Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 44 No. 12
December 2023

A Comprehensive Review of Load Balancing Algorithms, Strategies, and Performance Evaluation

R.Vimal Raja[1] and Dr.G.Santhi[2]

[1] Research Scholar, Department of Information Technology,
Puducherry Technological University

[2] Professor, Department of Information Technology,
Puducherry Technological University

Abstract

Cloud computing has emerged in recent years as a fundamental paradigm for delivering scalable and flexible computing resources. Cloud environments rely on load balancing to allocate resources and distribute workloads efficiently among servers. This survey paper provides researchers with an overview of load-balancing methods, algorithms, and approaches. Literature and research studies are reviewed in order to identify load-balancing strategies' strengths, limitations, and applicability. The paper also examines load-balancing mechanisms in major cloud platforms, including Amazon Elastic Compute Cloud (EC2), Google APP Engine, and Microsoft Azure. An overview of existing studies is presented, along with the performance metrics and evaluation methodologies commonly used in cloud computing research. There is also a discussion of open challenges in load balancing and possible directions for future research. In addition to providing a valuable resource for researchers and practitioners seeking a deeper understanding of load balancing in cloud environments, this survey paper highlights the need for continual innovation in this field

Keywords: *cloud computing, load balancing, Performance metrics, cloud platforms*

Introduction

In recent years, cloud computing has emerged as a fundamental paradigm for delivering scalable and flexible computing resources to users and organizations [1]. The use of virtualization technologies allows cloud environments to function efficiently and dynamically allocate resources based on varying workloads by providing on-demand access to shared computing resources. It has become increasingly difficult to determine optimal resource allocation and efficient utilization for cloud services due to exponential growth in demand. A cloud environment relies heavily on load balancing to evenly distribute the workload among multiple servers or virtual machines. As a result of load balancing, systems perform better, respond faster, utilize resources more effectively, and ensure high availability by preventing resource bottlenecks and overloading individual nodes. Essentially, it allows efficient and effective distribution of computing resources to meet user needs in cloud computing infrastructures. As part of this survey paper, we present a comprehensive overview of load balancing in cloud computing. There have been numerous load balancing methods, algorithms, and approaches proposed and studied within the research community. In this article, we will examine a few of them. This survey paper identifies strengths, limitations, and applicability of different load balancing strategies in cloud environments by analyzing existing literature and research studies.

A brief introduction to cloud computing and its associated concepts, including virtualization, scalability, and elasticity, is provided in Section 2 [2]. This section 3 describes and categorizes different load balancing techniques commonly employed in cloud computing, including static, dynamic load balancing [3]. Load balancing mechanisms in major cloud platforms are examined in Section 4, which discusses the features and algorithms utilized by Amazon Elastic Compute Cloud (EC2), Google APP Engine, and Microsoft Azure. [4].

In Section 5, performance metrics and evaluation methodologies commonly used in cloud computing research are examined in order to evaluate and compare load balancing algorithms. Performance results from existing studies are summarized and analyzed in this section. Section 6 discusses open challenges and directions for load balancing, emphasizing areas for further research and innovation.

In conclusion, this paper intends to serve as an informative resource for researchers and practitioners interested in load balancing in cloud computing. Our goal is to consolidate and analyze existing information in order to better understand load balancing techniques, their effectiveness, and the upcoming challenges. It is imperative to optimize load balancing algorithms in cloud computing environments in order to achieve efficient resource usage, scalability, and performance, ultimately enabling reliable and responsive cloud services.

Overview of Cloud computing

A brief overview of cloud computing is provided in this section, which includes concepts such as virtualization, scalability, and elasticity that are associated with cloud computing. Figure 1 shows the general architecture of cloud computing load balancer.

A. Cloud Computing

A cloud computing model provides online, on-demand access to a shared pool of computing resources for users. The service provides users with the ability to leverage a wide range of virtualized resources and services, such as servers, storage, and applications, without having to worry about a local infrastructure. Cloud computing has many benefits, such as virtualization, scalability, flexibility, and cost efficiency. As a result, users are able to access resources on demand and pay only for what they consume, making it easier for them to conserve money [5]. Further, cloud computing reduces the need for organizations to invest in costly hardware and software by allowing them to utilize the resources they already have. By doing so, organizations can keep their costs down and focus on their core activities, thereby reducing their overall business costs. As part of the cloud computing service, users have access to data and applications on any device, making it easier to collaborate and work remotely from anywhere.

