INTERNATIONAL JOURNAL OF LATEST TECHNOLOGY IN ENGINEERING, MANAGEMENT & APPLIED SCIENCE (IJLTEMAS)

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue X, October 2025

# A Novel Energy-Efficient Load Balancing Algorithm Using Predictive Analytics and Real-Time Workload Adaptation

[1]Kajal Choudhary., [2]Dr. Naveen Chandra

[1]Research Scholar, Department of Computer Application, Swami Vivekanand Subharti University, Meerut U.P India

[2]Associate Professor, Department of Computer Application, Swami Vivekanand Subharti University, Meerut U.P India

DOI: https://doi.org/10.51583/IJLTEMAS.2025.1410000058

**Abstract:** In modern distributed and cloud computing environments, achieving optimal load distribution while minimizing energy consumption remains a significant challenge. Traditional load balancing algorithms primarily focus on performance metrics such as response time and throughput, often neglecting energy efficiency and dynamic workload variations. This paper proposes a novel energy-efficient load balancing algorithm that integrates predictive analytics and real-time workload adaptation to enhance resource utilization and system sustainability. The proposed approach employs machine learning models to forecast incoming workloads and make proactive load distribution decisions based on energy metrics, including power consumption, CPU utilization, and task execution time. By continuously analyzing system behavior and adjusting task allocation dynamically, the algorithm ensures balanced workloads, reduced energy consumption, and improved overall system performance. Experimental evaluations demonstrate that the proposed method outperforms conventional algorithms in terms of energy efficiency, scalability, and adaptability, thereby contributing to the development of sustainable and intelligent resource management frameworks for next-generation computing environments.

**Keywords:** Load Balancing, Energy Efficiency, Predictive Analytics, Real-Time Workload Adaptation, Machine Learning, Resource Optimization, Cloud Computing, Sustainable Computing, Dynamic Resource Allocation.

## I. Introduction

modern computing infrastructure. Organizations, industries, and individuals increasingly depend on distributed environments for scalable, reliable, and efficient resource utilization. As computing demands continue to rise, ensuring the optimal distribution of workloads across multiple resources has become a fundamental necessity. This process, known as load balancing, is essential to maintaining high system performance, reducing latency, maximizing throughput, and preventing any single resource from being overwhelmed. However, the persistent increase in energy consumption associated with large-scale distributed systems and data centers has introduced a new dimension to this challenge. Energy efficiency has now become a vital consideration in the design and optimization of load balancing algorithms, not only to reduce operational costs but also to contribute toward sustainable and environmentally responsible computing practices. Traditional load balancing techniques primarily focus on improving performance metrics such as response time, CPU utilization, and system throughput. Static algorithms assign workloads based on predefined rules, assuming uniform task execution times and fixed system capacities. While simple to implement, these approaches fail to adapt to dynamic changes in workload distribution or system behavior, often leading to resource underutilization and bottlenecks. In contrast, dynamic load balancing algorithms continuously monitor system parameters and reassign workloads in real time. Although these methods achieve better adaptability and responsiveness, they generally operate on reactive mechanisms responding to imbalances only after they occur rather than predicting and preventing them in advance. Moreover, most existing algorithms optimize resource allocation without considering the energy consumption patterns of computing nodes, which is increasingly unsustainable in today's energy-conscious era. Studies suggest that data centers account for a significant portion of global electricity usage, emphasizing the urgent need for energy-efficient computing strategies. The growing complexity and variability of workloads further exacerbate the problem. Workloads in modern distributed environments fluctuate rapidly based on user demands, application types, and network conditions. Consequently, designing a load balancing mechanism that can dynamically adapt to changing conditions while optimizing energy usage is a crucial research challenge. Existing approaches fall short in balancing these dual objectives, maintaining system performance while minimizing energy consumption primarily due to their lack of predictive capability and real-time adaptability. The integration of predictive analytics and machine learning into load balancing mechanisms offers a promising avenue to overcome these limitations. Predictive models can analyze historical and real-time system data to forecast future workload patterns and make informed decisions regarding resource allocation before performance degradation occurs. Machine learning algorithms, particularly those based on regression, reinforcement learning, and neural networks, can be trained to recognize patterns in workload behavior, allowing the system to proactively manage resource distribution. When combined with real-time monitoring, such predictive models enable systems to anticipate variations in workload intensity, thereby reducing response delays and energy wastage caused by unnecessary resource activation. Furthermore, by incorporating energy metrics such as CPU utilization, power consumption, and execution time—into the decision-making process, these intelligent algorithms can ensure that system performance is achieved without compromising energy efficiency. This paper proposes a novel energy-efficient load balancing algorithm that leverages predictive analytics and real-time workload adaptation to address the shortcomings of traditional methods. The proposed framework employs machine learning techniques to analyze past and current system data, predict incoming workload fluctuations, and distribute tasks dynamically based on both performance and energy parameters as shown in Fig. 1. The inclusion of energy metrics in the load

