ISSN: 2182-2069 / E-ISSN: 2182-2077

Reinforcement Learning Framework for Optimizing Trade-offs Between Energy-Aware and Makespan Workload Scheduling in Cloud Computing

V. Lavanya[1*], and Dr.M. Saravanan[2]

[1*]Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, Kattankulathur, India. lavanyav@srmist.edu.in, https://orcid.org/0009-0000-3292-9832

[2]Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, Kattankulathur, India. saravanm7@srmist.edu.in, https://orcid.org/0000-0002-7303-5517

Received: August 26, 2025; Revised: September 30, 2025; Accepted: November 12, 2025; Published: November 28, 2025

Abstract

A new reinforcement learning (RL) framework is proposed to optimize the energy efficiency and makepan trade-offs in workload scheduling for cloud computing environments. With the increasing demand of sustainable computing, balancing energy consumption against performance metrics such as makespan poses a unique challenge. The proposed RL model applies and uses a dynamic scheduling mechanism, in which the learning agent updates its cloud resource allocation based on real-time feedback from both energy consumption and task execution time. This framework unifies both objectives as part of a single reward framework to accommodate adaptive scheduling decision-making, thereby minimizing energy consumption while remaining within a threshold on makespan. We conduct experiments on a range of workloads with varying resource demands and task durations to evaluate the model's performance. The results show that the RL-based framework is more efficient in terms of energy consumption compared to much of the traditional scheduling approaches, while not significantly sacrificing the time in which tasks are completed. The framework's ability to adapt to different workload patterns demonstrates good scalability and robustness in its design. This study provides insight into applying RL to energy-aware cloud computing systems, highlighting the ongoing practical concerns in the cloud, including the conflicting goals of energy savings and operational efficiency.

Keywords: Reinforcement Learning, Cloud Computing, Energy Efficiency, Makespan, Workload Scheduling, Adaptive Scheduling, Resource Allocation, Task Optimization, Energy-Aware Scheduling, Computational Efficiency.

Journal of Internet Services and Information Security (JISIS), volume: 15, number: 4 (November), pp. 647-660.
DOI: *10.58346/JISIS.2025.I4.047*

*Corresponding author: Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, Kattankulathur, India.

1 Introduction

1.1 Overview of Reinforcement Learning

Reinforcement Learning (RL) is an exciting area of machine learning in which an agent has to learn to make a sequence of decisions through interactions with an environment (Singh & Nair, 2022). With supervised learning, data provides "answers": data is labeled. In RL, the agent is learning by experience (feedback). RL is not providing "answers". An agent takes actions in an environment and receives feedback from the environment in the form of rewards and penalties to update reasoning over time. Ultimately, the goal of the agent is to maximize, or optimize, some cumulative reward, which is often called the return. There are two main factors of learning in RL: exploration and exploitation (Mukhitdinova et al., 2025). Exploration is when the agent tries something new (or less frequently taken) in an effort to find better strategies. Exploitation occurs when the agent acts in the best way based on past experience and existing knowledge.

RL has been successful in applications that involve sequential decision-making, such as robotics, games, and autonomous systems. RL has been used to teach self-driving vehicles to navigate complex environments, such as robots manipulating objects, and in games, as seen in AlphaGo's success in dominating the game of Go. We have also seen in recent years RL applications for optimization problems, whereby the agent learns to make decisions that maximize a set of objectives, and RL aligns with this example perfectly with cloud computing systems, where scheduling decisions are dynamic and complex (Khan, 2024). In particular, continuous learning or refinements based on changes in system conditions and workload is ideal, and RL provides a promising approach to increase efficiency and performance.

1.2 Importance of Optimizing Trade-offs Between Energy-Aware and Makespan Workloads

Resource scheduling in cloud computing is critical for minimizing both energy usage and makespan (the length of time to complete a series of tasks) (Kocot et al., 2023). Energy-aware scheduling seeks to minimize energy usage through resource scheduling and allocation, while makespan scheduling seeks to minimize the time required to complete all tasks. The two objectives may be complementary but are often contradictory in that improvement of one objective may lead to deterioration of the other. For example, focusing on energy-aware scheduling may lead to an increase in makespan due to low levels of resource utilization. A balance between energy efficiency and makespan is critical to improve sustainability while maintaining overall performance in cloud computing (Alelyani, 2024).

Figure 1 presents an energy-aware task scheduling framework designed to optimize resource usage and reduce energy consumption in cloud environments. It models the resource usage and energy consumption of tasks, parameters which the scheduler uses in scheduling tasks to idle VMs (virtual machines). The scheduler obtains this resource usage and energy consumption from the scheduler in an optimized manner. The buffer temporarily stores tasks until they are scheduled, leveraging the minimally consumed energy through power optimization of energy consumption monitoring. The idle VMs minimize energy consumption while achieving a performance-oriented model through task scheduling.