B. Virtualization

Cloud computing relies heavily on virtualization to abstract physical resources such as servers and storage into virtual counterparts.

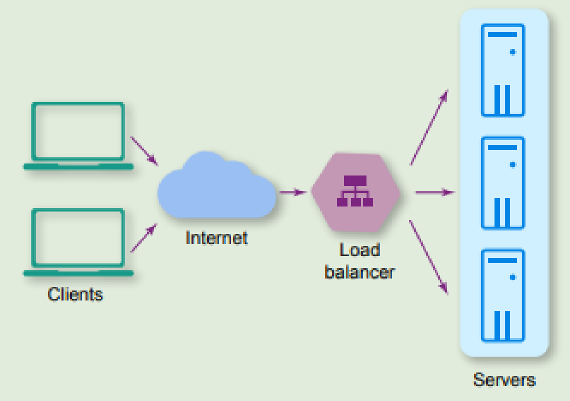

Fig 1. The general architecture of Cloud computing Load balancer

Running several virtual machines (VMs) or containers on a single physical machine has benefits such as effective resource management, isolation, and adaptable workload management. By doing so, organizations can scale and move workloads across different cloud providers with ease, as well as rapidly deploy applications and services at a time that is convenient for them. In addition to reducing the operational costs of businesses, virtualization allows them to optimize their IT infrastructures so as to reduce their operational costs. Moreover, virtualization also enables businesses to become more agile in their response to changing market conditions, which enables them to make quick decisions in response to these changes.

Furthermore, multiple virtualized machines can be securely isolated from each other, ensuring an increased level of security and privacy [6].

C. Scalability

In cloud computing, scalability refers to the ability to handle changes in workloads by adjusting the system's capacity according to the changes in workload. To meet the demand for resources, it is possible to add or remove them dynamically as the demand changes. There are two main ways in which scalability can be achieved: horizontal scaling - increasing the number of instances or vertical scaling - increasing the amount of resources available. By doing so, organizations will be able to optimize their resources and reduce the costs associated with them. The system can also handle spikes in traffic without any downtime or latency so that spikes in traffic are not a problem. By providing a better customer experience as well as saving time and money, businesses are able to provide better service to their customers. The ability to scale can also allow businesses to expand and contract quickly as they need to, without compromising quality or performance in the process [7].

D. Elasticity

A feature of cloud computing known as elasticity is the ability for resources to scale up and down automatically as workloads fluctuate based on the availability of resources. With the help of the platform, applications and services are capable of meeting changing demands while maximizing the use of resources. As workload changes, elasticity facilitates cost optimization and maintains performance levels to keep up with changes in workload. Furthermore, elasticity allows for greater agility in managing resources, since it allows organizations to quickly add or remove resources if the demand changes, which allows them to adjust their costs accordingly [8]. This makes cloud computing an attractive alternative to traditional on-premises solutions because of its flexibility. This enables organizations to remain competitive and responsive to the demands of their customers while also being able to reduce their costs at the same time. It has been a recognized fact that cloud computing has become the preferred solution for many businesses as a result

Various load balancing techniquesin cloud computing

Numerous load balancing strategies improve cloud computing's performance. Based on the fundamental environment, algorithms in this area can be divided into static and dynamic categories

E. STATIC LOAD BALANCING

A static environment requires prior knowledge of the system state and its capabilities and properties prior to the load balancing algorithm's processes. In addition to memory and storage capacity, processing power is another example of prior information [9This makes it possible to gauge the system's load. Algorithms based on static data do not take into account changes in the load that occur during runtime. This can lead to the inefficient use of computing resources and even system crashes. To prevent this, dynamic load balancing algorithms are needed to monitor the system in real time and adapt to any changes in load.

These algorithms suffer from low fault tolerance as a result of sudden changes in load, which is a major disadvantage of this algorithm [10]. Due to numerous restrictions, including uneven load distribution among nodes, which causes some machines to become overloaded while others remain unloaded, the use of static algorithms like Round-Robin in cloud environments is no longer acceptable or efficient.

