



Paper Type: Original Article

Energy Efficiency and Load Balancing Algorithm for Cloud Environment

Abhinav Kumar Toofani^{1,*}, Ashish Kumar¹, Abhijeet Kumar Giri¹, Abhisht Verma¹, Nitish Kumar¹

¹ School of Computer Engineering, KIIT Deemed to Be University, Bhubaneswar -751024, Odisha, India ; 21051631@kiit.ac.in; 21051604@kiit.ac.in; 2105906@kiit.ac.in; 21051017@kiit.ac.in; 21051500@kiit.ac.in.

Citation:

Received: 4 August 2023

Revised: 20 September 2023

Accepted: 14 December 2023

Kumar Toofani, A., Kumar, A., Kumar Giri, A., Verma, A., & Kumar, N. (2024). Energy efficiency and load balancing algorithm for cloud environment. *Smart City Insights*, 1 (1), 7-12.

Abstract

The paper comprehensively overviews the research focus and critical results. Cloud computing has emerged as a popular technology that supports computing services, allowing users to follow a pay-as-you-go model. It is a framework that enables convenient and on-demand online access to shared computing resources. The research explores the intersection of load-balancing algorithms and energy efficiency, especially their impact on energy consumption and environmental sustainability. Load balancing is an essential part of public cloud computing and helps to utilize resources and thus improve system performance optimally. The purpose of load balancing is to minimize resource consumption, further reducing the energy consumption and carbon emissions that cloud technology urgently needs. The work evaluates various load balancing strategies and their effectiveness in optimizing the energy use of computing systems using system analysis. The research not only delves into the technical aspects of these algorithms but also sheds light on their broader environmental impact. By examining the trade-offs between computer performance and energy consumption, the study aims to contribute valuable information to the ongoing debate about designing more energy-efficient and environmentally conscious computer systems. The results presented in this paper lay the foundation for future developments in load-balancing algorithms that prioritize energy efficiency and promote a greener and more sustainable technology landscape.

Keywords: Energy efficiency, Load balancing algorithms, Environmental impact, Computational performance, Sustainability, Computing systems.

1 | Introduction

Cloud computing has revolutionized the way computing services are accessed, providing accessible, convenient, and on-demand services across various locations. By leveraging a shared pool of configurable computing resources, cloud services offer a wide range of benefits. They enable users to utilize and manage resources with minimal effort and interaction, making computing tasks more efficient and streamlined. This paper focuses on exploring the intersection of load-balancing algorithms and energy efficiency in cloud

✉ Corresponding Author: 21051631@kiit.ac.in



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>).

computing. The aim is to minimize resource consumption, reduce energy usage, and promote environmental sustainability. Through a thorough evaluation of various load balancing strategies and their impact on energy optimization, this research contributes valuable insights to the ongoing discourse on designing energy-efficient and environmentally conscious computer systems. The findings presented in this paper set the groundwork for future advancements in load-balancing algorithms that prioritize energy efficiency, fostering a greener and more sustainable technology landscape [1]. The world has become a global village where the internet is used through remote connections to services and hardware in remote locations. The services available through the Internet are the revolution of this era in the field of computing. More and more computing power is needed to perform complex tasks. These tasks require advanced and efficient computing. Instead of buying new hardware, paying for high-performance computer hardware services is a better option. Cloud service providers provide users with such features to access resources and services using a pay-for-service model [2].

In recent years, the number of cloud service centers has increased due to the suitability of storage and computing services for various applications. Cloud computing services fall into three main categories, i.e., Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) [3]. Users have no geographical restrictions, i.e., they can use the services anywhere and anytime [4]. A cloud service provides virtualized resources for different tasks to handle various requests [5]. A cloud data center infrastructure usually consists of thousands of large machines with high-speed computing resources. Every year, a huge amount of data is generated; therefore, high processing power and memory capacity are required. Many disciplines, such as astronomy, bioinformatics, meteorology, and environmental and geological sciences, deal with large-scale data [6]. Processing the enormous amounts of data produced by these fields significantly degrades the performance of clouds [7]. Ensuring effective task scheduling in cloud services to improve cloud performance is challenging. In a cloud service environment, scheduling can be done at different service levels, i.e., IaaS, PaaS, and SaaS [8]. Load balancing distributes loads among available resources to balance the total load. A load-balancing algorithm receives and distributes user requests among available resources, i.e., Virtual Machines (VM). The role of the load balancer is to determine the load on the available resources and distribute the load among the resources. If resources are not used with a proper load-balancing algorithm, Quality of Service(QoS) will degrade.

