IJCSE International Journal of Computer Sciences and Engineering Open Access

Survey Paper Vol.-7, Issue-2, Feb 2019 E-ISSN: 2347-2693

A Survey of Load Balancing Algorithms in Cloud Environment

J. M. Tandel[1*], H. R. Patel[2]

[1,2]Dept. of Computer Engineering & Information Technology, Government Engineering Collage, Modasa, India

[*]*Corresponding Author: jaytandel993@gmail.com*

DOI: https://doi.org/10.26438/ijcse/v7i2.294299 | Available online at: www.ijcseonline.org

Accepted: 14/Feb/2019, Published: 28/Feb/2019

Abstract— Cloud computing provides storing and accessing of your data over the internet. Cloud computing delivers computing services (servers, databases, networking, software etc.) over the internet. There are various advantages of cloud computing including Virtual computing environment, On-demand services, Maximum resource utilization and easier use of services etc. Still, there are numerous issues in cloud computing related to Security, Resource provisioning, Server consolidation, and Virtual machine migration. Load Balancing is an essential task in the Cloud Computing environment to achieve maximum utilization of resources, minimize the response time and maximize the throughput of the overall system. Load balancing algorithms increase the efficiency of the system by equally distributing the workload among the completion process. In this paper, we have presented the performance analysis of various load balancing algorithms based on various dependent parameters by considering two main load balancing approaches: static and dynamic. The both types of the load balancing algorithm have some advantages as well as disadvantages. The main purpose is to analyze different algorithms based on the time factor.

Keywords— Cloud computing, Load balancing, Static load balancing, Dynamic load balancing.

I. INTRODUCTION

“Cloud is a parallel and distributed computing system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements (SLA) established through negotiation between the service provider and consumers [1]”.

Cloud computing gives an adaptable method to hold information and records which include virtualization, appropriated processing, and web services. It likewise has a few components like the customer and distributed servers. The point of cloud computing is to give the greatest services with the least expense whenever. These days, there are in excess of a huge number of PC gadgets associated with the internet. These gadgets present their demand, what's more, get the reaction immediately. The principal targets of the cloud are to lessen the cost, upgrade reaction time, give better execution, subsequently Cloud is additionally called a pool of services. The load has various types like CPU load, network load, memory capacity issue and so on [2]. With regards to cloud computing, load balancing is to share a heap of virtual machines overall hubs (end client gadgets) to enhance resources, service utilization and gives high fulfilment to clients. Because of load sharing, each hub can work productively, information can be gotten and sent immediately. Service Level Agreement (SLA) and client fulfillment could be given by phenomenal load balancing techniques. Hence, giving productive load-balancing algorithms and mechanisms is key to the achievement of a cloud computing environment.

A site or a web-application can be gotten to by numerous clients any time of time. It is exceptionally troublesome for a web application to deal with all these clients ask for at one time. Once in a while, it might result in system breakdowns. Here, the load balancer assumes a vital job.

The dynamic load balancing algorithm utilizes system data while distributing the load. A dynamic conspire is increasingly flexible and fault tolerant. Load balancing empowers advance network facilities and resources for better response and execution. A few algorithms are utilized to balance cloud information among hubs. All the client load is deal with by the cloud supplier for the smooth provisioning of services [3].

The organization of the paper as follows, Section I contains the introduction of cloud computing and load balancing, Section II contains the related work of load balancing strategy, Section III presents the conclusion.

II. RELATED WORK

Load balancing is a procedure or strategy that is utilized to convey an expansive handling load, uniformly and progressively to every one of the hubs on the cloud. The best possible load balancing gives great resource usage along these lines limiting the utilization of resources which further outcomes in higher client fulfilment. Aside from this, it spares vitality utilization which helps in the perfect and green environment.

Goals of Load balancing

1. To improve the performance substantially.
2. To have a backup plan in case the system fails even Partially.
3. To maintain system stability.
4. To accommodate future modification in the system.

