




Paper Type: Original Article

Load Balancing for Improved QoS in the Cloud Computing

Ramin Goudarzi Karim^{1,*}  Fatemeh Rasoulpour², Shamila Saeedi²

¹ Department of CIS, Stillman College, Tuscaloosa, Alabama, USA; rkarim@stillman.edu;

² Department of Computer Engineering, Institute of Higher Education, Tenekabon, Iran; Rasoulpour.72@gmail.com; shamila.saeedi@aihe.ac.ir.

Citation:

Received: 1 July 2023

Revised: 10 August 2023

Accepted: 10 October 2023

Goudarzi Karim, R., Rasoulpour, F., & Saeedi, Sh. (2024). Load Balancing for Improved QoS in the Cloud Computing. *Smart City Insights*, 1(1), 1-6.

Abstract

The emergence of cloud computing technology has led to the development of load-balancing algorithms. This paper presents the performance analysis of different load-balancing algorithms based on different metrics such as response time, processing time, scalability, throughput, system stability and power consumption. The primary purpose of this article is to help us propose a new algorithm by studying the behaviour of the various existing algorithms.

Keywords: Cloud computing, Load balancing, Throughput, Scalability, Power consumption.

1 | Introduction

The current cloud computing environment serves many fields but faces limitations such as security, authentication, fault tolerance, load balancing, and availability [1]. There are various advantages of cloud computing, such as virtualization, resource sharing, ubiquity, and utility computing, but there are also critical issues like security, privacy, load management and fault tolerance [2]. Load balancing is one of the main challenges in cloud computing, as it is required to distribute the dynamic workload across multiple nodes to ensure that no single node is overwhelmed [2]. This paper studies eleven load-balancing algorithms, and various parameters are used to check the results.

Cloud and its components

A cloud consists of a number of data centres, which are further divided into nodes and VMs [3]. A data centre controller controls the various activities, a cluster is a set of nodes, and a VM is a software program or operating system capable of performing tasks such as running applications and programs [4]. *Fig. 1* gives an overview of the generalized architecture of the cloud [5].

 Corresponding Author: rkarim@stillman.edu



Licensee System Analytics. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0>).

Problem statement

Load balancing is an essential part of the overall response time of the cloud, and we aim to design a new load balancer to improve quality of service by optimizing load balancing [6]. Load balancing is a computer networking method to distribute workload across multiple computers or a computer cluster, network links, central processing units, disk drives, or other resources to achieve optimal resource utilization, maximize throughput, minimize response time, and avoid overload [7]. It is classified into two types static and dynamic. Static algorithms decide how to distribute the workload according to prior knowledge of the problem and system characteristics, while dynamic algorithms use state information to make decisions during program execution [7]. Load balancing in cloud computing is based on the system's current state, with various metrics such as throughput, overhead, migration time, response time, resource utilization, scalability, performance, and fault tolerance [8]. These metrics help optimize the system's performance by shifting the load dynamically.

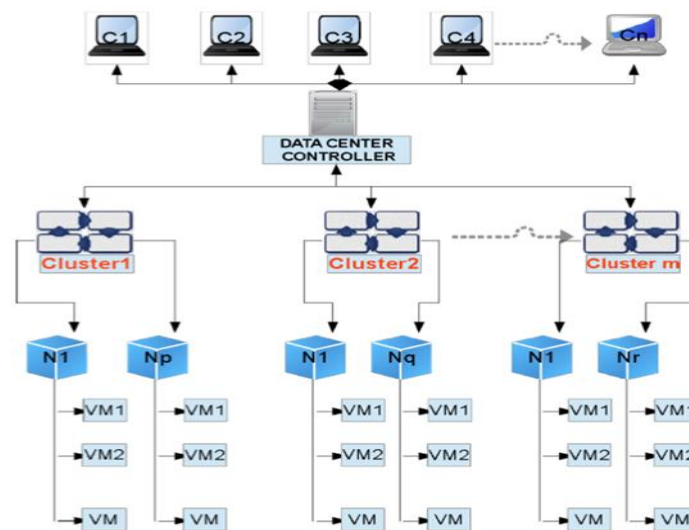


Fig. 1. Generalized architecture of cloud.

2 | Related Work

Load balancing is essential in cloud computing to ensure an efficient and fair allocation of computing resources. This section reviews existing load-balancing techniques and compares them to a new efficient Virtual Machine (VM) load balancing algorithm [9]. The proposed algorithm finds the expected response time of each resource. It sends the ID of a VM having a minimum response time to the data centre controller for allocation to the new request [10]. The experimental result compares the proposed VM load balancing algorithm with the throttled and active VM load balancers [11]. Researchers proposed the Weighted Active Monitoring Load Balancing (WALB) Algorithm, which creates VMs of different processing power and allocates weighted counts according to computing power [12].

