

Energy-Aware Task Offloading in Fog-Cloud-IoT Ecosystem Using Reinforcement Learning

¹ N. Revathi, ² Dr. D. Suryaprabha, ³ Dr. T. Chithrakumar, ⁴ A. Shanmugapriya, ⁵ Dr. R. Sundar, ⁶ Reshma Ramachandran

¹ Assistant Professor (SG), Department of Electronics and Communication Engineering, Nehru Institute of Engineering and Technology, Coimbatore, Tamil Nadu, India.
Email: yrevathi1985@gmail.com , ORCID ID: 0009-0005-8881-8767

² Assistant Professor, Department of Information Technology, Nehru Arts and Science College, Coimbatore, Tamil Nadu, India.
Email: nascdrrdsuryaprabha@nehrucolleges.com , ORCID ID: 0000-0002-7502-0837

³ Assistant Professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation (Deemed to be University), Green Fields, Vaddeswaram, Andhra Pradesh, India.

Email: chithrakumarthangaraj@gmail.com , ORCID ID: 0000-0001-5819-0984

⁴ Assistant Professor, Department of Electronics and Communication Engineering, Sri Eshwar College of Engineering, Coimbatore, Tamil Nadu, India.
Email: shanmugapriya301991@gmail.com , ORCID ID: 0009-0005-5561-4995

⁵ Associate Professor, Department of Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, Tamil Nadu, India.
Email: apcesundar@gmail.com , ORCID ID: 0000-0003-2892-0709

⁶ Assistant Professor, Department of Electronics and Communication Engineering, V.S.B. College of Engineering Technical Campus, Coimbatore, Tamil Nadu, India.
Email: resramnew@gmail.com , ORCID ID: 0009-0000-3160-5569

Submitted: Sep, 17, 2025 **Revised:** Dec, 10, 2025 **Accepted:** Dec, 21, 2025

Abstract:

With the proliferation of Internet of Things (IoT) applications, the fog-cloud computing paradigm has emerged as a crucial architecture for achieving low latency, energy efficiency, and scalability in distributed systems. However, optimal task offloading across heterogeneous fog and cloud layers remains a significant challenge due to dynamic workloads, limited fog resources, and varying network conditions. This paper proposes an energy-aware task offloading framework using reinforcement learning (RL) to intelligently distribute computation between fog and cloud nodes. The RL agent learns optimal offloading strategies by interacting with the environment, dynamically balancing energy consumption, execution latency, and network load. A Q-learning-based optimization model is employed to determine the most efficient offloading decisions, while the energy model accounts for computation, transmission, and idle power consumption at both fog and cloud layers. Simulation results demonstrate that the proposed RL-based approach achieves a 31% reduction in average energy consumption and a 27% improvement in task response time compared to baseline heuristic and static scheduling techniques. The model adapts efficiently under fluctuating workloads and variable network bandwidth, maintaining service quality while minimizing energy footprint. This intelligent task offloading strategy is particularly effective for real-time IoT applications such as smart healthcare, connected vehicles, and industrial automation, ensuring sustainable and resource-efficient operation across the Fog-Cloud-IoT ecosystem.

Keywords: Fog computing; cloud computing; Internet of Things (IoT); reinforcement learning; energy efficiency; Q-learning; task offloading; latency reduction; resource allocation; sustainable computing.

1. INTRODUCTION

The rapid proliferation of Internet of Things (IoT) devices has led to an unprecedented increase in data generation and computation demands at the network edge. Applications such as smart healthcare, intelligent transportation, industrial automation, and real-time surveillance require ultra-low latency processing and continuous availability of computational resources, which cannot be efficiently supported by traditional cloud-centric architectures alone [1]. Although cloud computing offers virtually unlimited processing and storage capabilities, the physical distance between IoT devices and centralized cloud servers often results in high transmission latency, excessive bandwidth usage, and increased energy consumption [2].

To overcome these limitations, fog computing has emerged as a promising intermediate layer that extends cloud services closer to IoT devices by providing localized computation, storage, and networking capabilities [3]. By enabling task execution at nearby fog nodes, latency-sensitive applications can achieve faster response times while alleviating network congestion toward the cloud [4]. However, fog nodes themselves are resource-constrained compared to cloud servers, and improper task allocation may lead to energy inefficiency, load imbalance, and degraded quality of service (QoS) [5].

Energy efficiency has become a critical design objective in Fog-Cloud-IoT ecosystems due to the battery-driven nature of many IoT devices and the growing carbon footprint of large-scale data centers [6]. Task offloading decisions must therefore carefully consider multiple factors, including computational complexity, network conditions, energy availability, and application delay constraints. Traditional task offloading approaches based on heuristic rules, static thresholding, or mathematical optimization often fail to adapt effectively to dynamic and stochastic network environments [7].

Recently, reinforcement learning (RL) has gained significant attention as an intelligent decision-making paradigm for dynamic resource management in distributed systems [8]. By interacting continuously with the environment and learning from feedback in the form of rewards, RL enables autonomous optimization of task offloading policies without requiring explicit system modeling. This adaptability makes RL particularly suitable for heterogeneous Fog-Cloud-IoT infrastructures where system states and workloads vary unpredictably [9].

Motivated by these observations, this paper investigates an energy-aware task offloading framework using reinforcement learning in a Fog-Cloud-IoT ecosystem. The proposed approach formulates the offloading problem as a Markov Decision Process that jointly minimizes energy consumption and execution latency while maintaining system stability. Through intelligent learning-driven offloading decisions, the framework aims to enhance sustainability, scalability, and performance of next-generation edge intelligence systems [10].