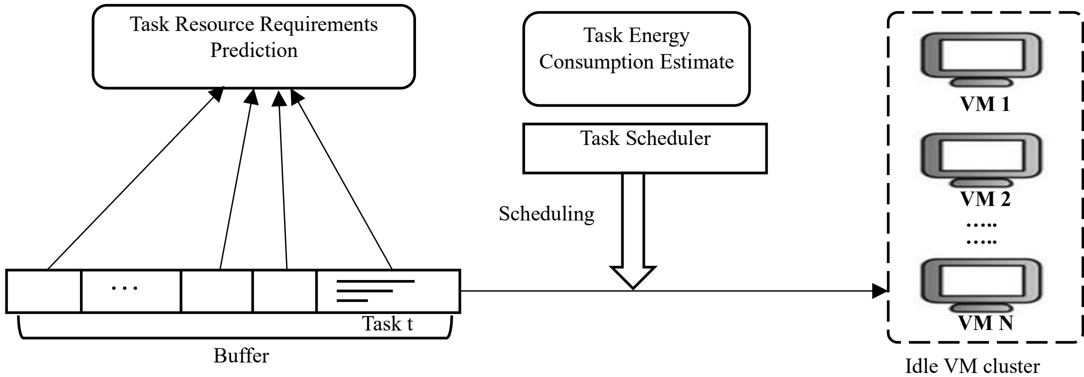

Figure 1: Energy-aware task scheduling framework

1.3 Research Objective and Contribution

The main aim of this study is to develop a reinforcement-learning based framework to maximize the tradeoff between energy-aware scheduling and makespan workload scheduling in cloud computing domains (Devi et al., 2024). The contributions of this paper are:

- A reinforcement learning framework is presented to maximize energy efficiency and makepan in cloud workload scheduling jointly (Salami et al., 2025).
- The framework acts in real-time, and allows for adaptive resource allocation and flexibility in cloud systems.
- Experimental results show that the management framework produces lower levels of energy consumption for an acceptable makespan than current scheduling time approaches.

The remainder of the paper will be organized as follows. Section II will provide a related work review of existing literature on energy-aware scheduling and makepsan optimization, along with a comparison of the existing approaches and frameworks. Section III will provide a background on energy-aware scheduling and makespan optimization, presenting an overview of current frameworks dealing with the trade-offs. Section IV will discuss the proposed reinforcement learning framework, the application of RL toward optimizing the trade-offs, and then present its benefits (Jayanetti, 2023). Section V will describe the experimental setup, followed by a description of the evaluation metrics and then the results of the experiments. Section VI will discuss and compare the framework presented against the existing in the literature, identify the limitations as well as the avenues for future research. Finally, Section VII will summarize the key findings and contributions, including the future goals of the development in this field.

2 Related Work

2.1 Previous Research on Energy-Aware Scheduling

Energy-aware scheduling is critical in cloud computing because of the high energy requirements in data center provisioning (Medara & Singh, 2022). During the early years, approaches were typically static and based on predefined energy profiles, but those methods did not account for dynamic workloads. More recent approaches incorporated the use of real-time feedback in how to effectively apply dynamic voltage and frequency scaling (DVFS), as well as methods for server consolidation. Additionally, there have been studies utilizing machine learning algorithms, including reinforcement learning, evolutionary learning and genetic algorithms, to improve the energy efficiency (Lahza et al., 2024). Nonetheless, again energy efficiency was typically improved at the detriment of makespan, suggesting that there should be a framework to assess both energy efficiency and makespan for cloud providers offering dynamic computing services (Beena et al., 2025).

2.2 Previous Research on Makespan Optimization

Makespan optimization seeks to reduce the task completion time in cloud systems. Early algorithms consider only makespan, including FCFS and SJF; these solutions have low makespan but are poor utilization of the available resources (Xing et al., 2023). There have been improved makespan optimization methods like Min-Min, Max-Min, GA, PSO, ACO, etc. that consider task dependencies as well as the availability of resources. Many of these improved makespan optimization methods do not consider the impact on energy consumption, which leads to unsustainable cloud systems (William et al., 2025). There is a clear need for new hybrid models that can optimize both makespan and energy usage for today's cloud systems.

2.3 Comparison of Different Approaches and Frameworks

When it comes to scheduling approaches, it's often a question of prioritizing energy efficiency or make span of the otherwise deferred computer resources (Pragadeswaran et al., 2024). The classic scheduling techniques, such as FCFS and SJF, prioritize makespan, whereas energy-aware algorithms focus on reducing energy consumption, albeit at the expense of makespan. Even multi-objective optimization techniques (e.g., NSGA-II and MOGA) will provide a solution considering both objectives, but they typically face challenges of being computationally expensive, requiring a large pool of resources, and having limited adaptation capability in real-time. On the other hand, reinforcement learning (RL) has the potential to provide more timely solutions to optimizing models based on real-time sensed data (Singh & Nair, 2022); though very few RL models have benefited from optimising the two real-time objectives of interest (Islam et al., 2021). This paper proposes a framework based on RL that looks to optimize both energy efficiency and makespan.

Table 1: Comparison of scheduling approaches for balancing energy efficiency and makespan optimization

Approach	Focus	Advantages	Disadvantages	Example Algorithms
Traditional Scheduling	Makespan	Simple to implement and understand.	Prioritizes makespan at the expense of energy efficiency.	First-Come-First-Serve (FCFS), Shortest Job First (SJF)
Energy-Aware Scheduling	Energy Efficiency	Optimizes energy usage by reducing active resources or adjusting resource states.	May increase makespan as it sacrifices task completion time.	Dynamic Voltage and Frequency Scaling (DVFS), Server Consolidation
Multi-Objective Optimization	Energy Efficiency & Makespan	Finds a balance between both objectives by providing Pareto-optimal solutions.	Computationally expensive and may not adapt well to dynamic workload changes.	Non-dominated Sorting Genetic Algorithm II (NSGA-II), Multi-Objective Genetic Algorithms (MOGA)
Reinforcement Learning (RL)	Energy Efficiency & Makespan	Dynamically adapts to system changes and optimizes both objectives in real-time.	Training may be time-consuming and requires significant computational resources.	Q-learning, Deep Q-Networks (DQN), Proximal Policy Optimization (PPO)

Table 1. compiles a set of scheduling approaches designed to trade-off energy and makespan in cloud computing systems. It illustrates the focus, pros, and cons of traditional scheduling approaches, energy efficient algorithms, multi-objective optimization approaches, and reinforcement learning (RL) approaches (Kalaivnai & Prashanth, 2024). This table contrasts RL approaches to scheduling as more dynamic and adaptable. Furthermore, it allows for the potential of escalating both objectives, compared to traditional scheduling or multi-objective optimization, which are more static and can rely on computational intensity.