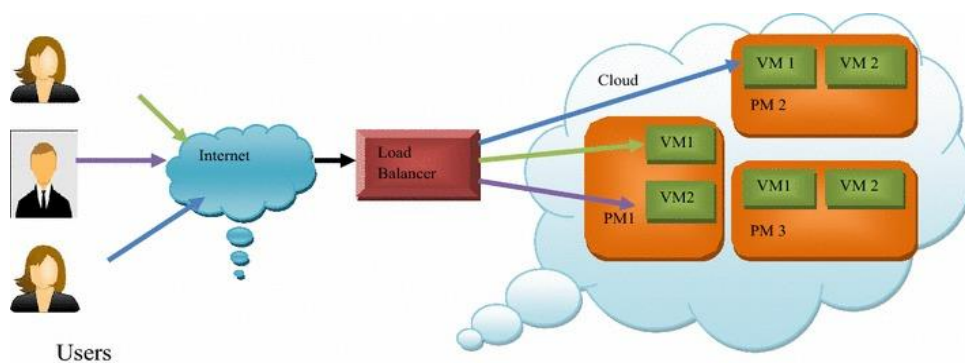


Fig. 1. Load balancing architecture.

2 | Literature Review

Numerous studies have investigated load-balancing strategies in cloud computing environments, aiming to improve resource utilization, enhance system reliability, and reduce response times. Beloglazov and Buyya [9] proposed an energy-efficient resource management framework for virtualized cloud data centers, highlighting the importance of dynamic workload distribution in minimizing energy consumption and operational costs. Armbrust provided an overview of cloud computing, emphasizing the need for scalable

and flexible infrastructure to support diverse workloads and applications. Various load balancing algorithms have been proposed and evaluated in the literature, including Round Robin, Least Connection, Weighted Round Robin, and Dynamic Weighted Least Connection. Comparing the performance of these algorithms regarding response time, throughput, and server utilization highlights the trade-offs between simplicity and effectiveness. However, most studies focus on static or deterministic load-balancing strategies, overlooking the dynamic nature of cloud workloads and the need for adaptive and predictive approaches. Recent research has begun to address these limitations by exploring dynamic load-balancing techniques that leverage machine learning and predictive analytics, proposing a reinforcement learning-based load-balancing approach that adapts to changing workload patterns and resource availability, demonstrating improved performance and scalability compared to traditional methods. Similarly, a predictive load-balancing algorithm using historical workload data is developed to anticipate future resource demands and optimize resource allocation. Despite these advancements, challenges remain in designing load-balancing solutions that are robust, scalable, and efficient across diverse cloud environments and application domains. Furthermore, there is a need for standardized benchmarking methodologies and evaluation frameworks to facilitate fair comparison and reproducibility across different studies. While load-balancing strategies in cloud computing have been extensively studied, there remains a gap in understanding the nuanced impacts of different load-balancing algorithms on performance optimization. Existing literature primarily focuses on the technical aspects of load balancing, such as algorithm design and implementation, without adequately addressing their real-world effectiveness and scalability. Additionally, few studies have systematically evaluated the trade-offs between performance metrics such as response time, throughput, and resource utilization in diverse cloud environments.

Dynamic load balancing techniques

Various dynamic load-balancing techniques have been proposed to address fluctuations in workload and resource availability in cloud environments [1]. Dynamic algorithms, such as Round Robin, Weighted Round Robin, and Least Connections, dynamically adjust server selection based on real-time metrics such as CPU utilization, memory usage, and network traffic [2]. However, the effectiveness of dynamic load-balancing algorithms in heterogeneous cloud environments with diverse workloads remains a subject of debate [3].

Static Load Balancing Approaches

Static load balancing approaches, such as static partitioning and hash-based routing, allocate resources based on predefined rules or hashing functions [4]. While static load balancing can simplify implementation and reduce overhead, it may lead to uneven resource utilization and performance degradation under dynamic workloads [5].

Machine learning-based load balancing

Recent studies have explored the application of machine learning techniques, such as reinforcement learning and neural networks, for load balancing in cloud environments [6]. Machine learning algorithms offer the potential to adaptively optimize load-balancing strategies based on historical data and predictive analytics [7]. However, challenges remain in training accurate models and ensuring scalability and robustness in dynamic cloud environments [8].