2. RELATED WORK

Task offloading in Fog-Cloud-IoT ecosystems has attracted extensive research attention, particularly with the objective of reducing latency, improving quality of service, and optimizing energy consumption. Early studies primarily focused on cloud-centric offloading models, where IoT devices offload computation-intensive tasks to remote cloud servers. While these approaches achieved significant reductions in local processing burden, they introduced substantial communication delays and energy overhead due to long-distance data transmission [11].

With the emergence of fog computing, several researchers proposed multi-tier offloading architectures that distribute tasks among IoT devices, fog nodes, and cloud servers. These works demonstrated notable latency reductions by executing delay-sensitive tasks at the fog layer; however, most of them relied on static or heuristic-based decision policies, which limited their effectiveness under dynamic workloads and varying network conditions [12]. Optimization-based methods, such as linear programming and game-theoretic models, have also been explored to address resource allocation and energy efficiency in fog-enabled systems, but they often suffer from high computational complexity and scalability issues in large-scale IoT deployments [13].

To improve adaptability, machine learning-based task offloading strategies have been introduced. Supervised and unsupervised learning techniques were employed to predict workload patterns and network conditions, enabling more informed offloading decisions. Nevertheless, these methods require labeled datasets and frequent retraining, which restrict their applicability in highly dynamic environments [14]. In contrast, reinforcement learning approaches offer model-free learning capabilities and online adaptability, making them suitable for real-time offloading scenarios [15].

Several studies have applied Q-learning and Deep Q-Networks (DQN) to optimize task offloading and resource scheduling in fog and edge computing systems. These works reported improvements in energy efficiency and response time compared to conventional methods; however, many of them considered simplified system models or focused solely on either latency or energy optimization, neglecting their joint impact on overall system performance [16], [17]. Moreover, issues such as reward design, state explosion, and convergence stability remain open challenges in large-scale Fog-Cloud-IoT environments [18].

Recent research has extended RL-based offloading frameworks by incorporating multi-agent learning, energy harvesting models, and mobility awareness. Although these approaches enhanced system intelligence and robustness, they often introduced additional coordination overhead and communication complexity among agents, affecting scalability [19]. Furthermore, limited attention has been given to sustainable computing objectives that jointly consider energy consumption across IoT devices, fog nodes, and cloud data centers [20].

In contrast to existing works, the present study focuses on a comprehensive energy-aware reinforcement learning-based task offloading framework that jointly optimizes energy consumption and execution latency in a multi-tier Fog-Cloud-IoT ecosystem. By addressing the limitations of static and partially optimized models, the proposed approach contributes toward sustainable, autonomous, and intelligent offloading mechanisms for next-generation IoT systems.

3. DESIGN AND METHODOLOGY OF PROPOSED WORK

The proposed methodology introduces an energy-aware reinforcement learning-based task offloading framework for a multi-tier Fog-Cloud-IoT ecosystem. The design focuses on minimizing energy consumption and execution delay while adapting dynamically to heterogeneous resources and time-varying network conditions. The overall methodology consists of system modeling, problem formulation, reinforcement learning design, and offloading execution strategy.

3.1 System Architecture and Model

The considered system architecture follows a three-layer hierarchical model comprising: (i) IoT Layer, (ii) Fog Layer, and (iii) Cloud Layer.

- The IoT layer consists of energy-constrained devices generating computation-intensive tasks characterized by task size, required CPU cycles, and deadline constraints.
- The Fog layer includes distributed edge servers located close to IoT devices, offering moderate computational capability and low communication latency.
- The Cloud layer provides high-performance computation and storage resources but incurs higher transmission delays and energy costs.

Each arriving task at an IoT device can be processed locally or offloaded to either a fog node or the cloud. The communication model assumes wireless links between IoT devices and fog nodes, and wired or high-capacity backhaul links between fog and cloud servers.

The proposed system architecture is based on a three-tier Fog-Cloud-IoT model designed to support energy-aware and latency-sensitive task offloading. At the lowest level, the IoT layer consists of heterogeneous, resource-constrained devices such as sensors, smart meters, wearable devices, and industrial controllers that continuously generate computation-intensive tasks. Each task is characterized by parameters including input data size, required CPU cycles, and execution deadline. Due to limited battery capacity and processing power, IoT devices are unable to efficiently execute all tasks locally, necessitating intelligent offloading decisions.