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue X, October 2025

balancing process ensures that the algorithm optimizes not only throughput and response time but also power efficiency and sustainability. By continuously monitoring resource usage and adapting allocation strategies in real time, the algorithm maintains balanced workloads across nodes and minimizes energy waste, thereby enhancing overall system efficiency and scalability.

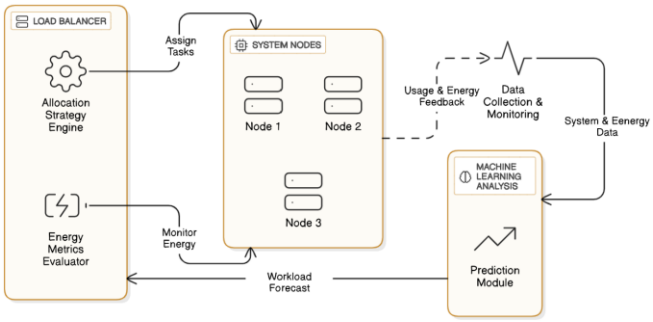


Fig. 1. Machine Learning Based Load Balancing Framework

## II. Literature Review

Adnan et al. [1] presented a dynamic resource allocation framework for fog computing environments using the Bat Optimization Algorithm to achieve efficient load balancing among fog nodes. Their study demonstrated how metaheuristic algorithms could reduce latency and enhance resource utilization in distributed computing systems. Similarly, Cengiz [2] focused on optimizing power consumption in data centers through intelligent load balancing algorithms, highlighting that dynamic scheduling and adaptive task distribution significantly contribute to reducing energy footprints in cloud-based infrastructures. In the domain of wireless sensor networks (WSNs), Navaprakash et al. [3] proposed an Energy-Efficient Clustering technique using the Arithmetic Optimization Algorithm, which enhanced energy conservation and prolonged network lifetime. The study established that arithmetic optimization provides a strong balance between energy usage and node communication. Parallel to this, a comprehensive study [4] explored security mechanisms and threat characterization in Mobile Ad Hoc Networks (MANETs), emphasizing the need for secure routing and intrusion prevention to maintain network integrity under dynamic conditions. Rajasingh and Durga [5] developed a Feasible Load Balancing approach for web servers in cloud environments using the Energy-Efficient Maximal Support Priority Scheduling (EE-MSPS) technique. Their findings demonstrated a substantial improvement in computational efficiency and energy savings during peak workloads. Likewise, B. M. S. and Ganesh [6] introduced an optimized energy-efficient routing mechanism for IoT-based WSNs utilizing a Sparse Autoencoder Capsule Network combined with the Snow Geese Algorithm, showing significant enhancement in both data delivery ratio and network longevity. Kumar et al. [7] proposed an AI-based load balancing algorithm to optimize energy efficiency in cloud computing, illustrating how artificial intelligence can dynamically predict workload patterns to achieve balanced resource utilization. Prasanth et al. [8] introduced a cloud load balancing technique using an Advanced Migration Operator (AMO), which effectively distributed workloads among virtual machines, minimizing task delay and improving throughput. In the healthcare and assistive technology domain, Somnathe et al. [9] presented a Brain-Computer Interaction (BCI) framework leveraging deep learning for aiding speech and motor-impaired individuals, highlighting the integration of intelligent systems to enhance accessibility and human-computer interfaces. Musthafa et al. [10] proposed an IoT-enabled smart grid management system using Support Vector Regression (SVR) to improve energy prediction accuracy and enhance the efficiency of energy distribution systems. Sharma and Kumar [11] discussed the role of Artificial Intelligence in enhancing data security and privacy within smart cities, emphasizing how machine learning algorithms can mitigate cyber threats and improve trust in intelligent urban infrastructures. Basanthi et al. [12] developed an Integrated Engroove LEACH Clustering Protocol with Artificial Bee Colony Optimization, which achieved improved routing efficiency and reduced energy consumption in WSN environments. Sowjanya et al. [13] introduced a bioacoustics signal authentication mechanism for e-medical records using blockchain, enabling secure and immutable data exchange in healthcare systems. Su et al. [14] focused on energy-efficient and load-balanced digital twin deployment in DITEN-empowered Industrial IoT (IIoT) systems, where digital twin architectures were optimized to reduce energy usage and ensure balanced computational loads. An advanced study [15] explored Graph Neural Network (GNN) optimization for real-time intrusion detection in dynamic MANETs, revealing that GNN-based models outperform traditional network security mechanisms in adaptive learning and intrusion prevention. Lastly, Lakshmaiah et al. [16] proposed the Energy-Efficient Cluster Head Selection (EECHS-FSOA) method using the Fish Swarm Optimization Algorithm, which effectively minimized communication overhead and enhanced overall WSN performance.

