Advances in Computational Sciences and Technology
ISSN 0973-6107 Volume 9, Number 1 (2016) pp. 51-59

http://www.ripublication.com

# A Comparison and Analysis of Load Balancing Algorithms in Cloud Computing

**Balwinder Kaur[1], Rachhpal Singh[2,*]**

*[1]COEM, Punjabi University Neighborhood Campus, Rampura Phul, Punjab, India.*

*[2]Punjabi University Regional centre for Information Technology and Management, SAS Nagar, Punjab, India.*

**Abstract**

Cloud computing is an initial technology that gives novel ideas for processing based on virtualization of resources. With increment in new applications on the cloud prompts increment the load on the servers. With the expanded burden on sever, the resources are not used proficiently, consequently load balancing has been presented. The primary objective of load balancing is to adjust the load equally among the nodes with the end goal that no node will be over-burden or under stacked. This paper comprises of a comparative study of the different load balancing algorithms in cloud environment.

**Keywords:** Cloud computing, virtualization, load balancing

## 1. INTRODUCTION

These days cloud computing is one of the broadly used technology in the zone of information technology and its empowered services. Due to its significant features and benefits such as high flexibility, scalability, and reliability several service providers and researcher are shifting towards it.

Cloud gives virtual resources and organizations with the objective of low cost. Cloud computing is popular for the most part due to its properties of giving virtualization. Cloud computing provides the resources to the customers according to their need. Cloud Computing has sure benefits, but there are some problems to deal with along with load balancing, scheduling of task, VM migration, security, and many.

* Corresponding author

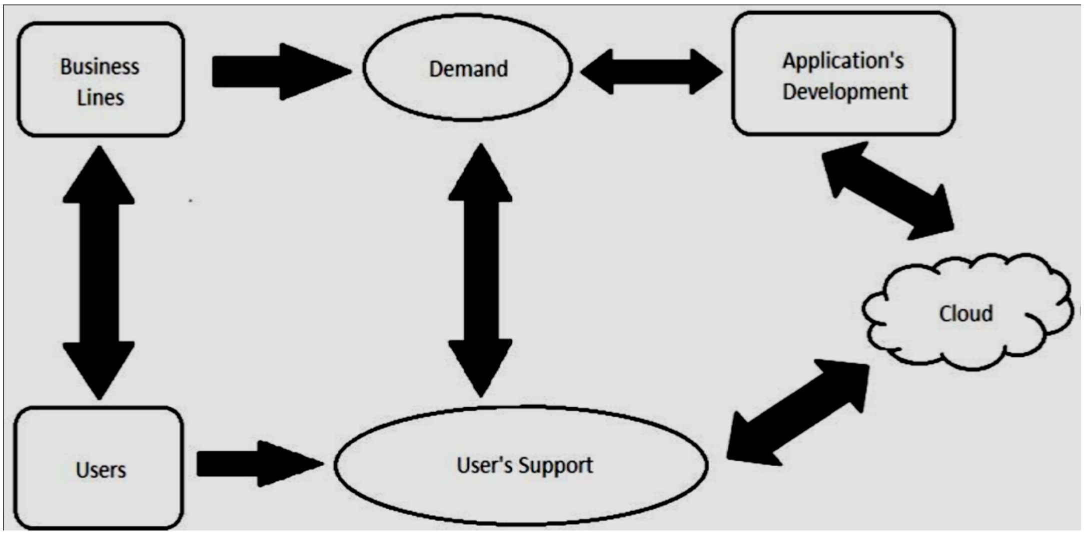


**Figure:** The typical cloud environment

This paper particularly highlights that how load balancing is done effectively. As the area of Cloud computing is obtaining a lot of hot, at a similar time, a lot of intensive task waiting to be processed, the way to assign cloud tasks fairly in order that the nodes within the cloud computing setting will have a balanced load become more grave, this task allocation strategy is named load balancing. Load balancing includes a extended influence on the performance in cloud computing as load balancing aims to reinforce resource utilization, increase throughput, reduce response time, and avoid overload of any single resource. The load balancing, cloud computing additional economical and enhance user satisfaction. Therefore, "it is that the procedure of confirming the equally distribution of work on the pool of system node or processor so the running task is completed with none disturbance". Cloud load balancing is a sort of load balancing that's performed in cloud computing which can be accomplished one by one additionally as on classified basis. There are numerous algorithms designed for balancing the load with totally different tasks.

