EAI Endorsed Transactions
on Scalable Information Systems
Research Article EAI.EU

A Distributed and Secure Resource Allocation Method for Power Communication Networks Based on Policy Distillation

Yue Zhang [1,*], Zongtao Li[1], Si Chen[2], Guoqiang Hu[2], Pengcheng Li[1] and Ruimei Wu[1]

[1] Inner Mongolia Power Communication Company, Hohhot 010020, China
[2] Inner Mongolia Power (Group) Co., Ltd., Hohhot 010020, China

Abstract

INTRODUCTION: In the next-generation smart grid communication architecture, how to achieve secure, dynamic, and fine-grained network resource allocation to ensure differentiated QoS for various services has become a key challenge. OBJECTIVES: Therefore, this study proposes a lightweight resource allocation method based on constrained policy distillation to address the challenge of balancing lightweight deployment with strong security assurance in power communication networks. METHODS: By integrating Graph Neural Networks (GNNs) and Bidirectional LSTM (Bi-LSTM), the model extracts three-dimensional features of topology, service, and resources to construct a 128-dimensional joint state representation. Moreover, a multi-objective reward function is designed that employs a double Q-network to mitigate value overestimation and generate a high-fidelity decision trajectory library. Through service-constrained policy distillation, the model innovatively combines KL divergence loss, a squared hard-constraint loss, and a soft-constraint L2 loss to compress the teacher model into a student model, subsequently compiled and deployed at the edge. Finally, a rule engine layer dynamically adjusts priorities for intercepting critical violations and ensures the security of the power system. RESULTS: Experimental results based on real-world power grid datasets demonstrate that our model achieves superior performance in resource efficiency, security, and edge effectiveness, effectively balancing lightweight deployment with strong security assurance in resource allocation for power communication networks. CONCLUSION: It can be seen that this method enables distributed and secure resource allocation in power communication network environments, thus providing reliable QoS guarantees for new-type power systems.

Keywords: Power Communication Networks, Reinforcement Learning, Policy Distillation, Distributed and Secure Resource Allocation

Received on 05 September 2025, accepted on 18 October 2025, published on 16 March 2026

doi: 10.4108/eetsis.11995

1. Introduction

With the rapid advancement of smart grids, the power system has transformed from a relatively static physical network into a complex cyber-physical system, driven by information technology. As its nerve center, the power communication network underpins critical services such as relay protection, wide-area measurement systems, smart meter data collection, and distribution automation. These services require extremely stringent and diverse demands on the power communication network's quality of service (QoS) and security. For instance, relay protection demands millisecond-level end-to-end latency and extremely high reliability. Conversely, the massive data collection from smart meters requires substantial bandwidth and throughput. Traditional resource allocation schemes struggle to adapt to the grid's volatile operational states and dynamic service demands, exhibiting inherent

[*] Corresponding author. Email: zhangyue199101@yeah.net

shortcomings such as sluggish response times, inefficient resource utilization, and insufficient flexibility. As a result, achieving dynamic, fine-grained network resource allocation to ensure power system security and differentiated QoS for various services has become a key challenge in building the next-generation smart grid communication architecture.

Academia and industry have recently focused on Deep Reinforcement Learning (DRL) to overcome the limitations of the traditional method [1]. By continuously interacting with the environment, DRL agents can autonomously learn and approximate optimal decision-making strategies in complex state spaces. Algorithms like Deep Deterministic Policy Gradient (DDPG) [2] and Proximal Policy Optimization (PPO) [3] have shown significant potential in simulated environments, improving network resource utilization and satisfying various constraints. However, deploying powerful DRL models from the “cloud” to the “edge” faces significant “last-mile” challenges in practical implementation, mainly shown in the following three aspects. First, there exists a conflict between computational overhead and real-time requirements. Advanced DRL algorithms, such as Soft Actor-Critic (SAC) [4] and DDPG, rely on deep neural networks with numerous parameters and high inference latency.

However, critical services within power communication networks—such as protection and control services—require decision-making within milliseconds. The expensive computational costs make it difficult to run these large models in real time on resource-constrained edge devices. Consequently, ensuring security constraints is difficult. Deep reinforcement learning (DRL) models act as significant “black boxes”, lacking interpretability and deterministic guarantees in their decision-making processes. Even when limitations are added while training, the policy can still produce non-compliant operations during deployment. Within the power industry, where security requirements are extremely strict, low-probability violations may have disastrous results. Most existing models employ penalty functions as “soft constraints” during training, not being able to provide “hard guarantees” during deployment.