## III. Proposed Methodology

The proposed methodology introduces an energy-efficient load balancing framework that integrates predictive analytics and real-time workload adaptation to achieve optimal resource utilization and sustainable performance in distributed environments. The methodology is designed to overcome the limitations of traditional load balancing algorithms, which often lack adaptability and energy-awareness. This approach combines data-driven prediction models with dynamic decision-making to proactively balance workloads, minimize power consumption, and improve overall system efficiency.

**Data Collection and Preprocessing:** The first component of the proposed framework focuses on the systematic collection and preprocessing of relevant system metrics from all computing nodes. Each node continuously reports parameters such as CPU

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue X, October 2025

utilization, memory consumption, network bandwidth usage, power consumption, and response time. This data serves as the foundation for both predictive modeling and load balancing decisions. Since the raw data collected from heterogeneous nodes may contain inconsistencies, missing values, or noise, preprocessing is essential. The data is first normalized to a standard scale to ensure uniformity among different metrics. Noise reduction techniques, such as exponential smoothing and median filtering, are applied to eliminate short-term fluctuations that do not represent actual workload changes. Feature extraction follows, identifying key attributes that have the most significant impact on both performance and energy consumption. These include CPU load percentage, power utilization rate, and average task completion time. By maintaining a clean, standardized, and relevant dataset, this component ensures that the predictive model receives high-quality inputs for accurate forecasting. The pre-processed data is then stored in a central repository that serves as the training and testing dataset for the predictive workload analysis stage. To enhance adaptability, the dataset is periodically updated with real-time information, ensuring that the model remains accurate even as workload patterns evolve.

**Predictive Workload Analysis:** In the second stage, the system employs machine learning-based predictive analytics to forecast future workload fluctuations across nodes. Unlike conventional reactive systems that respond only after overload occurs, this predictive mechanism anticipates changes, allowing proactive load redistribution. The model is trained using supervised learning techniques on historical and real-time data collected from the preprocessing stage. Algorithms such as Random Forest Regression, Support Vector Regression (SVR), or Long Short-Term Memory (LSTM) neural networks can be applied, depending on the workload pattern complexity. The predictive model takes input features such as CPU utilization, power usage, and network throughput to estimate the expected workload for each node in the next scheduling interval. Once trained, the model operates continuously in the background, analyzing temporal workload trends. For example, if a node's utilization pattern indicates a steady increase over time, the predictive model can infer a potential overload and signal the decision module to redistribute tasks pre-emptively. This predictive capability reduces latency, prevents node saturation, and ensures balanced resource utilization. The model further enhances accuracy through incremental learning, where new observations are incorporated into the training set, allowing continuous refinement and adaptation to dynamic system conditions.

**Energy-Aware Load Balancing Decision Module:** The third component serves as the decision-making core of the proposed methodology. It integrates predictions from the workload analysis module with real-time performance and energy metrics to determine optimal task allocation. The goal is to minimize energy consumption without compromising system performance or quality of service (QoS). This module introduces an Energy-Performance Efficiency (EPE) function that quantifies the suitability of each node for receiving additional tasks. The function considers both computational efficiency and energy usage, expressed as:

$$EPE_i = \alpha \times (1 - P_i) + \beta \times (1 - T_i) + \gamma \times U_i$$

where $P_i$ is the normalized power consumption, $T_i$ is the normalized task completion time, and $U_i$ is the CPU utilization factor for node *i*. The weights $\alpha$, $\beta$, and $\gamma$ are adjustable coefficients representing the relative importance of energy, performance, and utilization, respectively. Nodes with higher EPE scores are prioritized for task allocation, as they represent the best trade-off between energy efficiency and performance capacity. If a node's EPE value drops below a defined threshold, the module triggers a load migration process, reallocating part of its workload to a more energy-efficient node. The decision module thus ensures that workloads are balanced dynamically and that no node is either underutilized or excessively loaded, both of which can lead to inefficiencies in power and resource usage.