## 2. LOAD BALANCING IN CLOUD COMPUTING

The major aim of the load balancing is to balance the load expeditiously among the nodes in such the way that no nodes are full and below loaded. There are sure criteria for determining the effectiveness of the load balancing algorithm in Cloud computing atmosphere.[15]

- **Throughput:** The total number of jobs that have execution is called throughput. For the better performance of the system needs a high throughput.
- **Related Overhead:** the quantity of overhead that's created by the execution of the load balancing algorithm. smallest overhead is requiring for the thriving implementation of the algorithm.
- **Fault tolerant:** It is the ability of the algorithm to perform correctly and

uniformly even in conditions of failure at any arbitrary node in the system.

- **Response time:** It is the least time that a distributed system executing particular load balancing algorithm takes to react.
- **Resource Utilization:** It is the degree to that the resources of the system are utilised. a decent load reconciliation algorithmic program provides most resource utilization.
- **Performance:** It describes the usefulness of the system once acting load balancing. If all the higher than parameters are fulfilled optimally then it'll extremely improve the performance of the system.[15]

## 3. OBJECTIVES OF THE LOAD BALANCING

- Preserve the stability of the system
- Significant improvement in the performance
- Enhance flexibility of the system to adapt the adjustments
- Develop a fault tolerance [1][2]

## 4. CLASSIFICATION OF LOAD BALANCING ALGORITHM

### 4.1 Load balancing classification based on process origination they are classified as:

a) Sender Initiated:

In this the process is initiated by the sender; the client sends request until a receiver is assigned to receive the workload sends by client.

b) Receiver Initiated: The process is initiated by the receiver; the receiver sends a request to acknowledge a sender who is ready to share the workload

c) Symmetric: It is a mixture of both sender and receiver-initiated type of load balancing algorithm.

### 4.2 Load balancing classification based on the current state of the system they are classified as:

#### 4.2.1 Static Load Balancing

Static load balancing algorithms are suitable only for the system with minimum variations in load. In the static load balancing algorithm, the current state of the system does not affect the decision of shifting the load. It requires prior knowledge of the resources of the system. At the beginning of the execution, the performance of the virtual machines is determined. According to their performance, the master processor assigns the workload to other slave processors and the output is returned to the master processor by the slave processors.

Due to the non-pre-emption property of these static load balancing algorithms, each machine has at least one job assigned for itself. Minimizing the execution time of the task and limit communication overhead and delays is the primary goal of static load balancing algorithms. Static load balancing algorithms have a problem that the task can be assigned to the processors or machines only after it is produced, and that task cannot be shifted during its execution to any other machine .There are various kind of static load balancing techniques, i.e. Round Robin, Min-Min, Max-Min, CLBDM and Enhanced Map Reduce.[15]

### 4.2.2 Dynamic Load Balancing

In dynamic load balancing algorithms, the current state of the system is used to make any decision about load balancing. Thus, the load offset depends on the current state of the system. For the higher execution, this allows to dynamically move the process to the under loaded machine from the overloaded one. An important feature of this approach is that its load balancing solution is based on the current state of the system, which helps to improve overall performance of the system through dynamic transfer migration. The different types of dynamic load balancing algorithms are Honeybee, Ant colony, Carton, Throttle, Genetic Algorithm, Active Clustering, and OLB+ LBMM.

**Table1: Table of comparison for Static Load Balancing Algorithms based on Qualitative Matrix**

| Static Load Balancing Algorithms | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Load Balancing algorithms** | **Fairness** | **Response time** | **Throughput** | **Overhead** | **Fault tolerance** | **Performance** | **Resource utilization** | **Complexity** |
| **Round Robin [4]** | Yes | Fast | High | High | No | Fast | High | Low |
| **Min-Min [18] [7]** | No | Fast | High | High | No | Fast | High | Low |
| **Max-Min [21]** | No | Fast | High | High | No | Fast | High | Low |
| **CLBDM [22]** | Yes | Fast | High | High | No | Fast | High | Low |
| **Map Reduce [6]** | Yes | Low | High | High | Yes | Fast | High | Low |

**Table 2:** Table of comparison for Dynamic Load Balancing Algorithms based on Qualitative Matrix

| Dynamic Load Balancing Algorithm | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **LB algorithms** | **Fairness** | **Response time** | **Throughput** | **Overhead** | **Fault tolerance** | **Performance** | **Resource utilization** | **Complexity** |
| **Honeybee [3]** | No | Slow | High | Low | No | Slow | High | Low |
| **Ant colony [9][13]** | No | Slow | High | High | N/A | Slow | High | No |
| **Carton [8]** | Yes | Fast | High | N/A | N/A | Fast | High | High |
| **Throttle [13]** | No | Fast | Low | Low | Yes | Fast | High | Low |
| **Genetic Algorithm [23]** | Yes | Fast | Low | High | Low | Low | Les | Low |
| **Active Clustering [24]** | N0 | No | No | Yes | No | No | Yes | No |
| **OLB+LBMM [10]** | No | Slow | High | Low | No | Fast | High | High |

### 4.3 Load balancing classification based on the Spatial Distribution of Nodes:

#### 4.3.1 Centralized Load Balancing

The scheduling decision and allocation decision is performed by a particular node in centralized load balancing Algorithm. This node can apply static or dynamic approach for load balancing and it's liable for collecting knowledge base of whole cloud network. The centralized load balancing Algorithm creates a great overhead on the centralized node but reduces the time required to analyse different cloud resources also this kind of network, if the network is no longer fault tolerant, the recovery might not be easy in case of node failure in this scenario as failure intensity of the overloaded centralized node is high.

