Artificial Intelligence and Computing Innovations

Intelligence Science Publishing

https://www.ispublishing.com

ISSN: 3070-4944

Vo.4, No.4, 2024

Reinforcement Learning-Based Task Scheduling Under Energy Budget Constraints in Edge-Cloud Systems

Jianan Sun[1]*

[1]Syracuse University, Syracuse, USA

**Corresponding author: jianansun618@gmail.com*

Abstract: This paper addresses the energy constraint challenge in task scheduling within edge-cloud collaborative architectures by proposing a scheduling method named EA-BCRL (Energy-Aware Budget-Constrained Reinforcement Learning). The method integrates reinforcement learning with an energy-aware mechanism. It constructs an energy budget modeling module to dynamically estimate the current energy state and budget utilization of the system, enabling real-time modeling and feedback of budget constraints. A budget-guided reward shaping mechanism is introduced to embed energy constraint signals into the policy optimization process, guiding the reinforcement learning agent to avoid budget violations while ensuring task performance. The overall framework achieves coordinated optimization of energy modeling, policy updating, and budget control during task offloading and resource scheduling. Experiments are conducted on a real-world dataset to validate the method at scale, with evaluation metrics including task success rate, average energy consumption, budget violation rate, and average latency. Comparative results show that the proposed method outperforms existing mainstream scheduling approaches across multiple key performance indicators. Ablation studies are further carried out to analyze the independent contributions of each module. Multi-dimensional sensitivity tests are also performed on learning rate, task scale, data disturbance, and task heterogeneity. These results confirm the stability and adaptability of the method. Overall, the findings demonstrate that the proposed approach enables efficient and stable energy-constrained scheduling under complex resource environments and diverse workload conditions.

Keywords: Edge-cloud collaboration; energy consumption modeling; reinforcement learning scheduling; budget control

1. Introduction

With the rapid advancement of new-generation information technologies, edge computing and cloud computing have emerged as two pivotal paradigms to address the growing demand for data processing and intelligent services[1,2]. Edge computing provides low-latency and near-source processing capabilities, demonstrating significant value in scenarios such as intelligent transportation, industrial IoT, and smart healthcare[3]. In contrast, cloud computing offers powerful centralized computing resources and large-scale resource pooling, supporting data storage and complex model inference. The integration

of these two forms a collaborative edge-cloud architecture. This architecture aims to optimize computational efficiency and service quality through hierarchical task processing and dynamic resource coordination. However, in practical deployments of edge-cloud collaboration, energy constraints have become an increasingly critical issue. In energy-limited environments such as mobile devices and embedded terminals, balancing task performance and energy consumption has become a major challenge in edge-cloud scheduling[4].

Traditional task scheduling strategies primarily focus on optimizing computing performance, network latency, or resource utilization. They often overlook the critical role of energy management in ensuring sustainable system operation. In edge-cloud collaborative systems, task heterogeneity and environmental dynamics render static or heuristic scheduling methods insufficient for real-world demands. Moreover, energy consumption is influenced not only by computational load but also by device states, network transmission, and data distribution[5]. Developing accurate energy-aware models and incorporating them into the scheduling process is essential to achieving joint optimization of performance and energy efficiency. In many real-world systems, energy budgets are not unlimited but are subject to strict upper bounds. Under such constraints, scheduling methods must be capable of sensing and controlling energy budgets to ensure stable system operation within resource limits[6].

Reinforcement learning has shown great potential in complex decision-making problems and is gaining attention in task scheduling[7]. By interacting with the environment, reinforcement learning can adapt to dynamic system states and discover scheduling strategies that maximize long-term rewards. This approach provides a new perspective for resource allocation in edge-cloud collaboration. Especially when facing multi-objective optimization goals such as latency and energy consumption, and under dynamic energy budget constraints, reinforcement learning offers online learning and adaptive capabilities. These allow continuous strategy adjustment in response to changing task loads and energy budgets. However, effectively integrating global and dynamic energy constraints into reinforcement learning frameworks remains a major challenge. The goal is to develop policies that optimize system performance without exceeding energy limits, which requires careful model design and policy learning strategies[8].

From a system engineering perspective, energy-constrained task scheduling is not a single-objective problem. It involves a systemic collaboration process that includes state modeling, constraint representation, policy learning, and inference. Offloading decisions between edge and cloud must consider current device status and historical energy consumption. Scheduling strategies should balance short-term execution efficiency with long-term energy allocation. They should also recognize different task types and service levels. Introducing structured modeling, hierarchical scheduling, and distributed coordination can enhance the intelligence of scheduling systems. This supports the development of a new paradigm for energy-efficient edge-cloud task scheduling[9].

This study focuses on energy budget constraints in edge-cloud collaborative task scheduling. It aims to integrate reinforcement learning with energy modeling mechanisms. The goal is to explore unified scheduling strategies that balance performance and energy efficiency in complex heterogeneous systems. This research direction holds significant theoretical value and aligns with future trends in green computing, intelligent edge, and autonomous resource management. It contributes to the construction of energy-friendly intelligent systems and improves the operational efficiency and sustainability of smart infrastructures. By leveraging the capability of reinforcement learning in dynamic constraint modeling and policy optimization, this work is expected to provide general and intelligent scheduling solutions for energy-constrained systems. It will further promote the broad deployment and efficient operation of edge-cloud integrated systems in diverse application scenarios.

2. Related work

2.1 Research on energy-aware task scheduling under edge-cloud collaborative architecture

Edge-cloud collaboration, as a system architecture that integrates the advantages of edge computing and cloud computing, is gradually becoming a fundamental infrastructure for supporting intelligent applications and complex services[10]. Under this architecture, tasks can be flexibly offloaded and distributed between edge devices and cloud servers based on multidimensional characteristics such as computational intensity, data volume, and latency sensitivity. This enables a dynamic balance between service quality and resource utilization. However, with the rapid increase in the number of intelligent terminals and the continuous rise in computing demands, the overall energy pressure of the system has also grown rapidly. Especially at energy-constrained edge nodes, how to effectively control energy consumption while maintaining task performance has become a key issue in edge-cloud collaborative scheduling[11].