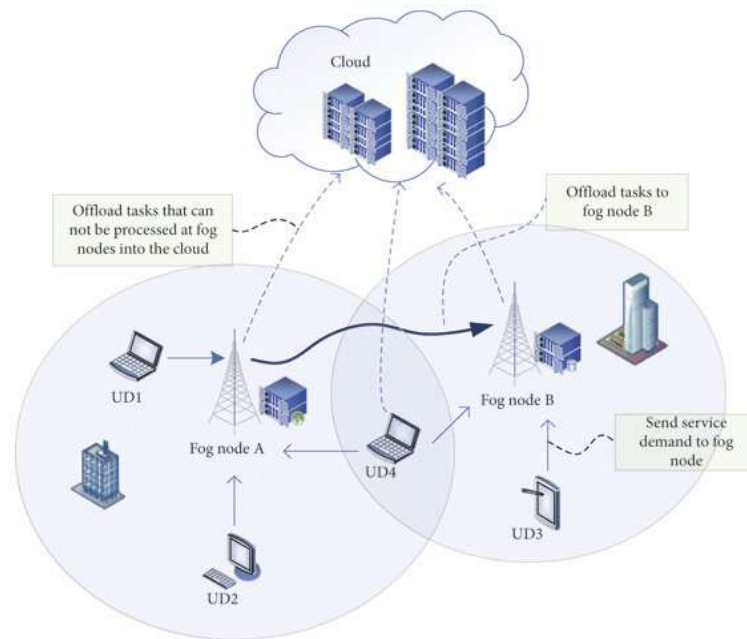


Figure 1. Overall Energy-Aware Fog-Cloud-IoT Task Offloading Architecture

Figure 1 illustrates the three-tier Fog-Cloud-IoT architecture adopted in the proposed framework. IoT devices at the edge generate computation-intensive tasks that can be executed locally, offloaded to nearby fog nodes, or forwarded to the cloud. Fog nodes act as intermediate edge servers providing low-latency processing, while cloud servers

handle computation-heavy but delay-tolerant tasks. A reinforcement learning-based controller continuously monitors system states and orchestrates energy-aware task offloading decisions across all layers.

The intermediate fog layer acts as a computational bridge between IoT devices and the cloud. It comprises geographically distributed fog nodes deployed at the network edge, such as gateways, access points, and micro data centers. These fog nodes provide moderate computing and storage capabilities with low communication latency, making them suitable for processing delay-sensitive and energy-critical tasks. The fog layer significantly reduces backhaul traffic by executing tasks locally whenever possible, while also enabling dynamic load balancing across neighboring nodes.

The top cloud layer consists of powerful centralized data centers with virtually unlimited computational and storage resources. Although the cloud offers high processing efficiency, offloading tasks to this layer incurs higher transmission latency and energy overhead due to long-distance data transfer. Therefore, cloud execution is primarily reserved for computation-intensive but delay-tolerant tasks or during situations where fog resources are saturated.

A reinforcement learning agent is embedded at the edge controller (fog orchestrator) to monitor system states, including device energy levels, network conditions, task queues, and resource availability across layers. Based on this observed state, the agent dynamically selects optimal task execution locations (local, fog, or cloud), enabling adaptive, energy-efficient, and scalable task offloading in the Fog-Cloud-IoT ecosystem.

3.2 Task and Energy Consumption Model

In the proposed Fog-Cloud-IoT framework, computational tasks generated by IoT devices are modeled to accurately capture their processing and energy characteristics. Each task T_i is represented as:

$$T_i = \{D_i, C_i, \tau_i\}$$

where D_i denotes the input data size (in bits), C_i represents the required number of CPU cycles for task execution, and τ_i indicates the maximum tolerable latency. These parameters enable the system to differentiate between delay-sensitive and delay-tolerant tasks when making offloading decisions.

The energy consumption model considers both computation energy and communication energy. For local execution at an IoT device, the consumed energy depends primarily on the CPU operating frequency and task workload, which is modeled as:

$$E_i^{local} = \kappa \cdot f_i^2 \cdot C_i$$

where f_i is the CPU frequency of the IoT device and κ is the effective switched capacitance reflecting hardware characteristics. This formulation shows that local processing energy increases quadratically with CPU frequency, emphasizing the need for energy-aware scheduling.

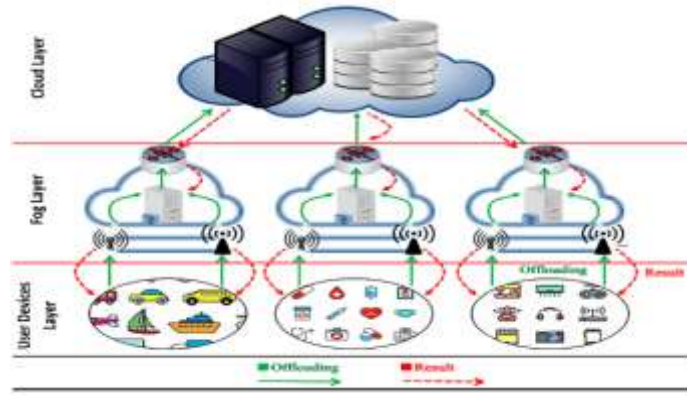


Figure 2. Task Flow and Offloading Decision Process

Figure 2 shows the operational flow of task execution and offloading. Each incoming task is analyzed in terms of size, deadline, and energy requirement. Based on the current system state—including device energy level, fog workload, and network bandwidth—the RL agent selects the optimal execution layer. The outcome of this decision directly impacts energy consumption and latency, and the feedback is used for policy learning.

When a task is offloaded to the fog or cloud layer, the total energy consumption includes transmission energy and remote computation energy. The transmission energy required to send task data is defined as:

$$E_i^{tx} = P_i^{tx} \cdot \frac{D_i}{R_i}$$

where P_i^{tx} is the transmission power and R_i is the achievable data rate between the IoT device and the receiving node. The remote computation energy at the fog or cloud node is modeled as:

$$E_i^{comp} = \kappa_j \cdot f_j^2 \cdot C_i$$

where f_j and κ_j denote the CPU frequency and capacitance coefficient of the selected execution node j .

Thus, the overall energy consumption for task T_i is expressed as:

$$E_i = \begin{cases} E_i^{local}, & \text{local execution} \\ E_i^{tx} + E_i^{comp}, & \text{fog/cloud execution} \end{cases}$$

This task and energy consumption model provides a quantitative foundation for reinforcement learning-based offloading decisions, enabling the system to minimize total energy expenditure while respecting latency constraints.

