

A Critical analysis on load balancing algorithms in Cloud Environment

Santosh Kumar Paul¹, Sunil Kumar Dhal²

^{1,2} Sri Sri University , Sri Sri Vihar, Ward No-3 Godi sahi, Cuttack, 754006, India

Abstract

Cloud computing is now growing as an emerging computing technology that enables user to avail the computing resources in the form of software, hardware, platform, database, communication, infrastructure working from own computer over internet. The main benefit of cloud computing technology is low cost, data backup, throughput, data privacy, security. To get started in cloud we need zero capital expenditure. The major benefit of cloud is you pay only for what you use. Load balancing in cloud environment is an essential operation. Due to increasing clients in cloud and computing services over internet, load balancing algorithms are essential in data centers. Many different types of algorithms are proposed by different researchers in different point of time. The central plan of load balancing algorithms is to dispense load between servers efficiently so that it improves task scheduling simultaneously with increasing system performance.

In this article, we have explained various load balancing algorithm's based on system state (static/Dynamic). We compared all these algorithms to find out the efficient algorithms depending on various parameters like throughput, response time, fault tolerance, resource utilization, power saving.

Keywords

Active, Throttled, Response Time, VM, Internet, throughput, Biased

1. Introduction

The term "cloud computing" refers to a technology where end users avail various computing services like hardware, software, platform, database, interface through internet [6]. Cloud can be different types like public, private, hybrid, community. Cloud offers different types of services like software, platform, database. Various companies are providing cloud services in these three categories. Cloud can remove many limitations from the traditional computing environment including time, space and cost [12]. The various characteristics of cloud are:

- Elasticity nature of cloud
- Self service provisioning of cloud technology
- API's
- Billing & metering of service usage in a pay-as-you-go.

This paper is structured into following sections: Section 2 gives an idea concerning different types of load balancing techniques. Section 3 explains the different existing algorithms implemented in the cloud environment. Section 4 gives a comparison of various load scheduling algorithms. Section 5 gives an overall ending of this paper.

2. Load Balancing

In cloud computing environment a field which has significant importance is load balancing on servers. Various algorithms are suggested by numerous researchers on load balancing. An important job of these algorithms is to allocate load efficiently between servers in such a means that, servers never be overloaded or underloaded. A key plan of most of the load scheduling algorithms is, to enhance the user happiness and maximum resource uses thus improves overall system performance. Load balancing algorithms can be classified on the basis of system state

2.1. Static Load Balancing Algorithm

These algorithms assign task to nodes based on prior knowledge of nodes capability[1]. These algorithms don't consider the dynamic load changes during execution time. Static load balancing algorithms never depends on networks current status.

2.2. Dynamic Load Balancing Algorithm

These algorithms works in the environment, where task is circulated between different virtual machines during execution time. It looks for the server having nominal load inside the network and after that assign proper load to it [2]. Dynamic load balancing algorithms consider the present status & traffic of systems. These algorithms well thought-out as difficult however enhanced typical throughput and fault tolerance.

2.2.1. Policies of dynamic load balancing algorithm

Dynamic Load balancing algorithms contains 6 major policies [4]:

- **Transfer Policy:** It is needed when any job must be shifted from one to other node.
- **Selection Policy:** Here importance is given for selecting right processor for load shift so that throughput & response time might be increased.
- **Location Policy:** It is the identification of critical resources for offering services and the selection of resource locations.
- **Information Policy:** Here information linked to load regarding the node is acquired.
- **Load Estimation Policy:** Here total load of a node inside a system is calculated.
- **Process Transfer Policy:** Under this policy a process execution is determined whether it is locally or remotely.

3. Review

This section describes different existing load balancing algorithms based on probabilistic approach and soft computing approach. These approaches can be implemented in cloud environment.

- **Round Robin algorithm**
Here a predetermined time slice is assigned to all virtual machines for executing a task[5,8]. After time slice over, again another virtual machine is given time slice for execution of job. In this way round robin algorithm gives time slice to each virtual machine for execution of task. Due to its uniform movement in a circular fashion some nodes are overloaded, while others are underloaded. It doesn't consider the current workload of the server.

- **Weighted Round robin algorithm**

Here every virtual machine is allotted weightage depending on the capacity of virtual machines. As per the weightage assigned to the virtual machine, it would receive appropriate requests. In real time cloud computing environment it is very difficult to forecast the accurate execution time. Therefore this algorithm is not a good choice [9].