Regarding the issue of inefficient knowledge transfer, knowledge distillation [5] has long been a widely accepted model compression method. However, its traditional form primarily emphasizes matching distributions of outputs without accounting for domain-specific constraints. Within the power communications domain, merely simulating the teacher model's outputs without embedding the physical rules governing grid operation and security regulations as concrete knowledge into the student model leads to high constraint violation rates in lightweight models. This results in an inability to meet the application requirements of power communications systems.

In conclusion, the primary difficulty in current research is how to effectively and securely compress the superior performance of large cloud-based DRL teacher models into lightweight models that are suitable for use on edge devices. In addition, we must ensure that the model's decision-making behavior strictly follows the hard constraints of grid security.

This study proposes a distributed resource allocation method for power communication networks based on constrained policies. Its main concept involves designing a ‘teacher-student’ collaborative evolution framework that achieves lightweight deployment while balancing performance and security. A large teacher model first learns optimal policies within a secure cloud-based simulation environment. Subsequently, its capabilities are securely transferred to edge student models by the injection of domain knowledge. The main contributions made by this research are as follows:

1) We integrate the Graph Attention Network (GAT) [6] and Bidirectional Long Short-Term Memory Network (Bi-LSTM) [7] to extract topological spatial features of the power communication network and temporal dynamic features of service flows, thereby building a thorough and low-dimensional state representation for decision-making.

2) We propose a reinforcement learning teacher model generation framework based on SAC to comprehensively suggest multiple objectives in the reward function, such as bandwidth utilization, latency, packet loss rate, and fairness, learning near-optimal resource allocation strategies in complex settings through offline training.

3) This study also develops a constrained policy distillation loss function, building on the traditional knowledge distillation loss to enforce the student model’s output and meet fundamental physical security rules through a hard constraint loss in a forward propagation. Additionally, using soft-constraint loss with backward-propagation gradients guides the student model to mimic the teacher’s priority scheduling strategies in complex service scenarios, unifying rule-based constraints and intelligent optimization.

4) We design a rule-engine-based policy verification process to guarantee that any non-compliant actions from the student model are intercepted and replaced with safe actions before output, providing a more reliable security guarantee for the model’s edge deployment.

2. Related Work

2.1 Network Resource Allocation Based on Reinforcement Learning

Network resource allocation constitutes a critical challenge within wireless communication networks, data centers, and edge computing scenarios. This issue requires the appropriate allocation of resources to users or tasks under the limitation of scarce bandwidth, power, computation, and storage, thereby maximizing system performance. In recent years, reinforcement learning (RL) has become a powerful tool for addressing dynamic network resource allocation problems. It achieves autonomous learning of optimal strategies through interaction with the environment, without requiring deterministic models. Moreover, DRL combines the strengths of deep learning and RL, enabling it to handle complex network environments and high-dimensional data, showing significant potential in resource allocation.

Considering resource allocation for cellular vehicle-to-everything (V2X) communication, Zhang et al. [8] discussed the difficulties of mode selection and resource allocation for V2X communication technologies based on cellular networks (4G LTE or 5G). They proposed a decentralized DRL algorithm based on Deep Q-Network (DQN), integrating federated learning [9] to design a dual-timescale collaborative DRL framework. This method effectively optimizes network capacity and satisfaction rate while reducing interference by minimizing transmission power, consequently lessening the burden of global channel state information and associated computational complexity. Ji et al. [10] proposed a distributed resource allocation method based on graph neural networks (GNN) and DRL to address resource allocation challenges in high-density, high-dynamic vehicle scenarios. Throughout this process, the GNN based on the Graph-SAGE framework employs dynamic graph structures, while DRL utilizes a double-depth Q-network (DDQN) [11] to interact with the environment. This approach innovatively employs graph construction methods to reduce computational complexity and fully leverages global information, thereby pioneering a novel approach to V2X network resource allocation.