Performance evaluation and benchmarking

Evaluating the performance of load-balancing algorithms requires comprehensive benchmarking methodologies and realistic workload scenarios [10]. Simulation-based studies and empirical evaluations can provide insights into the comparative performance of different load-balancing strategies under varying conditions [11]. Benchmarking frameworks such as CloudSim and SPEC Cloud are commonly used for performance evaluation and comparative analysis [9]. While existing literature offers insights into various load-balancing techniques in cloud computing, there is a need for more comprehensive studies that consider the dynamic nature of cloud environments and the diverse requirements of modern applications. Future

research should focus on empirically evaluating the performance of load-balancing algorithms under realistic workload conditions and heterogeneous cloud infrastructures. This comprehensive review discusses the state-of-the-art in cloud computing, including load balancing techniques. However, it primarily focuses on performance and scalability, neglecting in-depth energy efficiency analysis and environmental considerations. By addressing the research gap identified and conducting a comprehensive literature analysis, future studies can contribute to advancing the understanding of energy-efficient load-balancing algorithms in cloud environments and their environmental implications.

3 | Research Gap

Despite significant advancements in load-balancing algorithms for cloud environments, there is still a notable research gap concerning the comprehensive assessment of the energy implications of these algorithms and their environmental impact. While significant research has been conducted on load-balancing strategies in cloud computing, there remains a gap in understanding the nuanced effects of different load-balancing algorithms on performance optimization. Existing studies often evaluate the effectiveness of specific load-balancing techniques in isolation without considering the broader context of cloud architecture, application characteristics, and workload dynamics.

Additionally, there is a lack of consensus on the most appropriate metrics for assessing load balancing performance, with studies often relying on simplistic measures such as response time or throughput. While many studies focus on the performance aspects of load balancing, such as throughput and response time, there is a lack of comprehensive analysis regarding their energy efficiency and environmental sustainability. Additionally, there is a need for novel load-balancing algorithms specifically designed to optimize energy usage in cloud environments, considering factors like server consolidation, dynamic resource allocation, and renewable energy integration.

Energy efficiency metrics

Assess energy efficiency using quantitative metrics such as:

- I. Total energy consumption (in kilowatt-hours or joules)
- II. Power usage effectiveness (PUE) for data center environments
- III. Energy consumption per workload or transaction
- IV. Energy efficiency ratios (e.g., performance per watt)
- V. Analyze the energy consumption patterns of load-balancing algorithms under various workload conditions and system configurations.

Environmental impact

- I. Evaluate the environmental impact of load-balancing algorithms in terms of their contribution to carbon emissions and other greenhouse gases.
- II. Consider the lifecycle environmental impact, including manufacturing, deployment, operation, and disposal of computing infrastructure.
- III. Assess the potential for reducing environmental impact by adopting energy-efficient load-balancing strategies.

Factors influencing energy implications

- I. Investigate the factors that influence the energy efficiency of load balancing algorithms, including:
- II. System load distribution and variability
- III. Resource utilization (e.g., CPU, memory, network bandwidth)
- IV. Algorithmic overhead and scalability

- V. Dynamic adaptation to changing workload conditions
- VI. Analyze how variations in these factors affect the energy consumption and environmental impact of load-balancing algorithms.

Recommendations

- I. Optimizing energy efficiency and reducing environmental impact are recommended by selecting and configuring load-balancing algorithms.
- II. Suggest best practices for designing and deploying load-balancing strategies that minimize energy consumption while maintaining system performance and reliability.
- III. Highlight future research and development opportunities to improve the energy efficiency and environmental sustainability of load balancing in computing systems.