F. DYNAMIC LOAD BALANCING

As far as load balancing is concerned, these algorithms are more effective and adaptable. In dynamic environments, load balancing algorithms account for past system states, unlike static algorithms [10].

By adjusting and optimizing the system in real time, resources are more efficiently used. As well as being more cost-effective, these algorithms can reduce latency and improve performance. Furthermore, dynamic load balancing algorithms make systems more resilient to change by improving their scalability and reliability. The high availability and performance they provide make them ideal for distributed systems.Despite their complexity and overhead, these algorithms have the major benefit of flexibility.

As a result, new algorithms in this category should avoid such disadvantages. For distributed cloud systems, a dynamic algorithm is more effective than a static algorithm since it takes the system's current status into consideration. [11]. Additionally, dynamic algorithms eliminate overhead for storing previous system states and have higher runtime complexity than static.

G. LITERATURE REVIEW

In order to address the most relevant gaps in load balancing for cloud computing, a comprehensive literature review is taken from recent research papers. The purpose of this survey is to identify the strengths, limitations, and applicability of different load balancing strategies in cloud environments based on an analysis of existing literature and research studies. Table 1& 2 lists the strengths, limitations, and performance metrics used in existing works.

A hierarchical edge-cloud SDN controller method was suggested in [18] to improve scalability, load balancing, and computation speed. It includes a queuing model and a robust load-balancing algorithm to satisfy QoS requirements. The system allows for the implementation of a large-scale SDN network without compromising performance.

The paper [19] presents a novel architecture for the industrial Internet of Things (IIoT) that implements hierarchical control structures within the mobile edge cloud (MEC). Software-defined networking (SDN) separates the control plane and data plane and uses remote radio heads (RRHs) partitioned into clusters with servers for executing virtual machines (VMs). To boost edge intelligence, the MEC uses deep learning techniques. There are two control schemes, centralized and distributed, that offer a trade-off between performance and overhead. In this paper, a heuristic algorithm based on sub modular function maximization is proposed to solve a joint optimization problem. As metrics for evaluating performance, system delay, resource utilization, energy efficiency, and scalability are considered.

Load Balanced and Energy Aware Cloud Resource Scheduling (LBEACRS) [20] is presented as an approach for scheduling data-intensive applications in Software Defined Vehicular Clouds (SDVCs). A key feature of the algorithm is that it aims to improve energy efficiency and makespan performance compared to traditional cloud resource scheduling algorithms. As part of the paper, the load-balancing and energy-aware resource scheduling methodologies for SDN enabled VANET-Cloud are outlined. Using Montage and CyberShake workloads, SIMITS and CloudSimSDN simulation experiments evaluate LBEACRS against standard scheduling algorithms. The paper [21] presents a novel approach to service orchestration and data aggregation using software-defined networks (SDNs). Three layers are present in SODA: the data center, the middle routing, and the vehicle network layer. Redundancy of data and latency of service responses are the two goals of the framework. Evaluations of SODA show that it offers lower service response delay and data redundancy than traditional schemes. As a result of this study, service response delays and data redundancy are minimized while energy efficiency is maximized. In SODA, data orchestration as services and packet aggregation reduce redundancy and delay in service response.

The paper [22] presents a framework for improving cloud resource allocation for applications that require streaming data processing using virtual machines and software-defined networks. To optimize the processing of big data streams, a novel algorithm is proposed for cloud resource allocation. A simulation tool called CloudSIM is used to evaluate the algorithm and compare it with baseline algorithms. Based on the outcomes of the suggested method, the virtual machine could deal with 2000 requests in the most efficient 136 seconds. The framework improves cloud resource allocations for apps that need to process streaming data.