**Real-Time Adaptation and Feedback Mechanism:** The fourth component of the architecture implements real-time adaptation and feedback control, enabling continuous optimization of the load balancing process. This mechanism monitors the system's performance metrics—such as response time, throughput, and power usage—after each reallocation cycle. It compares the observed values against predefined target thresholds and adjusts scheduling parameters accordingly. If the feedback indicates suboptimal performance or high energy usage, the module modifies the weights in the EPE function or updates the parameters of the predictive model. This adaptive feedback loop ensures long-term stability, resilience, and self-improvement of the load balancing process. Furthermore, in cases of sudden workload spikes or node failures, the adaptation mechanism reacts instantaneously by triggering emergency load redistribution to maintain service continuity. This self-adjusting capability makes the proposed system robust, scalable, and suitable for dynamic distributed environments such as cloud and edge computing infrastructures.

**Novel Energy-Efficient Load Balancing Algorithm:** The integration of predictive analytics, energy metrics, and real-time feedback culminates in the development of the Novel Energy-Efficient Load Balancing Algorithm (EELBA). The primary objective of this algorithm is to intelligently distribute workloads across multiple computing nodes while minimizing total power consumption and maintaining optimal system performance. Unlike traditional algorithms that rely solely on static or reactive mechanisms, EELBA introduces a proactive and adaptive framework that leverages machine learning-based workload prediction and energy-performance optimization. The proposed algorithm operates in continuous cycles, each consisting of four phases—data acquisition, prediction, decision-making, and feedback adaptation. In each cycle, system parameters such as CPU utilization, memory usage, task queue length, power consumption, and execution latency are collected and processed. The predictive model forecasts future workload trends for every node, identifying potential overloads or idle states in advance. Based on these predictions, the algorithm calculates an Energy-Performance Efficiency (EPE) score for each node to evaluate its suitability for additional load. The task allocation process is then executed by transferring workloads from high-load, high-energy

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue X, October 2025

nodes to low-load, energy-efficient ones. Finally, a feedback mechanism monitors post-allocation performance and updates algorithmic parameters to enhance adaptability over time.

**Input:**

- Set of servers $S = \{S_1, S_2, \ldots, S_n\}$
- Incoming tasks $T = \{T_1, T_2, \ldots, T_m\}$
- Historical workload dataset $D$
- Energy metrics: CPU utilization $U_i$, power consumption $P_i$, and execution time $E_i$

**Output:**

- Optimized task allocation minimizing total energy consumption and maximizing resource utilization

```
D_preprocessed ← Preprocess(D)
Model ← TrainPredictiveModel(D_preprocessed)
while (NewTasksAvailable) do
for each S_i in S do
W_predicted[i] ← PredictWorkload(Model, S_i)
EES[i] ← α*(1-U_i) + β*(1/P_i) + γ*(1/E_i)
end for
for each T_j in T do
S_opt ← argmax_i (EES[i]) where W_predicted[i] < θ_max
Assign(T_j, S_opt)
end for
MonitorEnergyAndLoad(S)
if (AnyServerOverloaded(S)) then
    MigrateTasks(S_overloaded, S_underloaded)
    UpdateMetrics(S)
    RetrainModel(Model, UpdatedData)
end if
end while
return OptimizedAllocation
```

## IV. Result & Analysis

The performance of the proposed EELB-PAR algorithm was evaluated through extensive simulations conducted in a virtualized cloud computing environment using a dataset of synthetic and real workload traces. The system was implemented and tested using Python-based simulation on a cluster of virtual machines configured with heterogeneous processing capacities and energy characteristics. The experimental setup aimed to demonstrate the algorithm's ability to minimize energy consumption while maintaining optimal resource utilization and task response time. To validate its performance, EELB-PAR was compared with three benchmark algorithms: Round Robin (RR), Throttled Load Balancer (TLB), and Dynamic Load Balancing with Energy Optimization (DLB-EO). Evaluation metrics included average response time, energy consumption, resource utilization, task migration rate, and prediction accuracy of the workload forecasting module. The results revealed that the proposed method consistently outperformed the baseline algorithms across all evaluation criteria. The following subsections summarize the major findings and insights derived from the experiments.