Early research on task scheduling mainly focused on performance-driven objectives, such as minimizing response time, improving throughput, or reducing task queuing delays. These methods often assume sufficient computing resources and show limited sensitivity to energy consumption. In practice, edge nodes typically have limited resources and often operate on battery power. Without proper energy control mechanisms, systems face serious risks such as frequent power outages and thermal degradation[12]. In response, recent studies have attempted to introduce energy-aware mechanisms. These approaches incorporate device power models, task load predictions, and resource usage patterns into scheduling decisions to minimize overall energy consumption while maintaining service quality[13].

Based on this, two strategic approaches have emerged: energy efficiency optimization and energy budget constraint. The former aims to maximize task execution efficiency per unit of energy, while the latter focuses on completing as many tasks as possible within a limited energy budget. The latter is particularly suitable for systems with unstable energy supply or strict energy limits[14]. Compared with traditional performance-oriented scheduling strategies, these methods often involve task prioritization, budget allocation models, and energy tracking mechanisms. These components support resource scheduling under dynamic constraints[15].

However, current energy-aware scheduling methods still face limitations in edge-cloud collaborative systems. Many approaches rely on static rules or empirical parameters, which are not adaptable to the dynamic changes in complex system states. Additionally, energy modeling and scheduling strategies are often disconnected, lacking a unified optimization framework. In scenarios with significant task heterogeneity and frequent node state fluctuations, static scheduling strategies struggle to meet the joint requirements of real-time performance and energy efficiency[16]. Therefore, there is an urgent need for scheduling mechanisms with sensing, prediction, and adaptive capabilities. Such mechanisms are essential for efficient task orchestration under energy constraints and can offer stronger support for the sustainable development of edge-cloud collaboration systems.

2.2 Exploration of the application of reinforcement learning in resource scheduling and energy efficiency optimization

In recent years, reinforcement learning has emerged as a powerful tool for solving complex and dynamic decision-making problems. It has been gradually introduced into fields such as resource scheduling and energy efficiency optimization, showing strong modeling potential[17]. Traditional scheduling methods often struggle with rapid environmental changes and multi-objective coordination. In contrast, reinforcement learning can autonomously learn high-reward scheduling policies through interactions with the environment, adapting to dynamic system characteristics. Especially in scenarios involving multiple tasks, multiple nodes, and uncertain resource availability, reinforcement learning can construct a Markov decision process to continuously adjust scheduling behaviors from a global perspective[18]. This enables long-term optimization of system performance and provides a new

paradigm for addressing task allocation, resource coordination, and latency control in complex systems[19].

The application of reinforcement learning in energy efficiency optimization has also gained increasing attention, particularly in areas such as mobile edge computing and green data center management[20]. Studies have shown that by incorporating energy consumption into the reward function, reinforcement learning models can actively avoid high-energy scheduling paths while ensuring service quality. This enables energy-efficient task orchestration. Additionally, reinforcement learning has state-awareness and policy iteration capabilities. These allow adaptive control of energy consumption trends during system operation, making it especially suitable for resource-constrained environments that require real-time decision-making and budget control. This characteristic offers an effective supplement to traditional scheduling algorithms, which often suffer from outdated energy models and rigid policies[21].

Despite its advantages, the application of reinforcement learning in practical scheduling systems still faces several challenges[22]. On the one hand, there is a lack of general modeling frameworks that can effectively integrate multiple optimization objectives, such as delay minimization and energy budget constraints. On the other hand, learning efficiency and convergence speed are limited when dealing with large state spaces and sparse reward signals. In task scheduling scenarios where energy budgets serve as dynamic constraints, it remains a challenge to develop policies that can maximize long-term rewards without violating budget limitations. Therefore, integrating reinforcement learning with energy-aware mechanisms and constructing constraint-driven policy optimization frameworks is a critical step toward deploying this method in edge-cloud collaborative scheduling systems.

Recent studies on backend observability and cloud reliability further expand the technical context for energy-aware scheduling. Behavior representation learning for AI inference backend monitoring shows that runtime behavior traces can be transformed into compact representations for dependable state monitoring[23]. Transformer-based semantic log modeling strengthens robust and explainable cloud security analytics by extracting higher-level operational semantics from system logs[24]. Graph neural modeling of traffic propagation and topological interactions in cloud networks provides a useful basis for describing how workload movement affects neighboring nodes and links[25]. Fine-grained backend anomaly classification through structural dependency learning and version-aware feature alignment also indicates that scheduling states should encode dependency structure and software evolution rather than relying only on instantaneous resource counters[26]. These studies motivate richer state representations for EA-BCRL, especially when energy budget decisions must remain stable under inference-load variation, topology changes, and backend version drift.

3. Method

This study addresses the problem of edge-cloud collaborative task scheduling under energy budget constraints and proposes an energy-aware scheduling method based on reinforcement learning (EA-BCRL). This method aims to jointly optimize performance and energy efficiency in a dynamic resource environment. This method incorporates two key innovations: first, it introduces an Energy Budget Modeling Module (EBMM), which jointly models the remaining energy of devices and task consumption trends to enhance the scheduling strategy's global awareness and responsiveness to energy constraints. Second, it designs a Budget-Guided Reward Shaping (BGRS) mechanism, which explicitly incorporates energy constraint signals into the reinforcement learning framework. This guides the strategy to dynamically balance task completion efficiency and energy consumption during training, enhancing scheduling intelligence and adaptability while ensuring system stability. The detailed structure of the proposed model is illustrated in Figure 1.

