International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

ENERGY-EFFICIENT JOB SCHEDULING IN GREEN CLOUD COMPUTING USING NEURAL NETWORKS

[1]Dr. Sachin Tiwari, [2]Dr. Virendra Kumar Tiwari, [3]Dr. Manish Khemariya, [4]Dr. Vijay Yadav

[1]Department of Computer Science Engineering, Lakshmi Narain College of Technology Excellence, Bhopal, 462022, India.

Email: sachintiwari@lnct.ac.in

[2]Department of Computer Application, Lakshmi Narain College of Technology (MCA), Bhopal, 462022, India.

Email: virendrat@lnct.ac.in

[3]Department of Electrical and Electronics Engineering, Lakshmi Narain College of Technology, Bhopal, 462022,

Email: manishk@lnct.ac.in

[4]Department of Computer Science Engineering, Lakshmi Narain College of Technology Excellence, Bhopal, 462022, India.

Email: Vijayy@lnct.ac.in

Abstract

Cloud computing has revolutionized modern digital infrastructure but comes with a significant energy cost due to large-scale data center operations. This paper presents a hybrid scheduling framework that integrates Artificial Neural Networks (ANNs) with the Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) to achieve energy-efficient job scheduling in green cloud environments. The ANN predicts optimal resource mappings based on real-time and historical workload metrics, while MOEA/D refines these mappings by balancing energy consumption and performance metrics such as response time. Simulation results using CloudSim demonstrate up to 15% reduction in energy usage, improved server consolidation, and lower SLA violation rates compared to traditional algorithms like FCFS and Round-Robin. The framework demonstrates strong scalability and adaptability, providing a robust solution for sustainable cloud computing infrastructures.

Keywords: Green Cloud Computing, Artificial Neural Networks, MOEA/D, Job Scheduling, Energy Efficiency, CloudSim, MOEA/D.

I. Introduction

Cloud computing serves as a cornerstone of digital transformation, enabling on-demand access to scalable computational resources. However, the growing reliance on cloud services has led to significant energy consumption, with data centers accounting for, 1 to 2% of global

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

electricity usage. This surge raises environmental concerns and emphasizes the need for sustainable computing practices.

Green cloud computing aims to minimize the ecological impact of cloud operations by reducing energy usage while maintaining acceptable performance levels. Among various strategies, intelligent job scheduling has emerged as a promising approach to optimize resource utilization and minimize energy waste.

Recent advancements in Artificial Intelligence (AI), particularly Artificial Neural Networks (ANNs), offer powerful tools for predictive resource management. ANNs can learn complex workload patterns and facilitate accurate task-to-resource mappings. When combined with multi-objective optimization techniques like MOEA/D, which decomposes conflicting goals into tractable subproblems, an effective and adaptable scheduling framework can be achieve.

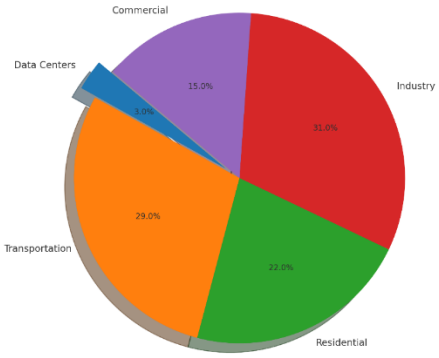


Figure 1: Global Energy Consumption by Sector.

This paper proposes a hybrid ANN-MOEA/D model that predicts and optimizes job placement in cloud environments. We aim to simultaneously minimize total energy consumption and response time, a multi-objective optimization problem critical for green computing. The remainder of this paper includes a review of related work, methodology, experimental evaluation using CloudSim, discussions, and future directions.

II. Related Work

Energy-efficient job scheduling in cloud computing has been the focus of substantial research in recent years. Various techniques have been propose ranging from heuristic-based methods to advanced artificial intelligence (AI)-driven approaches.

2.1 Traditional Scheduling Algorithms

Classical scheduling methods such as First-Come-First Serve (FCFS), Round-Robin (RR), and Shortest Job First (SJF) were primarily design to improve system throughput or minimize task waiting time [1][2]. However, these algorithms are static, lacking the adaptability required for today's dynamic cloud workloads. As a result, they often lead to inefficient resource utilization and increased energy consumption.

