

Load Balancing Algorithms in Cloud Computing Environment - A Methodical Comparison

J. Uma, V. Ramasamy, A. Kaleeswaran

Abstract : In cloud computing load balancing is required to achieve evenly distribute load among the nodes and to efficiently make use of the resources Load balancing ensures that all the processor in the system or every node in the network does approximately the equal amount of work at any instant of time. This technique can be sender initiated, receiver initiated or symmetric type (combination of sender initiated and receiver initiated types). This paper presents the features and discussed about the pros and cons of various load balancing algorithm in the authors style. Various consideration of the algorithm like throughput, performance, fault tolerance, migration time, response time etc has been discussed.

Keywords: Load Balancing, Resource Utilization, Performance, Job Scheduling

1. Introduction

Cloud computing is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers. It is an on-demand network access to a shared resource from geographically distributed systems all over the world through internet, which results in minimal management effort or service provider interaction.

Manuscript received Feb, 2014.

First Author name: J. Uma, Assistant Professor, Department of Computer Science and Engineering, Jansons Institute of Technology, Coimbatore, Tamilnadu, India.
Mobile: 9976488402

Second Author name: V. Ramasamy, Assistant Professor, Department of Computer Science and Engineering, Park College of Engineering and Technology, Coimbatore, Tamilnadu, India.
Mobile: 9894836977

Third Author name: A. Kaleeswaran, Assistant Professor, Department of Computer Science and Engineering, Park College of Engineering and Technology, Coimbatore, Tamilnadu, India.
Mobile: 9976142091

This can be easily said to be “Document Centric not PC centric” [1]. Cloud has an aspect of **five** characteristics, **four** deployment models and **three** service models in its architecture. The five characteristics are on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service. Private cloud, community cloud, public cloud and hybrid cloud are the four deployment models and as known the three service models are Software as a service (SaaS), Platform as a service (PaaS), and Infrastructure as a service (IaaS).

2. Research Issues in Cloud Computing

Indeed cloud computing provides several compelling features and attractive utilities there are some issues in this field which are to be carried out in research. Though there are many issues in Cloud Computing three important issues are discussed in this paper.

- i. **Load balancing:** Load Balancing is a method of distributing workload across multiple computing resources such as cluster of computers, network links [2]. The goal of LB is to optimize the resource usage, evade overload, maximize throughput and to minimize the response time. This was identified as a major concern in Cloud Computing to scale up the increasing demands. It is divided into two types Static Load Balancing and Dynamic Load Balancing, discussed in next Chapter.
- ii. **Job Scheduling:** Assigning an appropriate number of tasks to the nodes is Job Scheduling. Job scheduling is most important task in cloud computing environment because user have to pay for resources used based upon time. It is one of the major activities executed in many distributed networks, which gives maximum profit to the network. The objective of scheduling is spreading the load among the system equally by maximizing the utilization and minimizing the through-put (task execution time).
- iii. **Data Security:** Huge data are stored in a remote geographical network in a Cloud, so managing the big data is a major issue in Cloud Computing. There is a possibility where malicious user can penetrate into the cloud to hack the data, which infects the entire Cloud.

Thus many customers who are sharing the Cloud may lose their data.

3. Load Balancing

Among the above issues, many algorithms under load balancing are discussed in this paper. As Load Balancing is one of the major issues related to cloud computing, the load may represent a CPU capacity, memory, network load etc. It is necessary to distribute the load equally among the nodes in a network. This results in agile and efficient performance of the system. Thereby it avoids heavily loading or under loading of nodes in a network. It is divided into two types Static Load Balancing and Dynamic Load Balancing.

Static Load balancing Algorithm:

In static algorithm the traffic is divided evenly among the servers. This algorithm requires a prior knowledge of system resources, so that the decision of shifting of the load does not depend on the current state of system. Static algorithm is proper in the system which has low variation in load [4].

Dynamic Load balancing Algorithm:

In dynamic algorithm the lightest server in the whole network or system is searched and preferred for balancing a load. For this real time communication with network is needed which can increase the traffic in the system. Here current state of the system is used to make decisions to manage the load [4].

The goals of load balancing are [3]

- Performance
- Throughput
- Fault Tolerance
- Migration Time
- Overhead
- Response Time
- Scalability

4. Various Load Balancing Algorithms:

Opportunistic Load Balancing (OLB) and Load Balance Min-Min (LBMM)

S.-C. Wang et al. [5] proposed OLB and LBMM. In which OLB keeps each load busy regardless of the current workload assign to each node in the network. It assigns the job randomly. LBMM adopts a task to a respective node which has expected minimum completion time of this task over the other nodes. Since each task has different characteristics for execution and different capability to execute a task mere consideration of CPU remaining of the node is not enough.