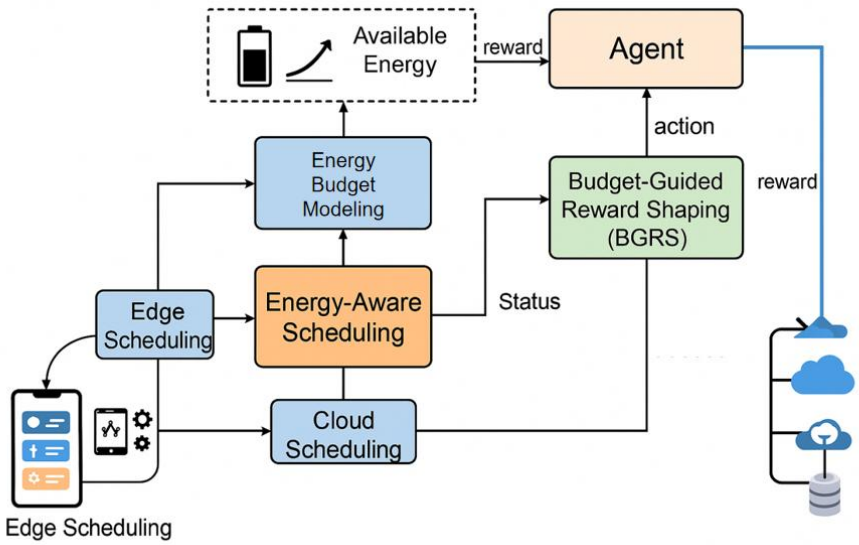


Figure 1. Overall model architecture diagram

3.1 Energy Budget Modeling Module

In the energy budget-constrained scheduling framework, the Energy Budget Modeling Module (EBM) serves as the foundation for policy decisions and undertakes key functions such as real-time estimation, dynamic feedback, and budget control. This module first collects the current available energy, historical energy consumption, and task distribution from edge devices. It then combines this information with device status to create a multidimensional feature vector representing the system's energy consumption status. Figure 2 shows the module architecture.

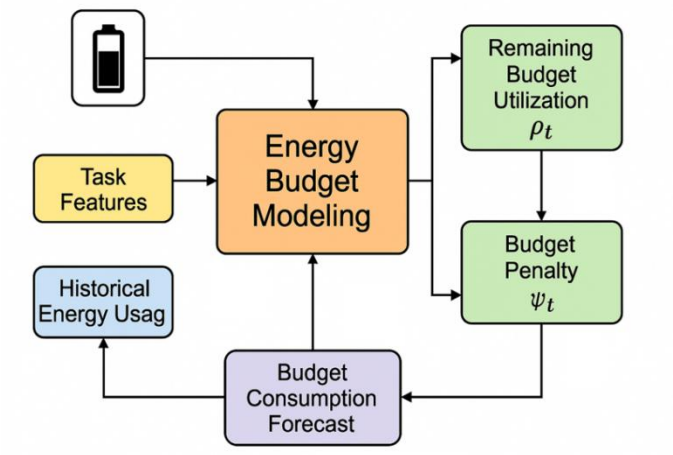


Figure 2. EBMM module architecture

Let the remaining power of the device at time t be E_t, the task set be $T_t = \{\tau_1, \tau_2, ..., \tau_n\}$, and the expected energy consumption of each task be ε_i. The remaining amount of the current energy budget can be expressed as:

$$B_t = E_t - \sum_{i=1}^{n} \varepsilon_i$$

To more accurately control the budget, a sliding window mechanism is introduced to model the energy consumption trend. The time window length is defined as Δ, and the average energy consumption at time Δ in the past is recorded as:

$$\varepsilon_i = \frac{1}{\Delta} \sum_{k=t-\Delta+1}^{t} \varepsilon_k$$

This estimate can be used to predict short-term future energy usage trends. At the same time, to enhance the model's awareness of energy consumption status, the budget utilization index ρ_t is introduced to measure the current budget consumption rate, which is defined as follows:

$$\rho_t = \frac{\sum_{k=1}^{t} \varepsilon_k}{E_0}$$

Where E_0 represents the initial total energy of the system. To prevent energy overconsumption, the system introduces a budget penalty factor φ_t, which is defined as the cost function when the budget usage exceeds the threshold θ:

$$\varphi_t = \begin{cases} 0, & \rho t \le \theta \\ a(\rho_t - \theta)^2, & \rho t > \theta \end{cases}$$

This penalty term will be used in subsequent reinforcement learning to reconstruct the reward signal to guide the strategy towards on-budget scheduling. Finally, to conduct a forward-looking evaluation of future scheduling behavior, the module introduces a prediction of the budget consumption rate within the remaining time:

$$\varepsilon_{t+1:t+H} = \frac{B_t}{H}$$

Where H is the period of the future scheduling horizon. Through the above modeling mechanism, the energy budget module can achieve efficient expression and feedback of the system energy consumption status, providing accurate energy support information for subsequent scheduling strategies.

3.2 Budget-Guided Reward Shaping

In energy-constrained scheduling tasks, effectively incorporating budget control factors into the reinforcement learning strategy optimization process is a key challenge in achieving intelligent scheduling. To this end, this study constructed a budget-guided reward shaping mechanism (Budget-Guided Reward Shaping). Structurally adjusting the original reward signal enables the agent to perceive and respond to changes in the energy budget during training. The model architecture is shown in Figure 3.

Assuming the original environment reward is r_t and the energy budget penalty is φ_t, the reconstructed reward function is defined as:

$$r_t = r_t - \lambda \cdot \varphi_t$$

Where λ is the budget sensitivity coefficient, which is used to adjust the system's penalty intensity for exceeding the budget. When the system budget is tight, the reinforcement learning agent will be incentivized to prioritize low-energy or local execution strategies, thus reflecting energy conservation at the policy level.

In addition to improving training stability and enhancing the generalization ability of the strategy, a state normalization and penalty smoothing mechanism is introduced. Let the state vector be s_t, and its normalized representation is:

$$s_t = \frac{s_t - \mu_s}{\sigma_s}$$

Where μ_s and σ_s are the expectation and standard deviation of the state, respectively. This normalization operation can suppress gradient oscillation during training and improve the convergence speed of policy updates. On this basis, to avoid the penalty impact caused by high-frequency fluctuations, an exponential moving average smoothing budget penalty term is introduced:

$$\widetilde{\varphi}_t = \beta \cdot \widetilde{\varphi}_{t-1} + (1-\beta) \cdot \varphi_t$$

Where $\beta \in [0, 1)$ is the smoothing factor. The reconstruction reward is also updated as follows:

$$r_t = r_t - \lambda \cdot \widetilde{\varphi}_t$$

In addition, to further improve the budget adaptability of the strategy, a budget threshold offset adjustment strategy is set. Let the target budget threshold be θ, the actual budget utilization be p_t, and the adjustment item δ_t be introduced:

$$\delta_t = \gamma(\rho_t - \theta)$$

The final policy function $\pi(a_t \mid s_t)$ will learn and make decisions based on the adjusted reward $\widetilde{r}_t$ and state s_t, dynamically balancing task completion and budget efficiency. This mechanism enables controllable expression and soft guidance of budget constraints, providing more interpretable and secure training signals for reinforcement learning scheduling strategies.