FOCALB, a fog computing architecture of load balancing, is introduced in the paper [23]. End-users benefit from reduced latency and better resource utilization through fog computing. Through load balancing, fog resources can be utilized more effectively by distributing workload evenly. The research suggests a hybridized load balancing technique that uses tabu search, Grey Wolf Optimization (GWO), and Ant Colony Optimization (ACO). FOCALB decreases energy usage, execution time, and implementation costs based on simulations. The Enhanced Dynamic Resource Allocation Method (EDRAM), which is a method of load balancing in fog nodes in the context of fog computing, is introduced in this work [24]. It seeks to enhance Quality of Experience (QoE) while lowering task waiting times, latency, and network bandwidth usage. Simulation experiments have shown that EDRAM performs better than existing methods. However, the paper acknowledges limitations, such as the absence of a detailed analysis of computational complexity and a lack of consideration for security issues. Despite the promise of the proposed method, further research is needed to address these limitations and assess its viability in the real world. In this article[25], the authors propose an automatic solution to load balancing in computer networks by utilizing a multi-agent actor-critic reinforcement learning algorithm. The algorithm blends a distributed execution framework with a centralized learning framework. A centralized "critic" uses the collective actions of all agents and the global network state to train, while distributed switches act on local observations. The algorithm outperforms baseline algorithms in terms of Jain's fairness index, average flow completion time, and network utilization metrics. The suggested algorithm outperforms state-of-the-art techniques in terms of flow completion time and fairness index.

In this paper [26], A3C3, a multi-agent actor-critic algorithm, is presented for cooperative multi-agent systems. For estimating value functions and policy functions, A3C3 uses centralized and decentralized critics. Information is shared within teams using decentralized communication networks. Scalability, dynamic agent counts, and noisy communication are all supported by A3C3. Compared to independent controllers, state-of-the-art algorithms, and centralized controllers, the algorithm outperforms partially observable multi-agent scenarios. A3C3's effectiveness in cooperative multi-agent systems is demonstrated in the paper, even though specific performance metrics are not stated.

The paper [27] presents IaaSP-SDN, a resource allocation framework for edge cloud data centers (ECDCs) using software-defined networking (SDN) to ensure quality of service (QoS) and efficient embedding of user applications. A coordinated provisioning approach for IaaS requests and SDN management modules comprise the framework. By contrasting IaaSP-SDN with generalized multi-protocol label switching (GMPLS) and looking at how it affects the placement of SDN controllers, the performance of IaaSP-SDN is assessed. The findings indicate that SDN is more reliable and scalable than GMPLS. The study advocates expanding the proposal to many controllers and using a proactive strategy for optimal position determination in order to account for traffic spikes and network disruptions.

Cloud computing environments require load balancing techniques to maximize performance and resource usage. There are two types of load balancing: static and dynamic. Dynamic load balancing techniques, such as MEC hierarchical control structures and hierarchical edge-cloud SDN controller systems, adapt dynamically to shifting network conditions and workload demands. Using algorithms and deep learning techniques, they increase scalability, decrease system latency, optimize resource usage, and increase energy efficiency. Real-time load distribution is achieved with the help of IoT edge intelligence. Load balancing methods based on predefined configurations, such as LBEACRS for Cloud Resource Scheduling and cloud-based frameworks for VM and SDN coupling, focus on dynamic load balancing. Their goal is to improve energy efficiency, durability, and the efficiency of use of resources.

Table I: Static Load Balancing Techniques

Ref	Methods/Algorithms	Metrics	Strengths	Limitations
[20]	LBEACRS for Cloud Resource Scheduling	Vehicular Edge Computing (VEC)	Makespan, Energy efficiency	Improved makespan, energy efficiency
[22]	Cloud-based framework for VM and SDN coupling	Resource allocation algorithm	-	Enhanced resource allocation, Improved VM performance

Load Balancing Mechanisms on Major Cloud Platforms

Load balancing mechanisms on major cloud platforms are examined in this section, which discusses the features and algorithms utilized by Amazon Web Services, Google Cloud Platform, and Microsoft Azure. In choosing a cloud provider, it is important to consider the algorithms used to allocate resources, which are unique to each platform that help address scalability and availability.

Performance and reliability are ensured by advanced load balancing features across all three cloud platforms.

The availability and/or scalability of these features are essential for applications requiring high levels of availability. Choosing the right cloud service provider also requires consideration of the provider's cost.

The features and algorithms of major cloud platforms are listed in Table I.