- **Central manager algorithm**

In the central manager algorithm, a node is treated as central manager node which selects a node which is least loaded depending on the server's current system load information [10]. As all the servers update their load condition to central manager, this results to excess inter process communications so network congestion occurs.

- **Active monitoring**

This type of technique identifies information about each Virtual Machine and the amount of tasks presently allotted to each virtual machine. Whenever a new request for a Virtual Machine enters, it determines which Virtual Machine is the least loaded. Now the new request allocated to the least loaded VM. If number of virtual machines are more than one detected then by default the 1st one is selected. The main disadvantage of this algorithm is whether a VM is previously utilized or not because some VM's are over utilized and some are remain ideal.

- **Throttled load balancing algorithm**

In this algorithm status of VM (idle/busy) is identified. At the load balancer, the data is saved in an index table. This algorithm find out appropriate VM for assignment of job. The job manager contains index list of VM's to find a machine so that it allot required job to the appropriate virtual machine [7].

A disadvantage of these algorithm is, when all machines contains analogous hardware configurations it can only work.

- **Threshold load balancing algorithm**

Here 1st it considers loading threshold of each server depending on its processing speed & storage capacity. After that we choose a method to distribute the load among servers depending on the threshold.

The threshold represents the storage percentage of every datacenter that should not be exceeded.

- **Opportunistic algorithm**

This algorithm never thinks about the present workload of each Virtual machine's. it always try to maintain every node busy. The tasks are assigned randomly to the nodes. It is a slow process as it never calculate the present work load during run time of the node.

- **Min-Min algorithm**

This algorithm chooses a number of unassigned tasks. Initially smallest completion time for al tasks is calculated. After that the task having less completion time is chosen for execution until all unassigned tasks are completed[3]. The major disadvantage of the algorithm is some tasks are suffered with starvation problem.

- **Max-Min algorithm**

Here it calculates smallest completion time of all tasks. Out of which the job having maximum execution time is allocated to the VM. Max -Min algorithm executes shortest jobs concurrently with longest jobs[11]. The main drawback is, jobs having minimum execution time suffer starvation problems.

- **Randomized algorithm**

This algorithm handles a node 'n' with probability 'p'. when processors are of equal load this algorithm works good. The major problem arises when loads are of different run time. It never shows deterministic approach.

- **Ant colony optimization**

This algorithm resembles the activities of ants finding for food. It aims to find optimal solution .it is a random search algorithm . Ants are moving from their nest to find food and join everyone by pheromone which is a volatile substance deposited on paths they travelled. The ant will reach near food by following the trace of pheromone. The concentration of pheromone may changes depending on many factors like the food quality source, distance from food. Now ants utilize the pheromone deposits so that to go to next node. The demerit of this algorithm is premature convergence and stagnating output.

- **Active clustering**

In active clustering more than two servers united & handle the network load and working as a team, they distribute it to the network servers. The load balancer keeps this information requests from users and store it in cache. This process minimizes network traffic load. Server clustering is a way of making multiple computer servers into a cluster, acts like a single server.

The disadvantage is it degrades as system diversity increases.

- **Biased random sampling**

This algorithm uses random sampling methods for load balancing between nodes. The nodes are selected which are servers. This algorithm uses virtual graph, where each side represents each node's load. All nodes are regarded as a directed graph's vertex. Now the node having minimum degree one assigned the task by load balancer. The server now execute the task showing the decrease of available free resources. Due to additional overhead when servers increases degrades the performance, is the biggest disadvantages of this algorithm.

- **Honeybee foraging**

Honeybee Foraging is inspired by honeybee foraging habits. There are numerous varieties of honeybees, such as finders and reapers. Initially, the finders move around and look for honey sources. They return to the honey comb after discovering the source and begin a waggle dance to demonstrate the quantity and quality of honey present. Then after, reapers go out and collect honey from various locations[13]. They perform a waggle dance and then return to harvest honey. The throughput is very less in this algorithm is the major disadvantage.

- **Central queue**

This algorithm stores newly arrived tasks & unfulfilled requests in a FIFO queue in a cyclic manner. All the new tasks inserted in the queue by the queue manager. The Manager after finding a new request, it removes the first operation from the queue and delivers this to the user. When there are no activities within queue, it is considered empty. When a request is made, a new activity is buffered. The queue manager scans the queue for any unanswered requests, assigns a new activity, and removes the first such request. When all remote servers are overloaded this algorithm assigned all processes locally, is the main disadvantages of this algorithm.