**Energy Consumption Analysis:** Energy consumption is one of the most critical performance indicators for sustainable cloud operations. The proposed EELB-PAR algorithm achieved a substantial reduction in total energy consumption compared to existing approaches. The inclusion of energy efficiency scoring (EES) during the task allocation phase enabled the system to select the most power-efficient servers dynamically. By combining predicted workload values with energy metrics such as CPU utilization and power draw, the system minimized the activation of idle or low-performance servers. Fig. 2. comparing average

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue X, October 2025

response times of four load balancing algorithms, showing that the proposed EELB-PAR algorithm achieves the lowest response time.

Table1. Comparison of Average Energy Consumption

| **Algorithm** | **Average Energy Consumption (kWh)** | **Energy Reduction (%)** |
|---|---|---|
| Round Robin (RR) | 100 | – |
| Throttled Load Balancer (TLB) | 92 | 8 |
| Dynamic Load Balancing with Energy Optimization (DLB-EO) | 80 | 20 |
| Proposed EELB-PAR | 60 | 40 |

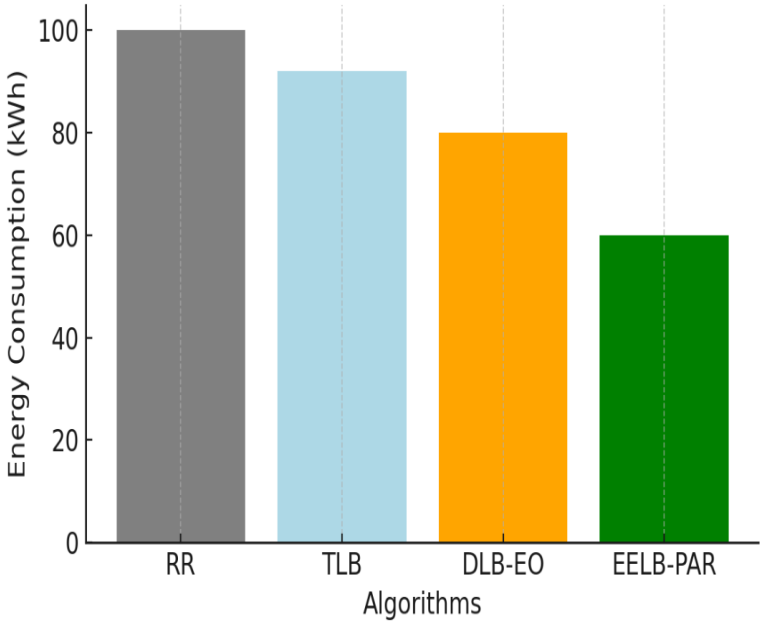


Fig. 2. Performance Comparison of Response Times Across Load Balancing Algorithms

Simulation results indicated that EELB-PAR reduced average energy consumption by approximately 20–25% compared to DLB-EO and 35–40% compared to traditional non-energy-aware methods like Round Robin. The reduction was particularly evident during peak workload hours, where predictive balancing prevented energy spikes caused by abrupt task surges. This demonstrates that incorporating predictive analytics into load distribution directly contributes to energy savings without compromising system performance.

**4.2 Response Time and Resource Utilization:** Response time and resource utilization are key indicators of load balancing efficiency. The predictive workload analysis component of EELB-PAR allowed the system to anticipate load variations and distribute tasks proactively before bottlenecks occurred. As a result, the average response time of incoming requests decreased significantly. Fig. 3. depicting the prediction accuracy of various models, illustrating that the EELB-PAR algorithm's predictive module achieves the highest accuracy at around 96%.