#### 4.3.2 Distributed Load Balancing

In this technique, for making resource provisioning or task scheduling decision no single node is accountable. Each machine in the network supports local knowledge base distribution to ensure efficient distribution of tasks in static environment. Even no specific domain is liable for monitoring the cloud network instead multiple domains monitor to make accurate load balancing decision. Hence, no single node is overloaded to make load balancing decision and the system is fault tolerant and balanced as well. The table of comparison of various static and dynamic load balancing algorithms is shown in Table 1. The table comparison is based on spatial distribution of nodes.

### 4.3.3 Hierarchical Load Balancing:

In this, it includes entirely diverse stages of the cloud in load balancing algorithm. This sort of load balancing generally operates in levels. It may be exhibited in tree structure where every node within the tree is balanced under the guidance of its parent node. Here the load balancing is done at the lower level of hierarchy and minimize the amount of information passed to the upper level of hierarchy, due to which there is a reduction in response time and delay time. Parent node will use lightweight agent task to urge statistics of slave nodes or child nodes. It is based mostly upon the knowledge collected by the parent node provisioning or scheduling processes created.

**Table 3:** Table of comparison for Load Balancing Algorithms in Cloud Computing Environment

| Load Balancing Algorithm | Static Environment | Dynamic Environment | Centralized Balancing | Distributed Balancing | Hierarchical Balancing |
|---|---|---|---|---|---|
| **Round Robin [4]** | Yes | No | Yes | No | No |
| **Map Reduce [6]** | Yes | No | No | Yes | Yes |
| **Ant colony [9][13]** | No | Yes | No | Yes | No |
| **Carton [8]** | No | Yes | No | Yes | No |
| **CLBDM [22]** | Yes | No | Yes | No | No |
| **Min-Min [18] [7]** | Yes | No | Yes | No | No |
| **Max-Min [21]** | Yes | No | Yes | No | No |
| **OLB [26]** | Yes | No | Yes | No | No |
| **Genetic Algorithm [23]** | No | Yes | Yes | No | No |
| **Particle Swarm Optimization [25]** | No | Yes | No | Yes | No |
| **Throttle [13]** | No | Yes | Yes | No | No |
| **OLB+ LBMM [10]** | Yes | No | Yes | No | Yes |
| **Active Clustering [24]** | No | Yes | No | Yes | No |

## 5. CONCLUSION

The prime goal of the load balancing is to maximize resource utilization and considerably increase the performance of the cloud system, to increase the customer satisfaction by reducing the response time and the amount of job rejection. This paper presented the comparison of various load balancing algorithms for cloud computing such as, Min-Min, Max-Min, round robin (RR), Carton, Ant colony, Honeybee etc.

The vital part of this paper is comparison of different algorithms considering the characteristics like fairness, performance, throughput, fault tolerance, overhead, and response time and resource utilization and these algorithms also compared based on the spatial distribution of nodes. Load balancing algorithms are depending upon the conditions in which job assigned during compilation time or execution time. The comparison result states that static load balancing algorithms are much secure than dynamic algorithms. However dynamic load balancing algorithms are better than static in line with overload rejection, response & waiting time reliability, fault tolerant, resource utilization, and throughput.

## REFERENCES

[1] A.kumar "Load Balancing in Cloud Data Center Using Modified Active Monitoring Load Balancer" IEEE2016.

[2] D. Escalnte and Andrew J. Korty, "Cloud Services: Policy and Assessment", EDUCAUSE Review, Vol. 46, July/August 2011.

[3] D. B. L.D. and P. Venkata Krishna, "Honey bee behavior inspired load balancing of tasks in cloud computing environments," *Appl. Soft Comput.*, vol. 13, no. 5, pp. 2292–2303, May 2013.

[4] D.Powar, S. Swaroop Moharana, R. D. Ramesh "analysis of load balancers in cloud computing," *International Journal of Computer Science and Engineering (IJCSE),* vol. 2, no. 2, pp. 101–108, 2013.