Table 1 compares the reviewed algorithms in terms of the challenges discussed. When a request to allocate a VM arrives, the algorithm identifies the least loaded and most powerful VM according to the weight assigned and returns its VM, ID to the data center controller [13]. The experimental result showed that the proposed algorithm achieves better performance factors such as response and processing time but does not consider process duration for each request [14]. Investigators proposed a new VM load balancing algorithm, modified WALB algorithm, which creates VMs of different processing power and allocates weighted count according to the computing power of the VM enhanced load balancing to avoid deadlock proposed a technique to avoid deadlock among VMs while processing a request by migrating the VM and maintaining a data structure containing VM, ID, job ID, and VM status [15]. Round robin load balancer is an algorithm that migrates jobs from an overloaded VM to an underutilized VM based on the least hop time. Weighted round robin is a

modified version of Round Robin, assigning a relative weight to all VMs [16]. It increases the number of jobs serviced by cloud providers, improving working and business performance.

Table 1. Synthesis table of existing load balancing algorithms.

Techniques	Metaphors	Conclusion
Throttled load balancer	This algorithm ensures only a pre-defined number of internet cloudlets are allocated to a single VM at any given time.	Response time improved but other parameters are not taken into account such as: weight of VM, processing time, etc.
Modified throttled algorithm	Focuses mainly on how incoming jobs are assigned to the available VMs. Load nearly distributed uniformly among VMs.	Resource utilization response time has improved.
Efficient VM load balancing algorithm	The proposed algorithm finds the expected response time of each resource.	Increases performance of the cloud environment. Decreases response time and cost.
Active VM load balancer	maintains information about each VM and the number of requests currently allocated to the VMs	Does not consider the hardware capacity of VMs.
WALB Algorithm	Allocates weighted count according to the computing power of the VM, but the algorithm does not consider process duration for each individual request.	Increase response time and processing time
Modified WALB Algorithm	This algorithm identifies VM with least load, least process duration and most powerful VM according to the weight assigned but it considers process duration.	Processing time: hence they tried best to consider the most affecting factor (process duration) in performance increase.
Enhanced load balancing to avoid deadlock	Propose a technique to avoid deadlock among VMs while processing a request by migrating the VM.	Improves: migration time performance response time
Round robin load balancer	Data center controller assigns first request to a VM, picked randomly from the group. It assigns requests to the rest of VMs in circular order.	There is a possibility that some nodes may get heavily loaded while others are overloaded. Decrease resource utilization
Weighted round robin algorithm	This algorithm assigns a relative weight to all the VMs.	Improvement of resource utilization
Dynamic load balancing: improve efficiency in cloud Computing	When the users send the request to the dynamic load balancer, it gathers the processor utilization and memory utilization of each active server.	Fault tolerance high scalability low overhead
Round robin with server affinity: a VM Load balancing algorithm for cloud based infrastructure	The limitation of round robin algorithm is that it does not save the state of the previous allocation of a VM to a request while the same state is saved in the proposed algorithm.	Improved response time data center processing time

The dynamic load balancer monitors the load of each VM in the cloud pool. If the processor and memory utilization are less than 80%, it instantiates a new VM on the following server with the lowest processor and memory utilization. The algorithm also checks the fault occurrence of a server, and if any fault occurs, the VMs will be shifted to another server with less than 80%. The proposed algorithm achieves high scalability, dynamic load balancing, fault tolerance and low overhead. The modified throttled algorithm [16] attempts to improve the response time and efficiency of load balancing in cloud computing. It uses a VM state list to store each VMs allocation status (i.e., Busy/Available) and an index table of VMs and the state of VMs. The

VM at first index is initially selected depending upon the state of the VM, and the VM at index next to the already assigned VM is chosen depending on the state of the VM. When compared to existing round-robin and throttled algorithms, the response time for proposed algorithm has improved considerably.

Enhanced load balancing algorithm using efficient cloud management system is based on the least hope time, while WALB identifies the least loaded and most powerful VM. In round robin LB, the request is assigned circularly, but a new algorithm, "weighted round robin", is proposed to improve load balancing and response time. Dynamic load balancer aims to improve efficiency in cloud computing by monitoring the load of each VM and saving the previous state of allocation of a VM to a request from a user. The modified throttled algorithm aims to improve response time and efficiency.

In round robin load balancer [17], [18] the data center controller randomly chooses a VM from the group to receive the initial request. It then distributes requests to the VMs in a circular fashion. A VM gets moved to the end of the list once a request has been assigned. The round robin algorithm's advantage is that it eliminates the need for inter-process communication. As the duration of each process cannot be predicted before it is performed, it is possible that some nodes could become overloaded.