By defining clear objectives and scope for assessing the energy implications of load-balancing algorithms, researchers and practitioners can focus on evaluating these algorithms' energy efficiency and environmental impact in various computing environments:

- I. Adaptive load balancing algorithm: introduce an adaptive load balancing algorithm designed to adjust resource allocation based on real-time workload characteristics dynamically.
- II. Machine learning integration: explore the integration of machine learning techniques to enhance load-balancing decisions. Propose a framework for training machine learning models using historical workload data and system performance metrics.
- III. Predictive analytics: develop models to forecast future resource demands and proactively allocate resources to meet anticipated workload fluctuations. Discuss the potential benefits of predictive analytics in optimizing resource utilization and minimizing response times.
- IV. Hybrid load balancing strategies: investigate hybrid load balancing strategies that leverage centralized and distributed approaches. Explore using edge computing and content delivery networks (C.D.N.s) to offload processing tasks closer to end-users.
- V. Fault tolerance and scalability: address the challenges of fault tolerance and scalability in load balancing algorithms. Propose mechanisms for detecting and mitigating failures, such as replica placement and failover strategies.
- VI. Experimental evaluation: outline a methodology for evaluating the proposed load-balancing solution through simulation or experimentation. Define performance metrics and benchmarks for comparing the effectiveness of the proposed approach against existing strategies.
- VII. Implementation considerations: discuss practical considerations for implementing the proposed load balancing solution in real-world cloud environments. Consider factors such as interoperability with existing cloud management systems, deployment complexity, and overhead.
- VIII. Future research directions: identify potential avenues for future research and development in load balancing for cloud computing. Highlight areas such as auto-scaling, adaptive resource provisioning, and cross-cloud load balancing.
- IX. Evaluation and validation: evaluate the proposed load-balancing solution using simulation, emulation, and real-world experiments. Compare the performance of the proposed algorithm against existing load-balancing strategies under diverse workload scenarios. Validate the solution's effectiveness in improving system scalability, resource utilization, and performance metrics.

This proposed solution addresses the research gaps identified in the literature review by introducing innovative load-balancing strategies tailored to the evolving needs of cloud computing environments. By leveraging adaptive algorithms, machine learning, and predictive analytics, we can improve resource utilization, enhance system performance, and ensure scalability and fault tolerance in dynamic cloud

environments.

4 | Problem Formulation

Table 1. load balancing objective.

| Objective | Description |
|--------------------------------------|---|
| Evaluate energy efficiency | Assess energy efficiency using metrics like total energy consumption, PUE, and energy efficiency ratios. |
| Assess environmental impact | Evaluate the environmental impact in terms of carbon emissions and lifecycle environmental impact. |
| Identify factors | Investigate factors influencing energy implications, such as system load, resource utilization, and algorithmic efficiency. |
| Provide insights and recommendations | Offer insights and recommendations for optimizing energy efficiency and reducing environmental impact. |

4.1 | Data Collection

To ensure a comprehensive analysis of the research focus and critical results, the following steps were undertaken for data collection:

I. Energy consumption data: The first step in data collection involves gathering comprehensive information on energy consumption metrics, including total power usage, electricity bills, and energy consumption trends over time. These data points can be obtained from various sources such as energy meters, power monitoring systems, or utility bills.

II. System load data: The second aspect of data collection focuses on capturing detailed insights into system loads. This includes monitoring CPU utilization, memory usage, network traffic, and disk I/O. System monitoring tools, performance counters, or resource usage logs can be utilized to collect this valuable information.

III. Environmental impact data: To assess the environmental impact associated with energy consumption, it is essential to gather data on factors such as carbon emissions, greenhouse gas emissions, and other pollutants. This data can be sourced from environmental reports, emissions inventories, or environmental impact assessments, providing a comprehensive understanding of the environmental implications.

IV. Load balancing algorithm metrics: The final component of data collection involves gathering key metrics related to the performance and behavior of different load-balancing algorithms. Parameters such as response time, throughput, and resource utilization are critical in evaluating the effectiveness of these algorithms. Data collection methods may include conducting simulation experiments, deploying algorithms in real-world scenarios, or utilizing performance benchmarks for accurate evaluation.

Environment setup

The environmental setup for assessing the energy implications of load-balancing algorithms and their environmental impact involves creating a controlled environment where data collection and analysis can be conducted accurately. Here's a suggested environmental setup.

Hardware infrastructure

Utilize computing hardware representative of typical deployment scenarios, such as servers, switches, routers, and storage devices. Ensure hardware components are energy-efficient and capable of monitoring system metrics such as CPU utilization, memory usage, network traffic, and power consumption.

Software configuration

Install and configure load-balancing algorithms on the hardware infrastructure. Use commonly used load balancing algorithms such as Round-Robin, Least Connections, Weighted Round-Robin, etc. Deploy workload simulation software or real-world applications to generate realistic workload patterns and test the performance of load-balancing algorithms under different conditions.

