlessly cluster multiple machines [2]. There are variety open source load balancers and load balancing software available for Linux, such as Linux virtual server, Ultra Monkey, Red Hat cluster suite, High availability Linux(LinuxHA), which can be used efficiently with most of the network services, including FTP, HTTP, DNS, SMTP, POP/IMAP, VOIP, etc[1].

1.1 Concept of Load balancing

"Load balancing "means an even distribution of the total load amongst all serving entities [3]. Load balancing is very essential in distributed computing systems to improve the quality of service by managing customer loads that are changing over time. The request demands of incoming requests are optimally distributed among available

system resources to avoid resource bottlenecks as well as to fully utilize available resources [4]. Load balancing also provides horizontal scaling e.g, adding computing resources in order to address increased loads.

2. LOAD BALANCING ALGORITHMS:

Load balancing algorithms can be divided into two categories, Static Algorithm and Dynamic Algorithm.

2.1 Static Algorithm:

Static algorithms divide the traffic equivalently between servers. By this approach the traffic on the servers will be disdain easily and consequently, it will make the situation more imperfectly. This algorithm which divides the traffic equally is announced as round robin algorithm [6]. In this algorithm each servers have been assigned a weight and according to the highest weight they received more connections. In the situation that all the weights are equal, servers receive balanced traffic [6].

2.2 Dynamic Algorithm:

Dynamic algorithm designated proper weights on servers and by searching in whole network a lightest server preferred to balance the traffic. However, selecting an appropriate server needed real time communication with the networks, which will lead to extra traffic added on system [6]. In comparison between these two algorithms, although round robin algorithms based on simple rule, more loads conceived on servers and thus unbalanced traffic discovered as a result [6].Dynamic algorithm predicated on query that can be made frequently on servers, but sometimes prevailed traffic will prevent these queries to be answered, and correspondingly more added overhead can be distinguished on network.

3. RELATED WORK

Cloud computing is growing technology in IT industry it is leading towards the research advances in many domains. Here one of the main challenges is load balancing in cloud computing,

Because the load balancing technique will sharing the resources across the network while we sharing the resources in cloud environment we will face some problems these problems will rectified through algorithms, there are many algorithms available for load balancing technique in cloud computing, in this paper we collect and study different category of algorithm and analysis which are the efficient algorithm for load balancing technique.

3.1 Round Robin algorithm

It is the simplest algorithm that uses the concept of time quantum or slices. Here the time is divided into multiple slices and each node is given a particular time quantum or time interval and in this quantum the node will perform its operations. The resources of the service provider are provided to the client on the basis of this time quantum. The following steps are define the simple RR algorithm.

Step 1: RR Vmload balancer maintains an index of VMs and states of the VMs in busy/available.At start all VMs has 0 allocations.

Step 2; The datacenter controller receives the user requests.

It stores the arrival time and burst time of the user requests.

The requests are allocated to vms on the basis of their states known from the VM queue.

The requests are allocated to vms on the basis of their states known from the VM queue.

Step:3The roundrobinVMloadbalancer will calculate the trun-around time of each process.

It also calculated the response time and average waiting time of user requests.

It decides the scheduling order.

Step:4 After the excecution of cloudlets, the VMs are de allocated by the RR Vm loadbalancer.
Step5: The datacentercontroller checks for new waiting/pending requests.
Step6: Continue from step-2

3.2 Equally spread current execution

Equally Spread current execution [9] is a dynamic load balancing algorithm, which handles the process with priority. It determines the priority by checking the size of the process. This algorithm distributes the load randomly by first checking the size of the process and then transferring the load to a Virtual Machine which is lightly loaded. The load balancer spreads the load on to different nodes, and hence, it is known as spread spectrum technique.

3.2 Throttled Load Balancing Algorithm.

Throttled load balancer [9] is a dynamic load balancing algorithm. In this algorithm, the client first requests the load balancer to find a suitable Virtual machine to perform the required operation. In Cloud computing, there may be multiple instances of virtual machine. These virtual machines can be grouped based on the type of requests they can handle. Whenever a client sends a request, the load balancer will first look for that group, which can handle this request and allocate the process to the lightly loaded instance of that group. Load balancer maintains an index table of virtual machines as well as their states whether virtual machine is available or busy. Client first makes a request to data centre to find the suitable virtual machine to perform the job. The data centre queries the load balancer for allocation of the LM. LB scans the index table from top. If the virtual machine is found, the LB inform to data centre, the data centre communicates the request to virtual machine. Further the data centre acknowledges the load balancer about the new allocation and revives the index table accordingly. While processing the request if virtual machine is not found, LB return to -1 to data centre, the data centre guesses the request. When vitrtual machine completes the task a request is acknowledged to data centre which is further sent to LB to de-allocate the same VM which is already communicated [9].

Merits

- Load balancer maintains an index table of VM, and as well as their states whether the VM is available or busy which decreases the time.

3.4 Biased Random Sampling

Biased Random Sampling [8] is a dynamic load balancing algorithm. It uses random sampling of system domain to achieve self-organization thus, balancing the load across all nodes of system. In this algorithm, a virtual graph is constructed with the connectivity of each node representing the load on server. Each node is represented as a vertex in a directed graph and each in-degree represents free resources of that node.