3 | Conclusions

Cloud computing provides everything to the user as a service, including application, platform, and infrastructure. Load balancing is required to distribute the load evenly among all servers in the cloud to maximize resource utilization, increase throughput, provide good response time, and reduce energy consumption. The threshold algorithm guarantees most of these metrics, except for overload rejection, fault tolerant, and process migration. We plan to improve throttled to make it more suitable for the cloud environment and more efficient regarding process migration.

Author Contributions

Conceptualization, R. G. K and Sh. S; methodology, R. G. K; software, F. R; validation, R. G.K. and F. R; formal analysis, Sh. S; investigation; writing—original draft preparation, R. G. K; writing—review and editing, Sh. S. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Data Availability

All the data are available in this paper.

Conflicts of Interest

The authors declare no conflict of interest.

References

- [1] Tennakoon, D., Chowdhury, M., & Luan, T. H. (2023). Cloud-based load balancing using double Q-learning for improved Quality of Service. *Wireless Netw* 29, 1043–1050 (2023). <https://doi.org/10.1007/s11276-018-1888-8>.
- [2] Zhou, J., Lilhore, U. K., Hai, T., Simaiya, S., Jawawi, D. N. A., Alsekait, D., ... Hamdi, M. (2023). Comparative analysis of metaheuristic load balancing algorithms for efficient load balancing in cloud computing. *Journal of cloud computing*, 12, 85 (2023). <https://doi.org/10.1186/s13677-023-00453-3>.

- [3] Ghafir, S., Alam, M. A., Siddiqui, F., & Naaz, S. (2024). Load balancing in cloud computing via intelligent PSO-based feedback controller. *Sustainable computing: informatics and systems*, 41, 100948.
- [4] Abdelhafeez, A., & Aziz, A. S. (2024). Multi-criteria decision-making model for rank dstrategy to overcome barriers to integrating the AI and cloud systems in the IT industry. *Soft computing fusion with applications*, 1(1), 1–9.
- [5] Ghasemi, A., Isaai, M., Bandarian, R., & Ekhtiarzadeh, A. (2022). designing a qualitative model of the micro foundations of dynamic capabilities in cloud computing service providers in Iran. *Innovation management and operational strategies*, 3(2), 150–159.
- [6] Kumar, A., & Thomaz, A. C. F. (2022). Smart bus ticketing system through IoT enabled technology. *Big data and computing visions*, 2(1), 1–8.
- [7] Muniz, R. D. F., Almaz Ali Yousif, B., & Shemshad, A. (2022). River water quality monitoring through IoT enabled technologies. *Computational algorithms and numerical dimensions*, 1(1), 35–39.
- [8] Gulbaz, R., Siddiqui, A. B., Anjum, N., Alotaibi, A. A., Althobaiti, T., & Ramzan, N. (2021). Balancer genetic algorithm—A novel task scheduling optimization approach in cloud computing. *Applied sciences*, 11(14), 6244.
- [9] Jyoti, A., & Shrimali, M. (2020). Dynamic provisioning of resources based on load balancing and service broker policy in cloud computing. *Cluster computing*, 23(1), 377–395.
- [10] Mohammadian, V., Navimipour, N. J., Hosseinzadeh, M., & Darwesh, A. (2021). Fault-tolerant load balancing in cloud computing: A systematic literature review. *IEEE access*, 10, 12714–12731.
- [11] Belgaum, M. R., Musa, S., Alam, M. M., & Su’ud, M. M. (2020). A systematic review of load balancing techniques in software-defined networking. *IEEE access*, 8, 98612–98636.
- [12]Ullah, A., Nawli, N. M., & Khan, M. H. (2020). BAT algorithm used for load balancing purpose in cloud computing: an overview. *International journal of high performance computing and networking*, 16(1), 43–54.
- [13]Sriram, G. S. (2022). Challenges of cloud compute load balancing algorithms. *International research journal of modernization in engineering technology and science*, 4(1), 1186–1190.
- [14]Siddiqui, S., Darbari, M., & Yagyasen, D. (2020). An QPSL queuing model for load balancing in cloud computing. *International journal of e-collaboration (ijec)*, 16(3), 33–48.
- [15]Kumar, M., & Sharma, S. C. (2020). PSO-based novel resource scheduling technique to improve QoS parameters in cloud computing. *Neural computing and applications*, 32(16), 12103–12126.
- [16]Talaat, F. M., Saraya, M. S., Saleh, A. I., Ali, H. A., & Ali, S. H. (2020). A load balancing and optimization strategy (LBOS) using reinforcement learning in fog computing environment. *Journal of ambient intelligence and humanized computing*, 11(11), 4951–4966.
- [17]Mohapatra, H., & Rath, A. K. (2019). Fault tolerance through energy balanced cluster formation (ebcf) in wsn. *Smart innovations in communication and computational sciences* (pp. 313–321). Singapore: Springer Singapore.
- [18]Panda, H., & Mohapatra, H. (2019). *WSN based water channelization: an approach of smart water* [Thesis].