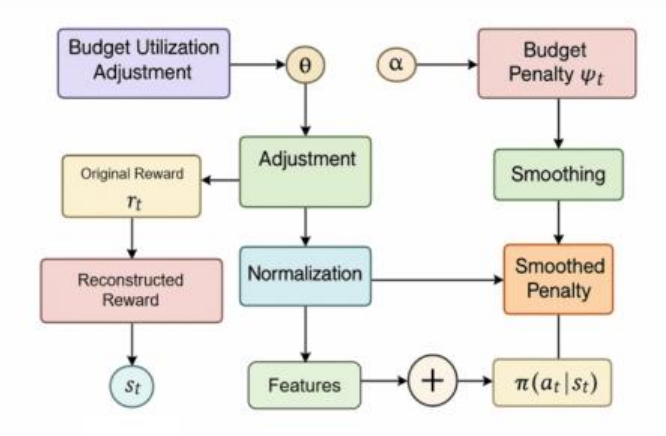


Figure 3. BGRS module architecture

4. Experimental Results

4.1 Dataset

This study uses the Alibaba Cluster Trace 2018 dataset as the primary source of experimental data. It is employed to simulate task scheduling behaviors and energy consumption patterns in large-scale edge-cloud collaborative computing scenarios. The dataset was collected from a real production environment. It contains hundreds of thousands of task scheduling records and resource usage information. It is widely used in distributed scheduling and system optimization research, offering high reliability and representativeness.

The dataset includes complete task lifecycle information. Key fields cover submission time, scheduling delay, execution duration, CPU and memory requests, and actual resource usage rates. It also provides server-level resource allocation data and failure annotations. These are useful for constructing realistic models of system dynamics. By extracting load sequences and energy consumption estimates for

edge nodes and cloud servers, the dataset provides detailed support for energy budget modeling and reinforcement learning training in this study.

To meet the requirements of this research, the task data is preprocessed through several steps. These include missing value imputation, outlier removal, time window resampling, and resource normalization. A multidimensional feature sequence is then constructed as input to the scheduling model. The dataset covers both high-frequency interactive tasks and low-frequency batch tasks. It also reflects resource fluctuations and uneven energy consumption in real systems. This provides a solid foundation for the design of scheduling strategies under budget constraints.

4.2 Experimental setup

The experiments were conducted on a high-performance computing server. The hardware configuration included an NVIDIA A100 GPU with 40 GB of memory, two Intel Xeon Gold 6348 processors, and 512 GB of RAM. The operating system was Ubuntu 22.04 LTS. In terms of software, all implementations were written in Python 3.10. The deep learning framework used was PyTorch 2.1, accelerated with CUDA 11.8 and cuDNN 8.6. All random seeds were fixed to ensure reproducibility.

For the reinforcement learning component, a continuous control algorithm based on the Actor-Critic architecture was adopted. Both the policy and value networks consisted of three fully connected layers, with hidden units set to 256, 256, and 128, respectively. The ReLU activation function was used. The optimizer was Adam, with an initial learning rate of $1x10^{-4}$. An exponential decay mechanism was applied to adjust the learning rate dynamically, with a decay rate of 0.98. Each training round included up to 20,000 environment interactions. The experience replay buffer size was 50,000, and the batch size was 256.

In the task scheduling simulation environment, the energy budget was initialized as a fixed upper limit within each window cycle. The sliding window length was set to 20. The budget threshold was set to 0.85. The penalty coefficient λ for the reshaped reward was set to 2.0. The smoothing factor β was set to 0.9. All training was conducted using a single GPU. The average training time per round was approximately 4.2 seconds. During the entire training process, logs and model parameters were saved to ensure reproducibility and support further visualization analysis.

4.3 Experimental Results

4.3.1 Comparative experimental results

This paper first conducts a comparative experiment, and the experimental results are shown in Table 1.

Table 1. Comparative experimental results

Model	TSR	AEC	BVR	AL
Ours	94.28	17.63	1.82	243.7
GOBI[27]	91.75	21.84	5.09	270.1
DeepEE[28]	89.92	19.71	3.44	286.5
BURL[29]	87.36	23.56	6.17	298.3
DRLEO[30]	90.12	20.95	4.73	264.2

From the overall results, the proposed method achieves the best performance in Task Success Rate (TSR), reaching 94.28%. This represents a clear improvement over existing reinforcement learning and energy-aware scheduling baselines. This result indicates that with the integration of energy budget modeling and reward reshaping mechanisms, the scheduling policy gains stronger global optimization capabilities under complex heterogeneous tasks and energy constraints. It significantly enhances the task execution stability and resource responsiveness of the edge-cloud collaborative system.

For the Average Energy Consumption (AEC) metric, the proposed method achieves the lowest consumption at 17.63 J, outperforming all comparison models. This confirms the effectiveness of the Energy Budget Modeling Module (EBMM) in expressing dynamic constraints and improving energy awareness. In contrast, methods such as BURL and GOBI exhibit some performance scheduling capabilities but fail to effectively restrict energy usage under fluctuating multi-task workloads, leading to lower overall energy efficiency. The proposed method dynamically adjusts scheduling actions based on remaining system budgets and task energy trends, enabling better trade-offs between performance and energy consumption.

The Budget Violation Rate (BVR) is a key metric for assessing whether a scheduling strategy can maintain stable operation under constraints. The proposed method keeps this value at 1.82%, significantly better than GOBI (5.09%) and BURL (6.17%). This highlights the effectiveness of the Budget-Guided Reward Shaping mechanism (BGRS) in integrating constraints into reinforcement learning. By explicitly introducing budget violation penalties during policy learning, the agent becomes more responsive to budget boundaries, effectively avoiding overconsumption and improving system controllability and stability.

In terms of Average Latency (AL), the proposed method also achieves the lowest latency among all models, at only 243.7 ms. This demonstrates that the proposed scheduling strategy does not sacrifice response speed while achieving high task success and low energy consumption. Instead, it optimizes delay by intelligently allocating task execution locations and scheduling order. This synergy across performance, energy efficiency, and stability highlights the broad deployment potential of the proposed EA-BCRL model in real-world edge-cloud systems.

4.4 Ablation Experiment Results

This paper further gives the results of the ablation experiment, which are shown in Table 2.