- **Weighted least connection**

It is created on the principle of Least connection algorithm used for different application servers. Based on traffic handling capacity the administrator allocates weight to each server. The computational time is very high in this algorithm[14,16].

- **Particle swarm optimization**

It is a stochastic optimization method works on the movement of intelligent swarms. Here for getting a solution from a problem social interaction is used. This algorithm uses many particles (agents) that forms a swarm moving around for search of space thus looking for best solution of a problem. Particle Swarm Optimization was proposed by Kennedy and Eberhart in 1995 [15]. As mentioned in the original paper, sociobiologists believe a school of fish or a flock of birds that moves in a group “can profit from the experience of all other members”[15].

The downsides of the algorithm are that it will be easy to slip into a local optimum in a high-dimensional region and that the iterative approach has a slow rate of convergence.

- **Least connection**

Least connection algorithm is dynamic in nature. It considers least number of active connections for client requests distributed in the application server. This algorithm always takes number of active connections into consideration. Here the algorithm distributes load by choosing least number of active connections. The demerits are system is heavily loaded and performance is degraded.

- **Stochastic Hill climbing**

Using a local optimization strategy, the allocation of arriving jobs toward servers or virtual machines is done via stochastic hill climbing. A stochastic and local optimization method is nothing more than a loop that travels uphill in the direction of increasing value. It comes to a halt when no neighbor has a better worth than it. The likelihood of selection varies with the steepness of the uphill motion in this form, which chooses at random from among the uphill moves. As a result, by making slight adjustments to the original assignment, it maps assignments to a collection of assignments. Each member of the set is assessed against a set of criteria in order to get closer to a valid assignment and enhance the state's evaluation score. The following assignment is given to the set's best element. This basic operation is continued until a solution or a stopping requirement is met. The disadvantages of the hill climbing algorithm include the fact that it is totally centralized and only run through one mode, resulting in a bottleneck as well as a failure.

4. Findings of various load balancing algorithms

This section gives an idea on different existing load balancing algorithms based on various factors. Table -1 explains different types of load balancing techniques (static, Dynamic) with different parameters used for load balancing. Different parameters like Fault tolerance, Migration time, Network overhead, Adaptability, Scalability, Throughput, Resource utilization, Response time, Power Saving, Reliability, Predictability, Process Migration, performance.

Table 1. Findings of various load balancing algorithms

Load Balancing Algorithms	Static / Dynamic nature	Fault tolerance	Migration time	Network overhead	Adaptability	Scalability	Throughput	Resource utilization	Response time	Power Saving	Reliability	Predictability	Process Migration	performance
Round Robin	Static	No	No	Yes	less	Yes	Yes	Yes	Yes	No	Less	More	No	Yes
Central manager	Static	Yes	No	No	Less	Less	No	No	No	No	Less	More	No	Yes
Throttled load balancing	Static	Yes	Yes	No	Less	Yes	No	Yes	Yes	No	Yes	more	Yes	Yes
Threshold	Static	No	No	No	Less	Yes	No	No	No	No	Less	More	No	Yes
Opportunistic load balancing	Static	No	No	No	Less	No	No	Yes	No	No	less	less	No	Yes
Min-Min	Static	No	No	Yes	Less	No	Yes	Yes	Yes	No	less	less	No	Yes
Max-Min	Static	No	No	Yes	Less	No	Yes	Yes	Yes	No	less	less	No	Yes
Randomized	Static	No	No	Yes	Less	No	No	No	No	No	less	More	No	Yes
Ant colony optimization	Dynamic	Yes	Yes	Yes	More	No	No	Yes	No	No	More	less	Yes	Yes
Active clustering	Dynamic	No	Yes	Yes	More	No	Yes	Yes	No	No	Yes	less	Yes	No
Biased random sampling	Dynamic	No	No	Yes	More	No	Yes	Yes	No	No	Yes	less	No	Yes
Honeybee foraging	Dynamic	Yes	No	No	More	No	No	Yes	No	No	More	less	Yes	Yes
Central queue	Dynamic	Yes	Yes	Yes	more	No	Yes	No	Yes	No	More	less	No	Yes
Least connection	Dynamic	No	Yes	Yes	More	Yes	Yes	Yes	No	No	More	less	No	Yes
Particle swarm optimization	Dynamic	No	No	No	More	No	Yes	Yes	Yes	No	Yes	less	No	Yes
Stochastic hill climbing	Dynamic	No	No	No	More	No	Yes	Yes	Yes	No	No	less	No	Yes