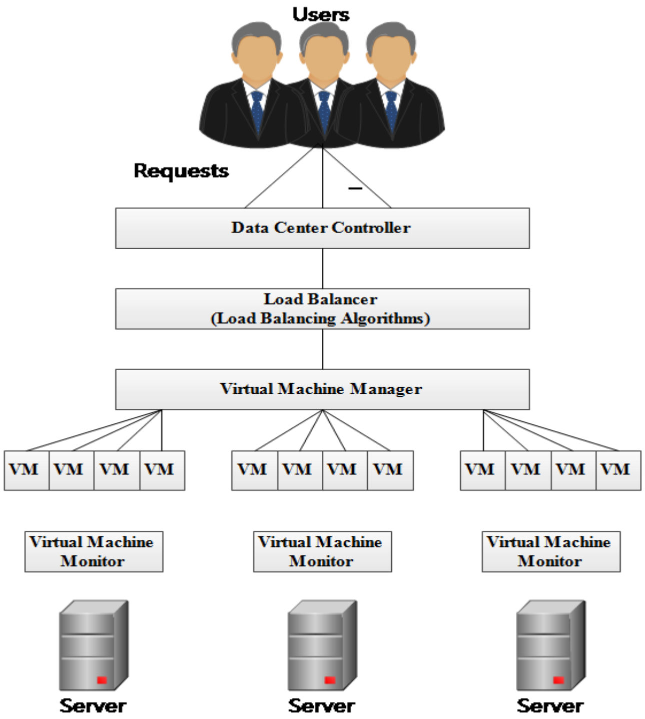

Figure 1. *Load Balancing in cloud computing*.

Load Balancing Metrics

There are a significant number of metrics that can be improved to acquire a progressively attractive cloud load balancing [3], [4].

Throughput- This is the entire task which includes finished execution inside the gave measure of the time limit. This time is required to have a superior throughput for the high efficiency of the system.

Overhead- It characterizes the computation of working cost, which is made while the execution of the load balancing methods. It ought to limit for better execution of the strategy.

Fault Tolerance- It is the capacity to manage effectively as well as continuing in the condition of breakdown at any one node in the system. A fault-tolerance parameter is useful in order to change to the different nodes that resolve faulty nodes.

Transfer Time- This is the time to relocate or move the resources of one specific node to another node for execution. This time will need to be minimized for better efficiency of the system.

Reaction Time- The measure of time in use by a specific load balancing system to react in a distributed environment. This time length will need to be minimal to help for better execution.

Resource Utilization- This specifies the measure to ensure the resources of the framework to be used. This parameter should be maximized for resource utilization.

Scalability- It specifies the capability of the framework to realize the technique for load balancing using a negligible amount of CPUs. This parameter can be progressed for more attractive system execution.

Performance- It stands to the entire efficiency of the system in the wake up of delivering load balancing. In case all the above metric are fulfilled, it will help better execution of the system.

Taxonomy of Load Balancing Algorithms

Load balancing algorithms were classified based on two factors: the state of the system and the person who initiated the process. Algorithms based on the state of the system are classified as static and dynamic.
Depending on who initiated the process, load balancing algorithms can be of three categories.

Sender Initiated- If the load balancing algorithm is initialized by the sender.

Receiver Initiated- If the load balancing algorithm is initialized by the receiver.

Symmetric- It is the combination of both sender-initiated and receiver initiated.

Depending on the current state of the system, load balancing algorithms can be of two categories.

Static- It doesn't depend on the current state of the system. Prior knowledge of the system is needed [3].

Dynamic- Decisions on load balancing are based on the current state of the system. No prior knowledge is needed. So it is better than static approach [3].

Static Load Balancing Algorithm

In static load balancing, it utilizes prior knowledge of the application and measurable data about the system and conveys the load proportionally between servers [4].

Round robin algorithm-

In the Round Robin algorithm, it is one of the most straightforward scheduling algorithms that use the standard of time slices. Here, time is isolated into various slices and every node is given a specific time interim. Every node is given a quantum and in this given quantum node needs to play out its tasks. In the event that the client asks for finishes inside time quantum, at that point, the client ought not to hang tight generally need to sit tight for its next opening. It implies that this algorithm chooses the load arbitrarily, while for some situation, some server is intensely loaded or somebody is softly loaded [6].

Pooja samal et al. examined some fundamentals of cloud computing, load balancing and proposed improvisation for both resource utilization and job response time by breaking down the RR, MRR, TSPBRR algorithm. It can be observed that the response time is better in the case of TSPBRR when contrasted with different variations. It can also be seen that the Avg TAT and Avg WT in TSPBRR better than that in MRR calculation [7].