Table 2. Ablation Experiment Results

Method	TSR	AEC	BVR	AL
Baseline	88.71	22.98	6.42	289.6
+ EBMM	91.36	19.84	3.95	266.2
+ BGRS	90.42	21.12	2.87	258.3
+ All	94.28	17.63	1.82	243.7

From the overall trend, the full model that incorporates both the Energy Budget Modeling Module (EBMM) and the Budget-Guided Reward Shaping (BGRS) mechanism achieves a significant improvement in Task Success Rate (TSR), with an increase of approximately 5.57% over the baseline. This result indicates that the proposed EA-BCRL framework provides stronger scheduling adaptability and global optimization ability. Compared to the performance gains from introducing either module alone, the complete model shows better synergy in policy learning. This suggests that the two components are complementary and interact effectively, jointly enhancing the system's robustness under complex constraint environments.

In terms of Average Energy Consumption (AEC), the introduction of the EBMM module reduces energy usage from 22.98 J to 19.84 J without affecting task completion. This shows that energy modeling provides more fine-grained guidance for energy allocation in the scheduling policy. When the BGRS module is added, the reduction in energy consumption is less pronounced. However, budget compliance improves significantly. This indicates that the reward shaping mechanism helps the agent focus more on long-term resource efficiency during learning, supporting energy-driven scheduling optimization.

Budget Violation Rate (BVR) is a key indicator of how well the system adheres to energy constraints. The introduction of either module leads to a notable decrease in BVR. Specifically, the BGRS module reduces BVR from 6.42% to 2.87%, showing that budget penalties play an important role in guiding

policy learning. When combined with EBMM, the BVR further drops to 1.82%. This demonstrates the model's capability to maintain a stable balance between task execution and energy limits. These results reinforce the theoretical soundness and practical effectiveness of the proposed dual-mechanism strategy that combines budget awareness and reward constraints.

Regarding latency, the complete model reduces the Average Latency (AL) to 243.7 ms, which is nearly 46 ms lower than the baseline. This indicates that integrating energy and budget information into the reinforcement learning framework does not introduce additional performance overhead. On the contrary, it improves scheduling precision and reduces task reassignment, which shortens response time. This result further confirms the EA-BCRL framework's advantage in jointly optimizing performance, energy efficiency, and system stability. It demonstrates the practical feasibility of deploying this approach in large-scale edge-cloud collaborative systems.

4.5 The impact of different learning rate settings on model performance

This paper also gives the impact of different learning rate settings on model performance, and the experimental results are shown in Figure 4.

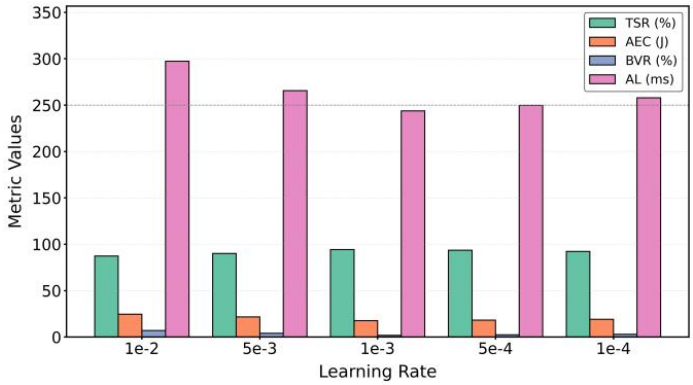


Figure 4. The impact of different learning rate settings on model performance

The results in the figure show that the choice of learning rate has a significant impact on the overall scheduling performance of the model. When the learning rate is set to 1×10^{-3}, the model achieves the highest Task Success Rate (TSR). This indicates that the training process of the reinforcement learning policy is more stable at this setting, allowing the model to better capture long-term optimal behaviors in edge-cloud collaborative scheduling. In contrast, when the learning rate is set to 1×10^{-2}, TSR drops significantly. This suggests that an excessively high step size may lead to policy oscillation or even training failure.

For Average Energy Consumption (AEC), as the learning rate decreases from higher values, energy consumption first drops noticeably and then levels off. Specifically, at 1×10^{-3}, the model achieves the lowest energy consumption. This shows that the policy has learned to allocate computing resources efficiently under budget constraints and avoids scheduling decisions that lead to redundancy or high energy usage. This trend confirms the value of the Energy Budget Modeling Module during reinforcement learning, as proper learning rate settings enhance its modeling capability.

From the perspective of Budget Violation Rate (BVR), the violation rate generally decreases with lower learning rates. Between 1×10^{-3} and 5×10^{-4} , BVR remains at a low level, reflecting the strong constraint reinforcement effect of the Budget-Guided Reward Shaping mechanism under medium and low learning rates. In contrast, extremely high or low learning rates tend to cause policy misjudgments regarding budget constraints. This often results in improper resource allocation and frequent budget violations.

In terms of Average Latency (AL), the model also achieves the best performance when the learning rate is set to 1×10^{-3}. This indicates that this setting not only improves task execution efficiency but also achieves a good balance in ensuring timely task responses. Overall, the experimental results demonstrate that a moderate learning rate helps the scheduling policy achieve robust adherence to budget constraints while ensuring task completion, reducing energy consumption, and minimizing latency. This provides practical guidance for parameter selection in the real-world deployment of reinforcement learning-based scheduling frameworks.

4.6 Testing model transfer capabilities at different task scales

This paper also presents a comprehensive test of the model's migration capability under varying task scales, aiming to evaluate its generalization performance across different workload intensities. By systematically adjusting the number of tasks in the scheduling environment, the study examines how the proposed model adapts to increasing or decreasing computational demands. This test is crucial for assessing the scalability and robustness of the scheduling policy when applied to systems with diverse task volumes, ranging from small-scale deployments to large-scale distributed settings. The evaluation considers the model's ability to maintain stable performance and consistent decision-making behavior as task complexity and resource contention levels change. Such analysis offers valuable insights into the practical applicability of the model in dynamic edge-cloud environments. The corresponding experimental results are illustrated in Figure 5.