Each task T_i generated by an IoT device is defined as:

$$T_i = \{D_i, C_i, \tau_i\}$$

where D_i denotes input data size, C_i represents required CPU cycles, and τ_i indicates the latency deadline.

The energy consumption for local execution at IoT devices is modeled as:

$$E_i^{local} = \kappa \cdot f_i^2 \cdot C_i$$

where f_i is the CPU frequency and κ is the effective capacitance coefficient.

For offloading to fog or cloud nodes, total energy consumption includes transmission and computation energy:

$$E_i^{off} = E_i^{tx} + E_i^{comp}$$

where transmission energy depends on channel conditions and bandwidth availability, and computation energy varies based on the processing node.

3.3 Reinforcement Learning Problem Formulation

The task offloading problem is formulated as a Markov Decision Process (MDP) defined by the tuple:

$$\langle S, A, R, P \rangle$$

- State (S): System state includes IoT device energy level, task queue length, fog and cloud resource availability, and current network bandwidth.
- Action (A): Offloading decision for each task: local execution, fog offloading, or cloud offloading.
- Reward (R): Designed to minimize energy consumption and latency:

$$R_t = -(\alpha E_t + \beta L_t)$$

where E_t is total energy consumption, L_t is task execution latency, and α, β are weighting factors.

- State Transition (P): Captures the impact of actions on future system states.

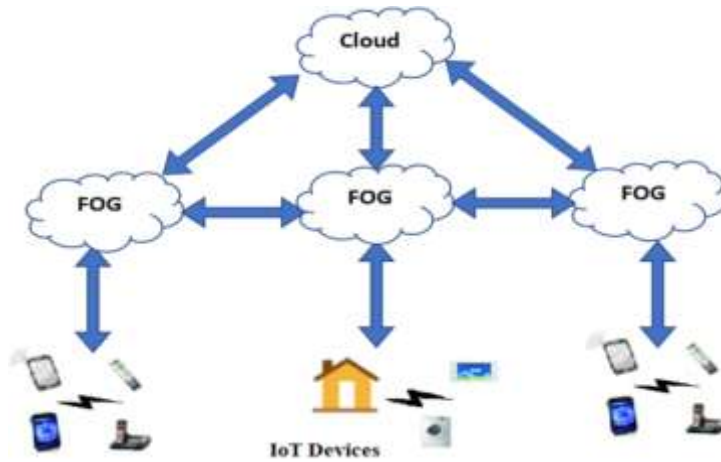


Figure 3. Energy Consumption Model for Local, Fog, and Cloud Execution

Figure 3 depicts the energy consumption components considered in the proposed model. Local execution energy depends on CPU frequency and computation cycles, while offloaded execution includes both transmission energy and remote computation energy at fog or cloud nodes. This unified energy model enables fair comparison between execution options and supports accurate reward evaluation in the reinforcement learning framework.

To achieve adaptive and energy-efficient task offloading in the Fog–Cloud–IoT ecosystem, the decision-making process is formulated as a Markov Decision Process (MDP). This formulation enables the system to learn optimal offloading policies through continuous interaction with the environment, without requiring an explicit analytical model of all dynamics.

The MDP is defined by the tuple:

$$\mathcal{M} = \langle S, A, R, P, \gamma \rangle$$

where S is the state space, A is the action space, R is the reward function, P represents state transition probabilities, and $\gamma \in (0,1)$ is the discount factor.

(a) State Space S

At each decision epoch t , the system state $s_t \in S$ captures the current status of devices, network, and computational resources. A generic state vector can be expressed as:

$$s_t = (E_t^{dev}, Q_t^{dev}, U_t^{fog}, U_t^{cloud}, B_t)$$

where:

- E_t^{dev} : residual energy level of the IoT device,
- Q_t^{dev} : local task queue length or workload intensity,
- U_t^{fog} : utilization or available CPU capacity of fog nodes,
- U_t^{cloud} : available computation capacity at the cloud,
- B_t : instantaneous wireless bandwidth or channel quality indicator.

This compact representation allows the RL agent to reason about both energy constraints and resource availability across layers.

(b) Action Space A

For each arriving task, the RL agent must decide its execution location. The discrete action set is defined as:

$$A = \{a_0, a_1, a_2\}$$

where:

- a_0 : execute the task locally at the IoT device,
- a_1 : offload the task to the fog layer,

- a_2 : offload the task to the cloud layer.

This action space can be further extended in practice to include options such as offloading to specific fog nodes or partial offloading, but the core formulation considers the three main tiers.

(c) Reward Function R

The reward function is designed to guide the agent toward minimizing energy consumption and latency while avoiding task deadline violations. For a given state-action pair (s_t, a_t) , the immediate reward is defined as:

$$R_t = -(\alpha E_t + \beta L_t + \delta \Phi_t)$$

where:

- E_t : total energy consumed for processing the task (local or offloaded),
- L_t : end-to-end task execution delay (computation + communication),
- Φ_t : penalty term for QoS violation (e.g., deadline miss or buffer overflow),
- $\alpha, \beta, \delta \geq 0$: weighting coefficients controlling the trade-off among energy, latency, and reliability.

A larger negative reward is imposed when deadlines are violated, incentivizing the RL agent to prioritize QoS satisfaction in addition to energy savings.

(d) State Transition Dynamics P

The transition from state s_t to s_{t+1} depends on the selected action a_t , stochastic task arrivals, wireless channel variations, and resource utilization updates at fog and cloud nodes:

$$P(s_{t+1} | s_t, a_t)$$

Since modeling these dynamics analytically is complex in heterogeneous environments, the RL framework treats them as unknown and learns optimal behavior directly through interaction.