The Merits of this algorithm are fixed time slice which gives better execution for short CPU blasts. It likewise utilizes running time and landing time. Demerits of this algorithm are Resource Utilization which isn't better and large task takes more time for completion. As a result, at any moment some node may possess heavy load and others may have no request. Be that as it may, this issue was overwhelmed by the weighted round robin algorithm [8].

Min-Min algorithm-

Min-Min Load Balancing Algorithm in the cloud administrator distinguishes the execution and finishing time of the unassigned tasks holding up in a line. This is a static load balancing algorithm so the parameters identified with the activity are known ahead of time. In this kind of algorithm, the cloud supervisor first manages the occupations having least execution time by doling out them to the processors as indicated by the capacity to complete the activity in the predefined finishing time. The occupations having the most extreme execution time need to sit tight for the unspecific period. Until every one of the tasks is appointed in the processor, the doled out tasks are refreshed in the processors and the task is expelled from the holding up the line [9].

Huankai chen et al. proposed that two new systems of the task scheduling algorithm can diminish the activity's consummation time, which improves the load balance and fulfill client's need requests in the cloud. As indicated by the outcome, the proposed algorithm beats the Min-Min calculation as far as makespan, load balancing and client need mindful. [10].

The Merits of the algorithm is less completion time value and in the proximity of all the tasks which demonstrates the best outcome. Demerits are starvation, machine and undertakings variety which can't be anticipated.

Max-Min algorithm-

The Max-Min algorithm works like the Min-Min algorithm aside from the accompanying: in the wake of discovering the base execution time, the cloud manager manages tasks having most extreme execution time. The doled out task is expelled from the rundown of the tasks that are to be appointed to the processor and the execution time for every other task is refreshed on that processor. As a result of its static methodology, the necessities are known ahead of time then the algorithm performed well. An upgraded variant of the max-min algorithm was proposed. It depends on the situations where meta-tasks contain homogeneous tasks of their finish and execution time. Improvement in the productivity of the algorithm is accomplished by expanding the chance of simultaneous execution of tasks on resources. [11].

O. M. Elzeki et al. presented RASA, which offers an improved task scheduling algorithm based on Max-min to determine the above referenced issue with both Max-min and Min-min. The fundamentals of an improved adaptation of the Max-min allots the task with most extreme execution time to resource to produce least total time as opposed to the first Max-min algorithm which allocates the assignment with great completion time to the resource with least execution time [12].

The Merits of the algorithm is that the requirements are prior known as results it works better. Only demerit is it takes more completion time.

OLB algorithm-

This Opportunistic Load Balancing Algorithm is a static load balancing algorithm so it doesn't consider the present outstanding task at hand of the VM [13]. It endeavors to keep every node occupied. This algorithm bargains rapidly with the unexecuted tasks in an arbitrary request to the presently accessible node. Each task is allocated to the node arbitrarily [14]. It gives a load balance plan without great outcomes. The task will process it gradually in a way since it doesn't ascertain the present execution time of the node and the mix of OLB (Opportunistic Load Balancing) and LBMM (Load Balance Min-Min) Scheduling algorithms to use better execution productivity and keep up the load balancing of the system [15].

The Merits of the algorithm deals rapidly with the unexecuted tasks in an irregular request to the current accessible node and each task is allocated to the node haphazardly. Demerits, it gives load balance schedule without great outcomes. The task will be processed moderately in the way since it doesn't calculate the present execution time of the node.

Table 1: Various metrics have been considered to compare different techniques

	RR	**Min Min**	**Max Min**	**OLB**
Throughput	High	Moderate	High	High
Response time	High	High	High	Low
Overhead	High	High	High	Low
Fault tolerance	-	-	-	-
Performance	Fast	High	High	Moderate
Resource utilization	High	High	High	High
Speed	High	High	Low	Low
Complexity	Low	Low	Low	High

Dynamic Load Balancing Algorithm

Dynamic load balancing algorithms are those algorithms that scan for the lightest server in the system and after that assigned a proper load on it. The algorithms in this class are thinking about complex, yet have better adaptation to internal failure and general execution [4].

Ant Colony Optimization algorithm-
Ant colony optimization is a population-based meta-heuristic that can be utilized to discover estimate answer for troublesome advancement issues [16].