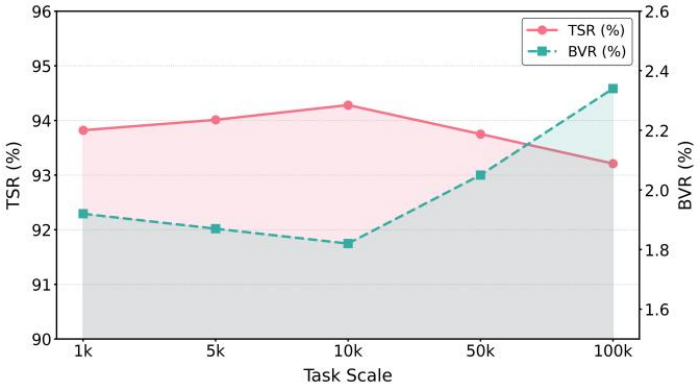


Figure 5. Testing model transfer capabilities at different task scales

The results in the figure indicate that the proposed scheduling model demonstrates strong transferability across different task scales. As the task volume increases from 1k to 10k, the Task Success Rate (TSR) steadily improves and reaches a peak of 94.28% at the 10k scale. This suggests that the model exhibits good generalization and scheduling adaptability in medium-scale task scenarios. It also shows that the energy budget modeling mechanism can maintain an accurate representation under varying load densities, providing stable energy guidance for the policy.

As the task scale further increases to 50k and 100k, the TSR experiences a slight decline but remains above 93%. This indicates that the model continues to maintain high task completion efficiency under high-load conditions. Despite increased task congestion and intensified resource contention, the dynamic energy control mechanism introduced in EA-BCRL retains a certain degree of stability and effectively prevents large performance fluctuations.

In terms of Budget Violation Rate (BVR), the model shows strong budget control in small-scale task scenarios, with the violation rate remaining below 2%. However, as the task scale expands to 100k, the BVR rises to approximately 2.4%. This suggests that under extreme task pressure, the policy may sacrifice some budget compliance in pursuit of higher task throughput, leading to more frequent budget violations. It reflects that the current mechanism still has room for improvement under extreme load conditions.

Overall, the experimental results confirm the transfer stability and scheduling resilience of EA-BCRL across various task scales. The model maintains a high TSR while effectively controlling the energy budget in most cases, demonstrating strong scalability and applicability. This analysis further validates the practicality and generality of the proposed mechanism in dynamic and resource-fluctuating environments.

4.7 Impact of data missing and noise injection on the energy consumption perception module

This paper also provides a detailed analysis of the impact that data missing and noise injection have on the energy consumption perception module. Specifically, it explores how variations in data quality, such as incomplete system observations or corrupted input signals, influence the accuracy and stability of the module's output. By simulating different levels of data disturbance, the study investigates the module's ability to maintain reliable energy state estimation and budget utilization feedback under non-ideal conditions. This analysis is essential for understanding the sensitivity of the perception mechanism to real-world uncertainties, including sensor faults, communication delays, and measurement errors. The corresponding experimental results, which illustrate these effects in detail, are presented in Figure 6.

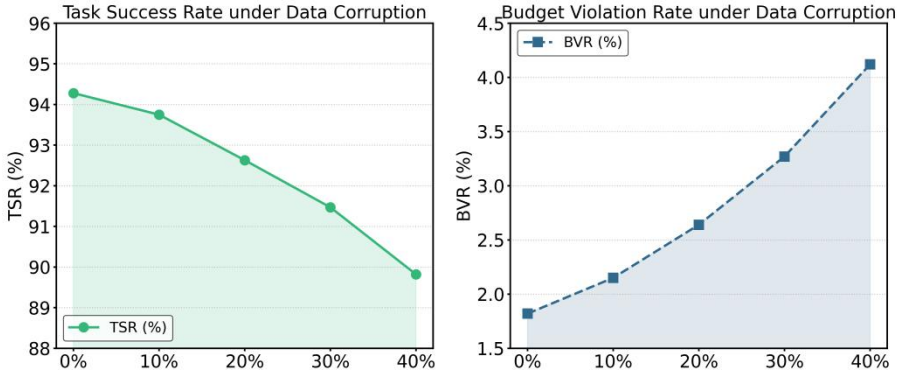


Figure 6. Impact of data missing and noise injection on the energy consumption perception module

The experimental results show that data loss and noise injection have a significant impact on the energy-aware module, which in turn affects overall scheduling performance. When the data is complete (0% perturbation), the Task Success Rate (TSR) remains above 94%, indicating that the energy budget modeling module can effectively construct system states and support efficient policy execution. However, as the perturbation ratio increases, TSR shows a continuous decline, dropping to 89.82% at 40% perturbation. This suggests that damaged perception information weakens the model's ability to accurately assess resource conditions, reducing scheduling accuracy and task success.

The Budget Violation Rate (BVR) shows an opposite trend. As the perturbation increases, BVR rises from an initial 1.82% to 4.12%. This result indicates that data loss and noise weaken the effectiveness of budget constraints within the scheduling policy. The model becomes less capable of identifying high-risk energy consumption zones during training, which leads to more frequent budget violations. This trend confirms that the reward shaping mechanism is highly sensitive to data quality and depends on the stable output of the perception module.

It is worth noting that when the perturbation ratio is between 10% and 20%, TSR and BVR fluctuate slightly, but overall performance remains within an acceptable range. This demonstrates that the EA-BCRL model has a certain level of disturbance resistance. It can maintain scheduling stability under mild information loss or observation errors. This result reflects the redundancy and fault tolerance built into the proposed mechanism, making it suitable for edge scenarios with occasional anomalies and environmental uncertainties.

In summary, the experiment highlights the critical importance of maintaining high-quality data input for effective energy perception and accurate policy decision-making in edge-cloud collaborative systems.

Reliable input data serves as the foundation for the energy budget modeling module to correctly estimate system states and provide trustworthy feedback to the reinforcement learning agent. Any degradation in data quality, such as missing values or noisy observations, may compromise the model's ability to perceive real-time energy usage patterns and enforce budget constraints effectively. Therefore, the findings suggest that future system deployments should consider integrating auxiliary mechanisms such as anomaly detection, data imputation, or self-correction techniques to proactively identify and mitigate data disturbances. These enhancements would enable the perception module to deliver more stable outputs, even under uncertain or degraded conditions. As a result, the overall scheduling framework would benefit from improved adaptability, resilience, and operational consistency when deployed in real-world, dynamic environments characterized by fluctuating workloads and imperfect sensing.