In dynamic spectrum access resource allocation for 5G networks, Ramin et al. [12] addressed the issue of spectrum sharing and coexistence between secondary and primary users. The approach employs an Echo State Network (ESN) as the core of the Q-network architecture and proposes a DRL-based spectrum access policy. This method significantly improves training efficiency and sampling efficiency by updating output weights without freezing the input and recurrent weights.

In wireless networks, Amjad et al. proposed the DDQN framework [13] to optimize energy efficiency following cloud-based radio access networks. This framework decouples action selection from target Q-value generation to deliver superior returns. Compared with traditional DQN models, DDQN avoids overestimating Q-values, thereby enhancing optimization performance and energy efficiency. Harun et al. [14] designed a centralized resource allocation strategy based on DRL to address resource allocation challenges in multi-cell networks (RA). The DQN model, utilizing deep neural networks and Q-learning, effectively approximates the action-value function through a two-layer hidden configuration. This approach substantially improved data transmission rates and average utility, thereby improving the user experience.

Current research has greatly improved the efficiency of network resource allocation. However, when network topology or traffic patterns change, the generalizability of these policies proves insufficient, thereby compromising model performance. Furthermore, these models have a limited degree of integration with existing network protocols, posing challenges for realistic deployment.

2.2 Lightweight Resource Allocation Based on Knowledge Distillation

The rapid expansion in the scale and complexity of networks has greatly improved the performance of deep learning-based resource allocation models. However, owing to their considerable size and high inference latency, these models struggle to meet the demands of energy-efficient networking scenarios [15]. Knowledge distillation, as a model compression method, tackles this challenge by enabling small-scale student models to obtain knowledge from high-performance teacher models through imitation of their predictive distributions. This method substantially reduces model complexity and computational burden while maintaining efficiency, making it extensively used for constructing online-deployable network resource allocation solutions.

Studies have investigated the use of knowledge distillation in 5G networks [16], especially in the radio access network resource allocation and trading problems. Daniel et al. [17] proposed a federal learning-based DRL method, which allows for decentralized data processing for resource trading in multi-entity environments. Specifically, they designed a mutual strategy distillation scheme that distills complex mobile virtual network operator teacher strategies into infrastructure provider student models, achieving personalized and global optimal resource trading decisions. In the wireless communication domain, Ma et al. [18] used knowledge distillation to treat traditional high-performance optimization as “teacher models.” By transmitting knowledge, they improved the model’s ability to maximize the rate of all users in multi-user scenarios. This method supports different architectures of neural networks and training techniques, making it flexible and suitable for a variety of situations and resource management problems. Zhang et al. [19] proposed a privilege-guided knowledge distillation network to optimize resource allocation in edge intelligence scenarios. This knowledge distillation not only condenses intricate models for effective implementation on edge devices but also optimizes resource allocation methods to enhance system performance. Chen et al. [20] focused on the issue of common model parameter transmission between nodes and central nodes in IoT edge cloud computing. They employed knowledge distillation as an auxiliary tool to transfer complex model knowledge to resource-constrained IoT devices, enhancing device performance by optimizing computational and energy resource allocation.

Overall, distillation methods significantly reduce the model’s computational and storage overhead while maintaining network resource allocation accuracy, providing a feasible approach for online deployment. However, existing models still have some shortcomings. First, they lack adaptability in dynamic network environments. Additionally, the distillation process heavily relies on labels and intermediate representations, making it challenging to guarantee performance when high-quality outputs from the teacher model are unavailable.

3. Method

The proposed framework primarily comprises a teacher model and a student model. The teacher model is trained using RL based on the SAC algorithm and is compressed into a student model through policy distillation. Ultimately, the student model is deployed to edge devices. As shown in Figure 1, the system architecture mainly comprises four components: a multimodal perception module, a teacher model based on the SAC algorithm, a student model based on policy distillation, and a security verification module.