Table Ii Dynamic Load Balancing Techniques

Table Iii Features And Load Balancing Algorithms Utilized By Major Cloud Platforms

Ref	**Major cloud Platforms**		
	Cloud Platform	*Features*	*Load Balancing (LB)Algorithms Cloud Platform*
[12]	Amazon EC2	Elastic Load Balancing	Application Load Balancer (ALB) Network Load Balancer (NLB) Classic Load Balancer (CLB)
[13]	Google App Engine	Traffic Splitting Automatic Scaling	Automatic Load Balancing
[14]	Microsoft Azure	Azure Load Balancer	Round Robin Source IP Affinity Leat Connections

Amazon Elastic Compute Cloud (EC2)

Elastic Compute Cloud (EC2) is a cloud offering from Amazon that provides a web service that can be scaled on demand and can be used to host different types of software. By creating, starting, and terminating server instances, it helps software designers create web-scale computing. It is also known as Elastic Compute because they can pay an hourly rate to active servers. The scalable non-relational data store that makes database management easier. In order to facilitate high availability and data durability, SimpleDB creates geographically distributed data automatically. Data storage or computing power consumed for queries, reading, or writing is charged solely for this service. A public and a private IP address are assigned to facilitate the access of different user instances. For example, the public IP address of the replacement instance will be different. Dynamic cloud computing is also possible with Amazon EC2's elastic IP addresses (static IP addresses).). Additional features offered by Amazon EC2 include load balancing,

Ref	Methods / Algorithms	Metrics	Strengths	Limitations
[18]	Hierarchical edge-cloud SDN controller system	Fairness allocation, load balancing	Scalability, Computation delay, QoS	Enhances scalability, Efficient load balancing
[19]	MEC hierarchical control structure	Deep learning techniques	System delay, Scalability, Resource utilization, Energy Efficiency	IoT edge intelligence enhanced
[21]	SODA (Service orchestration and data aggregation)	Correlation-based routing	Response time, Data redundancy, Network traffic	Reduced redundancy,
[24]	Enhanced Dynamic Resource Allocation Method (EDRAM)	Particle swarm optimization	Task waiting time, Network bandwidth, QoE	Reduced waiting time, Improved QoE
[25]	Reinforcement learning algorithm with multi-agent actors and critics	Centralized learning framework	AFCT, JFI, Network utilization	Automated load balancing, Distributed execution
[26]	A3C3 multi-agent actor-critic algorithm	Decentralized communication networks	Analyzed in partially observable scenarios	Outperforms other multi-agent algorithms
	SDN technology for QoS and		Acceptance ratio, Response time,	Stable, and

Moreover, any operating system can be used to run compute instances through its API [12].

H. Microsoft Windows Azure

Windows Azure is another cloud platform offered by Microsoft. On-demand enterprise-level computing capacity, including computing power and storage, can be accessed in this development, hosting, and management environment [13]. It is necessary to use Azure APIs in order to take advantage of Azure Cloud features. Using Windows Azure, developers can build web-based applications that are hosted in Microsoft data centers and communicate with local devices. In order to develop these programs, the .NET framework and Visual Studio are employed. Additionally, it supports HTTP, REST, SOAP, and plain XML protocols. There are a number of components that support it, including:

- SQL Azure supports structured, semi-structured, and unstructured data storage in cloud applications using Microsoft SQL Server.
- Application developers can purchase and sell code, components, training, service templates, and other features needed to develop Windows Azure applications on the Windows Azure Marketplace.
- By simply maintaining domain security across domains, Windows Azure services facilitate cross-organizational collaboration. With its powerful, secure, standards-based infrastructure, it provides authentication and access control functions.
- HPC applications run and are managed within Windows Azure through the Windows Azure HPC Scheduler.

I. Google App Engine

With Google App Engine, developers can design, develop, and deploy Java and Python-based applications. It is available in Java, Go, and Python environments. The same level of reliability, availability, and scalability is guaranteed for your applications. The user interface is programmed in software. The platform also offers a comprehensive programming environment regardless of the size of the user (small or large). For cloud-based applications, there are many useful features, including application templates and endpoints [14].