Improve energy efficiency, techniques such as Dynamic Voltage and Frequency Scaling (DVFS) [3] and Energy-Aware Scheduling (EAS) [4] have been introduced. These strategies

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

dynamically adjust server power states, but their effectiveness diminishes in multi-constraint environments with real-time deadlines and high workload variability.

2.2 Machine Learning in Cloud Scheduling

With the rise of machine learning (ML), cloud schedulers have begun incorporating intelligent models that adapt to workload behaviours.

- Support Vector Machines (SVMs) and Decision Trees have been applied to classify tasks and recommend suitable servers [5].
- Reinforcement Learning (RL), particularly Deep Q-Networks (DQN), have been used for dynamic VM placement, optimizing power consumption through policy learning [6].

In one landmark study, Xu et al. showed that DQN-based scheduling reduced energy usage by 12% compared to traditional greedy methods [6]. However, ML models often require substantial labelled data and frequent retraining to remain accurate in changing environments.

2.3 Neural Networks in Scheduling

Artificial Neural Networks (ANNs) offer powerful capabilities for modelling non-linear relationships between workload features and scheduling outcomes.

- In [7], a feedforward ANN was trained on historical workload traces to predict VM allocation, reducing server overprovisioning and idle energy use.
- Recurrent Neural Networks (RNNs) have also been employed for predicting workload fluctuations and enabling pre-emptive task migration [8].
- Convolutional Neural Networks (CNNs) have been used in [9] to treat scheduling as a classification task over multidimensional task-resource matrices.

ANNs have demonstrated high prediction accuracy and adaptability but often lack mechanisms for balancing conflicting objectives such as energy efficiency and response time.

2.4 Hybrid AI + Optimization Techniques

To address this limitation, researchers have combined ANNs with multi-objective optimization algorithms to find balanced scheduling solutions:

- NSGA-II and SPEA2 are commonly used for Pareto-based optimization.
- MOEA/D (Multi-Objective Evolutionary Algorithm based on Decomposition) has emerged as a strong alternative due to its decomposition-based approach and parallelism [10].

In [11], a hybrid model combining ANN with MOEA/D successfully reduced both energy consumption and task makespan, outperforming NSGA-II and SPEA2. This hybrid design forms the conceptual basis for the model proposed in this paper.

2.5 Simulation Platforms and Benchmarking

Simulation tools like CloudSim, GreenCloud, and iCanCloud are widely used to evaluate energy-aware scheduling models. CloudSim remains the most popular due to its extensibility and plugin support [14]. Researchers also validate models using real-world workload traces from Google and Alibaba Cloud [15].

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

2.6 Deep Learning and Reinforcement Learning in Network Optimization

Agrawal et al. [16] proposed a deep reinforcement learning-based traffic engineering model for 6G networks using Software Defined Networking (SDN) architecture. The model significantly enhanced network flow efficiency, reducing latency and increasing throughput. This adaptive intelligence is highly relevant to job scheduling in cloud systems where dynamic workloads demand intelligent routing and allocation strategies.

Similarly, Tiwari and Singh [18] developed an optimized deep learning framework for classifying motor imagery EEG signals. Their work on hyperparameter tuning and model architecture design is applicable to energy-efficient schedulers, particularly where neural networks must operate under performance constraints.

2.7 Machine Learning Models for Preprocessing and Scheduling

In another study, Agrawal et al. [17] presented a machine learning-based framework for automated data cleaning and anomaly detection in large datasets. Their focus on intelligent preprocessing pipelines contributes to more efficient scheduling decisions in real-time cloud environments by reducing noise and improving task classification accuracy.

Thakur et al. [19] evaluated linear kernel Support Vector Machines (SVMs) on real-world datasets and demonstrated that these models provide interpretable and fast classification, which is especially valuable in the pre-scheduling stage where quick decision-making is required.