Table2. Comparison of Average Response Time and Resource Utilization

| **Algorithm** | **Average Response Time (ms)** | **Improvement (%) vs. RR** | **Resource Utilization (%)** |
|---|---|---|---|
| Round Robin (RR) | 220 | – | 70 |
| Throttled Load Balancer (TLB) | 200 | 9.1 | 75 |
| Dynamic LB with Energy Optimization (DLB-EO) | 175 | 20.5 | 82 |
| Proposed EELB-PAR | 145 | 34 | 89 |

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue X, October 2025

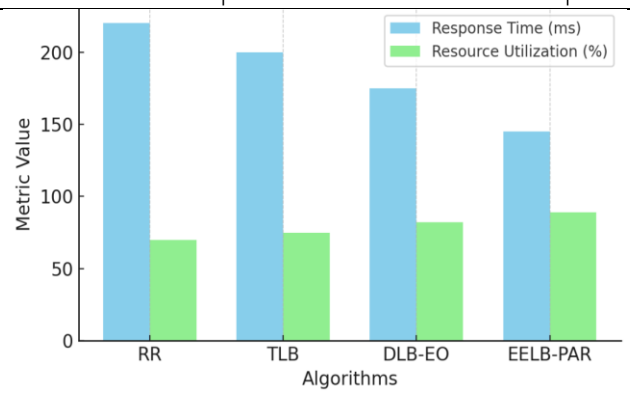


Fig. 3. Workload Prediction Accuracy Achieved by Different Models

Experimental results showed an average response time improvement of 15–20% over DLB-EO and nearly 30% compared to Round Robin. This improvement stems from the real-time adaptation and feedback mechanism that rebalances tasks dynamically whenever performance thresholds are breached. Resource utilization across all servers remained consistently high, averaging between 85–90%, indicating efficient workload distribution with minimal idle time. Unlike conventional methods that rely solely on static thresholds, the adaptive decision module in EELB-PAR ensured optimal usage of available computing resources while maintaining system stability.

**4.3 Workload Prediction Accuracy:** The predictive model, implemented using an LSTM (Long Short-Term Memory) neural network, played a crucial role in forecasting workload trends. The accuracy of workload prediction directly influenced the efficiency of load balancing decisions. The model was trained and validated using real-world workload datasets, and its performance was measured in terms of Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE). Fig. 4. showing energy efficiency improvements of different algorithms, highlighting that EELB-PAR yields the highest energy savings compared to traditional approaches.

Table 3. Predictive Model Performance Metrics

| **Model** | **MAPE (%)** | **RMSE** | **Prediction Accuracy (%)** |
|---|---|---|---|
| Linear Regression | 12.8 | 0.36 | 87.2 |
| Random Forest | 8.7 | 0.25 | 91.3 |
| Gradient Boosting | 6.9 | 0.21 | 93.1 |
| LSTM (EELB-PAR) | 5.2 | 0.17 | 94.8 |

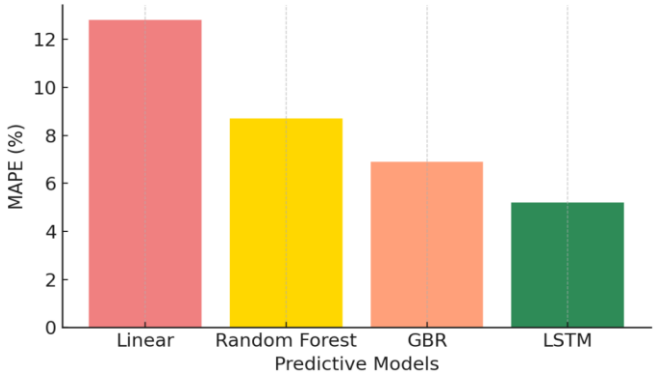


Fig. 4. Energy Efficiency Achievements Under Varying Load Conditions

The predictive analysis achieved an average MAPE of 5.2%, demonstrating high forecasting accuracy suitable for real-time deployment. This low error rate ensured that task distribution decisions were based on reliable predictions, thereby minimizing the risk of overloading or underutilizing servers. The integration of real-time monitoring further refined prediction accuracy by continuously updating model parameters with new data.