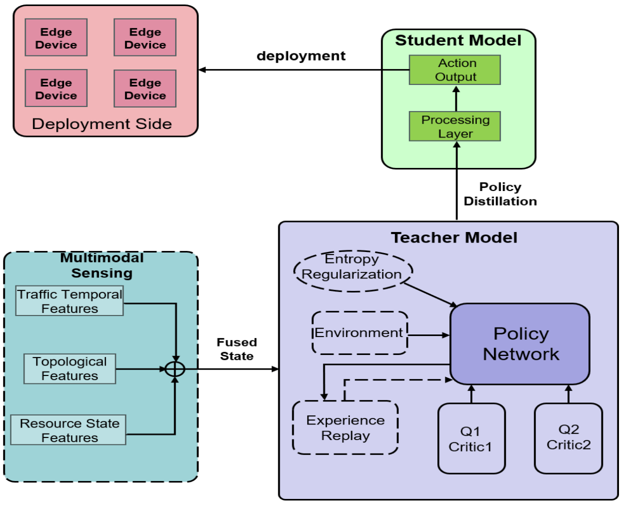

Figure 1. System Architecture Diagram

3.1 Multimodal Perception Module

The multimodal perception module is the input end of the entire method, integrating spatial topology, temporal service, and resource state data. Fusing various heterogeneous data in the power communication network provides a comprehensive and accurate network description for subsequent teacher and student models.

First, a communication network attribute graph model must be constructed to extract topological spatial features. This attribute graph model is denoted as $G = (V, E, F_v, F_e)$, where Vrepresents the set of nodes, such as substations, in the communication network; E signifies the set of edges corresponding to communication links between nodes; F_v is the node state-space vector, including metrics like CPU utilization. F_e denotes edge features, such as link bandwidth or bit error rate.

Next, a two-layer neural network is designed for neighborhood feature aggregation. We define $\hat{A}$ as the adjacency matrix with self-loops added, $\hat{D}$ as the degree matrix of $\hat{A}$, X as the original node features, and h_v as the spatial topological features. The computation process for neighborhood feature aggregation is as follows:

$$h_v = \varphi_G(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}X) \qquad (1)$$

where φ_G represents the two-layer graph neural network. Through continuous parameter updates, it obtains a 64-dimensional topological feature vector h_v.

Following this, a bidirectional LSTM is employed to extract temporal features for the historical traffic data X_t, obtaining the forward final state $\vec{h}_{forward}$ and the backward initial state $\overleftarrow{h}_{backward}$. The bidirectional states are combined to form the temporal feature vector h_t via vector concatenation.

The standardized resource state feature vector h_s is obtained by monitoring the computing resources of the node—CPU utilization μ_{cpu}, CPU volatility σ_{cpu}, average memory usage μ_{men} , memory usage P95 quantile mem $_{p95}$, bit error rate (log-normalized) (BER), average latency τ_{avg} , latency volatility $\Delta\tau$, and signal-to-noise ratio SNR, standardizing them using a 24-hour sliding mean/standard deviation.

The obtained topological, temporal, and resource state features are combined through concatenation and compressed using a single-layer linear transformation to yield the fused feature vector h_{out} to construct a multidimensional state space.

3.2 Teacher model based on the SAC algorithm

RL is a computational approach that achieves goals through interactions between a machine and its environment, emphasizing actions that maximize rewards. This scheme employs the SAC algorithm for reinforcement learning on the teacher side, with its architecture shown in Figure 2.

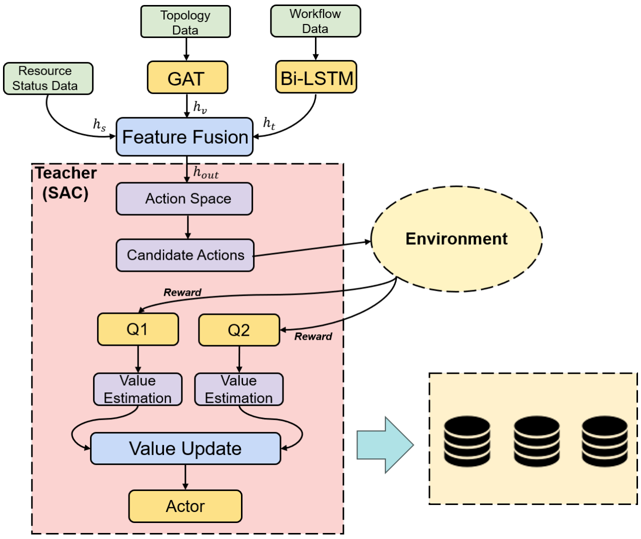

Figure 2. Teacher Model Architecture

Due to its benefits for exploration, training stability, and optimization in complex environments, this model can generate efficient and reliable resource allocation strategies in dynamic and complex power communication environments, providing high-quality decision trajectories for subsequent policy distillation.