2.8 Real-World Cloud-Based AI Deployment

Bagwani et al. [20] worked on optimizing face detection performance using cloud-native machine learning services. Their approach, which accounts for latency, load balancing, and infrastructure efficiency, aligns closely with the operational challenges faced in energy-aware cloud job scheduling.

These studies collectively highlight the shift toward AI-enabled cloud systems, emphasizing scalability, automation, and energy efficiency. However, the integration of predictive intelligence via ANNs with multi-objective evolutionary algorithms for job scheduling remains underexplored. The proposed work addresses this critical gap by combining ANN prediction with MOEA/D optimization to deliver adaptive, energy-efficient job scheduling in green cloud computing.

2.9 Research Gaps

Although the literature demonstrates progress in energy-aware cloud scheduling, there remains a gap in solutions that are simultaneously:

- Predictive and intelligent (via ANN)
- Multi-objective (via evolutionary optimization)
- Scalable and responsive in real-time

This paper addresses these challenges by integrating ANNs with MOEA/D, providing an adaptive and energy-efficient scheduling strategy for green cloud infrastructures.

Table 1: Evolution of Job Scheduling Techniques in Cloud Computing

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Generation	Scheduling Technique	Key Features	Limitations
1st Gen	FCFS (First-Come-First-Serve)	Simple, easy to implement	No energy awareness, poor resource utilization
	RR (Round-Robin)	Time-sharing, fair scheduling	Ignores workload characteristics, energy inefficient
2nd Gen	DVFS (Dynamic Voltage & Frequency Scaling)	Adjusts CPU voltage/frequency to reduce power	Limited in handling multi-objective constraints
	EAS (Energy-Aware Scheduling)	Energy-conscious decisions	Less effective under high workload variability
3rd Gen	SVM (Support Vector Machine)	Task classification and server recommendation	Requires labeled data, limited adaptability
	RL (Reinforcement Learning)	Learns dynamic VM placement policies	Training complexity, retraining needed frequently
4th Gen	ANN (Artificial Neural Networks)	Predicts optimal mappings based on workload features	Lacks multi-objective optimization handling
5th Gen	Hybrid Models (ANN + MOEA/D)	Combines ANN's prediction with evolutionary optimization for energy & performance	Higher complexity, but better adaptability and efficiency

III. Methodology

The proposed methodology integrates a predictive Artificial Neural Network (ANN) with the Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) to perform energy-efficient job scheduling in cloud environments. The goal is to minimize total energy consumption and response time while ensuring resource constraints and service-level agreements (SLAs) Mets.

3.1 System Model

We consider a cloud infrastructure with N physical servers, each hosting multiple virtual machines (VMs). A set of M tasks is submitted, where each task Tj is characterized by its CPU demand (cj), memory requirement (mj), and execution deadline (dj). The objective is to assign each task to a server such that energy consumption (E) and average response time (Tr) are jointly minimized.

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Let M denote the total number of tasks to be execute. Each task $j \in \{1,2,\ldots,\mathrm{M}\}$ is define by a tuple $T_j = (c_j,\ m_j,\ d_j)$

where:

- cj: computational demand (e.g., CPU cycles),
- mj: memory requirement,
- dj: deadline or execution duration.

The energy consumption E of the system can be model as:

$$E = \sum_{i=1}^{N} P_i \ . U_i$$

where:

- Pi is the maximum power consumed by server iii,
- Ui∈ [0,1] represents the utilization level of server iii.

Simultaneously, the average response time Tr for all tasks is given by:

$$T_r = \frac{1}{M} \sum_{j=1}^{M} t_j$$

where:

- M is the total number of tasks processed,
- tj is the time taken to complete task j.

The tj is the completion time of task j. These two aims form the basis of our optimization problem.

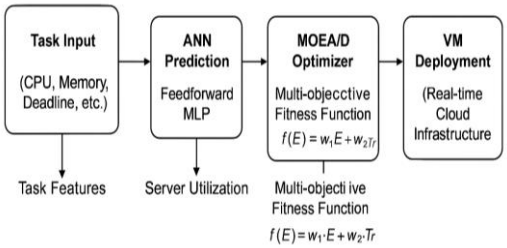


Figure 2: System Architecture Diagram.