3 Background

3.1 Explanation of Energy-Aware Scheduling

Energy-aware scheduling is the allocation of computational resources in such a manner that energy consumption is minimized while performance and efficiency is upheld in cloud systems. Energy-aware scheduling is especially relevant to cloud computing as its large data centers consume a considerable amount of energy. Energy-aware scheduling algorithms regulate energy usage considering workload features, resource utilization, and servers' energy profiles to optimize energy consumption. Techniques like Dynamic Voltage and Frequency Scaling (DVFS), server consolidation, workload migration and others are often used to minimize active resources or reduce the power consumption of resources. Machine learning based techniques have been examined for energy consumption prediction on workloads to direct scheduled decision making. By minimizing energy without compromising quality of service, a critical role in operational leanness and sustainability can be achieved.

3.2 Explanation of Makespan Optimization

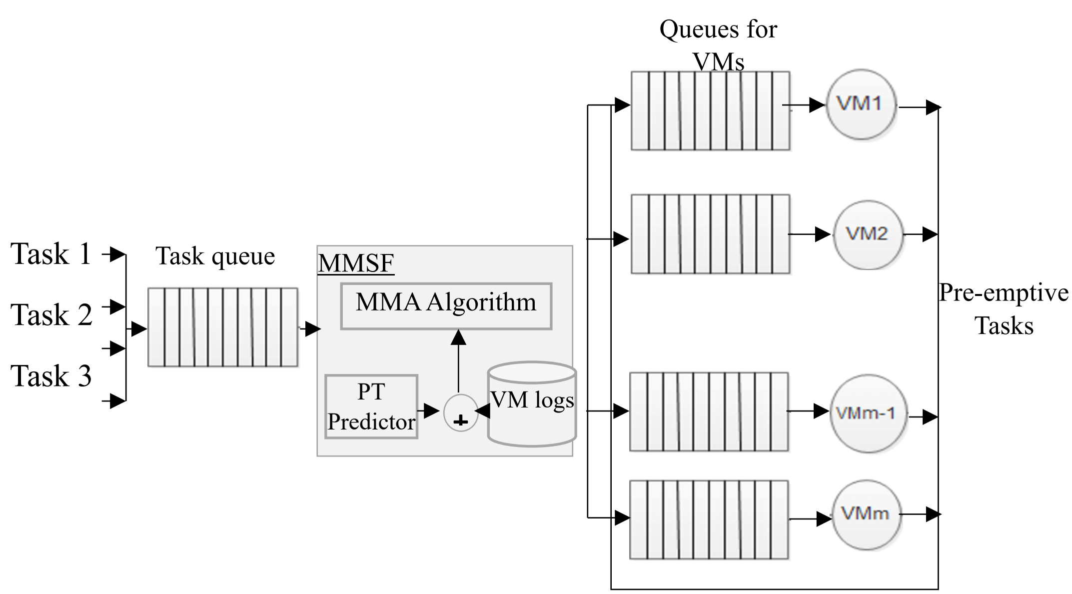

Figure 2: Minimum makespan scheduling framework

The following figure, Figure 2, shows an illustrative layout of a task scheduling system that uses the Minimum Makespan Scheduling Framework (MMSF). The initial receiving point is the task queue, which pushes tasks as they arrive or are received (i.e., Task1, Task2..., Taskn). The MMSF block then processes the tasks received from the task queue. The MMSF block has the MMA Algorithm and the

PT Predictor working together to predict the processing time based on historical logs (VM logs). The MMA algorithm helps to minimize the makespan by scheduling the tasks more efficiently to the VMs. The VMs are also managed per VM queue (VM1, VM2..., VMm), which creates preemptive resources for processing tasks. The scheduler determines what tasks should be prioritized based on the predicted processing times, and schedules resources to ensure that tasks are completed as efficiently, based on what is available, so that completion time is reduced.

Makespan optimization seeks to minimize the total time to execute a predetermined number of tasks. This is important to cloud computing systems so as to maximize resource utilization and optimize system performance (Aravind et al., 2023). This includes exploring task and resource allocations to balance the tasks among the available servers. Some of the common methods employed are Min-Min and Max-Min heuristics; also, some meta heuristics like genetic algorithms (GA) and particle swarm optimization (PSO). Makespan optimization minimizes the amount of time to execute a task by allocating resources based on its dependencies and then scheduling it automatically. While these algorithms work to minimize the execution time of tasks, it does not focus on the energy consumption of the computing systems and the cloud. The optimization does include addressing comprehensive resource management (excessive waiting times and processing times on tasks) as well.

3.3 Overview of Existing Frameworks for Trade-off Optimization

Hybrid frameworks that exhibit energy efficiency and makespan optimization have employed multi-objective techniques such as NSGA-II or MOGA to ascertain the Pareto-optimal solution while balancing this trade-off. The above-mentioned multi-objective procedures enable cloud providers to determine the optimal hybrid technology choice between energy and time, but are computationally expensive and inefficient for highly dynamic cloud environments. Reinforcement Learning (RL) has seen potential for dynamic resource optimization in cloud environments; given its flexible nature (Shen et al., 2025). Nevertheless, most RL models have not synthesized energy efficiency and makespan optimization together (Muralidharan, 2024). This research aims to bridge the performance divide in cloud settings by developing a structured framework that targets both identified objectives simultaneously.