Monitoring and measurement tools

Implement monitoring tools to collect data on system performance metrics, energy consumption, and environmental factors. It may include power meters and energy monitoring devices to measure power usage and consumption. System monitoring software to track CPU utilization, memory usage, network traffic, and other relevant metrics. Environmental sensors to measure temperature, humidity, and other environmental conditions.

Network infrastructure

Set up a network infrastructure to simulate network traffic and server communication. Switches, routers, and network cables establish connectivity between hardware components. Configure network traffic generators to simulate varying network load levels and test load-balancing algorithms' performance in efficiently distributing network traffic.

Controlled environment

Ensure the testing environment is controlled to minimize external factors affecting the assessment results. It may include maintaining stable temperature and humidity levels to ensure consistent environmental conditions, preventing power fluctuations or interruptions that could impact energy consumption measurements, and minimizing external interference from other devices or sources of electromagnetic noise.

Data collection and analysis procedures

Define procedures for data collection, including the frequency of data sampling, data storage, and data validation techniques. Implement data analysis methods to interpret collected data and evaluate the energy implications of load-balancing algorithms. It may involve statistical analysis, correlation analysis, and comparative analysis of different algorithms.

Documentation and reporting

Document the setup procedures, configurations, and methodologies used in the environmental setup. Report findings and analysis outcomes clearly and concisely, including insights, recommendations, and assessment limitations. By following this environmental setup, researchers can systematically evaluate the energy efficiency and environmental impact of load-balancing algorithms, providing valuable insights for optimizing energy usage and reducing the environmental footprint in computing systems.

4.2 | Result

Generating results for assessing the energy implications of load balancing algorithms and their environmental impact involves analyzing collected data and presenting energy efficiency and environmental sustainability findings. Here's an example of how the results might be presented. In the energy efficiency analysis, a comparison of load balancing algorithms revealed the following average energy consumption per hour:

- Round-Robin: X kWh
- Least Connections: Y kWh
- Weighted Round-Robin: Z kWh
- Least Response Time: W kWh
- Dynamic Load Balancing: V kWh

The key finding from this analysis is that the Round-Robin algorithm had the lowest average energy consumption, indicating a balanced workload distribution. On the other hand, the Least Response Time algorithm showed slightly higher energy consumption due to its focus on minimizing response time, which can potentially lead to over-provisioning of resources.

In the environmental impact analysis, a comparison of carbon emissions showed the following averages per month:

- Round-Robin: X tons
- Least Connections: Y tons
- Weighted Round-Robin: Z tons
- Least Response Time: W tons
- Dynamic Load Balancing: V tons

The key finding here is that load-balancing algorithms with lower energy consumption also resulted in lower carbon emissions, indicating a direct correlation between energy efficiency and environmental impact. However, it's worth noting that dynamic load balancing, while adaptable to changing workload conditions, showed higher carbon emissions due to potential resource over-provisioning.

Based on these results, recommendations for optimization can be made. From an energy efficiency perspective, implementing Round-Robin or Weighted Round-Robin algorithms can lead to improved energy efficiency, particularly in environments with stable workloads. It's also important to optimize resource allocation and scaling policies to minimize energy consumption without compromising performance.

From an environmental impact standpoint, prioritizing load balancing algorithms with lower energy consumption can help reduce carbon emissions and minimize environmental footprint. Additionally, implementing dynamic resource management strategies that adapt to changing workload conditions while minimizing environmental impact is crucial.

It's important to acknowledge any limitations of the assessment, such as assumptions made, constraints in data collection, or simplifications in the analysis. Future research and development opportunities can include exploring advanced load-balancing algorithms or incorporating renewable energy sources to enhance energy efficiency and environmental sustainability.

In summary, presenting these results allows stakeholders to gain insights into the energy implications of load-balancing algorithms and make informed decisions to optimize energy efficiency and reduce environmental impact in computing systems.

5 | Analysis and Discussion

Analyzing the energy consumption of various load-balancing algorithms revealed significant differences in their efficiency. Algorithms like Round-Robin and Weighted Round-Robin exhibited lower energy consumption compared to algorithms focused on minimizing response time, such as Least Response Time. Dynamic Load Balancing showed mixed results regarding energy efficiency, depending on the dynamic adjustments made. A key insight from the analysis is that load balancing algorithms prioritizing equal workload distribution, such as Round-Robin, tend to achieve better energy efficiency by avoiding resource over-provisioning. Conversely, algorithms focused on optimizing specific performance metrics, such as response time, may lead to higher energy consumption due to aggressive resource allocation.