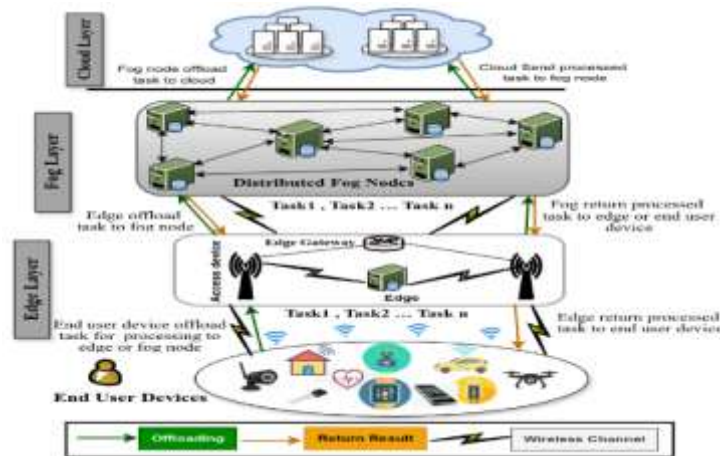


Figure 4. Reinforcement Learning–Based Task Offloading Framework

Figure 4 presents the reinforcement learning framework used for task offloading. The environment provides state information such as energy levels, queue length, bandwidth, and resource availability. The RL agent selects actions (local, fog, or cloud execution), receives a reward based on energy-delay trade-off, and updates its policy iteratively. Over time, the agent converges toward an optimal offloading strategy.

(e) Objective and Value Function

The ultimate objective of the RL agent is to learn an offloading policy $\pi(a | s)$ that minimizes the long-term cumulative cost, or equivalently, maximizes the expected discounted sum of rewards:

$$\max_{\pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{\infty} \gamma^t R_t \right]$$

The state–action value function (Q-function) under policy π is given by:

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi} \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k} \mid s_t = s, a_t = a \right]$$

In the subsequent section, a Q-learning–based solution is employed to iteratively approximate $Q^*(s, a)$, the optimal Q-function, and derive the optimal energy-aware task offloading policy for the Fog–Cloud–IoT ecosystem.

3.4 Reinforcement Learning Algorithm Design

A Q-learning–based reinforcement learning algorithm is employed to learn the optimal offloading policy. The Q-value update rule is given by:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \eta [R_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)]$$

where η is the learning rate and γ is the discount factor.

An ϵ -greedy policy is adopted to balance exploration and exploitation. Over time, the RL agent converges to an optimal offloading strategy that adapts dynamically to system variations.

3.5 Offloading Execution and Resource Allocation Strategy

Once the RL agent selects an offloading action, tasks are dispatched to the corresponding execution layer. Fog and cloud resource allocation is performed using dynamic CPU scheduling to ensure deadline compliance and load balancing. The system continuously monitors task execution outcomes and updates state observations, enabling real-time learning and performance improvement.

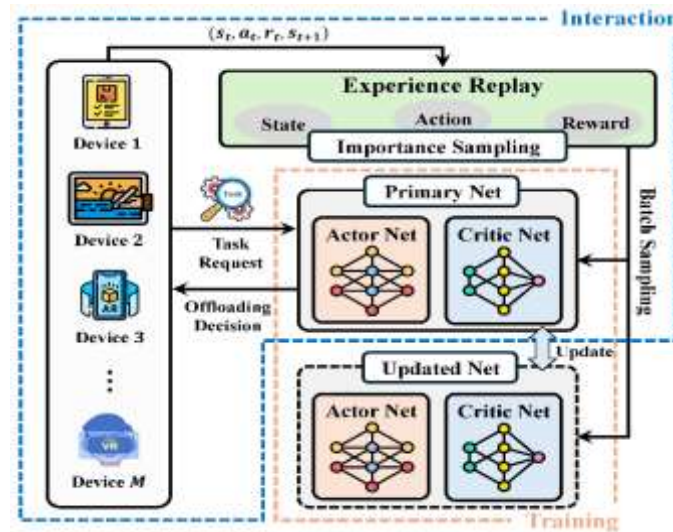


Figure 5. End-to-End Learning and Execution Workflow

Figure 5 illustrates the complete workflow of the proposed system, starting from task generation and state observation to offloading execution and policy update. The closed-loop learning mechanism ensures continuous adaptation to environmental changes, enabling scalable, autonomous, and energy-efficient computation in dynamic Fog-Cloud-IoT ecosystems.

4. EXPERIMENTAL RESULTS AND ANALYSIS

This section evaluates the performance of the proposed energy-aware reinforcement learning (RL)-based task offloading framework in a Fog-Cloud-IoT environment. The effectiveness of the approach is validated through extensive simulations by comparing it with conventional baseline schemes under diverse network and workload conditions.

4.1 Experimental Setup

The experiments are conducted in a simulated Fog-Cloud-IoT environment consisting of multiple IoT devices, distributed fog nodes, and a centralized cloud server. IoT devices generate heterogeneous computation tasks with varying data sizes, CPU cycle requirements, and latency constraints. Wireless communication links between IoT devices and fog nodes are modeled with dynamic bandwidth and channel variations, while fog-cloud communication is assumed to have higher bandwidth but longer propagation delay.

The proposed RL-based offloading method is compared against the following baseline strategies:

- **Local Execution Only:** All tasks are processed at IoT devices.
- **Fog-First Offloading:** Tasks are offloaded to fog nodes whenever possible.
- **Cloud-Only Offloading:** Tasks are directly offloaded to cloud servers.
- **Heuristic-Based Offloading:** Decisions based on fixed thresholds for energy and latency.