M.Padmavati talked about the essential presentation of cloud computing and load balancing algorithms. Quickly depict the standard ant colony load balancing algorithm ACO to perform load balance among the system existing in server farms. Proposed dynamic and elasticity ant colony load balancing algorithm by applying the pheromone updating and VM choosing for next task modifications. Another load balancing algorithm had been proposed and the calculation normal makespan not exactly existing ACO and it was assessed by the open source cloudsim toolkit. [17].

Sabit et al. proposed the fundamental thought of Ant colony improvement and subtleties of different ACO based scheduling algorithm. Proposed an Ant Colony Optimization (ACO) based task scheduling (ACOTS) algorithm to streamline the makespan of the system and reducing the average waiting time. The designed algorithm is executed and reproduced in CloudSim simulator. Consequences of reproductions are contrasted with Round Robin and Random algorithms which show agreeable yield [18].

The Merits of the algorithm is, faster information can be collected by the ants, minimizes make span, independent tasks, computationally intensive. Demerit is network overhead so search takes a long time and no clarity about the numbers of ants.

Honey Bee algorithm-
Honeybee Foraging Load Balancing Algorithm It's a nature-inspired decentralized load balancing technique, which helps to achieve load balancing across heterogeneous virtual machines of cloud computing environment through local server action and maximize the throughput [19].

Harshit gupta et al. proposed a stream outline for load balancing in cloud computing conditions dependent on the conduct of honey bee foraging strategy. The tasks are to be sent to the under loaded machine and like searching honey bee, the following tasks are likewise sent to that virtual machine till the machine gets over-load as flower patches abuse is finished by scout honey bees. Honey bee conduct propelled load balancing improves the general throughput of preparing and need put together offsetting concentrations with respect to diminishing the measure of time a task needs to look out for a queue of the VM. In this way, it reduces the response of time of VMs. [20].

Ashish Soni et al. displayed the proposed algorithm, In that fitness values are concluded by minimizing differences of requested load and served load, requested priority and served priority, and also minimizing total execution error. According to the outcomes, the proposed calculation can give a superior answer for the cloud systems. [21].

The merits of this algorithm are increased throughput and minimized response time. Demerit is high priority task can't work without VM machine.

Genetic algorithm-
Genetic Algorithm is based on the biological concept of generating the population [22]. GA is considered a rapidly growing area of Artificial Intelligence [23]. By Darwin's theory of evolution was inspired by the Genetic Algorithms (GAs). According to Darwin's theory, the term "Survival of the fittest" is used as the method of scheduling in which the tasks are assigned to resources according to the value of the fitness function for each parameter of the task scheduling process [24].

Safwat hamad et al. presented a task scheduling algorithm in the Cloud computing environment dependent on a Genetic Algorithm for allotting and executing autonomous tasks to improve task finish time, decline the execution cost, just as, maximize resource utilization [23].

Garima et al. proposed an algorithm which improves the overall response time of tasks while scheduling them in various VM. In author's work, they considered the wellness capacity of each task before scheduling them to a specific preparing node. Improvised Genetic Algorithm (IGA) was simulated in the MATLAB R 2010 toolkit [25] by the authors.

The Merits of this strategy is that it can handle a vast search space applicable to the complex objective function and can avoid being trapped in a locally optimal solution. Negative marks of this algorithm is that there no assurance to locate the ideal arrangement.

***Particle Swarm Optimization**-*

Particle Swarm Optimization (PSO) as a meta-heuristics strategy is a self-versatile worldwide inquiry-based enhancement system presented by Kennedy and Eberhart [28].

Ajeena beegom et al. displayed the issue as an imperative bi-target optimization problem, where the goals are make span, cost and have utilized the Particle Swarm Optimization algorithm to solve the equivalent, where the Pareto optimality was accomplished through weighted aggregate methodology. A variation of PSO technique (Integer-PSO) was proposed by authors, whose outcomes are promising [26].

Kai pan et al. described that an improved particle swarm optimization (IPSO) is straightening out the meaning of the particle's position and speed and principles for updating, correspondingly altering its fitness value. The experiment was directed by contrasting this current paper's improved particle swarm optimization (IPSO) against the Max-Min algorithm and the IABC algorithm which were aforementioned [27].