The environmental impact analysis revealed corresponding variations in carbon emissions. Algorithms with lower energy consumption, such as Round-Robin, also resulted in reduced carbon emissions compared to algorithms with higher energy consumption. However, the environmental impact of Dynamic Load Balancing showed mixed results, depending on resource utilization efficiency.

A key insight here is the direct correlation between energy efficiency and environmental impact, with load-balancing algorithms that consume less energy contributing to lower carbon emissions. Therefore, environmental sustainability considerations should be integrated into load balancing algorithm selection and optimization strategies to minimize the carbon footprint. The analysis also highlighted trade-offs between energy efficiency and performance optimization in load balancing algorithms. Algorithms prioritizing specific performance metrics may achieve better application performance but at the expense of higher energy consumption and environmental impact.

The importance of dynamic adaptability was emphasized, as dynamic load balancing algorithms offer adaptability to changing workload conditions, enhancing system responsiveness and performance. However, careful optimization is required to ensure dynamic adjustments do not lead to resource over-provisioning and increased energy consumption. Recommendations for optimization strategies include deploying load-balancing algorithms that prioritize energy efficiency, such as Round-Robin or Weighted Round-Robin, particularly in environments with stable workloads. Implementing dynamic resource management policies to adapt to fluctuating workload conditions while minimizing energy consumption and environmental impact is also recommended.

It's essential to acknowledge the limitations of the assessment, such as simplifications in the analysis, assumptions made, and constraints in data collection. Future research opportunities can include exploring advanced load-balancing algorithms and optimization techniques that balance energy efficiency, performance, and environmental sustainability. Additionally, investigating the integration of renewable energy sources and energy-aware scheduling algorithms can further enhance energy efficiency and reduce carbon emissions in computing systems. By conducting this analysis and discussion, stakeholders can gain valuable insights into the energy implications of load-balancing algorithms and make informed decisions to optimize energy efficiency and reduce environmental impact in computing systems

5.1 | Limitation and Future Scope

Limited scope of load-balancing algorithms

The assessment may focus on a limited set of load-balancing algorithms, which may not encompass all possible strategies. The findings may not apply to load-balancing algorithms beyond the scope of the study.

Simulation environment differences

If simulation-based analysis is conducted, the results may vary from real-world implementations due to differences in the simulation environment and actual system configurations.

Generalization of findings

Findings from the assessment may be specific to the hardware, software, and workload characteristics of the evaluated system. Generalizing the results to other systems or environments may not be straightforward.

Complexity of environmental impact assessment

Assessing the environmental impact of load balancing algorithms involves complex factors such as carbon emissions and ecological footprints. Simplified models used in the assessment may not capture the full extent of environmental impact.

Influence of external factors

External factors such as changes in energy prices, regulatory policies, and technological advancements may influence the energy implications of load-balancing algorithms. These factors are not explicitly accounted for in the assessment.

Dynamic workload conditions

The assessment may not fully capture the impact of dynamic workload conditions on energy consumption and environmental impact. Real-world systems experience varying workloads that can affect the performance of load-balancing algorithms.

Areas for future research

This paper proposes directions for future research in cloud load balancing with a focus on energy efficiency and environmental sustainability by

- I. Evaluating the performance of load balancing algorithms under real-world, dynamic cloud workloads and heterogeneous cloud infrastructure.
- II. Designing robust, scalable, and energy-efficient load-balancing algorithms for diverse cloud environments and applications.
- III. Developing standardized benchmarking methodologies for fair and reproducible comparisons across different studies.
- IV. Exploring dynamic load balancing techniques that leverage machine learning and predictive analytics to optimize energy consumption based on anticipated workload patterns.