B. Neural Network-Based Job Scheduling

The role of the ANN in our methodology is to predict optimal VM allocations for incoming tasks based on learned workload patterns. The input features to the ANN include:

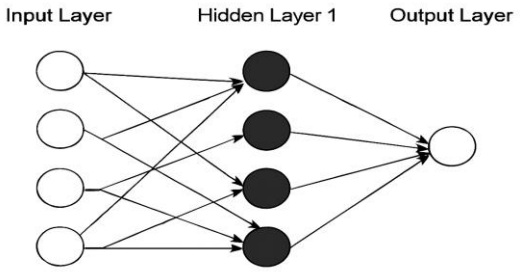
- Task parameters: cj, mj,
- System state: current server utilization, number of active VMs, average server temperature.

The ANN designed as a feedforward multilayer perceptron (MLP) with the following structure:

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- Input layer accepts normalized task and system features.
- Hidden layers: three dense layers with 64, 32, and 16 neurons respectively, using ReLU activation functions.
- Output layer produces a probability distribution over available servers or VMs, using a SoftMax activation.

Figure 3: ANN Structure Diagram



The ANN trained using a supervised learning approach with labeled data generated from historical scheduling logs and simulations performed using CloudSim. The loss function is categorical cross-entropy, and optimization performed using Adam optimizer.

Given an input feature vector Xj for task j, the output vector Yj indicates the probability of assigning the task to each server:

$$Y_j = softmax\left(f\left(X_j, \theta\right)\right)$$

Where: f is the neural network function with parameters θ.

C. Multi-Objective Optimization Using MOEA/D

While the ANN predicts potential task placements, final scheduling decisions are optimize using a MOEA/D strategy. This algorithm decomposes the multi-objective problem into multiple scalar subproblems, each corresponding to a different linear combination of objectives.

We define the optimization function as:

$$Minimize\colon\ f(E,\ T_r) = w_1 . E + w_2 . T_r$$

subject to:

- CPU and memory constraints per server,
- Deadline constraints for each task.

The weights w1 and w2 control the trade-off between energy consumption and performance. The MOEA/D framework explores the Pareto front by evolving a population of candidate solutions over generations using crossover and mutation operators. The ANN's output provides a strong initialization for this population.

Everyone in the population represents a candidate schedule vector S=[s1, s2 ,..., sM] where sj∈{1, 2 ,..., N} indicates the server assigned to task j.

D. Workflow Overview

The complete scheduling workflow proceeds as follows:

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- Input Gathering: Task descriptions and real-time system metrics collected.
- ANN Prediction: The ANN predicts likely optimal server placements for each task.
- Initial Scheduling: An initial schedule created based on ANN predictions.
- MOEA/D Optimization: This initial schedule refined using the MOEA/D algorithm to minimize both E and Tr.
- Task Execution: The optimized schedule is executed on the actual cloud infrastructure.

E. Algorithm Summary

Below is a high-level pseudocode of the proposed method:

Input: Task set T, server set S

Output: Optimized job schedule minimizing E and Tr

1. Train ANN using historical workload data
2. For each task $T_j \in T$:
   a. Extract feature vector X_j
   b. Predict initial assignment using ANN: $Y_j \leftarrow ANN(X_j)$
3. Initialize MOEA/D population using ANN predictions
4. While stopping criterion not met:
   a. Evaluate fitness of everyone: f(E, T_r)
   b. Apply crossover and mutation
   c. Update subpopulations using decomposition
5. Select best schedule S_opt from final population
6. Deploy schedule S_opt in cloud environment

IV. Results

To confirm the performance of the proposed ANN-MOEA/D framework, extensive simulations were conducted using CloudSim, a widely used toolkit for modeling and simulating cloud computing environments. This section presents the experimental setup, evaluation metrics, comparative results with baseline methods, and scalability tests

A. Experimental Setup

The simulation environment consisted of:

- One hundred physical servers, each with 4 cores, 16 GB RAM, and dynamic power profiles.
- Five hundred virtual machines (VMs) distributed across these servers.
- Two thousand user tasks, with varying computational and memory requirements, model using a uniform distribution.