[5] Garima Gupta, Vimal Kr.Kumawat, P R Laxmi, Dharmendra Singh, Vinesh Jain," A Simulation of Priority Based Earliest Deadline First Scheduling for Cloud Computing System", IEEE 2014

[6] Google App Engine, http://appengine.google.com (April 18, 2010).

[7] H. Chen and F. Wang, "User-priority guided min-min scheduling algorithm for load balancing in cloud computing."*Parallel Computing Technologies (PARCOMPTECH), National Conference IEEE,* 2013

[8] 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.

[9] J. Kaur and S. Kinger, "A survey on load balancing techniques in cloud computing," *International Journal of Science and Research (IJSR)*, vol. 3, no. 6, pp. 2662–2665, 2014.

[10] J. Uma, V. Ramasamy, A. Kaleeswaran "Load Balancing Algorithms in Cloud Computing Environment - A Methodical Comparison" *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, Volume 3, Issue 2, February 2014 (39)

[11] K. Li, G. Xu, G. Zhao, Y. Dong, and D. Wang, "Cloud task scheduling based on load balancing ant colony optimization," in *Sixth Annu. Chinagrid Conf*, pp. 3–9, Aug. 2011.

[12] Li, K., Xu, G., Zhao, G., Dong, Y. & Wang, D. (2011). Cloud task scheduling based on load balancing ant colony optimization. 2011 Sixth Annual ChinaGrid Conference 978-0-7695-4472-4/11 $26.00 © 2011 IEEE. DOI 10.1109/ChinaGrid.2011.

[13] M. Randles, D. Lamb, and a. Taleb-Bendiab, "A Comparative study into distributed load balancing algorithms for cloud computing," *IEEE 24th Int. Conf. Adv. Inf. Netw. Appl. Work.*, pp. 551–556, 2010.

[14] P. V. Patel, Hitesh. D. Patel, Pinal. J. Patel, "A Survey on Load Balancing in Cloud Computing" IJERT, Vol. 1, Issue 9, November 2012.

*[15]* P.Geetha and Dr.C.R.Rene Robin "A Comparative-Study of Load-Cloud Balancing Algorithms in Cloud Environments" *International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS-2017)*

[16] Ravindra A. Vyas, Hardik H. Maheta/Vipul K. Dabhi, H. B. Prajapati" Load balancing using process migration for linux based distributed system" IEEE 2015.

[17] Reena Panwar, Prof. Dr. Bhawna Mallick" Load Balancing in Cloud Computing Using Dynamic Load Management Algorithm"IEEE 2015

[18] S. S. Chauhan and R. C. Joshi, "A weighted mean time min-min maxmin selective scheduling strategy for independent tasks on grid," *Advance Computing patiala Conference (IACC), 2010 IEEE 2nd International,* pp. 4–9, 2010.

[19] Shang-Liang Chen, Yun-Yao Chen ∗, Suang-Hong Kuo "CLB: A novel load balancing architecture and algorithm for cloud services.

[20] Surbhi Kapoor, Dr. Chetna Dabas "Cluster Based Load Balancing in Cloud Computing" IEEE 2015.

[21] U. Bhoi, P. N. Ramanuj, and W. S. Email, "Enhanced max-min task scheduling algorithm in cloud computing," *International Journal of Computer and Information Technology*, vol. 2, no. 4, pp. 259–264, 2013.

[22] Z. Chaczko, V. Mahadevan, S. Aslanzadeh, and C. Mcdermid, "Availability and load balancing in cloud computing," *--International Proceedings of Computer Science and Information Technology*, vol. 14, pp. 134–140, 2011.

[23] Zhao, C., Zhang, S., Liu, Q., Xie, J. & Hu, J. (2009). Independent Tasks Scheduling Based on Genetic Algorithm in Cloud Computing.

[24] Randles, M., Lamb, D., Bendiab, A. T. (2010). A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing. 2010 IEEE 24th International Conference on Advanced Information Networking and

Applications Workshops 978-07695-4019-1/10 $26.00 © 2010 IEEE. DOI 10.1109/ WAINA.2010.85.

[25] Vesna Sesum-Cavic Institute of Computer Languages Vienna University of Technology Wien, Austria, Eva Kühn. Applying swarm intelligence algorithms for dynamic load balancing to a Cloud Based Call Center" 2010 Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems. 978-0-7695-4232-4/10 2010 IEEE DOI 10.1109/SASO.2010.19.

[26] Al Nuaimi, K., Mohamed, N., Al Nuaimi, M. & Al-Jaroodi, J. (2012). A survey of load balancing in cloud computing: challenges and algorithms. 2012 IEEE Second Symposium on Network Cloud Computing and Applications 978-0-7695-4943-9/12 2012 IEEE DOI 10.1109/NCCA.2012.29.