Central Load Balancing Decision model

B. Radojevic et al [6] proposed this static LB algorithm, overcomes the drawback of Round Robin Technique. This works based on the application layer session switching. In this algorithm calculation of connection time between the client and the node is calculated. If the connection is above the threshold level, then the connection between the client and the node will be

terminated and the previous model Round Robin technique will be continued to towards completion of the task.

Map Reduced Based Entity Resolution model

Lars Kolb et al [7] introduced the Map Reduce Based Entity Resolution model which has two parts – Map() and Reduce(). Map() procedure performs filtering and sorting of tasks with the help of Part() method. This partition the large datasets into sub-tasks and Comp() method is used to compare the similar task and group it using Group() with the help of Reduce(). Overloading of task is reduced due to parallel processing of task by the Part() method.

Even Driven algorithm

V. nae [8] proposed this algorithm for load balancing real time massively multiplayer online games. It receives capacity events as input and analyses its component in the context of resources and the global state of game session, thereby generating the game session load balancing actions. According to the variable user load in multiple resources this algorithm is capable of scaling the game session. But this algorithm reaches occasional QoS Breaches.

Ant Colony optimization

Z. Zhang et al. [9] investigated a static load balancing algorithm called Ant Colony Optimization. As the request is sent the ant initiates its movement. This algorithm work based on the Ants behavior to collect information of cloud node to assign task to the particular node. Once the task is initiated from the “head node”, the ant and the pheromone starts the forward movement in the pathway. The ant moves in forward direction from an overloaded node looking for next node to check whether it is an overloaded node or not. Now if ant find under loaded node still it move in forward direction in the path. And if it finds the overloaded node then it starts the backward movement to the last under loaded node it found previously. In the algorithm [8] if ant found the target node, ant will commit suicide so that it will prevent unnecessary backward movement.

Honey Bee Foraging

This algorithm was proposed by Dhinesh B.L.D, P.V.Krishna. This algorithm was derived from the behavior of honey bees in finding their food [10]. Among the classes of bees the forager bees forage for food sources. Once the food source has been found the forager bees come back to the bee hive and advertise the food source by a dance called “waggle dance”. The kind of dance shows the quality and quantity of the food and the distance of the source from the bee hive. This makes the scout bees to race for the food.

In case of load balancing the servers are grouped into virtual servers. Each Virtual servers will has its own Virtual Server (VS) request queue. Each V S will calculate its profit which is similar to the bees waggle dance.

Biased Random Sampling

M. Randles et al. [11] proposed this distributed load balancing algorithm. Load balancing can be achieved efficiently across the nodes in this approach, by using random sampling method. Here servers are treated as nodes. In this method a virtual graph is constructed representing the load on the nodes and with each in-degree directed to the respective resources to the server.

While the server starts executing the job it reduces the in-degree which indicated the reduction in availability of free resources. Similarly while the server completes the job the incoming degree gets incremented which in turn indicates the increase in availability of resources. This process is called random sampling. The execution starts at any node and the random neighboring node will be selected for the next job to be executed. Thus the load balancing technique used here is fully decentralized and chosen apt for many cloud networks.

Active Clustering

Active clustering is an enhanced method of random sampling, where this algorithm works on the principle of grouping similar nodes together and start working on these group nodes [12]. This method uses the resources efficiently thereby increases the throughput and performance of the system by using high resources. In this approach a technique called match-maker is introduced. When an execution starts in a network, the process gets initiated and searches for the next matching node said to be match-maker which should satisfy the criteria that it should be the different one from the former one.

Once the match-maker is found the process gets initiated and as soon as the process gets over the match-maker gets detached from the network. Thus this is an iterative process in the network to balance the load efficiently.

5. Comparative Study

OLB:

- (+) Works based on framework of the system
- (+) Keeps each node busy
- (-) Keeps the node busy irrespective of the of the execution time, which results in bottle neck of the system
- (-) Depends on LBMM

LBMM:

- (+) This algorithm divides the framework into 3 and each layer splits the task.
- (-) The major drawback is it divides the task into many subtasks which takes huge time to complete even a simple task.

CLBDM:

- (+) Session switching.
- (-) Since this method works based on the round robin technique, if the connection goes above the threshold level problem arises.

Map Reduce ER model

- (+) This executes the job parallel. So no overhead.

- (-) Complexity is more thereby low throughput is achieved.

Event Driven:

- (+) Generates the game session load balancing algorithm.
- (-) Limited Applications

Ant Colony optimization:

- (+) Allow shared computation and storage over long distances.
- (+) Achieves optimal job scheduling
- (+) Global optimization using the crossover operator which guarantees the diversity of the solution.