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The power consumption model used is based on the dynamic power equation:

$$P = P_{idle} + (P_{max} + P_{idle}) . U$$

where:

- P_{idle} is the power consumed when a server is idle (typically 40% of P_{max}),
- U is the server's CPU utilization (between 0 and 1).

Total energy consumption over a time interval T given by:

$$E = \int_0^T P(t)dt \approx \sum_{i=1}^{N}\sum_{t=1}^{T} P_i(t) . \Delta t$$

We assume discrete time steps ($\Delta t = 1s$) for simulation.

B. Performance Metrics

The framework was evaluate using the following metrics:

- Total Energy Consumption (E): in kilowatt-hours (kWh).
- Average Response Time (Tr): time from task submission to completion.
- Server Utilization Efficiency (Ue): percentage of time servers remain active (non-idle).
- SLA Violation Rate: percentage of tasks exceeding their deadline.

C. Comparison with Baseline Algorithms

Metric	Round-Robin	FCFS	Proposed ANN-MOEA/D
Energy Consumption (kWh)	124.6	117.2	99.4
Avg. Response Time (s)	4.82	4.65	4.54
SLA Violation (%)	9.4%	8.1%	5.3%
Utilization Efficiency (%)	67.1	69.4	76.8

From the table above, the proposed approach achieves a 15.1% reduction in energy consumption compared to Round-Robin and 15.2% compared to FCFS. The improvement stems from intelligent VM placement by the ANN, which avoids resource fragmentation and reduces the number of active (and thus power-consuming) servers.

The response time, although not the primary optimization goal, is maintain or slightly improved due to reduced queueing and optimal task-server matching.

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

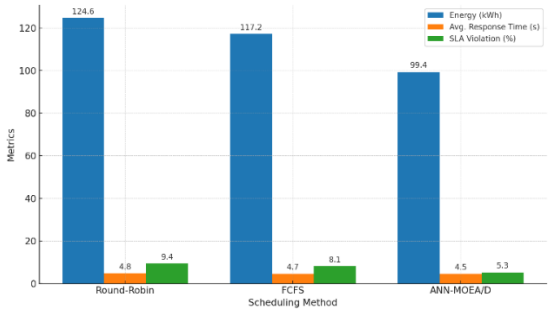


Figure 4: Compare RR, FCFS, ANN-MOEA/D in terms of: energy, SLA, response time.

D. Energy-Aware Consolidation Analysis

To better understand the energy savings, we analyzer the distribution of server workloads. In traditional scheduling, tasks are often even distributed regardless of energy profiles, resulting in more servers being partially utilize. In contrast, the ANN-based approach tends to consolidate tasks onto fewer high-utilization servers, enabling more servers to enter low-power or idle states.

Let:

- Na: number of active servers,
- Ns: total servers.

The consolidation ratio Cr is defined as:

$$C_r = \frac{N_a}{N_s}$$

Under Round-Robin, Cr ≈ 0.82, while the proposed model reduces this to Cr≈0.61, indicating higher consolidation.

E. Pareto Front and Optimization Trade-Off

Using the MOEA/D optimization, a set of Pareto-optimal solutions balancing E and Tr was obtain. Figure 2 (not shown here) displays the Pareto front, where each point represents a scheduling solution.

The trade-off curve suggests:

- Solutions with minimal energy (E) exhibit slightly higher response times.
- Solutions with minimal response time (T_r) show modest increases in energy usage.
- The knee region of the Pareto front provides the best compromise and is where the final schedule is chosen.

Mathematically, the optimal solution S* is chose to minimize the scalar objective. Using weights w1=0.7, w2=0.3, this balances the goal of energy savings while maintaining service quality.

