Mody University International Journal of Computing and Engineering Research
Vol. 1 Issue 1, 2017, pp.51-55
ISSN: 2456-9607 (Print) 2456-8333(Online)

A Comparative Study of Different Load Balancing Algorithms in Cloud Computing

Divya Rani Mittal, Manmohan Sharma

Department of Computer Science & Engineering
Mody University of Science & Technology, Lakshmangarh, India

divyamittal70@gmail.com, manmohansharma.cet@modyuniversity.ac.in

Received 30 Nov. 2016, Published 31 March. 2017

Abstract: Cloud computing is very popular because of the features it provides. It has changed the field of parallel and distributed computing system today. It is very much in use because of the features it provides like pay per usage, resource sharing, rapid elasticity, broad network access etc. Along with many advantages, cloud computing comes with many challenges. Load balancing is one of the biggest challenges of cloud computing. If not handled properly, it leads to degradation of business performance. For handling load balancing many algorithms have been proposed such as Min-Min, Max-Min, Genetic Algorithm, Honey Bee etc. In this paper we have performed a brief review of some of load balancing techniques along with their merits and demerits.

Keywords: Cloud computing, load balancing algorithms, comparitive study

1. INTRODUCTION

Cloud computing is an interconnected high speed network which provides facilities like elasticity, on demand resource provisioning. It has several elements like client and servers [1]. The infrastructure of cloud computing is used by business and users to access application services on demand. It moves jobs from private PC to remote computers for further processing. It provides maximum services in minimum time. It is used as a group of processing nodes that can cooperate to perform a specific service together [2]. It addresses the computational needs of users. It delivers three kinds of services: Infrastructure as a Service (IaaS), Platform as a Service(PaaS), and Software as a Service(SaaS). Users access these services on demand and pay as per usage. Though Cloud Computing has a bright future but still there are many problems associated with it. One of the main problems is load balancing. Load balancing stands for distributing load evenly among all the nodes such that no node is over loaded and no node is idle. There are many characteristics of Load balancing like equally dividing work among all nodes, user satisfaction, improving overall performance, minimize response time and resource utilization. There are several approaches which are used for load balancing. In this paper we haveprovided a brief review of all the processes. Each algorithm deals with different issues and has some limitations too. The existing algorithms for load balancing deals with many issues like performance issues, large processing time, starvation and are limited to the environment. A good load balancing algorithm ensured that no node is overloaded or idle at any time. Through this paper our aim is to evaluate the performance of some common known load balancing algorithms. There are some goals of load balancing algorithms:

- *Cost Effectiveness*: It aims on improving the overall performance on reasonable cost.
- *Scalability and Flexibility:* The cloud computing may change in size or network. So the algorithm must be able to handle and adapt these changes.
- *Priority:* Resources should be assigned on priority by the algorithm for better service.

2. RELATED WORK

Load balancing is a technique or a process in which load is divided between different servers or nodes equally [3] [4]. For this purpose, various algorithms are developed known as load balancing algorithms. The main goal is to make the overallsystem more efficient and to increase the performance. Various load balancing algorithms are: Min-Min, Max Min, Round Robin etc. Basically load balancing algorithms are divided in two categories, static and dynamic. We have provided a detailed study of these algorithms.

A. Static Algorithms

These algorithms are based on completion time of a task [3][5]. All the decisions are made during compile time. In this type of algorithm work is divided equally among all the servers. System information is not used while distributing the workload [6][7]. Servers are assigned weight and the server with maximum weight receives more connections. In this type of algorithm changes are not allowed during run time. One of the biggest advantages of static algorithm is that they are not dependent on present conditions of the system.It does not use system information while distributing the load. Along with advantages static algorithms do have limitations like they can be used only when load variation is low and do not have the ability to handle workload during run time. It transfers only fixed amount of data and no prior knowledge of system is required.

Some of the algorithms which fall under this category are:

I. Round Robin Algorithm

In this algorithm a fixed quantum is assigned to each job. It uses equal time to complete each task. If any task goes beyond that fixed quantum the process is dropped in between so that no process leads to starvation. In case of heavy loads, this algorithm takes long time to complete the entire task [7] [8]. As the name suggests, it works in a circular pattern. Every node has a fixed time size and has to perform its work in that given time only. This algorithm is used because of its advantages like fixed time quantum, easy to understand, fairness. Moreover it performs better for short CPU burst. Some limitation of these algorithms is that larger tasks take longer time and more context switches due to short quantum time.