4.8 The impact of changes in the proportion of heterogeneous tasks on strategy stability

This paper also gives a detailed investigation into the impact of changes in the proportion of heterogeneous tasks on the stability of the scheduling strategy. By varying the ratio of tasks with differing resource demands, execution times, and computational characteristics, the study explores how increased task diversity influences the model's ability to maintain consistent scheduling performance. This analysis is particularly relevant in edge-cloud environments, where workloads are often composed of a mixture of real-time, interactive, and batch processing tasks. Evaluating the strategy under different levels of heterogeneity helps to assess its robustness in handling dynamic and complex task distributions. It also provides insight into the limits of policy generalization and the degree to which the model can preserve budget compliance and decision reliability under diverse operating conditions. The corresponding experimental results are presented in Figure 7.

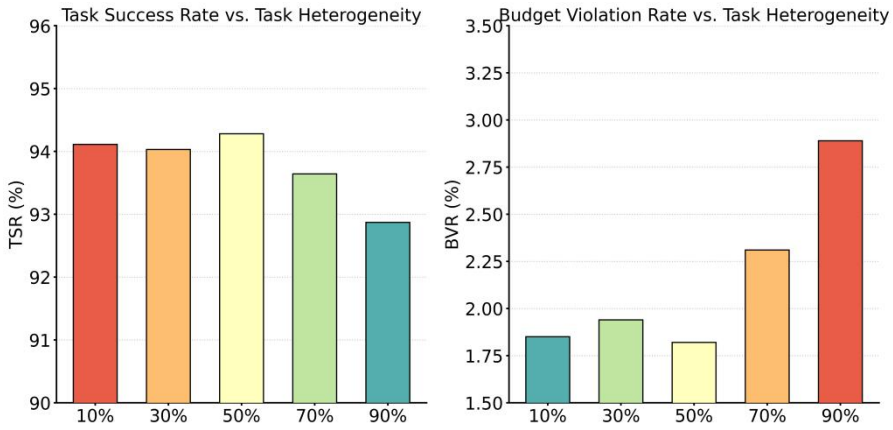


Figure 7. The impact of changes in the proportion of heterogeneous tasks on strategy stability

The results in the figure show that increasing the proportion of heterogeneous tasks has a significant impact on the model's scheduling stability and compliance with resource constraints. The Task Success Rate (TSR) remains high when the heterogeneity ratio is low, between 10% and 50%. The model achieves its peak TSR of 94.28% at 50%, indicating that it can effectively adapt to task feature differences under moderate heterogeneity. This demonstrates good generalization in scheduling and convergence of the learned policy. It reflects the robustness and adaptability of the energy modeling and policy learning mechanisms when facing changes in task structure.

As the heterogeneity ratio further increases to 70% and 90%, TSR begins to decline. This suggests that excessive task diversity poses challenges to constructing a global scheduling policy. In highly heterogeneous scenarios, tasks have significantly different resource demands and execution behaviors. The reinforcement learning policy may accumulate errors in perception and allocation, which leads to reduced scheduling efficiency. This trend reveals the existence of a "stability window" for task structure complexity in edge-cloud systems. Beyond this window, model performance starts to fluctuate.

The Budget Violation Rate (BVR) curve further confirms this conclusion. When task heterogeneity is low, from 10% to 50%, BVR remains at a low level, indicating that the model can control energy budgets and avoid resource overload. However, when the proportion of heterogeneous tasks exceeds 70%, BVR increases significantly. At 90%, it approaches 2.89%. This shows that the scheduling policy becomes less capable of maintaining budget compliance under highly diverse workloads. This is partly due to the difficulty of accurately perceiving system states when feedback from tasks is highly variable, which leads to overload in resource allocation.

In summary, the experimental results verify the stability boundaries of the EA-BCRL method in heterogeneous environments. The model maintains high performance and strong compliance under low to moderate heterogeneity. However, in highly complex task structures, there is a risk of policy oscillation and increased budget violations. These findings provide useful parameter references for large-scale heterogeneous system deployment and motivate future work on stronger structural modeling mechanisms.

5. Conclusion

This paper addresses the problem of edge-cloud collaborative task scheduling under energy budget constraints and proposes an energy-aware scheduling method named EA-BCRL, which integrates reinforcement learning. By introducing an Energy Budget Modeling Module (EBMM) and a Budget-Guided Reward Shaping (BGRS) mechanism, the method enables joint policy learning under both performance optimization and energy budget constraints. It effectively enhances the resource sensitivity and energy controllability of the scheduling strategy. The proposed approach not only structurally bridges energy awareness and reinforcement policy learning but also provides a feasible solution for low-carbon scheduling in edge computing environments.

Through systematic experiments on a real-world task workload dataset, the proposed method achieves significant advantages in multiple key metrics, including task success rate, budget compliance, and average latency. It demonstrates strong robustness and transferability in heterogeneous task environments and under complex resource fluctuation scenarios. Ablation studies further reveal the complementary and synergistic effects of the two core modules in improving model performance. They also confirm the independent effectiveness and cooperative tuning capabilities of budget modeling and reward constraint mechanisms. These results are of practical importance for promoting energy-efficient management strategies in large-scale distributed computing.

Additionally, this study conducts multi-dimensional sensitivity experiments under different environmental disturbances and structural settings. It systematically evaluates the model's stability and adaptability in scenarios involving learning rate tuning, task scale variation, perception degradation, and task heterogeneity. These evaluations enhance the credibility of the model's real-world deployment in edge-cloud collaborative systems. The results also reveal the stability boundaries of the scheduling strategy in highly heterogeneous and noisy environments, providing both theoretical support and practical guidance for future algorithmic robustness design.

Future research can further extend the current method to complex task systems involving multi-objective scheduling, resource negotiation, and cross-platform coordination. Adaptive perception mechanisms and game-theoretic reasoning frameworks can be introduced to enhance the system's responsiveness to sudden resource conflicts and energy shocks. In addition, reinforcement learning can be combined with graph modeling techniques to build structure-aware energy prediction systems. This will support the development of more general and real-time scheduling solutions for ultra-large-scale intelligent systems and advance the deployment of edge-cloud computing in industrial IoT, smart manufacturing, and low-carbon computing infrastructures.

References

[1] Ben Sada A, Khelloufi A, Naouri A, et al. Multi-agent deep reinforcement learning-based inference task scheduling and offloading for maximum inference accuracy under time and energy constraints[J]. Electronics, 2024, 13(13): 2580.