The environment must first be modeled to train the teacher model using RL. The state space uses the fused feature vector h_{out} from the multimodal perception layer. The defined action space includes continuous actions, such as bandwidth adjustments, and discrete actions, such as priority adjustments for business agents. In power communication resource allocation, multiple objectives must be optimized simultaneously, as a single reward function cannot fully capture the complex demands of the power communication network. Therefore, a reward function is designed by weighting and combining multiple sub-reward terms to balance different objectives, including positive rewards and negative penalties. The specific definition of the reward function is as follows:

$$R = \omega_1 \cdot U_{bandwidth} - \omega_2 \cdot \triangle_{delay} - \omega_3 \cdot I_{violate} + \omega_4 \cdot F_{fairness} \quad (2)$$

where $U_{bandwidth}$ is the network-wide bandwidth utilization; $\triangle_{delay}$ is the excess latency for critical services; $I_{violate}$ signifies the constraint violation indicator function. $F_{fairness}$ is the resource allocation fairness index based on the Gini coefficient, and $\omega_1 - \omega_4$ represents the weighting coefficients.

Compared to traditional RL, the SAC algorithm avoids premature convergence to local optima through its entropy regularization mechanism [21], making it more suitable for dynamic and complex power communication network scenarios. The SAC algorithm primarily uses the Actor-Critic framework, consisting of an Actor network that generates action policies and a Critic network that evaluates action values. As a result, we use a three-layer MLP network with LeakyReLU activation functions as the Actor, employing a reparameterization trick to generate continuous actions. The Critic also uses a three-layer MLP to output value estimates. Notably, our SAC algorithm uses a dual-Critic architecture to mitigate Q-value overestimation by employing the minimum Q-value when calculating the target Q-value, thereby reducing overestimation bias. Specifically, the Critic network is divided into an online Critic network that directly participates in training and updates, and a target Critic network that slowly integrates parameters from the online Critic through soft updates. The specific update method is given by

$$\theta_{target} \leftarrow \tau\theta_{online} + (1-\tau)\theta_{target} \quad (3)$$

where θ_{online} represents the online Critic network parameter, θ_{target} is the target Critic network parameter, and τ is the soft update parameter.

During training, the Actor and Critic networks, along with the experience replay buffer used to store interaction data between the agent and the environment, are first initialized. After initializing the Actor network's policy, actions are chosen based on the given state, and the state and reward returned by the environment are stored in the experience replay buffer. Next, a batch of data is randomly sampled from the experience replay buffer to compute the target Q-value as the online Critic network learning target. The target Q-value is defined as follows:

$$y = r + \gamma \cdot (1-d) \cdot \min\left(Target\ Q1(s', a'), Target\ Q2(s', a')\right) + \alpha \cdot H(\pi) \quad (4)$$

where a' is the action sampled from the Actor network, and $H(\pi)$ is the entropy of the policy at the next state s'. By incorporating an entropy regularization term, the agent is encouraged to maximize rewards while avoiding convergence to local optima. Additionally, using the target Critic network's minimum output to calculate the target Q-value provides a more conservative estimate of future returns, effectively reducing the overestimation risks.

After calculating the target Q-value, the parameters are updated based on the defined Critic mean squared error $\mathcal{L}_{Critic} = \mathbb{E}[(Q(s_t, a_t) - y)^2]$, followed by a soft update strategy to update the corresponding parameters of the target Critic network. Subsequently, the Actor network parameters are updated based on the Actor loss defined as follows:

$$\mathcal{L}_{Actor} = -\mathbb{E}[Q_\varphi(s_t, a_t) - \alpha \log \pi(a_t|s_t)] \quad (5)$$

where $Q_\varphi(s_t, a_t)$ is the expected cumulative return output by the online Critic network; $\pi(a_t|s_t)$ is the policy function output by the Actor network, and α is the entropy coefficient.

This loss function encourages selecting actions with high expected returns while incorporating an entropy regularization term to promote a more random action distribution. As a result, the policy balances exploration and exploitation of known high-reward actions, thereby avoiding the premature convergence to local optima often encountered in conventional reinforcement learning algorithms. The final teacher model can achieve maximum rewards under various operating conditions through continuous iteration.