II. PERFORMANCE METRICS OF COMMONLY USED CLOUD COMPUTING

In cloud computing, performance metrics are used to evaluate the efficiency and effectiveness of load-balancing mechanisms. The metrics provide insight into how load balancers perform, scale, and utilize resources. The following are some commonly used cloud computing performance metrics:

1. **Request Distribution:** Using this metric, you can determine how evenly resources are distributed among incoming requests. By measuring how many requests or connections each resource handles, it measures the balance achieved by the load balancer. When distributing requests evenly, a load balancer should avoid overloading specific resources.
2. **Response Time:** The response time refers to the time it takes the load balancer to receive a request, distribute it to a resource, and receive a response. This includes the time spent executing load balancing algorithms, routing requests, and performing any additional processing. Having a lower response time indicates better load balancing.
3. **Throughput:** Load balancer performance is measured by the amount of requests or connections it can handle in a given period of time. In other words, it indicates how efficiently the load balancer distributes traffic. As throughput values increase, load balancing performance is improved, and high requests can be handled more efficiently [15].
4. **Scalability:** In order to accommodate increased traffic and resource demands, load balancers should be scalable. By dynamically adding or removing resources and adjusting their configurations, scalability metrics assess the efficiency of load balancers in adapting to changing workloads. The ability to scale across multiple nodes, handle more connections, and maintain performance over time are all key scalability metrics.
5. **Latency:** The latency of a load balancer is the delay that requests experience while passing through it. This includes time spent in load balancing operations, network transmissions, and backend processing. The user experience is improved with low latency, since it contributes to better response times.
6. **Resource Utilization:** Optimizing performance and minimizing waste is the responsibility of load balancers. In addition to CPU, memory, and network bandwidth, resource utilization metrics evaluate the load balancer's efficiency in utilizing these resources. When utilization values are higher, resources are effectively allocated and available capacity is being utilized optimally [16].
7. **Fault Tolerance:** Load balancer fault tolerance metrics assess how well they can handle failures and maintain service availability. When a load balancer fails, these metrics assess its ability to automatically redirect traffic to healthy resources.
8. **Overhead:** To achieve high availability or horizontal scalability, cloud computing may reroute requests from one node to another. Data may need to be moved from one node to another even over the network due to this reallocation of requests. Network bandwidth is used to move data or requests. As one of the important measures of a load balancing algorithm, this kind of overhead must be considered [17].

Discussion of open challenges and future directions

The purpose of this review is to analyze existing research in the area and identify potential solutions. Moreover, it should provide directions for future research and recommend best practices for cloud computing load balancing. Additionally, it should provide guidelines for practitioners to ensure that they are using technology to its full potential. A review of load balancing in cloud computing should also consider the economic and environmental impacts and the potential reduction of these impacts through new technologies. Additionally, it should identify potential mitigation strategies for the risks associated with cloud computing load balancing.

1. **Dynamic Workload Management:** Cloud computing environments present a significant challenge to load balancing techniques for managing dynamic workloads. It is an ongoing research goal to develop algorithms that can adapt dynamically to workload fluctuations, scale resources accordingly, and optimize performance in real-time.

2. **Heterogeneous Resource Management:** In order to balance cloud resources effectively, algorithms must be able to handle heterogeneous characteristics, such as processing power, memory capacity, and network bandwidth variations. For maximizing resource utilization and ensuring fairness, load balancing in heterogeneous environments is vital.
3. **Energy Efficiency:** The development of load balancing strategies that take energy consumption into account as an optimization criterion is an emerging research field. By optimizing load balancing algorithms, cloud infrastructures can consume less energy and be more sustainable.
4. **Quality of Service (QoS) Provisioning:** It is critical to ensure quality of service, such as response time, throughput, and reliability, while balancing load. In order to meet user expectations and meet performance expectations, load balancing algorithms need to consider QoS requirements.
5. **Security and Privacy:** Load balancing techniques that address security and privacy concerns in cloud computing environments are essential. Protecting sensitive data, preserving data confidentiality and integrity, and preventing unauthorized access during load balancing operations are ongoing research challenges.
6. **Cost Optimization:** Algorithms that optimize cost by allocating resources efficiently and minimizing wasted resources can greatly benefit users and cloud service providers. When it comes to load balancing for cloud computing, it is imperative that performance requirements are balanced with cost considerations.
7. **Multi-Cloud and Hybrid Cloud Environments:** It poses unique challenges to develop load balancing techniques that can handle load distribution across multiple clouds. The development of algorithms that take into account the diverse characteristics of different cloud providers is an active area of research.
8. **Edge and Fog Computing Integration:** In addition to traditional cloud infrastructures, cloud load balancing approaches incorporating edge and fog computing resources are gaining traction. Current research efforts are focused on developing load balancing techniques that take edge and fog computing devices into account.
9. **Autonomic Load Balancing:** Research on self-adaptive load balancing mechanisms that can adjust load distribution autonomously based on changing system conditions and performance metrics continues. The use of autonomous load balancing algorithms can help optimize resource utilization and increase system efficiency.
10. **Scalability and Elasticity:** It is challenging to develop load balancing algorithms that can efficiently scale and adapt to dynamic resource demands and workload fluctuations in highly scalable and elastic cloud environments. There is a critical research gap in ensuring load balancing mechanisms can handle large scale deployments and sudden spikes in workload demand.