3.5 Min-Min

It is a static load balancing algorithm. So, all the information related to the job is available in advance. Some terminology related to static load balancing Min-Min algorithm [9] begins with a set of all unassigned jobs. First of all, minimum completion time for all jobs is calculated. The job with minimum completion time is selected. Then, the node which has the minimum completion time for all jobs is selected. Finally, the selected node and the selected job are mapped. The ready time of the node is updated. This process is repeated until all the unassigned jobs are assigned. The advantage of this algorithm is that the job with the smallest execution time is executed. The drawback of this algorithm is that some jobs may experience starvation.

Demerits

It leads to starvation mostly

3.6 **Max-min**

Same as min-min algorithm, except after finding out the minimum execution time, the maximum value is selected from the minimum execution times. The machine that has the minimum completion time for all the jobs is selected. Finally the selected node and the selected job are mapped. Then the ready time of the node is updated by adding the execution time of the assigned task [9].

3.7 Honeybee Foraging Algorithm

The main idea behind the Honeybee Foraging algorithm [8] is derived from the behavior of honeybees. There are two kinds of honeybees: finders and reapers. The finder honeybees first goes outside of the honey comb and find the honey sources. After finding the source, they return to the honey comb and do a waggle dance indicating the quality and quantity of honey available. Then, reapers go outside and reap the honey from those sources. After collecting, they return to beehive and does a waggle dance. This dance indicates how much food is left. M. Randles proposed a decentralized honeybee based algorithm for self-organization. In this case, the servers are grouped as virtual server and each virtual server have a process queue. Each server, after processing a request from its queue, calculates the profit which is analogous to the quality

3.8 Token Routing

The main objective of the algorithm is to minimize the system cost by moving the tokens around the system. But in a scalable cloud system agents cannot have the enough information of distributing the work load due to communication bottleneck. So the workload distribution among the agents is not fixed.

Merits

- The drawback of the token routing algorithm can be removed with the help of heuristic approach of token based load balancing.

Demerits

- As agents have their own knowledge base derived from previous tokens, no communication overhead is generated.

3.9 Opportunistic Load Balancing Algorithm

This is static load balancing algorithm so it does not consider the current workload of the virtual machine. It attempts to keep each node busy. This algorithm deals quickly with the unexecuted tasks in random order to the currently available node. Each task is assigned to the node randomly. It provides load balance schedule without good results. The task will process in slow in manner because it does not calculate the current execution time of the node.

Merits

- Deals quickly with the unexecuted tasks in random order to available node.

Demerits

- It provides load balance schedule with bad results.
- The task processing is slow in manner because it does not calculate the current execution time of the node

CONCLUSION

In this paper, we have studied different load balancing techniques in the cloud environment, we have discussed main issues of these algorithms which must be taken into consideration while designing any load balancing algorithms we have discussed the already proposed algorithms by various researchers in literature, their advantages and disadvantages. A comparison has been done on the basis of different criteria like Throughput, Overhead, Response time, Scalability, Response time, etc. In future we will focus on designing better algorithms for

maintain a best trade-off-among all performance parameters.

REFERENCES

[1] Network Load Balancing in Microsoft Windows 2000 Advanced Server and Datacenter Server Operating Systems,http://technet.microsoft.com/en-us/library/bb742455.aspx.

[2] Yixin Diao, Chai Wah Wu, Joseph L. Hellerstein, Adam J.Storm, Maheswaran Surendra, Sam Lightstone, Sujay Parekh, Christian Garcia-Arellano, Matthew Carroll, Lee Chu, and Jerome Colaco, “Comparative Studies of Load Balancing With Control and Optimization Techniques”,
2005 American Control Conference Portland, OR, USA,June 8-10, 2005.

[3] Raman a Ku m ar K., Mah e sh V. G h at age, “Lo ad Balancing of Services with Server In it iated Connections”, ICPWC, 2005.

[4] BrankoRadojevic, Mario Žagar, “Analysis of Issues with Load Balancing Algorithms in Hosted Cloud Environments”, Opatija, Croatia, MIPRO 2011, May 23-27, 2011.

[5]Z. Zhang, and X. Zhang, “A Load Balancing Mechanism Based on Ant Colony and Complex Network Theory in Open Cloud Computing
Federation”, Proceedings of 2nd International Conference on Industrial Mechatronics and Automation ICIMA), Wuhan, China, May 2010, pages 240- 243.

[6] R. X. T. and X. F.Z.A Load Balancing Strategy Based on the Combination of Static and Dynamic, in Database Technology and Applications (DBTA), 2010 2nd International Workshop (2010), pp.1-4.

[7]P.Beaulah Soundarabai,Sandhya Rani A.,Ritesh Kumar Sahai, Thriveni j.,K.R.Venugopal and L.M.Patnaik,”Comparative Study on
Load balancing technique in Distributed Systems”,IJITKM,December 2012,Volum 6,No-1,PP53-60

[8]Shantiswaroop Moharana, Rajadeepan.D,Ramesh and Digamber power,”Analysis of Load Balancers in cloud computing”,IJCSE,Vol.2,Issue 2,May2013,101-108.

[9] Dr.G.N.K.Sureshbabu and Dr.S.K.Srivatsa,” A Review of Load Balancing Algorithms for Cloud computing”,IJECS,ISSN:2319-7242,Volume-3,Issue-9,September-2014,Page no.8297-8302.