Honey Bee Foraging Algorithm:

- (+) Achieves global load balancing through local server action
- (+) Based on naturally occurring phenomenon.
- (-) Throughput decreases when variety of nodes increases.

Biased Random Sampling:

- (+) Achieves load balancing across all system nodes using random sampling of the system domain
- (-) Degrades when the load increases.

Active clustering:

- (+) Optimizes job assignment by connecting similar services by local re-wiring.
- (-) This degrades its performance when increase in diversity of nodes.

6. Conclusion

This paper focused on a methodological analysis on various load balancing algorithm. The primary objective of load balancing is to satisfy the customer requirement by distributing the load equally among the nodes and to make maximum utilization of resources in the stipulated time. This paper investigated on the major aspects of load balancing and compared the existing algorithms and its features. The findings from the discussed algorithm is that to develop an energy efficient load balancing algorithm which focuses less on energy consumption and carbon emission factors. Thus to achieve Green Computing.

REFERENCES

- [1] Michael Miller, “*Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate*”, Que Publishing, First Edition, 2008
- [2] [http://en.wikipedia.org/wiki/Load_balancing_\(computing\)](http://en.wikipedia.org/wiki/Load_balancing_(computing))
- [3] D. Escalante, Andrew J. Korty, “Cloud Services: Policy and Assessment”, *Educause review* July/August 2011.
- [4] T-Y., W-T. Lee, Y-S. Lin, Y-S. Lin, H-L. Chan and J-S. Huang, “Dynamic load balancing mechanism based on cloud storage” in proc. *Computing, Communications and Applications Conference (ComComAp), IEEE*, pp: 102-106, January 2012.
- [5] Wang S., Yan K., Liao W. and Wang S. (2010) *3rd International Conference on Computer Science and Information Technology*, 108-113.
- [6] Radojevic, B. and M. Zagar, "Analysis of issues with load balancing algorithms in hosted (cloud) environments." In proc. *34th International Convention on MIPRO, IEEE, 2011*.
- [7] Lars Kolb, Andreas Thor, Erhard Rahm, “Load Balancing for MapReduce-based Entity Resolution”, in proc. *28th International Conference on Data Engineering (ICDE), IEEE*, pp: 618-629, 2012.
- [8] Nae V., Prodan R. and Fahringer T. (2010) *11th IEEE/ACM International Conference on Grid Computing (Grid)*, 9-17.
- [9] J. Hu, J. Gu, G. Sun, and T. Zhao, “A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud computing Environment”, *Third International Symposium on Parallel Architectures, Algorithms and Programming (PAAP)*, 2010.
- [10] Dhinesh B. L.D , P. V. Krishna, “Honey bee behavior inspired load balancing of tasks in cloud computing environments”, in proc. *Applied Soft Computing*, volume 13, Issue 5, May 2013, Pages 2292-2303.
- [11] Randles M., Lamb D. and Taleb-Bendiab A.,”A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing “,(2010) *24th International Conference on Advanced Information Networking and Applications Workshops*, 551-556.
- [12] O. Abu- Rahmeh, P. Johnson and A. Taleb-Bendiab, A Dynamic Biased Random Sampling Scheme or Scalable and Reliable Grid Networks, *INFOCOMP - Journal of Computer Science*, ISSN 1807-4545, 2008, VOL.7, N.4, December, 2008, pp. 01-10.



J. Uma, is currently working as an Assistant Professor, in Jansons Institute of Technology, Tamilnadu, India received her B. Tech Degree in Information Technology, from Anna University, Tamilnadu, India during 2009 and M. E in Computer Science and Engineering from Karpagam University, Tamilnadu, India during 2013. Her area of interest includes Load Balancing and Scheduling in Cloud Computing. She has published few papers in both National and International Conferences in many fields.



V. Ramasamy received his B. E Degree under Anna University Chennai, during 2006 and M. E Software Engineering from Anna University Trichy during 2009. He is currently working as an Assistant Professor in the Department of Computer Science and Engineering at Park College of Engineering and Technology. His area of interest includes Load Balancing, Job Scheduling and Security Issues in Cloud Computing and Data mining. He has published many papers in the Cloud Computing in various Conferences and journals.



Kaleswaran.A received his B.Sc (Physics) and MCA from Bharathiar University, Coimbatore and M.E (SE) in Anna University Coimbatore during 2011. Now he is doing Ph.D in Anna University Chennai. He has four years of teaching experience. He is now working as a Assistant Professor in Department of Computer Science and Engineering in Park College of Engineering and Technology, Coimbatore. He has many publications to his credit in various international conferences and journals. He has attended a number of conferences, seminars and workshops His research interest is mainly focused on Data mining and Image Processing.