Conclusion

Load balancing is an essential component of cloud computing, as it allows efficient resource allocation and workload distribution. In this survey paper, various methods, algorithms, and approaches have been explored to provide a comprehensive overview of load balancing strategies in cloud environments. A literature review and research study were conducted in order to examine the strengths, limitations, and applicability of different load balancing techniques. Moreover, we examined the load balancing algorithms and features utilized by major cloud platforms. Furthermore, the paper summarized existing studies' results and discussed performance metrics and evaluation methodologies commonly used in cloud computing research. The article also discussed open challenges and future research directions in load balancing. As a result of the survey paper, researchers and practitioners can benefit from a valuable resource emphasizing the need for ongoing innovation and advancement in load balancing techniques in order to meet the changing needs of cloud computing environments.

References

[1] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing. National Institute of Standards and Technology.

[2] Dinh, H. T., Lee, C., Niyato, D., & Wang, P. (2013). A survey of mobile cloud computing: architecture, applications, and approaches.

Wireless Communications and Mobile Computing, 13(18), 1587-1611.

[3] Liu, J., & Buyya, R. (2020). A comprehensive study of load balancing algorithms in cloud computing environments. IEEE Transactions on Parallel and Distributed Systems, 31(3), 675-689.

[4] Sharma, V., Singla, M., & Kumar, N. (2019). Comparative analysis of load balancing techniques in cloud computing. 2019 International Conference on Information Networking (ICOIN), 25-30.

[5] Pradhan, A., Bisoy, S.K., Mallick, P.K. (2020). Load Balancing in Cloud Computing: Survey. In: Sharma, R., Mishra, M., Nayak, J., Naik, B., Pelusi, D. (eds) Innovation in Electrical Power Engineering, Communication, and Computing Technology. Lecture Notes in Electrical Engineering, vol 630. Springer, Singapore. https://doi.org/10.1007/978-981-15-2305-2_8

[6] M. Singh, "Virtualization in Cloud Computing- a Study," 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, 2018, pp. 64-67, doi: 10.1109/ICACCCN.2018.8748398.

[7] Al-Said Ahmad, A., Andras, P. Scalability analysis comparisons of cloud-based software services. *J Cloud Comp* 8, 10 (2019). https://doi.org/10.1186/s13677-019-0134-y

[8] Ahmed Barnawi, Sherif Sakr, Wenjing Xiao, Abdullah Al-Barakati, The views, measurements and challenges of elasticity in the cloud: A review, Computer Communications, Volume 154, 2020,Pages 111-117, ISSN 0140-3664.

[9] Dalia, Abdulkareem, Shafiq., Noor, Zaman., Azween, Abdullah. (2021). Load balancing techniques in cloud computing environment: A review. Journal of King Saud University - Computer and Information Sciences, doi: 10.1016/J.JKSUCI.2021.02.007

[10] Alam, M., Ahmad Khan, Z., 2017. Issues and challenges of load balancing algorithm in cloud computing environment. Indian J. Sci. Technol. 10 (25), 1–12. https:// doi.org/10.17485/ijst/2017/v10i25/105688

[11] Adaniya, A., Paliwal, K., 2019. A Proposed Load Balancing Algorithm for Maximizing Response Time for Cloud Computing, Int. J. Res. Appl. Sci. Eng. Technol., 7(Iv).

[12] A. Choudhary, P. K. Verma and P. Rai, "The Proposed Pre-Configured Deployment Model for Amazon EC2 Cloud Services," 2022 6th International Conference on Electronics, Communication and Aerospace Technology, Coimbatore, India, 2022, pp. 794-799, doi: 10.1109/ICECA55336.2022.10009551.