The Merits of this algorithm is that the training speed is fast, the efficiency is high and simple algorithm. Demerits, are easy to fall into local optimal solution and poor handling of discrete optimization problems.

Table 2: Various metrics have been considered to compare different techniques

	ACO	**HONEY BEE**	**GA**	**PSO**
Throughput	High	High	Low	High
Response time	Low	Low	Low	High
Overhead	High	Low	Low	Low
Fault tolerance	High	Low	Low	Low
Performance	Low	Low	High	Moderate
Resource utilization	High	High	High	High
Speed	Moderate	High	Moderate	High
Complexity	Low	Low	Low	High

III. Conclusion

Load balancing is one of the vexing challenge in cloud computing. We have studied different load balancing algorithms in the Cloud environment. The different load balancing algorithms are additionally being thought about here based on various types of the parameter. The motivation behind this paper was to gather information about various load balancing algorithms dependent on distinguished subjective parameters. In this paper, we have completed the examination of various load balancing algorithms along with different parameters which can be utilized in order to check the outcomes. Load balancing algorithms are absolutely reliant upon the assignment of the workload at compile time or execution time. As per shown in above comparison table, static load balancing algorithms are steadier than dynamic. Based on description of the overload dismissal, reliability, adaptability, cooperativeness, fault tolerant, resource utilization, response & waiting time and throughput we can infer that Dynamic approach is far superior to static one.

Acknowledgment

I would like to thank Prof. Hiren Patel and Prof. Anil Prajapati Assistant Professor of Department of Computer Engineering & Information Technology, Government Engineering College from Modasa who have contributed towards the development of the template.

References

[1] R. Buyya, J. Broberg, A.M. Goscinski, "*Cloud computing: Principles and paradigms*", John Wiley & Sons, 2010.

[2] Y. Jadeja, K. Modi, "Cloud computing-concepts, architecture and challenges", In Computing, International Conference on Electronics and Electrical Technologies, Kumaracoil, India, pp. 877-880, 2012.

[3] G. Rastogi, R. Sushil, "Analytical literature survey on existing load balancing schemes in cloud computing", International Conference on In Green Computing and Internet of Things, Noida, India, pp. 1506-1510, 2015.

[4] A. S. Milani, N. J. Navimipour, "*Load balancing mechanisms and techniques in the cloud environments: Systematic literature review and future trends.*" Journal of Network and Computer Applications, Vol.71, pp.86-98, 2016.

[5] N. Rathore, I. Chana, "*Load balancing and job migration techniques in grid: a survey of recent trends.*" Wireless personal communications, Vol.79, No.3, pp. 2089-2125, 2014.

[6] K. Mahajan, A. Makroo, D. Dahiya, " *Round robin with server affinity: a VM load balancing algorithm for cloud based infrastructure*", Journal of information processing systems, Vol.9, No.3, pp. 379-394, 2013.

[7] P. Samal,P. Mishra, "*Analysis of variants in Round Robin Algorithms for load balancing in Cloud Computing*", International Journal of computer science and Information Technologies, Vol.4, No.3, pp.416-419, 2013.

[8] V. Shinde, A. Dange, M.A. Lambay, *"Load Balancing Algorithms in Cloud Computing*", International Journal of computer science trends and technology, No.4, pp.75-81, 2016.

[9] U. Bhoi, P.N. Ramanuj, "*Enhanced max-min task scheduling algorithm in cloud computing*", International Journal of

Application or Innovation in Engineering and Management, Vol.2, No.4, pp.259-264, 2013.

[10] H. Chen, F. Wang, N. Helian, G. Akanmu, "User-priority guided Min-Min scheduling algorithm for load balancing in cloud computing", National Conference on Parallel Computing Technologies, Bangalore, India, pp.1-8, 2013.

[11] S. Ray, A. De Sarkar, "*Execution analysis of load balancing algorithms in cloud computing environment*", International Journal on Cloud Computing: Services and Architecture, Vol.2, No.5, pp.1-13, 2012.

[12] O. M. Elzeki, M. Z. Reshad, M. A. Elsoud, "*Improved max-min algorithm in cloud computing*", International Journal of Computer Applications, Vol.50, No.12, pp.22-27, 2012.