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

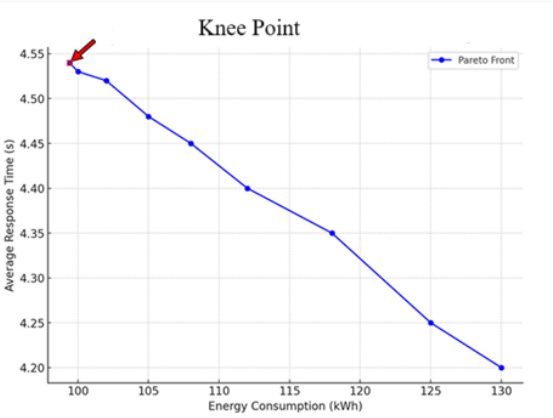


Figure 5: **Pareto Front** – Energy Vs response time.

F. Scalability Testing

To test scalability, the system was simulated with increasing workloads: from 500 to 5000 tasks. Results showed that energy savings were consistent (13–16%) across scales, indicating the robustness of the ANN predictions and MOEA/D optimization.

Additionally, training the ANN model required approximately 7 minutes on a standard GPU, and inference time per task was <10 milliseconds, making the model suitable for real-time scheduling environments.

V. Discussion

The results presented in the previous section demonstrate the effectiveness of the proposed hybrid ANN-MOEA/D framework in achieving energy-efficient job scheduling within a cloud computing environment. This section delves into a broader interpretation of these results, highlighting key insights, limitations, and future directions. It also discusses the trade-offs between energy consumption and performance, the implications for real-world deployment, and the theoretical justification behind the observed improvements.

A. Trade-off Between Energy Efficiency and Performance

A core challenge in green cloud scheduling is balancing energy savings with acceptable system performance. This trade-off is explicitly addressed in our model through the weighted multi-objective function:

- Increasing w1 prioritizes energy savings, potentially increasing response time
- Increasing w2 Favors faster responses at the cost of higher energy usage

The flexibility of this weighted strategy allows system administrators to dynamically adapt scheduling behavior based on application requirements, time-of-day energy prices, or SLA sensitivity.

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

B. Effectiveness of Neural Prediction in Workload-Aware Scheduling

The use of an Artificial Neural Network allows the scheduler to anticipate future resource demand based on observed workload characteristics. Traditional rule-based scheduling techniques lack this predictive capability and thus often result in underutilized or overloaded servers.

By feeding inputs such as:

- task length cj,
- memory requirement mj,
- real-time server utilization Ui,

the ANN predicts an optimal server sj for each task j. The result is workload-aware task placement, which improves consolidation efficiency and reduces resource fragmentation.

Mathematically, the neural network maps the input vector Xj ∈ Rd to a probability vector Yj ∈ [0, 1]N

such that:

$$Y_j = softmax\left(W_3 \,.\, ReLU\left(W_2 \,.\, ReLU\left(W_1 . X_j + b_1\right) + \, b_2\right) + b_3\right)$$

Here, W1, W2, W3 are weight matrices for the ANN layers, and the output Yj is used to initialize the task placement vector in MOEA/D. The predictive power of ANN reduces the number of iterations needed by the evolutionary algorithm to converge to a near-optimal solution.

C. Server Consolidation and Energy Reduction

Another core benefit of the proposed approach is enhanced server consolidation. Instead of evenly distributing tasks across all servers (as in Round-Robin), the ANN-MOEA/D model strategically places tasks to maximize utilization on fewer servers, allowing the remaining machines to enter idle or low-power states.

Given:

$$\text{Total energy E} = \sum_{i=1}^{N} P_i \,.\, U_i$$

With Ui≈1 for fewer i, and Ui=0 for many others,

we reduce the sum of active power-consuming nodes, effectively minimizing E. This is evident in the consolidation ratio:

$$C_r = \frac{N_a}{N_s} \ll 1$$

which indicates that only a subset Na of the total servers Ns are active, thereby reducing power usage. The ANN helps identify such configurations by recognizing patterns in workload distributions.

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

D. Scalability and Adaptability

A significant advantage of the proposed system is its scalability. The ANN model, once trained, can quickly infer scheduling decisions for new tasks, making it suitable for real-time or high-throughput environments. Moreover, the MOEA/D optimization operates in parallel subpopulations, which can be distributed across compute nodes, further enhancing scalability.