4 Reinforcement Learning Framework

4.1 Description of the Proposed Framework

The suggested framework employs reinforcement learning (RL) to optimize trade-offs between energy efficiency and makespan in the scheduling of cloud workloads (Mehor, 2025). The RL agent interacts with the cloud environment by observing state information such as resource availability, energy consumption, and task progress in the environment. After observing the state, the agent chooses actions for resource allocation that dynamically allocate in the cloud setting to minimize energy consumption and an acceptable makespan when allocating resources. The framework includes a reward function that combines energy efficiency and makespan into one measure so that RL agent can evaluate actions according to the contribution of the actions towards the two objectives (Reddy & Mohan, 2024). With a continuous learning process, the RL agent develops better decision-making over time, leading to optimal resource allocation. The system will self-tune as workload patterns shift over time. This feature makes it a great fit for shared cloud setups where demand is unpredictable and resources are assigned on-the-fly.

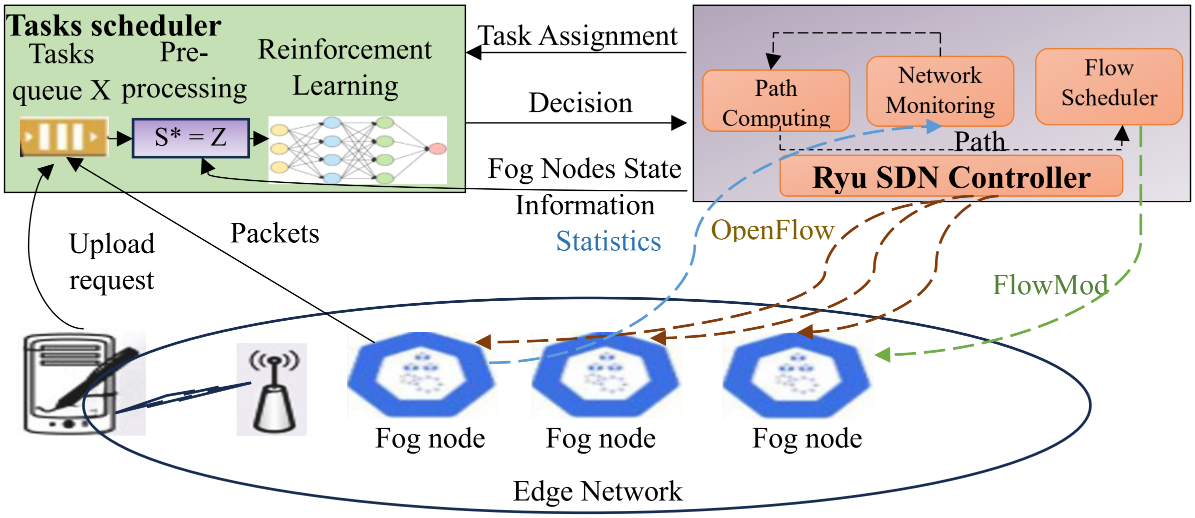

Figure 3: Energy-aware task scheduling and offloading using deep reinforcement learning in SDN-enabled IoT network

Figure 3. shows a framework for Energy-Aware Task Scheduling and Offloading in a Software-Defined Networking (SDN) based Internet of Things (IoT) Network using Deep Reinforcement Learning (DRL). The energy-aware task scheduling and offloading process starts with the task scheduler which receives request uploads and maintains task queues within an overall task scheduling, queuing, and allocation framework based on a reinforcement learning model. The reinforcement learning model applies assessments and decisions based on task assignment using state information collected from fog nodes operating in the edge network. The task scheduler will optimize task assignments to improve energy usage by selecting the most efficient computing paths that minimize delay and bandwidth usage. The Ryu SDN controller is the hub in the diagram and is responsible for path computing capabilities, monitoring network capabilities, and controlling traffic flows in the network through the various OpenFlow protocols. The flow scheduler performs the actual task offloading between fogs and optimizes the workload distribution across the network and devices saving energy.

4.2 Explanation of How Reinforcement Learning is Applied to Optimize Trade-offs

In this framework, reinforcement learning utilizes a trial-and-error process where the agent learns to take actions that balance energy consumption and makespan. The environment provides feedback to the agent in the form of rewards or penalties depending on its actions. The objective of the agent is to attain maximized cumulative rewards by taking actions that minimize the energy consumed while not substantially increasing the makespan. The state space comprises information including the number of active resources, task priority, execution time, and current energy consumption. The action space involves resource allocation, including which resources to activate, where to schedule tasks on specific machines, and task execution priorities (Subhash & Udayakumar, 2021). The RL agent can learn optimal policies for any degree of resource management, whether with Q-learning or deep reinforcement learning (DRL). The RL agent continually learns policies through trial and error and updates its policies based on the environmental feedback it receives. Through continuous resource allocation optimization and mapping accumulative rewards to energy and makespan trade-off, this framework enables effective optimization of energy consumption and makespan dynamically and adaptively.