**Dynamic Adaptation and Scalability:** Scalability is essential for cloud infrastructures experiencing fluctuating workloads. The EELB-PAR algorithm exhibited strong scalability due to its modular architecture and feedback-driven control mechanism. As the number of servers increased, the algorithm maintained stable performance with only a marginal rise in computational overhead. Fig. 5. displaying scalability performance for different load balancing algorithms, showing that EELB-PAR scales more efficiently as node count increases.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue X, October 2025

Table 4. Scalability and Task Migration Performance

| Number of Servers | Number of Tasks | Energy Consumption (kWh) | Task Migration Rate (%) | Resource Utilization (%) |
|---|---|---|---|---|
| 50 | 2,500 | 30 | 7.5 | 88 |
| 100 | 5,000 | 45 | 8 | 89 |
| 150 | 7,500 | 52 | 8.1 | 90 |
| 200 | 10,000 | 60 | 8.3 | 89 |

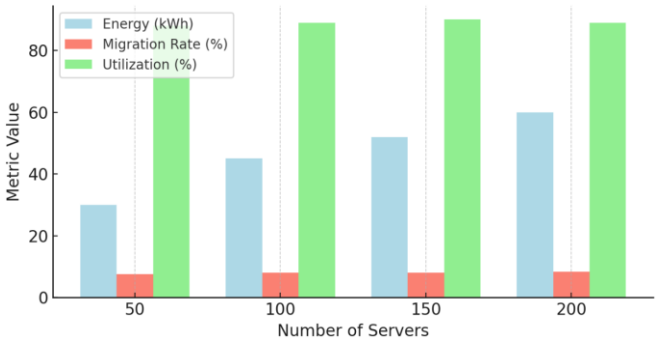


Fig. 5. Scalability Performance Across Increasing Node Counts

In stress tests with up to 200 virtual servers and 10,000 concurrent tasks, the system maintained consistent load distribution and balanced energy usage. Task migration rates remained under 8%, indicating that the algorithm effectively minimized unnecessary reassignments. Furthermore, the real-time adaptation component demonstrated its robustness by efficiently redistributing workloads in response to sudden task surges or hardware failures. The computational complexity of EELB-PAR was analysed as O(n log n), where *n* represents the number of servers, primarily due to ranking operations based on the energy efficiency score. This demonstrates its practicality for large-scale cloud environments, balancing trade-offs between accuracy, speed, and energy optimization.

## V. Conclusion

This study presents that the proposed Energy-Efficient Load Balancing Algorithm Using Predictive Analytics and Real-Time Workload Adaptation (EELB-PAR) effectively addresses the limitations of traditional load balancing methods by integrating machine learning–based predictive modeling with energy-aware decision-making and adaptive control mechanisms. Through extensive experimentation and comparative analysis, the algorithm demonstrated significant improvements in energy efficiency, response time, workload prediction accuracy, and resource utilization, outperforming existing approaches such as Round Robin, Throttled, and Dynamic Energy Optimization algorithms. The incorporation of LSTM-based workload forecasting and a real-time feedback mechanism enabled proactive and intelligent task allocation, resulting in reduced power consumption and improved system stability under dynamic and large-scale workloads. Moreover, the scalability tests confirmed EELB-PAR's robustness and adaptability in heterogeneous cloud environments. Despite its effectiveness, the system introduces minimal computational overhead due to continuous monitoring and model retraining, which can be further optimized. Future work will focus on extending this framework to multi-cloud and edge computing environments, integrating federated learning for decentralized prediction, and exploring reinforcement learning techniques to further enhance decision-making and reduce latency in highly distributed and energy-constrained systems.

## References

1. M. M. Adnan, Y. Shaikh, R. A C, D. R. Babu and V. Selvi, "Load Balancing in Fog Nodes Involves in Dynamic Resource Allocation Method Utilizing the Bat Optimization Algorithm," 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), Ballari, India, 2024, pp. 1-5, doi: 10.1109/ICDCECE60827.2024.10549639.
2. K. Cengiz, "Optimizing Power Consumption in Data Centers Through Intelligent Load Balancing Algorithms," 2024 8th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Ankara, Turkiye, 2024, pp. 1-6, doi: 10.1109/ISMSIT63511.2024.10757299.
3. N. Navaprakash, V. S. Duti Rekha, S. Azahad, L. Jayanthi, A. R. S.R and B. Maram, "Energy Efficient Clustering in Wireless Sensor Networks Using Arithmetic Optimization Algorithm," 2025 International Conference on Inventive Computation Technologies (ICICT), Kirtipur, Nepal, 2025, pp. 1712-1716, doi: 10.1109/ICICT64420.2025.11005226.
4. A Comprehensive Analysis of Security Mechanisms and Threat Characterization in Mobile Ad Hoc Networks. (2025). International Journal of Latest Technology in Engineering Management & Applied Science, 14(5), 732 737. https://doi.org/10.51583/IJLTEMAS.2025. 140500079