II .Weighted Round Robin Algorithm

One major drawback of Round Robin algorithm was that at some moment some node possesses heavy load and others with no load. This issue was tackled weighted round robin. In this algorithm each node is allowed to receive specific number of requests according to the assigned weight only [9] [10] [11] [12]. Each instance of server gets the load assigned depending on its processing capability, which depends on how that instance is behaving. One can assign a weight to each server in the group so that if one server is capable of handling twice as much load as the other, the powerful server gets a weight of 2. Weighted round robin has many advantages over round robin because each node receives task according to the assigned weight and the server receives balanced traffic. But again this algorithm also suffered some limitations likewise no precise prediction of execution time is possible and hence this algorithm is not preferred.

III .Min-Min Algorithm

The execution time of each task is calculated. The task is arranged in ascending order on the basis of their completion or execution time. The running time for all other tasks is also updated [13] [14]. In this there is a set of unassigned tasks.Theminimum completion time for all tasks is calculated. The task with minimum value is selected and is schedule on machine and the process is followed until all the tasks are assigned on the resources. This algorithm served better because it works according to smallest completion time and shows best result in presence of small tasks but was not taken in note because of some limitations likeprocess with maximum completion time often leads to starvation and variations of machine and tasks cannot be predicted.

IV. Max-Min Algorithm

Max-Min is almost same as Min-Min algorithm [15] [16]. But in Max-Min the task with maximum execution time is assigned first to the machine. After assigning the task, machine works according to the updates. The assigned tasks which get completed are then removed from the list. These algorithms served better over min-min algorithm because in this algorithm requirements are priory known. Some limitations associated with this algorithm are that it takes long time for task completion and process with short execution time leads to starvation.

B. Dynamic Algorithms

This type of algorithms is based on some criteria's like capabilities and network bandwidth. These algorithms require constant check of nodes and very difficult to implement. Server with the lightest weight is searched in the network and is preferred. If the algorithm finds high usage of CPU,the load is sent to some other node[17]. To handle the load, current state of system is used. These algorithms are better over static algorithms because of the advantages like work is distributed at run time, fault tolerant, only current state is required, high performance. Apart from the advantages these too have some limitations like constant check of nodes is required, complicated and more difficult to implement.

The algorithms which come under dynamic algorithms areHoney Bee, Ant Colony, Genetic algorithm, Carton etc.

Table 1 Comparison of load balancing

LB algorithms	Fairness	Response time	Throughput	Overhead	Fault tolerance	Performance	Resource utilization	Speed	Complexity
Static	Yes	Fast	High	N/A	No	Fast	High	Fast	Low
Round Robin	Yes	Fast	High	High	No	Fast	High	N/A	Low
Min-Min	No	Fast	High	High	No	Fast	High	Fast	Low
Max-Min	No	Fast	High	High	No	Fast	High	Slow	Low
Dynamic	No	Slow	High	High	Yes	Slow	High	Fast	High
Honey bee	No	Slow	High	low	No	Slow	High	Fast	Low
Ant colony	Yes	Slow	High	High	N/A	Slow	High	Fast	High
Carton	No	Fast	Low	N/A	N/A	Fast	High	Fast	High
Throttle	No	Fast	High	low	No	Fast	High	Fast	Low

I. Ant colony algorithm

This is based on the nature rule of ants. Basically when all ants look for food, they leave a chemical pheromone besides them which makes other ants to follow the path and reach the food. The amount of pheromone depends on the quality of food. Similar approach is followed in load balancing. Ant colony approach was very much in use because of the benefits like less time span, mutual independence, high computation power. Although with the passing time several limitations were discovered in this nature's based algorithm like increased network overhead and no prior knowledge of number of ants.

II. Honey Bee Foraging Algorithm

It is again based on nature phenomenon of honey bees. When a virtual machine which is underweight assigns a task, priority tasks and load of other virtual machine is updated [18]. A task with highest priority, selects a virtual machine with minimum number of priority tasks. This algorithm came with benefits like increased throughput and minimum response time and with some limitations like tasks with high priority are dependent on virtual machines.