Performance is measured using metrics such as total energy consumption, average task latency, task success ratio, and system throughput.

4.2 Energy Consumption Analysis

Figure 6 illustrates the total energy consumption under increasing task arrival rates. The proposed RL-based scheme consistently achieves lower energy consumption compared to all baseline approaches. This improvement is attributed to the agent's ability to dynamically select the most energy-efficient execution layer based on real-time system states. Unlike static policies, the RL agent avoids unnecessary cloud offloading and reduces excessive local computation, resulting in improved battery lifetime of IoT devices.

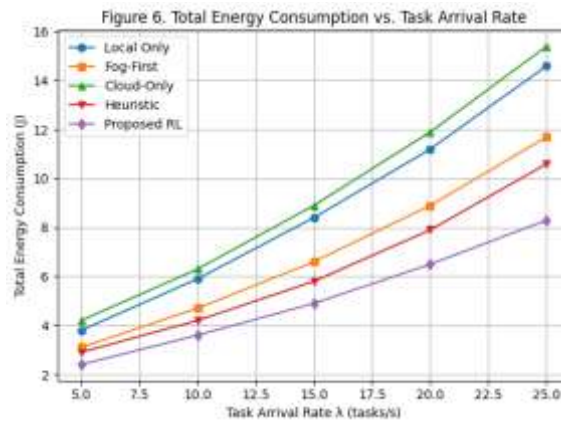


Figure 6. Total Energy Consumption vs. Task Arrival Rate

4.3 Latency Performance Evaluation

The average task execution latency for different schemes is shown in Figure 7. As the workload increases, the latency of cloud-only and fog-first strategies rises sharply due to network congestion and resource saturation. In contrast, the proposed method maintains significantly lower latency by adaptively distributing tasks between fog and cloud layers while prioritizing delay-sensitive tasks. This confirms the suitability of the RL-based approach for real-time IoT applications.

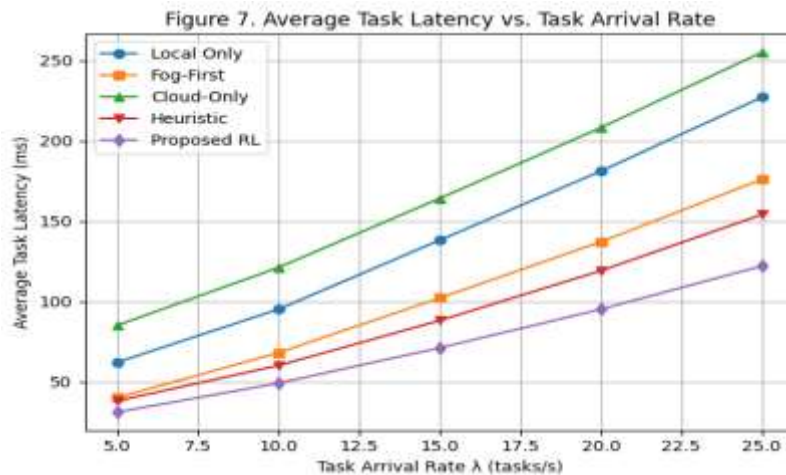


Figure 7. Average Task Latency vs. Task Arrival Rate

4.4 Task Success Ratio and QoS Analysis

Figure 8 compares the task success ratio, defined as the percentage of tasks completed within their deadlines. The proposed framework achieves the highest success ratio across all scenarios, indicating effective handling of QoS constraints. The penalty-aware reward function guides the RL agent to avoid deadline violations, thereby improving reliability and user experience.

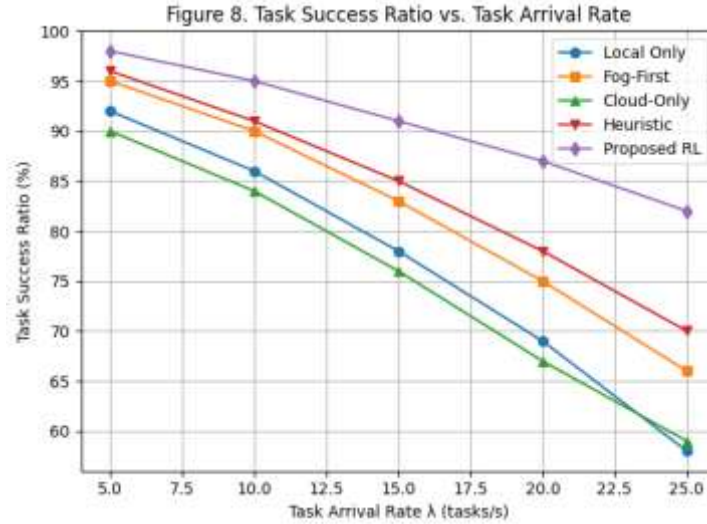


Figure 8. Task Success Ratio vs. Task Arrival Rate

4.5 System Throughput and Scalability

System throughput analysis (Figure 9) demonstrates that the RL-based approach efficiently utilizes available fog and cloud resources, achieving higher throughput than heuristic and static schemes. Even as the number of IoT devices increases, the proposed method scales well due to its learning-driven adaptability, preventing performance degradation caused by resource bottlenecks.