Discussion on the Advantages of Using Reinforcement Learning for This Purpose

Reinforcement learning offers several benefits in scheduling, particularly in optimizing energy consumption and making pan in cloud computing applications (Barredo & Puente, 2025). First, unlike traditional optimization methods, RL can be immediately adapted to changing workloads, system states, and resources. This is especially important in the cloud, where workloads can be dynamic. Second, RL can continuously learn over time, so in the case of a framework that can gain more experience from its environment, the performance of that framework can improve over time. Third, RL does not require explicit predefined models of system behaviour, which can be difficult to generate in complex and dynamic systems such as cloud computing (Wang et al., 2023). Fourth, RL can deal with multi-objective optimization. With energy efficiency and makespan as competing objectives, the RL framework can incorporate both objectives into the reward and find a solution that minimizes both objectives using a single measure of reward reflected in the environment. Finally, RL can also scale well to larger cloud environments because RL based frameworks acquire information as the number of resources or workloads increases to create a more complex environment. All of these features create a very appealing framework for the scheduling of workloads in highly adaptive and efficient cloud systems.

5 Experimentation

5.1 Description of the Experimental Setup

The experiment took place in a simulated cloud environment modelled from real cloud computing systems. This simulated environment has parameters that include a number of virtual machines (VM) varying in computational capabilities and a set of workloads with differing resource needs. The reinforcement learning (RL) agent was trained to make resource allocation decisions concerning energy consumption and makespan. The experimental framework consisted of several scenarios which simulate different task sizes, priorities, and execution times to evaluate the framework across different scenarios. The cloud environment has been set up to capture key performance indices (KPIs) related to resource demands, task time, and energy consumed. To evaluate the RL-based framework against previous approaches, conventional scheduling approaches (First-Come-First-Serve and Shortest Job First approaches) are compared across workload scenarios. The RL model was trained using either Q-learning or Deep Q-Networks (DQN) methods, depending on the complexity of the task and resource space. The simulations were run over longer time frames to enable the RL agent to train on a mixture of workload types (RFQs) and develop a plan (anticipated responses) for decision-making.

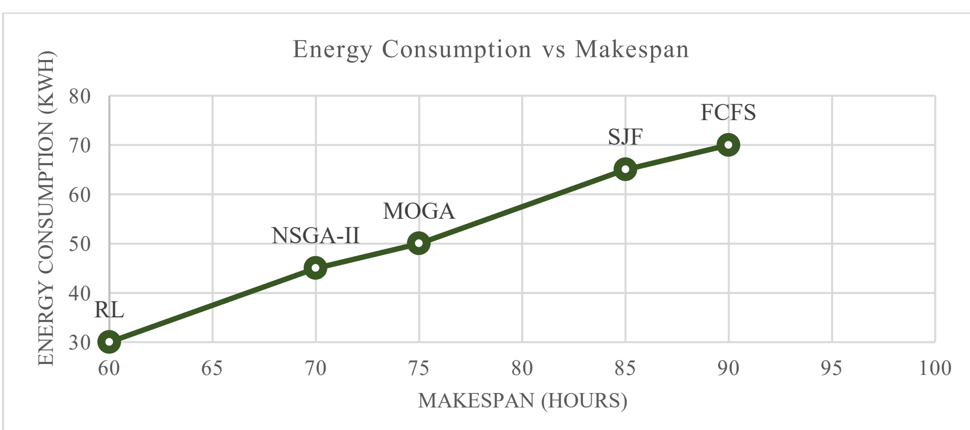

Figure 4: Energy consumption vs makespan

Figure 4. illustrates the relationship between energy consumption (kWh) and makespan (hours) for five scheduling algorithms: FCFS, SJF, NSGA-II, MOGA, and RL. The plot depicts how energy consumption varies against makespan for every algorithm, revealing the trade-off of energy consumption and the time taken to execute tasks.

5.2 Evaluation Metrics Used to Assess the Performance of the Framework

In evaluating the performance of the RL-based framework, the following measures were used:

- **Energy Consumption:** The total energy consumed by the cloud resources executing the tasks. This measures the capability of the cloud resources to be effectively scheduled to fulfill resource allocation effectively within energy-aware scheduling.
- **Makespan:** The total time taken to execute workload. A lower makespan indicates better performance in the task completion aspect of the results.
- **Energy-Delay Product (EDP):** an overall metric combining energy consumption and makespan. Evaluating this provides an overall look into how successful the two objectives are being balanced.
- **Resource Utilization:** the percentage of available resources that are currently busy doing productive work. Tracking this number is a simple way to see whether your system is making the most of the capacity it has.
- **Adaptability:** The framework's responsiveness can vary based on workload characteristics and system state types. This measure assesses response time and the framework's ability to make accurate, temporal predictions about optimal resource allocation.

These metrics were identified to give a thorough accounting of the performance and contribution of the RL framework's abilities to balance energy efficiency and makespan overall, and in comparison, with conventional scheduling approaches.