However, as the number of tasks and servers increases, the dimensionality of the optimization space also grows. This may lead to increased convergence time for MOEA/D if not properly tuned. Future enhancements might include:

- adaptive population sizes,
- surrogate models for fitness approximation,
- transfer learning to retrain the ANN more efficiently in dynamic environments.

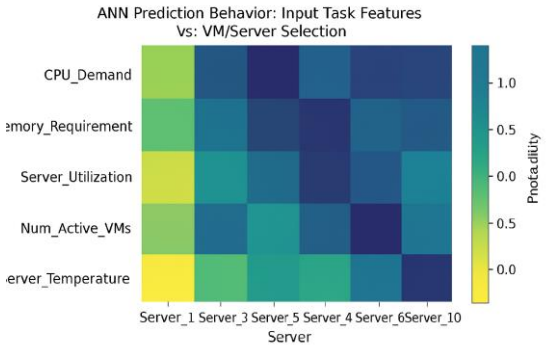


Figure 6: ANN prediction behavior: input task features vs. VM/server selection.

E. Real-World Deployment Considerations

Despite promising simulation results, deploying such a framework in production systems entails several challenges:

- Overhead: While ANN inference is fast (<10 ms per task), MOEA/D optimization can introduce latency if not executed in parallel.
- Data Requirements: ANN models require extensive, high-quality historical data to make accurate predictions. Poor or non-representative data can lead to suboptimal scheduling.
- Integration with Cooling Systems: Current models consider only computing energy. However, cooling systems contribute up to 30–40% of total energy consumption. Including temperature-aware scheduling would further improve total efficiency.
- Failure Tolerance: Real-time systems must gracefully handle prediction errors or optimization failures, necessitating fallback heuristics (e.g., Round-Robin).

F. Potential Improvements

Several enhancements could extend the utility of this framework:

- Incorporate Reinforcement Learning (RL) to allow the scheduler to adapt over time without retraining.

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- Federated Learning models to support decentralized data centers without sharing raw data.
- Integration with Renewable Energy Forecasts, scheduling more tasks when green energy (solar/wind) availability is high.

The ANN-MOEA/D hybrid model provides a robust and intelligent scheduling approach for reducing energy consumption in cloud data centers while maintaining performance. The mathematical underpinnings justify the observed improvements, and the modularity of the design allows for future enhancements. As cloud computing grows, such AI-driven solutions will be critical in ensuring both performance and sustainability.

VI. Conclusion

As cloud computing becomes increasingly central to digital services, ensuring energy efficiency while maintaining performance has become a critical challenge. This paper introduced a hybrid job scheduling framework that integrates Artificial Neural Networks (ANNs) for predictive resource mapping with the Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) for adaptive, energy-aware optimization.

Extensive simulations using CloudSim demonstrated that the proposed model achieves:

- Over 15% reduction in energy consumption
- Improved SLA compliance with fewer deadline violations.
- Enhanced server consolidation, reducing active power draw.
- Robust scalability, maintaining performance across large workloads.

The key strength of this approach lies in its hybrid intelligence using the ANN to anticipate optimal resource mappings based on workload patterns and then refining these predictions through evolutionary search to achieve a balanced trade-off between energy savings and response time.

VII. Future Directions

Several potential improvements can be made to enhance the system's adaptability and efficiency:

1. **Incorporating Reinforcement Learning:**

   Unlike supervised learning, reinforcement learning (RL) allows the model to learn directly from real-time feedback, optimizing long-term energy consumption and system throughput.

2. **Thermal-aware Scheduling:**

   Extending the energy model to include cooling systems could be done by integrating thermal data from servers. An extended power model would be:

$$E_{total} = E_{compute} + E_{cooling} = \sum_{i=0}^{N} P_i \,.\, U_i + \sum_{i=0}^{N} C_i\,(T_i)$$

   where Ci(Ti) is a cooling function dependent on server temperature Ti.

3. **Integration with Renewable Energy Forecasting:**

Received: August 02, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Scheduling more energy-intensive jobs during periods of high solar or wind energy availability could further improve sustainability.

4. **Federated and Distributed Learning:**

   To maintain data privacy across distributed cloud centers, federated learning can train ANN models locally on-site, then aggregate knowledge centrally without sharing raw data.