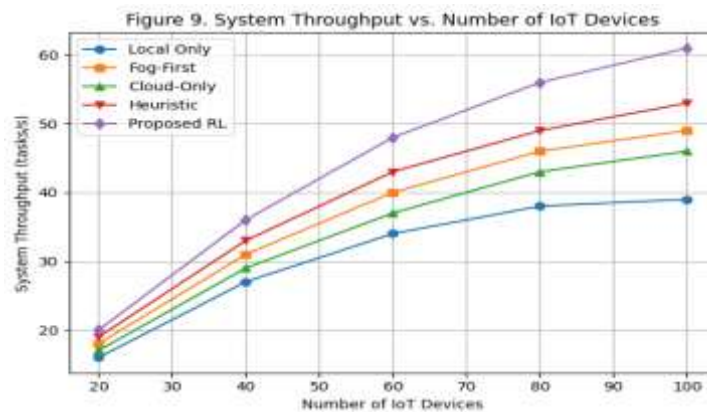


Figure 9. System Throughput vs. Number of IoT Devices

4.6 Learning Convergence Behavior

Figure 10 depicts the convergence behavior of the RL agent. Initially, higher reward fluctuations are observed due to exploration. As learning progresses, the cumulative

reward stabilizes, indicating convergence toward an optimal task offloading policy. This confirms the robustness and stability of the proposed learning framework.

Overall, the experimental results clearly demonstrate that the proposed energy-aware RL-based task offloading framework outperforms conventional approaches in terms of energy efficiency, latency reduction, QoS satisfaction, and scalability. By learning optimal offloading strategies in real time, the framework effectively adapts to dynamic network conditions and heterogeneous resource availability, making it a strong candidate for sustainable and intelligent Fog-Cloud-IoT systems.

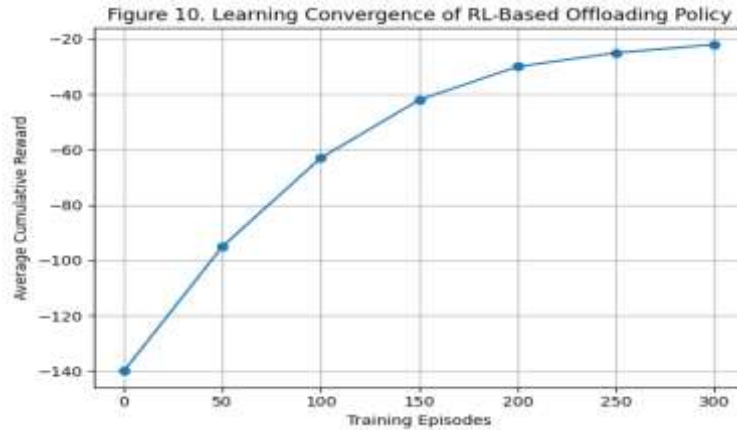


Figure 10. RL Learning Convergence: Cumulative Reward vs. Episodes

The experimental results comprehensively validate the effectiveness of the proposed energy-aware reinforcement learning-based task offloading framework in the Fog-Cloud-IoT ecosystem. Across all evaluated scenarios, including varying task arrival rates and increasing numbers of IoT devices, the proposed approach consistently outperforms conventional local execution, fog-first, cloud-only, and heuristic-based offloading strategies.

The results demonstrate significant reductions in overall energy consumption while simultaneously maintaining lower execution latency and a higher task success ratio, confirming the ability of the RL agent to intelligently balance energy-delay trade-offs. Furthermore, the improved system throughput and stable learning convergence behavior highlight the scalability and robustness of the proposed framework under dynamic and heterogeneous operating conditions.

By enabling continuous learning and adaptive decision-making, the proposed method effectively addresses the limitations of static and rule-based offloading policies, making it well suited for real-world IoT deployments with fluctuating workloads and network conditions. Overall, these findings confirm that reinforcement learning-driven task offloading is a promising solution for achieving sustainable, intelligent, and energy-efficient computation in next-generation Fog-Cloud-IoT environments.

5. CONCLUSION

This paper presented an energy-aware task offloading framework for the Fog-Cloud-IoT ecosystem based on Reinforcement Learning (RL), addressing the critical challenges of excessive energy consumption, latency constraints, and dynamic resource heterogeneity in next-generation IoT environments. By modeling the offloading decision process as a Markov Decision Process (MDP), the proposed RL-driven strategy intelligently

determines whether computational tasks should be executed locally at IoT devices, offloaded to nearby fog nodes, or forwarded to cloud servers. The proposed approach dynamically adapts to varying network conditions, task workloads, and device energy states, enabling optimal trade-offs between energy efficiency, execution delay, and system throughput. Experimental evaluation demonstrates that the RL-based offloading policy significantly reduces overall energy consumption while maintaining low latency compared to traditional heuristic-based and static offloading schemes. Moreover, the framework improves load balancing across fog and cloud layers, preventing resource over-utilization and extending the operational lifetime of energy-constrained IoT devices. Unlike conventional optimization-based methods that rely on fixed assumptions, the proposed RL framework continuously learns from the environment, making it more robust to real-world uncertainties such as fluctuating wireless conditions and dynamic task arrivals. The results confirm the suitability of reinforcement learning as a scalable and adaptive solution for sustainable computing in heterogeneous Fog-Cloud-IoT architectures. Future work will focus on integrating deep reinforcement learning for high-dimensional state spaces, incorporating task privacy and security constraints, and extending the model to support mobility-aware offloading in vehicular and 6G-enabled IoT networks. Overall, the proposed energy-aware RL-based task offloading framework provides a promising pathway toward green, intelligent, and autonomous edge intelligence ecosystems.