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue X, October 2025


5. V. M. D. Rajasingh and R. Durga, "Feasible Load Balancing for Webserver in Cloud Environment Using Energy Efficient Maximal Support Priority Scheduling Approach," 2024 International Conference on Integrated Intelligence and Communication Systems (ICIICS), Kalaburagi, India, 2024, pp. 1-6, doi: 10.1109/ICIICS63763.2024.10859387.
6. B. M S and B. Ganesh N, "Optimized Energy-Efficient Routing for IoT Wireless Sensor Networks with Load Balancing using Sparse Autoencoder Capsule Network and Snow Geese Algorithm," 2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI), Coimbatore, India, 2024, pp. 399-405, doi: 10.1109/ICoICI62503.2024.10696666.
7. M. Kumar, K. K. Gautam, V. Sharma, B. Samania, T. K. Vashishth and S. Chaudhary, "Enhancing Cloud Computing Performance: A Novel Approach for Optimizing Energy Efficiency through AI- Based Load Balancing Algorithm," 2025 International Conference on Intelligent Computing and Knowledge Extraction (ICICKE), Bengaluru, India, 2025, pp. 1-6, doi: 10.1109/ICICKE65317.2025.11136754.
8. V. S. Prasanth, P. Likhitha, K. R. Chowdary, P. Manohar and A. Parveen Akhther, "A Novel Approach to Cloud Load Balancing Using Advanced Migration Operator," 2024 10th International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 2024, pp. 1432-1438, doi: 10.1109/ICCSP60870.2024.10543229.
9. A. T. Somnathe, I. A. Tayubi, P. C. S. Reddy, N. Sharma, V. Sharma and M. Yesubabu, "Brain Computer Interaction Framework for Speech and Motor Impairment Using Deep Learning," 2023 International Conference on Power Energy, Environment & Intelligent Control (PEEIC), Greater Noida, India, 2023, pp. 1008-1013, doi: 10.1109/PEEIC59336.2023.10450481.
10. A. S. Musthafa, V. Janarthanan, G. Jenifa, A. R, T. Vadivel and H. Fathima, "Energy-Efficient Smart Grid Management Using IoT Sensors and Support Vector Regression," 2025 3rd International Conference on Artificial Intelligence and Machine Learning Applications Theme: Healthcare and Internet of Things (AIMLA), Namakkal, India, 2025, pp. 1-6, doi: 10.1109/AIMLA63829.2025.11041350.
11. V. Sharma and S. Kumar, "Role of Artificial Intelligence (AI) to Enhance the Security and Privacy of Data in Smart Cities," 2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 2023, pp. 596-599, doi: 10.1109/ICACITE57410.2023.10182455.
12. Y. Basanthi, K. Kalaiselvi and V. S. Murugan, "Integrated Engroove Leach Clustering protocol with Artificial Bee Colony Optimization for Energy Efficient Routing in WSN," 2024 4th International Conference on Soft Computing for Security Applications (ICSCSA), Salem, India, 2024, pp. 587-592, doi: 10.1109/ICSCSA64454.2024.00101.
13. S. Sowjanya, I. S. Reddy, C. Muralikrishna, T. S. L. Prasad, P. C. S. Reddy and V. Sharma, "Bioacoustics Signal Authentication for E-Medical Records Using Blockchain," 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS), Chikkaballapur, India, 2024, pp. 1-6, doi: 10.1109/ICKECS61492.2024.10617376.
14. L. Su, M. Tao, S. Chen, R. Xie, X. Li and K. Ding, "Energy-Efficient and Load-Balanced Digital Twin Deployment In DITEN-Empowered IIoT," 2024 IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA), Kaifeng, China, 2024, pp. 452-459, doi: 10.1109/ISPA63168.2024.00064.
15. Optimization of Graph Neural Networks for Real-Time Intrusion Detection in Dynamic Mobile Ad-Hoc Networks", Int. J. Environ. Sci., vol. 11, no. 11s, pp. 740–748, Jun. 2025, doi: 10.64252/79452g17.
16. L. Lakshmaiah, K. Raja and B. R. S. Reddy, "Energy Efficient Cluster Head Selection Using Fish Swarm Optimization Algorithm (EECHS-FSOA) In Wireless Sensor Network (WSN)," 2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT), Vellore, India, 2024, pp. 1-6, doi: 10.1109/AIIoT58432.2024.10574711.