II.Throttled Load Balancing Algorithm

In this algorithm a suitable search of virtual machines is made. The list of virtual machines is managed by the task manager. With the help of this list, request is assigned to the machines. If size and capability is suitable task is assigned to the machine [19]. This algorithm is better than Round Robin and has good performance where list of virtual machine is managed in proper way. But the tasks need to be waited.

IV .Carton

It is a combination of load balancing and distributed rate limiting. In this jobs are fairly assigned to servers with equal distribution of resources. The algorithm can be implemented as low communication required [19]. This algorithm has a high fairness rate along with good performance, low communication need and equal distribution of resources. One thing which was lacking in this algorithm is that it depends on lower costs.

V. Genetic Algorithm

This algorithm is best among the entire proposed algorithms. It is based on nature selection process and consists of four steps namely population generation, selection, mutation andcrossover [20]. The chromosomes generated are tested if they are healthy or not. They are a part of further process only if healthy. This approach is followed in load balancing stating that only virtual machines which are capable of handling load perfectly are used. By using the features of this, fit machines new machines are produced which are better. This algorithm is best among all. It provides various features like effectiveness, low cost, high performance, high degree load balancing, and minimum response time. Some of the limitations associated with this algorithm is this that resources with no capability are also considered and it makes overburden on cloud environment. Moreover resources are not handled properly.

3. Performance Evaluation

There are various parameters to evaluate the performance of these load balancing algorithms. These parameters are as follows:

- Throughput: It calculates number of tasks whose execution has been completed. In order for a good performance of a system, throughput should be high.
- Overhead: It is associated with movement of tasks, inter-process and inter-processor communication, for a load balancing algorithm to perform well, and should be minimum.
- Fault tolerance: A good system is the one which is fault tolerant means that it can work efficiently even if one of the nodes fails. Load balancing is the best technique to provide this feature. It migrate the tasks from one server to another when necessary.
- Response time: It refers to the time taken by a particular load balancing algorithm in the cloud environment. This metric should be minimized.
- Resource utilization: It should be maximum for an efficient system. Resources should be utilized efficiently.
- Scalability: It states that the performance of the system would not be affected.

Based on these parameters, table 1.1 of the above algorithms is made. This table evaluates the performance of these algorithms based on different metrics. This table shows positive and negative aspects of the load balancing algorithms. We can clearly see from the table that static algorithms are fair to distribute and are less complex and not fault tolerant. Min-Min algorithm is not fair and in case of Max-Min requirements are known in advance so that they work better giving high throughout. In case of Dynamic algorithms current state of system is required. They have more overhead and are fault tolerant. Honey bee has low response time with high throughput. Ant colony is very simple to implement and has high throughout. Working of Carton Algorithm is simple.

4. Conclusion

We have presented comparisons of different load balancing algorithms. We have also described merits and demerits of these different algorithms. The important part is that the comparison is made on different metrics of load balancing like fairness, throughput, fault tolerances, overhead, performance, and response time given in Table 1.1 The limitation of the existing algorithms is this that no algorithm addresses the issues like fairness throughout etc. We have reached on a conclusion that round robin is more efficient than other approaches as it distributed the workload fairly with high throughput and good response time, moreover it is less complex. The biggest advantage of this algorithm is its time limitation and how it uses equal period to complete each task. But there are some disadvantages too. So our future work is to mitigate the drawbacks of this algorithm.

References

[1] Sanjay K. Dhurandher, Mohammad S. Obaidat," A Cluster-Based Load Balancing Algorithm in Cloud Computing," *IEEE ICC -Mobile and Wireless Networking Symposium*, October 2014, pp.2921-2925.

[2] Jia Zhao, Kun Yang, Xiaohui Wei, Yan Ding, Liang Hu, and GaochaoXu, "A Heuristic Clustering-BasedTask Deployment Approach for Load Balancing Using Bayes Theorem in Cloud Environment," *IEEE Transactions On Parallel and Distributed Systems*, Vol. 27, No. 2, February 2016, pp. 305-316.

[3] Matthias Sommer, Michael Klink, Sven Tomforde, JorgHahner, "Predictive Load Balancing in Cloud Computing Environments based on Ensemble Forecasting," *IEEE International Conference on Autonomic Computing*, September 2016, pp.300-307.

[4] Sidra Aslam, Munam Ali Shah," Load Balancing Algorithms in Cloud Computing: A Survey of Modern Techniques," *National Software Engineering Conference (NSEC 2015) IEEE*, May2015, pp. 30-35.