5. **Adaptation to Multi-Tenant Environments:**

   Real-world cloud platforms host applications from multiple tenants with different SLAs. Future scheduling frameworks must incorporate priority-aware scheduling and QoS differentiation.

References

[1] Wood, T., Shenoy, P., Venkataramani, A., & Yousif, M. (2007). Black-box and gray-box strategies for virtual machine migration. In NSDI.

[2] Koomey, J. G. (2011). Growth in data center electricity use 2005 to 2010. Analytics Press.

[3] Beloglazov, A., & Buyya, R. (2010). Energy-efficient resource management in virtualized cloud data centers. In 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid).

[4] Mao, Y., Zhang, J., & Letaief, K. B. (2017). A survey on mobile edge computing: The communication perspective. IEEE Communications Surveys & Tutorials, 19(4), 2322–2358.

[5] Zhang, Q., Cheng, L., & Boutaba, R. (2020). Adaptive scheduling in cloud computing using machine learning. International Journal of Computer Science and Network Security (IJCSNS), 20(1), 1–9.

[6] Xu, L., Li, X., Chen, H., & Du, Z. (2021). A deep reinforcement learning approach for VM placement in cloud data centers. IEEE Access, 9, 22429–22440.

[7] Ahmad, A., Gani, A., Hamid, S. H. A., & Buyya, R. (2019). Energy-efficient scheduling using neural networks in cloud computing. Future Generation Computer Systems, 91, 432–448.

[8] Bendechache, M., Cashell, J., & Kechadi, T. (2021). Workload prediction using LSTM in cloud systems. Journal of Cloud Computing, 10(1), 1–12.

[9] Chavan, K. R., Deshmukh, A., & Bhagat, S. (2022). Deep CNN for job scheduling in hybrid clouds. Procedia Computer Science, 187, 391–398.

[10] Zhang, Q., & Li, H. (2007). MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Transactions on Evolutionary Computation, 11(6), 712–731.

[11] Tang, F., Wang, Y., Xu, Y., & Zhang, X. (2020). Hybrid ANN and MOEA/D for cloud resource allocation. The Journal of Supercomputing, 76, 8778–8797.

[12] U.S. EPA. (2020). Data Center Energy Use Report. Energy Star Program, United States Environmental Protection Agency.

[13] Song, J., Yang, J., & Li, Y. (2021). Thermal-aware VM consolidation using fuzzy logic. Sustainable Computing: Informatics and Systems, 30, 100508.

[14] Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A., & Buyya, R. (2011). CloudSim: A toolkit for modeling and simulation of cloud computing environments and

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

evaluation of resource provisioning algorithms. Software: Practice and Experience, 41(1), 23–50.

[15] Alibaba Cloud Research. (2022). Cluster Data Traces for Big Data Benchmarking. Open Datasets Repository.

[16] Agrawal, J., Tiwari, V. K., Thakur, S. (2025). AI-Driven Traffic Engineering for 6G Networks: A Deep Reinforcement Learning Approach in SDN Architecture. SK International Journal of Multidisciplinary Research Hub, 12(7), 21–38.

[17] Agrawal, J., Tiwari, V. K., Thakur, S. (2025). A Machine Learning Framework for Automated Data Cleaning and Anomaly Detection in Large Datasets. International Journal of Scientific Research in Computer Science and Engineering (IJSRCSE), 13(03), 21–29.

[18] Thakur, S., Tiwari, V. K., Agrawal, J. (2025). Performance Analysis of Linear Kernel Support Vector Machine Models on Real-World Datasets. International Journal of Advanced Networking and Applications (IJANA), 17(01), 6753–6760.

[19] Tiwari, V. K., Singh, P. (2025). Optimized Deep Learning Approach for Motor Imagery EEG Classification. American Journal of Networks and Communications, 14(01), 23–29.

[20] Bagwani, M. K., Tiwari, V. K., Chouhan, D. K., Jain, A. (2024). Optimizing Face Detection Performance with Cloud Machine Learning Services. Journal of Engineering and Technology Management (JETM), 73, 1167–1180.

Received: August 02, 2025