[13] J. Uma, V. Ramasamy, Kaleeswaran, "*A Load Balancing Algorithms in Cloud Computing Environment-A Methodical Comparison*", International Journal of Advanced Research in Computer Engineering & Technology, Vol.3, No.2, pp.79-82, 2014.

[14] C. L. Hung, H. H. Wang, Y. C. Hu, "Efficient load balancing algorithm for cloud computing network", International Conference on Information Science and Technology, pp.28-30, 2012.

[15] K. Kaur, A. Narang, K. Kaur, "*Load balancing techniques of cloud computing*", International Journal of Mathematics and Computer ReseDori, Vol.1, No.3, pp.103-108, 2013.

[16] M. Dorigo, "*Optimization, Learning and Natural Algorithms"*. Ph.D. Thesis, Politecnico di Milano, Italy, 1992.

[17] M. Padmavathi, S. M. Basha, "Dynamic and elasticity ACO load balancing algorithm for cloud computing", International Conference on Intelligent Computing and Control Systems, Madurai, India, pp.77-81, 2017.

[18] S. K. Mishra, B. Sahoo, P. S. Manikyam, "Adaptive scheduling of cloud tasks using ant colony optimization", In Proceedings of the 3rd International Conference on Communication and Information Processing, Tokyo, Japan, pp.202-208, 2017.

[19] P. V. Krishna, "*Honey bee behavior inspired load balancing of tasks in cloud computing environments*", Applied Soft Computing, Vol.13, No.5, pp.2292-2303, 2013.

[20] H. Gupta, K. Sahu, "*Honey bee behavior based load balancing of tasks in cloud computing*", International journal of Science and Research, Vol.3, No.6, 2014.

[21] A. Soni,G. Vishwakarma,Y. K. Jain, "*A bee colony based multi-objective load balancing technique for cloud computing environment*", International Journal of Computer Applications, Vol.114, No.4, 2015.

[22] K. Dasgupta,B. Mandal, P. Dutta,J. K. Mandal,S. Dam, "*A genetic algorithm (ga) based load balancing strategy for cloud computing*", Procedia Technology, No.10, pp.340-347, 2013.

[23] S. A. Hamad, F. A. Omara, "*Genetic-based task scheduling algorithm in cloud computing environment*", International Journal of Advanced computer Science and Applications, Vol.7, No.4, pp.550-556, 2016.

[24] T. Wang, Z. Liu, Y. Chen, Y. Xu, X. Dai, "Load balancing task scheduling based on genetic algorithm in cloud computing", In International Conference on Dependable, Autonomic and Secure Computing, Dalian, China, pp.146-152, 2014.

[25] G. Joshi,S. K. Verma, "*Load balancing approach in cloud computing using improvised genetic algorithm: a soft computing approach*", International Journal of Computer Applications, Vol.122, No.9, 2015.

[26] A. A. Beegom, M. S. Rajasree, " A particle swarm optimization based pareto optimal task scheduling in cloud computing", In International Conference in Swarm Intelligence, Springer, Cham, pp.79-86, 2014.

[27] K. Pan, J. Chen, "Load balancing in cloud computing environment based on an improved particle swarm optimization", In International Conference on Software Engineering and Service Science, Beijing, China, pp.595-598, 2015.

[28] M. A. Rodriguez, R. Buyya, "*Deadline based resource provisioningand scheduling algorithm for scientific workflows on clouds*", IEEE transactions on Cloud Computing, Vol.2, No.2, pp.222-235, 2014.

Authors Profile

Mr. Jaykumar Maheshbhai Tandel pursed Diploma in Computer Engineering from Vallabh Budhi Polytechnic, Navsari, India in 2013 and Bachelor of Engineering degree from GIDC Degree Engineering Collage, Navsari, India in 2016 and currently pursuing Master of Engineering Computer Engineering degree from Department of Computer Engineering & Information Technology, Government Engineering Collage (GEC), Modasa, India in year 2018.

Mr. Hiren R. Patel received the Bachelor of Engineering degree from SCET, Surat, India in 2004 and Master Of Engineering Computer Engineering at GEC Modasa in 2016. He is now with Government Engineering Collage, Modasa as the assistant professor since 2009. Currently he is pursuing PhD from Gujarat Technological University, Ahmedabad. His main research work focuses on Cloud Computing, Web Development. He has 12 years of teaching experience .