After completing the teacher model's training, a decision trajectory library must be generated. The decision trajectory library stores optimal decision patterns across various operating conditions, systematically summarizing the diverse decisions generated during the teacher model's training, thus preventing knowledge loss during the subsequent distillation of the student model. Specifically, the corresponding state and action sequences and constraint satisfaction flags are recorded after the teacher model completes training and enters the testing phase. Trajectories with a violation rate exceeding 20% are filtered out, retaining only the optimal strategy samples. This process ensures that the trajectories stored in the decision trajectory library satisfy the standard of power communication.

3.3 Student Model Based on Policy Distillation

Although models like DRL excel in dynamic resource allocation, their numerous parameters and high computational demands make direct deployment on resource-constrained edge devices challenging. Policy distillation effectively bridges this gap by compressing the complex model and transferring its knowledge to a lightweight student model, significantly reducing computational complexity and resource requirements while maintaining performance. In addition, a lightweight student model reduces inference latency, fulfilling the stringent millisecond-level response and hard business-rule constraints of power communication networks. Through supervised learning and constraint injection, the student model reduces the trial-and-error cost and mitigates security risks. To this end, we propose a specific policy distillation scheme that designs a student model with a 3-layer MLP structure and incorporates continuous and discrete action heads at the output layer. These heads produce continuous actions, such as communication and resource allocation, and discrete decisions, such as business priority adjustments, to achieve efficient, business-compliant resource allocation. A constraint-aware layer is also introduced to incorporate hard business-rule constraints, thereby ensuring the security and compliance of decisions for efficient and business-compliant resource allocation. Figure 3 shows the student-side model architecture.

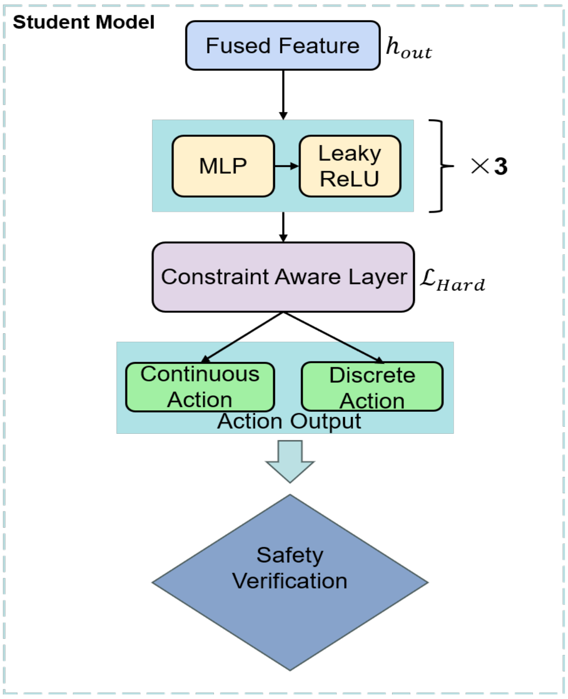

Figure 3. Student-Side Model Architecture

To guide the student model's learning of the teacher model's policy, we designed a composite loss function comprising two parts: the base distillation loss and the constraint loss, with their influences balanced by weights. The base distillation loss aims to enable the student model to learn the teacher model's output distribution under given states. Specifically, the KL divergence is used to measure the difference between the two policy distributions, which is defined as follows:

$$\mathcal{L}_{KL} = D_{KL}(\pi_T(a|s)||\pi_S(a|s)) \qquad (6)$$

where $\pi_T(a|s)$ represents the probability distribution of the teacher model over action a, given state s; and $\pi_S(a|s)$ represents the corresponding policy distribution of the student model. By minimizing such losses, the student model gradually approximates the teacher model's strategy, which leads to superior guidance. To avoid overfitting, we introduce constraint losses to promote the generation of diverse policies while improving the model's exploration capabilities. This ensures the student model satisfies other critical performance metrics, leading to more robust learning. Constraint losses comprise hard and soft constraints. These constraints limit the action output range of the student model, ensuring it remains within threshold boundaries and thus avoids straying from safe or reasonable policy regions. Specifically, the hard constraint loss function is defined as:

$$\mathcal{L}_{hard} = \sum_{c=1} \max\,(0, \hat{a}_S - a_{threshold})^2 \qquad (7)$$

where $\hat{a}_S$ represents the output action of the student model, and $a_{threshold}$ is the action threshold. Once the student model's actions exceed the threshold, a penalty term is activated to impose constraints. Furthermore, the soft constraint introduces a continuous penalty to encourage the strategy distribution of the student model to align with that of the teacher model, thereby achieving smooth and robust generalization. The loss function for soft constraints is defined as:

$$\mathcal{L}_{soft} = \|V \cdot (\pi_S(a|s) - \pi_t(a|s))\|_2 \qquad (8)$$

where V is the constraint violation indicator vector, used to adjust the severity of the penalty under different circumstances. Soft constraints provide a continuous penalty term, enabling the student model to simulate the teacher's decision-making while maintaining flexibility in bias. This enhances the policy's rationality and generalization capability. Ultimately, the total loss function comprises a base distillation loss and a constraint loss. Specifically, the total loss function is defined as follows:

$$\mathcal{L}_{total} = \alpha\mathcal{L}_{KL} + \beta\mathcal{L}_{hard} + \gamma\mathcal{L}_{soft} \qquad (9)$$

Where α, β, and γ are adjusted, balancing the impact of the base distillation loss and the constraint loss. We adopt a two-stage training method to leverage the teacher model's guidance and enhance the student model's generalization performance. In particular, the first stage is the pre-training phase, where only the KL divergence loss $\mathcal{L}_{KL}$ is used to align the student model with the teacher model, enabling the student model to quickly grasp the task's core features. This stage provides the student model with well-initialized parameters, establishing a foundation for the subsequent optimization process. The second stage is the fine-tuning phase, where the constraint losses $\mathcal{L}_{hard}$ and $\mathcal{L}_{soft}$ are introduced. Whilst maintaining imitation of the teacher's strategy, the student model autonomously explores and optimizes its strategy by incorporating feedback from the environment, thus improving the strategy's robustness and adaptability. Through persistent engagement with the surroundings, the strategy undergoes continual refinement, further improving performance. This two-stage training strategy combines the advantages of imitative learning and autonomous optimization, significantly improving training but also effectiveness, in addition to the performance and generalization capabilities of the student model's strategy.

3.4 Security Verification Module

Given the strict security requirements for resource allocation processes within power communication networks, we have introduced a security verification mechanism based on operational standards. This guarantees the system effectively intercepts and prevents non-compliant operations, thereby protecting the security and reliability of power communications. At the execution layer, real-time security validation modules have been specifically designed in accordance with predefined power communication standards (such as IEC 61850 and IEEE C37.94). This ensures all resource allocation and control commands comply with relevant industry rules and operational requirements. Specifically, the verification module compares generated control commands against the local business rules repository and power communication standards. When the module detects a command violating resource allocation rules or standards, the system immediately triggers security interception measures to stop the execution of commands. Meanwhile, non-compliant commands are automatically replaced with the latest valid operations or predefined security policies. This ensures system continuity and operational stability, making certain that the rational scheduling of critical resources and system security is maintained. This mechanism improves the security protection capabilities of power communication systems. It effectively mitigates potential risk resulting from non-compliant operations, providing a robust safeguard for the secure operation of power communication networks.

4. Experimental Results and Analysis

4.1 Dataset Description

We conducted experiments using a publicly available dataset to validate the effectiveness of the constrained policy refinement structure in simulating provincial-level power grid communication networks. Within the network topology, we employed a 50-node hybrid structure (comprising three optical transport ring networks and twelve branches) with 200 communication links (bandwidth ranging from 100 Mbps to 10 Gbps). This topology changes dynamically under fault conditions, such as link disconnections. Based on the power data service scenario and its performance requirements, the selected service traffic types are shown in Table 1. With respect to resource metrics, six months of operational data (with a granularity of 1 second) were utilized, and network performance monitoring was conducted across multiple indicators, including CPU/memory utilization, link bandwidth utilization, packet loss rate, and end-to-end latency.