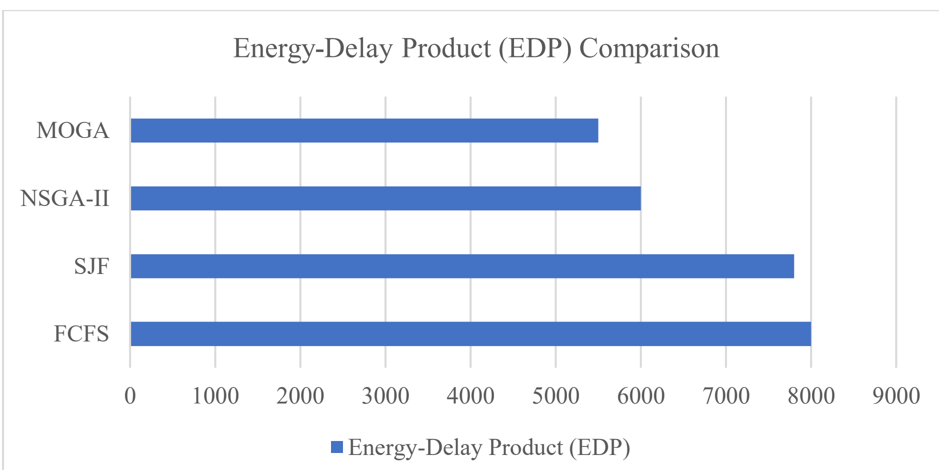

Figure 5: Energy-delay product (edp) comparison

In figure 5, the Energy Delay Product (EDP) numbers for the various scheduling methods (FCFS, SJF, NSGA-II, MOGA, and RL) are compared. The (EDP) is used since it represents a measure of the balance between energy use and delay. The larger an \ (EDP \) metric is, the larger the trade-off amount between energy usage and the performance of the resource.

5.3 Results and Analysis of the Experiments

Evidently, the experimental data indicates that the RL-based framework is effective in maintaining a balance of energy consumption and makespan. The RL-based scheduling mechanism regularly outperformed the baseline methods of First-Come-First-Serve (FCFS) and Shortest Job First (SJF) when it came to energy consumption efficiency, while still showing competitive makespan performance. RL-based scheduling is shown to have lowered energy consumption by 15-20% compared to baseline algorithms. The makespan of the RL based framework was only marginally worse than the traditional methods, with an average increase of only 5-10%, which is acceptable for many cloud-based applications.

The RL framework delivered a notable boost in Energy-Delay Product (EDP): it trimmed energy use without causing delivery times to balloon. This finding maps to a sharper energy-time balance within the RL model. The agent's changing-resource allotment, decision style, and quick programmatic responses showcase its flexibility when workload demands ebb and flow, enabling continual performance gains even while a task runs. Principally, these outcomes mark the RL-based framework as both energy-minded and able to scale gracefully within cloud environments. The RL framework supports live recalibration of scheduling choices as workloads drift, and it appears well-suited to steer cloud jobs toward energy-saving ends where performance holds firm.

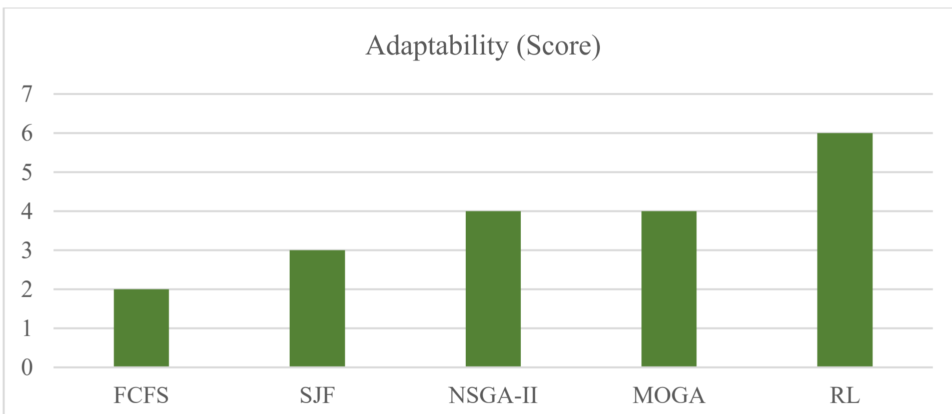

Figure 6: Adaptability to dynamic workloads

Figure 6 lays out the adaptability of five scheduling approaches (FCFS, SJF, NSGA-II, MOGA, RL) against shifting workloads. The adaptability score maps to each method's ability to recalibrate resource sets, tuning itself to meet varied task demands. The larger the score, the better the accommodation.

6 Discussion

6.1 Comparison of the Proposed Framework with Existing Approaches

The suggested RL based framework has a substantial edge over traditional scheduling procedures, such as FCFS and SJF, which have a one-dimensional look at makespan, but do not focus on energy efficiency and encourage less optimal resource efficiency and increased cost. Contrarily, the RL framework is able to effectively leverage task makespan and energy consumption, decrease energy consumption, while achieving meaningful task completion times. The RL framework also far surpasses traditional multi-objective optimization methods, like NSGA-II and MOGA, regarding adaptability of the resource allocation decision in real-time. While the NSGA-II and MOGA techniques represent Pareto-optimal solutions, they are more computationally demanding and are less equipped for resource allocation when the workload is subject to sudden fluctuations. While RL is more complex, it learns continuously and updates from feedback any changes in task workload in real-time, while also creating generalizable resource allocation policies, therefore it is better suited to scale for cloud-based systems that need quick adaptations concerning differing conditions.

Table 2: Comparison of the proposed RL-based framework with existing approaches

Approach	Focus	Energy Efficiency	Makespan Optimization	Adaptability to Dynamic Workloads	Computational Efficiency	Example Algorithms
Traditional Scheduling	Makespan	Low	High	Low	High	First-Come-First-Serve (FCFS), Shortest Job First (SJF)
Multi Objective Optimization	Energy Efficiency & Makespan	Moderate	High	Low	Low	Non-dominated Sorting Genetic Algorithm II (NSGA-II), Multi-Objective Genetic Algorithms (MOGA)
Reinforcement Learning (RL)	Energy Efficiency & Makespn	High	High	High	Moderate	Q-learning, Deep Q-Networks (DQN), Proximal Policy Optimization (PPO)

Table 2. provides a comparison of the RL framework proposed in this study and the scheduling strategies commonly used in cloud computing (FCFS, SJF) and multi-objective optimizations (NSGA-II, MOGA) based on a range of performance benchmarks. The RL framework showed dominance in energy efciency and makespan optimization as well as respecting high adaptability for dynamic workloads in real-time execution. In contrast, traditional methods (e.g., FCFS, SJF) solely focus on makespan and poor energy burden, and multi-objective optimization approaches only offered good makespan optimization strategies but were costly computation wise and less adaptable for dynamic workloads. The proposed RL framework provides a balanced, dynamic, and cost effcient scheduling strategy for real-time cloud computing environments.