[5] Qi Liu, WeidongCai, JianShen, Xiaodong Liu, Nigel Linge," An Adaptive Approach to Better Load Balancing in a Consumer-centric Cloud Environment,"*IEEE Transactions on Consumer Electronics*, Vol. 62, No. 3, August 2016, pp. 243-250.

[6] Martin Randles, David Lamb, A. Taleb-Bendiab," A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing," *IEEE 24th International Conference on Advanced Information Networking and Applications Workshop,* September,2010, pp. 325-329,.

[7] Liuhua Chen, HaiyingShen, KaranSapra.," RIAL: Resource Intensity Aware Load Balancing in Clouds," *IEEEINFOCOM - IEEE Conference on Computer Communications,* August 2014, pp.1294-1302.

[8] Lahar Singh Nishad, Sarvesh Kumar, Sumit Kumar Bola," Round Robin selection of datacenter for load balancing in cloud computing," *International Conference on Computing for*

Sustainable Global Development (INDIACom), IEEE, February,2016, pp 2901-2905.

[9] Suriya Begum, Dr. Prashanth C.S.R,” Review of Load Balancing in Cloud Computing,” *IJCSI International Journal of Computer Science Issues*, Vol. 10, Issue 1, No 2, January 2013, pp. 343-352.

[10] Seungmin Kang, BharadwajVeeravalli, KhinMiMiAung,” Scheduling Multiple Divisible Loads in a Multi-Cloud System,” *IEEE/ACM 7th International Conference on Utility and Cloud Computing*, July 2014, pp. 371-378.

[11] Filipe Fernandes S B de Matos, JoaquimCelestinoJúnior, André Ribeiro Cardoso,” VBalance: A Selection Policy of Virtual Machines for Load Balancing in Cloud Computing,” *20th IEEE Symposium on Computers and Communication (ISCC),* December 2015, pp. 770-775.

[12] Soumya Ray and Ajanta De Sarkar, “Execution Analysis of Load Balancing Algorithms in Cloud Computing Environment,” *International Journal on Cloud Computing: Services and Architecture (IJCCSA)*, Vol.2, No.5, October 2012, , pp. 1-13.

[13] Pavithra B, Ranjana R,” A Comparative Study on Performance of Energy Efficient Load Balancing Techniques in Cloud,” *IEEE WiSPNET conference*, July 2016, pp. 1192-1196.

[14] SoumenSantra, Kalyani Mali,” A New Approach to Survey on Load Balancing in VM in Cloud Computing: usingCloudSim,” *IEEE International Conference on Computer, Communication and Control (IC4-2015)*, March 2015, pp. 452-461.

[15] GeethuGopinath P P, Shriram K Vasudevan.,” An in-depth analysis and study of Load balancing techniques in the cloud computing environment.,” *2nd International Symposium on Big Data and Cloud Computing (ISBCC’15) ScienceDirect, Procedia Computer Science 50,* May 2015, pp. 427-432.

[16] BrototiMondal, KousikDasgupta, ParamarthaDutta,” Load Balancing in Cloud Computing using Stochastic Hill Climbing-A Soft Computing Approach,”*ScienceDirect,Procedia Technology 4*, July 2012*,)*, pp. 783 – 789.

[17] Chun-Cheng Lin, Hui-Hsin Chin, and Der-Jiunn Deng, “Dynamic Multiservice Load Balancing in Cloud-Based Multimedia System,” *IEEE Systems Journal*, Vol. 8, No. 1, MARCH 2014, pp. 225-234.

[18] Zhen Xiao, Weijia Song, and Qi Chen,” Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment,” *IEEE Transactions On Parallel and Distributed Systems*, Vol. 24, NO. 6, June2013, pp. 1107-1117.

[19]] BrototiMondal, KousikDasgupta, ParamarthaDutta,” Load Balancing in Cloud Computing using Stochastic Hill Climbing-A Soft Computing Approach,”*ScienceDirect,Procedia Technology 4*, July 2012*,)*, pp. 783 – 789.

[20] Ronak R Patel, Swachil J Patel, Dhaval S Patel, Tushar T Desai,” Improved Ga Using Population Reduction for Load Balancing in Cloud Computing*,” 2016 Intl. Conference on Advances in Computing, Communications and Informatics (ICACCI) IEEE*, Sept. 21-24, 2016, pp. 2372-2374.