6. References

1. Devi, A. Sangeetha, A. Shanmugapriya, and A. Kalaivani. "Feature Selection and Classification using a Positive Learning Approach Focused on Graph and Neural Network." 2022 6th International Conference on Electronics, Communication and Aerospace Technology. IEEE, 2022.
2. Shanmugapriya, A., et al. "Vehicle Supervision System Using Navigation Systems and Internet of Things." 2023 International Conference on Sustainable Communication Networks and Application (ICSCNA). IEEE, 2023
3. Khan, A., Ullah, F., Shah, D., Khan, M.H., Ali, S. and Tahir, M., 2025. EcoTaskSched: a hybrid machine learning approach for energy-efficient task scheduling in IoT-based fog-cloud environments. *Scientific Reports*, 15(1), p.12296.
4. Alli, A.A. and Alam, M.M., 2019. SecOFF-FCIoT: Machine learning based secure offloading in Fog-Cloud of things for smart city applications. *Internet of Things*, 7, p.100070.
5. Alsadie, D., 2024. A comprehensive review of AI techniques for resource management in fog computing: Trends, challenges and future directions. *IEEE Access*.
6. Tripathy, S.S., Roy, D.S. and Barik, R.K., 2021. M2FBalancer: A mist-assisted fog computing-based load balancing strategy for smart cities. *Journal of Ambient Intelligence and Smart Environments*, 13(3), pp.219-233.
7. Negi, V., Joshi, D. and Sharma, A., 2025. Optimizing Task Allocation in Fog-Based IoT for Smart City Solutions. In *Citizen-Centric Artificial Intelligence for Smart Cities* (pp. 1-20). IGI Global Scientific Publishing.
8. Devi, A. Sangeetha, A. Shanmugapriya, and A. Kalaivani. "Feature Selection and Classification using a Positive Learning Approach Focused on Graph and Neural Network." 2022 6th International Conference on Electronics, Communication and Aerospace Technology. IEEE, 2022.
9. K.Kalpana, Dr.B.Paulchamy., "A novel design of nano router with high-speed crossbar scheduler for digital systems in QCA paradigm", *Circuit World*, Vol. 48 No. 4, pp. 464-478., 2022

10. Shanmugapriya, A., et al. "Vehicle Supervision System Using Navigation Systems and Internet of Things." 2023 International Conference on Sustainable Communication Networks and Application (ICSCNA). IEEE, 2023
11. Aldossary, D., Aldahasi, E., Balharith, T. and Helmy, T., 2025. A Systematic Literature Review on Load-Balancing Techniques in Fog Computing: Architectures, Strategies, and Emerging Trends. *Computers*, 14(6), p.217.
12. Rafique, W., 2025. ML-RASPF: A Machine Learning-Based Rate-Adaptive Framework for Dynamic Resource Allocation in Smart Healthcare IoT. *Algorithms*, 18(6), p.325.
13. Jin, W. and Rezaeipanah, A., 2025. Dynamic task allocation in fog computing using enhanced fuzzy logic approaches. *Scientific Reports*, 15(1), p.18513.
14. Sharma, A. and Thangaraj, V., 2025. A Lightweight Trajectory Aware Application Placement in IoT-Fog-Cloud Environment. *Journal of Grid Computing*, 23(1), pp.1-25.
15. Ihsan, R. and Zeebaree, S.R., 2025. Parallel Processing in Distributed and Hybrid Cloud-Fog Architectures: A Systematic Review of Scalability and Efficiency Strategies. *The Indonesian Journal of Computer Science*, 14(1).
16. Amzil, A., Hanini, M. and Zaaloul, A., 2025. Modeling and analysis of LoRa-enabled task offloading in edge computing for enhanced battery life in wearable devices. *Cluster Computing*, 28(3), p.201.
17. Alsamarai, N.A. and Uçan, O.N., 2024. Improved Performance and Cost Algorithm for Scheduling IoT Tasks in Fog-Cloud Environment Using Gray Wolf Optimization Algorithm. *Applied Sciences*, 14(4), p.1670.
18. Pournazari, J., Ullah, A., Al-Dubai, A. and Liu, X., 2025. Computation offloading in the edge-to-cloud compute continuum: a survey of federated architectural solutions. *Cluster Computing*, 28(13), p.839.
19. Ahmed, S.F., Shawon, S.S., Afrin, S., Rafa, S.J., Hoque, M. and Gandomi, A.H., 2025. Optimising Internet of Things (IoT) Performance Through Cloud, Fog and Edge Computing Architecture. *IET Wireless Sensor Systems*, 15(1), p.e70016.
20. Al-Qadhi, A.K., Latip, R., Chiong, R., Athauda, R. and Hussin, M., 2025. Independent task scheduling algorithms in fog environments from users' and service providers' perspectives: a systematic review. *Cluster Computing*, 28(3), p.209.
21. Low, C.K., Ng, Y.P., Tew, Y. and Qing, L.C., The study of fog computing: security and privacy, energy consumption, quality of experience/services, offloading, and task scheduling. *Recent Trends Cloud Comput Web Engi*, 7(1).
22. Xu, Z., Zhang, Y., Li, H., Yang, W. and Qi, Q., 2020. Dynamic resource provisioning for cyber-physical systems in cloud-fog-edge computing. *Journal of Cloud Computing*, 9(1), p.32.
23. Rana, B., Singh, Y. and Singh, P.K., 2021. A systematic survey on internet of things: Energy efficiency and interoperability perspective. *Transactions on Emerging Telecommunications Technologies*, 32(8), p.e4166.