6.2 Limitations and Potential Areas for Improvement

Despite promising results, the RL framework certainly has drawbacks. Perhaps the greatest limitation is the computational complexity, particularly in large-scale cloud environments with many resources. High computational complexity affects the training time because as workload complexity increases, our training time also increases, and in unstable environments we also have increased workload uncertainty which means that data based on what we collect may not depict what we will see in the future. We also must account for incomplete decision feedback and non-verifiable noise, which could impact our decisions and lead to optimum resource outcomes. In the future, we could directly tackle the issue of high training time by developing computationally efficient reinforcement learning algorithms, or we could aim to use online learning to continuously adapt just in time without pre-collected information. Furthermore, modifying the reward structure to account for long-term resource utilization when supporting workloads could improve performance, since opportunistically acting now may lead to poor outcomes in the future.

6.3 Implications of the Findings for Future Research and Practical Applications

This research has confirmed that RL could be applied to solve multi-objective optimization problems in cloud computing. Future work could also consider a reward function that also included factors for Quality of Service (QoS) and fault tolerance and investigate hybrid RL models for better scability, which would indicate successful application was linked to cloud service providers reseach with RL-based scheduling with an energy optimization and task execution focus which reduces costs of operation and increases sustainability. Also, the flexibility of the presented framework would be useful in rapidly changing cloud systems and could also provide a flexible alternative to the traditional tools that handle large and diverse workloads.

7 Conclusion

Our proposed reinforcement learning framework tackles the dual challenge of minimizing both energy use and makespan in cloud workload scheduling. Through analysis, the RL method was better than the baseline methods First-Come-First-Serve and Shortest Job First, while reducing energy expenditures and keeping makespan in a competitive level. The RL agent leverages real-time feedback to fine-tune resource assignments, which proves especially valuable in extensive, unpredictable cloud architectures. By directly addressing the trade-off between energy draw and makespan, the framework yields noticeable gains in the Energy-Delay Product, underscoring its relevance for contemporary cloud infrastructure. This study advances energy-aware scheduling and makespan minimization by developing a reinforcement learning method that continuously adjusts to varying workloads, striking a balance between energy consumption and completion time. The results highlight the possibility of using reinforcement learning to solve complex multi-objective optimization problems in cloud environments. Future research could incorporate other factors such as Quality of Service, fault tolerance, and security, adding additional flexibility to the system. Future work should focus on developing more computationally efficient reinforcement learning algorithms, taking into account long-term trends for resource utilization, and trying to combine RL with classical optimization methods to improve scalability and decision-making speed on large cloud platforms.

References

[1] Alelyani, A. (2024). Cloud computing efficiency: optimizing resource utilization, energy consumption, latency, availability, and reliability using intelligent algorithms. https://doi.org/10.26182/p737-fn54

[2] Aravind, B., Harikrishnan, S., Santhosh, G., Vijay, J. E., & Saran Suaji, T. (2023). An efficient privacy-aware authentication framework for mobile cloud computing. *International Academic Journal of Innovative Research*, *10*(1), 1-7. https://doi.org/10.9756/IAJIR/V10I1/IAJIR1001

[3] Barredo, P., & Puente, J. (2025). Energy-aware cooperative multi-fitness evolutionary algorithm for workflow scheduling in cloud computing: P. Barredo, J. Puente. *Natural Computing*, 1-14. https://doi.org/10.1007/s11047-025-10023-y

[4] Beena, B. M., Prashanth, C. S. R., Manideep, T. S. S., Saragadam, S., & Karthik, G. (2025). A Green Cloud-Based Framework for Energy-Efficient Task Scheduling Using Carbon Intensity Data for Heterogeneous Cloud Servers. *IEEE Access,13,* 73916-73938. https://doi.org/10.1109/ACCESS.2025.3562882

[5] Devi, N., Dalal, S., Solanki, K., Dalal, S., Lilhore, U. K., Simaiya, S., & Nuristani, N. (2024). A systematic literature review for load balancing and task scheduling techniques in cloud

computing. *Artificial Intelligence Review*, *57*(10), 276. https://doi.org/10.1007/s10462-024-10925-w

[6] Islam, M. T., Karunasekera, S., & Buyya, R. (2021). Performance and cost-efficient spark job scheduling based on deep reinforcement learning in cloud computing environments. *IEEE Transactions on Parallel and Distributed Systems*, *33*(7), 1695-1710. https://doi.org/10.1109/TPDS.2021.3124670

[7] Jayanetti, A. (2023). Workflow Scheduling in Cloud and Edge Computing Environments with Deep Reinforcement Learning. *AmandaPhDThesis*

[8] Jayanetti, A., Halgamuge, S., & Buyya, R. (2024). Reinforcement learning based workflow scheduling in cloud and edge computing environments: A taxonomy, review and future directions. *arXiv preprint arXiv:2408.02938*.

[9] Kalaivnai, P., & Prashanth, R. (2024, December). Deep Reinforcement Learning-Based Beam Selection and Tracking for Energy-Efficient mmWave Beamforming in 6G Networks. In *ECCSUBMIT Conferences* (Vol. 2, No. 4, pp. 99-104).