5. Acknowledgements

Load balancing [1] is considered as a vital work within cloud environment for best deployment of cloud resources .Due to the proper load balancing approach the data centers can manage VM's efficiently. We have described and performed comparative analysis of a range of load balancing algorithms along with their static or dynamic nature based on different parameters like fault tolerance ,throughput, response time, resource utilization. After analyzing algorithm based on various parameters we found that static algorithms are stable in comparison to dynamic .However dynamic load balancing algorithms[1] are preferred choice compared to static as it considers the present workload of the servers. From the above observation by considering various parameters we found

that , Ant colony algorithm, honey bee foraging algorithm , central queue algorithms are better in fault tolerance approach where as Round robin, Weighted round robin, Min-min, max-min are better in throughput, resource utilization and response time. During our *subsequently* future effort we must compare algorithms having simulation work and mathematical investigation on various additional parameters to determine the best algorithm on load balancing in cloud environment.

6. References

- [1] Al Nuaimi, K., Mohamed, N., Al Nuaimi, M., Al-Jaroodi, J.: A survey of load balancing in cloud computing: Challenges and algorithms. In *2012 second symposium on network cloud computing and applications* (pp. 137-142), IEEE (2012).
- [2] Ghutke, B., Shrawankar, U.: Pros and cons of load balancing algorithms for cloud computing. In *2014 International Conference on Information Systems and Computer Networks (ISCON)* (pp. 123-127). IEEE, (2014).
- [3] Gupta, S., Rani, S., Dixit, A., & Dev, H.: Features exploration of distinct load balancing algorithms in cloud computing environment. *International Journal of Advanced Networking and Applications*, 11(1), 4177-4183. (2019)
- [4] Kumar, S., & Rana, D. S. (2015). Various dynamic load balancing algorithms in cloud environment: a survey. *Int. J. Comput. Appl.*, 129(6), 0975-8887.
- [5] Kashyap, D., & Viradiya, J.: A survey of various load balancing algorithms in cloud computing. *Int. J. Sci. Technol. Res.*, 3(11), 115-119. (2014)
- [6] Ray, S., & De Sarkar, A.: Execution analysis of load balancing algorithms in cloud computing environment. *International Journal on Cloud Computing: Services and Architecture (IJCCSA)*, 2(5), 1-13. (2012)
- [7] Ramya, R., Kriushanth, M., & Arockiam, L.: A state-of-art load balancing algorithms in cloud computing. *International Journal of Computer Applications*, 95(19)(2014)
- [8] Meenakshi, Comparative Study of Load Balancing Algorithms in Cloud Computing Environment, *International Journal of engineering research & technology (IJERT)* Volume 02, Issue 10 (October 2013)
- [9] Beaulah Soundarabai, P., Thriveni, J., Venugopal, K. R., & Patnaik, L. M. Comparative study on load balancing techniques in distributed systems. *International Journal of Information Technology and Knowledge Management*, 6(1), 53-60, (2012)
- [10] Rajguru, A. A., & Apte, S. S.: A comparative performance analysis of load balancing algorithms in distributed system using qualitative parameters. *International Journal of Recent Technology and Engineering*, 1(3), 175-179. (2012)
- [11] Gopinath, P. G., & Vasudevan, S. K.: An in-depth analysis and study of Load balancing techniques in the cloud computing environment. *Procedia Computer Science*, 50, 427-432. (2015)
- [12] Mandal, S., Khan, D. A., & Jain, S. : Cloud-based zero trust access control policy: an approach to support work-from-home driven by COVID-19 pandemic. *New Generation Computing*, 39(3), 599-622. (2021)
- [13] Milan, S. T., Rajabion, L., Ranjbar, H., & Navimipour, N. J. (2019). Nature inspired meta-heuristic algorithms for solving the load-balancing problem in cloud environments. *Computers & Operations Research*, 110, 159-187.
- [14] Zhang, M., Mao, J., Chen, L., Li, N., Fan, L., & Shuang, L. (2021, July). A Load-Balancing Algorithm for Power Internet of Things Resources Based on the Improved Weighted Minimum Number of Connections. In *International Conference on Artificial Intelligence and Security* (pp. 563-574). Springer, Cham.
- [15] <https://machinelearningmastery.com/a-gentle-introduction-to-particle-swarm-optimization/>
- [16] Mesbahi, M., & Rahmani, A. M. (2016). Load balancing in cloud computing: a state of the art survey. *Int. J. Mod. Educ. Comput. Sci.*, 8(3), 64.