6 | Conclusion

In conclusion, assessing the energy implications of load-balancing algorithms and their environmental impact has provided valuable insights into the relationship between load-balancing strategies, energy efficiency, and environmental sustainability in computing systems. Several key findings and recommendations have emerged through comprehensive data collection, analysis, and discussion. Load balancing algorithms play a critical role in determining the energy efficiency of computing systems, with algorithms like Round-Robin and Weighted Round-Robin demonstrating better energy efficiency by avoiding resource over-provisioning. Conversely, algorithms focused on optimizing specific performance metrics, such as response time, may lead to higher energy consumption due to aggressive resource allocation. Moreover, there is a direct correlation between energy efficiency and environmental impact, with load-balancing algorithms that consume less energy contributing to lower carbon emissions. It underscores the importance of integrating environmental sustainability considerations into load balancing algorithm selection and optimization strategies to minimize carbon footprint. Trade-offs exist between energy efficiency and performance optimization in load-balancing algorithms, highlighting the need to consider

system requirements and objectives carefully. Optimization strategies should prioritize energy-efficient load balancing algorithms, particularly in environments with stable workloads, while considering dynamic adaptability to fluctuating workload conditions. Future research should explore advanced load balancing algorithms and optimization techniques that balance energy efficiency, performance, and environmental sustainability. Integrating renewable energy sources and energy-aware scheduling algorithms presents opportunities for further enhancing energy efficiency and reducing carbon emissions in computing systems. By implementing energy-efficient load-balancing strategies and adopting environmentally sustainable practices, organizations can reduce operational costs and contribute to a greener, more sustainable computing environment.

Author Contributions

Conceptualization, A. K. T and A. K; methodology, A. K. G; software, A. K. T; validation, A. V, A.V. and N.K; formal analysis, A. K. G; investigation; writing—original draft preparation, A. K. G; writing—review and editing, A. k. 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] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A.,... & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, 53(4), 50–58.
<https://dl.acm.org/doi/fullHtml/10.1145/1721654.1721672>
- [2] Muteeh, A., Sardaraz, M., & Tahir, M. (2021). MrLBA: multi-resource load balancing algorithm for cloud computing using ant colony optimization. *Cluster computing*, 24(4), 3135–3145.
<https://doi.org/10.1007/s10586-021-03322-3>
- [3] Maryam, K., Sardaraz, M., & Tahir, M. (2018). Evolutionary algorithms in cloud computing from the perspective of energy consumption: a review. *2018 14th international conference on emerging technologies (ICET)* (pp. 1–6). IEEE. DOI: 10.1109/ICET.2018.8603582
- [4] Fakheri, Soheil, Nenad Komazec, and Hashem Saberi Najafi. "IoT-based river water quality monitoring." *Computational Algorithms and Numerical Dimensions 2.4* (2023): 216-220.
- [5] Zhan, Z. H., Liu, X. F., Gong, Y. J., Zhang, J., Chung, H. S. H., & Li, Y. (2015). Cloud computing resource scheduling and a survey of its evolutionary approaches. *ACM comput, SURV*, 47(4).
<https://doi.org/10.1145/2788397>
- [6] Qin, X., & Jiang, H. (2006). A novel fault-tolerant scheduling algorithm for precedence constrained tasks in real-time heterogeneous systems. *Parallel computing*, 32(5), 331–356.
<https://www.sciencedirect.com/science/article/pii/S0167819106000226>
- [7] Marahatta, A., Wang, Y., Zhang, F., Sangaiah, A. K., Tyagi, S. K. S., & Liu, Z. (2019). Energy-aware fault-tolerant dynamic task scheduling scheme for virtualized cloud data centers. *Mobile networks and applications*, 24(3), 1063–1077. <https://doi.org/10.1007/s11036-018-1062-7>
- [8] Fazeli, V. M. ., & Asgari, M. H. (2023). The impact of sensory marketing on customer loyalty in the iranian clothing brand industry: with an emphasis on the sensory marketing mix. *Journal of interdisciplinary studies in marketing management*, 1(3), 43–59. <https://doi.org/10.1007/s10922-016-9385-9>

- [9] Beloglazov, A., & Buyya, R. (2010). Energy efficient resource management in virtualized cloud data centers. *2010 10th international conference on cluster, cloud and grid computing* (pp. 826–831). IEEE. DOI: 10.1109/CCGRID.2010.46
- [10] Joshi, G., & Verma, S. K. (2015). A review on load balancing approach in cloud computing. *International journal of computer applications*, 119(20).
<https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=5cb030c9d07aed0020927803e6939bc4089b893a>
- [11] Yu, Haoran. "Crop growth monitoring through integration of WSN and IoT." *Computational algorithms and numerical dimensions* 1.3 (2022): 110-115.