[2] Wang Z, Chen S, Bai L, et al. Reinforcement learning based task scheduling for environmentally sustainable federated cloud computing[J]. Journal of Cloud Computing, 2023, 12(1): 174.

[3] Li K. Energy-Constrained DAG Scheduling on Edge and Cloud Servers with Overlapped Communication and Computation[J]. Journal of Grid Computing, 2024, 22(3): 60.

[4] Pandey N K, Diwakar M, Shankar A, et al. Energy efficiency strategy for big data in cloud environment using deep reinforcement learning[J]. Mobile Information Systems, 2022, 2022(1): 8716132.

[5] Xun M, Yao Y, Yu J, et al. Deep Reinforcement Learning for Delay and Energy-Aware Task Scheduling in Edge Clouds[C]//CCF Conference on Computer Supported Cooperative Work and Social Computing. Singapore: Springer Nature Singapore, 2023: 436-450.

[6] Wang Z, Goudarzi M, Gong M, et al. Deep reinforcement learning-based scheduling for optimizing system load and response time in edge and fog computing environments[J]. Future Generation Computer Systems, 2024, 152: 55-69.

[7] Ale L, King S A, Zhang N, et al. D3PG: Dirichlet DDPG for task partitioning and offloading with constrained hybrid action space in mobile-edge computing[J]. IEEE Internet of Things Journal, 2022, 9(19): 19260-19272.

[8] Kim D Y, Lee D E, Kim J W, et al. Collaborative policy learning for dynamic scheduling tasks in cloud–edge–terminal IoT networks using federated reinforcement learning[J]. IEEE Internet of Things Journal, 2023, 11(6): 10133-10149.

[9] Xu X, Li H, Xu W, et al. Reinforcement Learning-Based Computation Offloading and Resource Allocation in MEC: A Survey[J]. ACM Computing Surveys, 2023, 55(13): 1-38.

[10] Miao Z, Liu L, Nan H, et al. Energy and carbon-aware distributed machine learning tasks scheduling scheme for the multi-renewable energy-based edge-cloud continuum[J]. Science and Technology for Energy Transition, 2024, 79: 82.

[11] Sheng S, Chen P, Chen Z, et al. Deep reinforcement learning-based task scheduling in iot edge computing[J]. Sensors, 2021, 21(5): 1666.

[12] Wang Z, Chen S, Bai L, et al. Reinforcement learning based task scheduling for environmentally sustainable federated cloud computing[J]. Journal of Cloud Computing, 2023, 12(1): 174.

[13] Zhao T, Zhou S, Guo X, et al. A Cooperative Computation Offloading and Resource Allocation Framework in Mobile Edge Computing Based on Deep Reinforcement Learning[J]. IEEE Transactions on Vehicular Technology, 2020, 69(10): 12212-12225.

[14] Sheng S, Chen P, Chen Z, et al. Deep reinforcement learning-based task scheduling in iot edge computing[J]. Sensors, 2021, 21(5): 1666.

[15] Reddy P V, Reddy K G. An energy efficient RL based workflow scheduling in cloud computing[J]. Expert Systems with Applications, 2023, 234: 121038.

[16] Jayanetti A, Halgamuge S, Buyya R. Deep reinforcement learning for energy and time optimized scheduling of precedence-constrained tasks in edge–cloud computing environments[J]. Future Generation Computer Systems, 2022, 137: 14-30.

[17] Jayanetti A, Halgamuge S, Buyya R. Deep reinforcement learning for energy and time optimized scheduling of precedence-constrained tasks in edge–cloud computing environments[J]. Future Generation Computer Systems, 2022, 137: 14-30.

[18] Jawaddi S N A, Ismail A, Shafian S. Enhancing energy efficiency in cloud scaling: A DRL-based approach incorporating cooling power[J]. Sustainable Energy Technologies and Assessments, 2023, 60: 103508.

[19] Jayanetti A, Halgamuge S, Buyya R. Deep reinforcement learning for energy and time optimized scheduling of precedence-constrained tasks in edge–cloud computing environments[J]. Future Generation Computer Systems, 2022, 137: 14-30.

[20] Ke H, Wang H, Sun H. Multi-agent deep reinforcement learning-based partial task offloading and resource allocation in edge computing environment[J]. Electronics, 2022, 11(15): 2394.

[21] Ramezani Shahidani F, Ghasemi A, Toroghi Haghighat A, et al. Task scheduling in edge-fog-cloud architecture: a multi-objective load balancing approach using reinforcement learning algorithm[J]. Computing, 2023, 105(6): 1337-1359.

[22] Ebrahim M, Hafid A. Privacy-aware load balancing in fog networks: A reinforcement learning approach[J]. Computer Networks, 2023, 237: 110095.

[23] Ding T, Yuan Y. Behavior Representation Learning for Reliable Monitoring of AI Inference Backends[J]. 2024, 4(1).

[24] Yang Z. Transformer-Based Semantic Log Modeling for Robust and Explainable Cloud Security Analytics[J]. 2024, 4(5).

[25] Huang X. Capturing Traffic Propagation and Topological Interactions in Cloud Networks via Graph Neural Networks[J]. Transactions on Computational and Scientific Methods, 2024, 4(5).

[26] Liu W. Fine-Grained Backend Anomaly Classification through Structural Dependency Learning and Version-Aware Feature Alignment[J]. 2024, 4(4).

[27] Jayanetti A, Halgamuge S, Buyya R. Deep reinforcement learning for energy and time optimized scheduling of precedence-constrained tasks in edge–cloud computing environments[J]. Future Generation Computer Systems, 2022, 137: 14-30.

[28] Mach P, Becvar Z. Mobile Edge Computing: A Survey on Architecture and Computation Offloading[J]. IEEE Communications Surveys & Tutorials, 2017, 19(3): 1628-1656.

[29] Mounesan M, Lemus M, Yeddulapalli H, et al. Reinforcement learning-driven data-intensive workflow scheduling for volunteer edge-cloud[C]//2024 IEEE 8th International Conference on Fog and Edge Computing (ICFEC). IEEE, 2024: 79-88.

[30] Hao Y, Yang S, Li F, et al. Edgetimer: Adaptive multi-timescale scheduling in mobile edge computing with deep reinforcement learning[C]//IEEE INFOCOM 2024-IEEE Conference on Computer Communications. IEEE, 2024: 671-680.