[13] A. Verma, D. Malla, A. K. Choudhary and V. Arora, "A Detailed Study of Azure Platform & Its Cognitive Services," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 129-134, doi: 10.1109/COMITCon.2019.8862178.

[14] K, Mahesh and Laxmaiah, Dr. M. and Sharma, Dr. Yogesh Kumar, A Comparative Study on Google App Engine Amazon Web Services and Microsoft Windows Azure (2019). International Journal of Computer Engineering and Technology, 10(1), 2019, pp. 54-60, Available at SSRN: https://ssrn.com/abstract=3537564

[15] S. Abed, D. S. Shubair, "Enhancement of task scheduling technique of big data cloud computing", presented at Int. Conf. on Advances in Big Data, Comput. and Data Commun. Syst., Durban, South Africa, 2018, pp. 1-6.

[16] O. Kaneria, R. Banyal, "Analysis and improvement of load balancing in cloud computing", presented at Int. Conf. on ICT in Business Industry & Government, 2017.

[17] E. K, M. Naghibzadeh, “A min-min max-min selective algorithm for grid task scheduling”, presented at Int. Conf. in Central Asia on Internet,2007, pp. 1–7

[18] F. P. -C. Lin and Z. Tsai, "Hierarchical Edge-Cloud SDN Controller System With Optimal Adaptive Resource Allocation for Load-Balancing," in IEEE Systems Journal, vol. 14, no. 1, pp. 265-276, March 2020, doi: 10.1109/JSYST.2019.2894689.

[19] W. Xia, J. Zhang, T. Q. S. Quek, S. Jin and H. Zhu, "Mobile Edge Cloud-Based Industrial Internet of Things: Improving Edge Intelligence With

Hierarchical SDN Controllers," in IEEE Vehicular Technology Magazine, vol. 15, no. 1, pp. 36-45, March 2020, doi: 10.1109/MVT.2019.2952674.

[20] Shalini. S and Annapurna P Patil, "Load Balanced and Energy Aware Cloud Resource Scheduling Design for Executing Data-intensive Application in SDVC" International Journal of Advanced Computer Science and Applications(IJACSA), 12(10), 2021. http://dx.doi.org/10.14569/IJACSA.2021.0121040

[21] Y. Liu, Z. Zeng, X. Liu, X. Zhu and M. Z. A. Bhuiyan, "A Novel Load Balancing and Low Response Delay Framework for Edge-Cloud Network Based on SDN," in IEEE Internet of Things Journal, vol. 7, no. 7, pp. 5922-5933, July 2020, doi: 10.1109/JIOT.2019.2951857.

[22] A. Al-mansoori, J. Abawajy and M. Chowdhury, "BDSP in the cloud: Scheduling and Load Balancing utlizing SDN and CEP," 2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID), Melbourne, VIC, Australia, 2020, pp. 827-835, doi: 10.1109/CCGrid49817.2020.000-2.

[23] Kaur, M., Aron, R. FOCALB: Fog Computing Architecture of Load Balancing for Scientific Workflow Applications. *J Grid Computing* 19, 40 (2021). https://doi.org/10.1007/s10723-021-09584-w

[24] Baburao, D., Pavankumar, T. & Prabhu, C.S.R. Load balancing in the fog nodes using particle swarm optimization-based enhanced dynamic resource allocation method. *Appl Nanosci* 13, 1045–1054 (2023). https://doi.org/10.1007/s13204-021-01970-w

[25] T. Mai, H. Yao, Z. Xiong, S. Guo and D. T. Niyato, "Multi-Agent Actor-Critic Reinforcement Learning based In-network Load Balance," GLOBECOM 2020 - 2020 IEEE Global Communications Conference, Taipei, Taiwan, 2020, pp. 1-6, doi: 10.1109/GLOBECOM42002.2020.9322277.

[26] David Simões, Nuno Lau, Luís Paulo Reis, Multi-agent actor centralized-critic with communication, Neurocomputing, Volume 390, 2020,Pages 40-56,ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2020.01.079.

[27] F. A. Zaman, A. Jarray and A. Karmouch, "Software Defined Network-Based Edge Cloud Resource Allocation Framework," in IEEE Access, vol. 7, pp. 10672-10690, 2019, doi: 10.1109/ACCESS.2018.2889943.