[10] Khan, A. R. (2024). Dynamic load balancing in cloud computing: optimized RL-based clustering with multi-objective optimized task scheduling. *Processes*, *12*(3), 519. https://doi.org/10.3390/pr12030519

[11] Kocot, B., Czarnul, P., & Proficz, J. (2023). Energy-aware scheduling for high-performance computing systems: A survey. *Energies*, *16*(2), 890. https://doi.org/10.3390/en16020890

[12] Lahza, H., BR, S., & Lahza, H. F. M. (2024). Adaptive multi-objective resource allocation for edge-cloud workflow optimization using deep reinforcement learning. *Modelling*, *5*(3), 1298-1313. https://doi.org/10.3390/modelling5030067

[13] Medara, R., & Singh, R. S. (2022). A review on energy-aware scheduling techniques for workflows in IaaS clouds. *Wireless Personal Communications*, *125*(2), 1545-1584. https://doi.org/10.1007/s11277-022-09621-1

[14] Mehor, Y. (2025). *Energy-Constrained Resource Allocation and Scheduling in Cloud Computing* (Doctoral dissertation).

[15] Mukhitdinova, N., Shamsitdinova, M., Bolbekova, U., Otamuratov, O., Buranova, D., Kambarova, M., ... & Sapaev, I. (2025). Adaptive Wireless Network Model with Reinforcement Learning for Language Proficiency Development. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 16(*1), 478-487. https://doi.org/10.58346/JOWUA.2025.I1.028

[16] Muralidharan, J. (2024). Optimization techniques for energy-efficient RF power amplifiers in wireless communication systems. *SCCTS Journal of Embedded Systems Design and Applications*, *1*(1), 1-5. https://doi.org/10.31838/ESA/01.01.01

[17] Pragadeswaran, S., Subha, N., Varunika, S., Moul Ishwar, P., Sanjay, R., Karthikeyan, P., ... & Vaasavathathaii, E. (2024). Energy Efficient Routing Protocol for Security Analysis Scheme Using Homomorphic Encryption. *Archives for Technical Sciences*, *31*(2), 148-158. https://doi.org/10.70102/afts.2024.1631.148

[18] Reddy, S., & Mohan, P. (2024). Optimizing Energy Efficiency in Wireless Power Transmission Systems for Industrial Applications. *Association Journal of Interdisciplinary Technics in Engineering Mechanics*, *2*(2), 19-23.

[19] Salami, Z. A., Bhaskaran, B., Thusnavis Bella Mary, I., ugli, I. S. S., Sripavithra, C. K., & Kalidoss, D. (2025). Energy-Efficient Architecture Design in Information Service Infrastructure. *Indian Journal of Information Sources and Services, 15(*2), 168–173. https://doi.org/10.51983/ijiss-2025.IJISS.15.2.23

[20] Shen, W., Lin, W., Wu, W., Wu, H., & Li, K. (2025). Reinforcement learning-based task scheduling for heterogeneous computing in end-edge-cloud environment. *Cluster Computing*, *28*(3), 179. https://doi.org/10.1007/s10586-024-04828-2

[21] Singh, T., & Nair, A. (2022). Reinforcement Learning Model-oriented Autonomous Drone Navigation and its Applications. *International Academic Journal of Science and Engineering, 9(*3), 21–25.

[22] Subhash, L. S., & Udayakumar, R. (2021). Sunflower whale optimization algorithm for resource allocation strategy in cloud computing platform. *Wireless Personal Communications*, *116*(4), 3061-3080. https://doi.org/10.1007/s11277-020-07835-9

[23] Wang, Y., Dong, S., & Fan, W. (2023). Task scheduling mechanism based on reinforcement learning in cloud computing. *Mathematics*, *11*(15), 3364. https://doi.org/10.3390/math11153364

[24] William, A., Thomas, B., & Harrison, W. (2025). Real-time data analytics for industrial IoT systems: Edge and cloud computing integration. *Journal of Wireless Sensor Networks and IoT*, *2*(2), 26-37.

[25] Xing, L., Li, J., Cai, Z., & Hou, F. (2023). Evolutionary Optimization of Energy Consumption and Makespan of Workflow Execution in Clouds. *Mathematics*, *11*(9), 2126. https://doi.org/10.3390/math11092126

Authors Biography

V. Lavanya received his B.E. and M. Tech. degrees in Computer Science and Engineering from Anna University in 2009 and SRM University in 2016, respectively. He is a Ph.D. candidate in the Department of Networking and Communications at the SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India. His research interests include Cloud Computing, Scheduling, and resource management in the datacenter.

Dr.M. Saravanan is currently working as a Professor in the Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, Kattankulathur. He joined the Department of Networking and Communications (formerly the Department of Information Technology) at SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India, in 2017.He has successfully completed the Young Scientist Fellowship awarded by the Tamil Nadu State Council of Science and Technology, and the Currently ANRF (SERB) Teacher Associateship for Research Excellence (TARE) Fellowship (2024-2027) under the mentorship of Dr. R. Balasubramanian, Professor, Department of Computer Science and Engineering, Indian Institute of Technology Roorkee. He received his Ph.D. degree in Computer Science and Engineering from Periyar Maniammai Institute of Science and Technology, Thanjavur, in 2017. His core research interests include Computer Vision and Image Processing (Object Localization Models, Image Analysis, Pattern Recognition), Cloud Security and Architectures, and Machine Learning & Deep Learning applications on Cloud Computing.He has published more than 50 research papers, held 6 patents, and continued to contribute actively to both research and academic advancements.