Table 1. Power System Service Traffic and Its Performance Metrics in the Dataset

Service Type	Example	Proportion	Latency Requirement	Bandwidth Requirement
Protection Service	Relay Protection, Differential Protection	15%	≤4 ms	≥2 Mbps
Control Service	SCADA, Automatic Generation Control	25%	≤50 ms	≥1 Mbps
Acquisition Service	PMU, Smart Meter	60%	≤1 s	≥0.5 Mbps

4.2 Evaluation Indicators

Three metrics were used in the experiments to comprehensively assess the effectiveness of the proposed scheme for resource allocation in the power communication network.

First, in terms of resource efficiency, the primary focus was on two metrics: bandwidth utilization and task completion rate. Bandwidth utilization is the average ratio of used bandwidth to total available bandwidth across all network links within a specific period, reflecting the overall efficiency of network resource usage. Its calculation method is as follows:

$$U_{bw} = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{\sum_{l=1}^{L} B_{used}^{(l)}(t)}{\sum_{l=1}^{L} B_{total}^{(l)}}\right)\times 100\% \tag{10}$$

where T is the total number of time steps (or sampling moments); t is the index of the current time step; L represents the total number of communication links in the network; and l indicates the link index. $B_{used}^{(l)}(t)$ denotes the actual bandwidth used on the l-th link at time step t; and $B_{total}^{(l)}$ is the total available bandwidth of the l-th link, which is a fixed value.

The Task Completion Rate (TCR) is the ratio of service flows that complete resource allocation and meet their minimum quality-of-service requirements to the total number of service flows. Its calculation method is as follows:

$$TCR = \frac{N_{completed}}{N_{total}}\times 100\% \tag{11}$$

where $N_{completed}$ is the number of service flows completed within the evaluation period. Here, “completed” must simultaneously satisfy: 1) the allocated bandwidth B_{min} the minimum required bandwidth. 2) The end-to-end latency τ_{max} the maximum tolerable latency. N_{total} represents the total number of service flow requests within the evaluation period.

Second, to evaluate the performance of the proposed scheme in ensuring Quality of Service (QoS), the experiments focused on two QoS metrics for power communication services: the Critical Service Violation Rate (CSVR) and the 95th percentile latency of protection services. Among these, CSVR is a core metric for assessing system security, representing the proportion of critical service flows (e.g., relay protection, stability control) whose QoS parameters violate their preset thresholds. The specific calculation method is as follows:

$$CSVR = \frac{1}{N_{critical}} \sum_{i=1}^{N_{critical}} \text{II}(\tau_i > \tau_{\max}^{(i)} ORB_i < B_{\min}^{(i)}) \tag{12}$$

where $N_{critical}$ Represents the total number of critical service flows evaluated; i is the index of the vital service flow. τ_i denotes the actual measured end-to-end latency of the i-th critical service flow; and $\tau_{max}^{(i)}$ is the maximum allowable latency required by the i-th critical service flow. B_i represents the actual allocated bandwidth for the i-th critical service flow, and $B_{min}^{(i)}$ indicates the minimum guaranteed bandwidth required by the i-th critical service flow. II is an indicator function that returns 1 if the condition in parentheses is true, and 0 otherwise.

To verify the performance of the proposed scheme in model light weighting, the experiments conducted a comparative analysis of edge deployment efficiency by analyzing the inference latency of the edge model. Here, inference latency is the average time from when the student model receives input states until it outputs decision actions.

4.3 Analysis of Experimental Results

4.3.1 Overall Performance Comparison

Comparative experiments were conducted on a 50-node power communication network test platform over 24 h to comprehensively evaluate the overall performance of the proposed lightweight resource allocation method based on constraint policy distillation. The performance was compared with four mainstream methods: the traditional telecom-grade solution (MPLS-TE)[22], a mainstream deep reinforcement learning algorithm (DDPG) [2], and the original distillation method without constraint processing (Vanilla Distill) [5].

The experimental results for bandwidth utilization are shown in Figure 4. The results indicate that MPLS-TE, which employs a static strategy, cannot adapt to dynamic fluctuations in traffic demand due to its static bandwidth reservation approach, leading to significant resource idleness during low-load periods and an inability to fully utilize resources during high-load periods due to the lack of flexibility in the reservation mechanism, failing to meet the smart grid's demand for efficient resource utilization. Within the DDPG algorithm, the DRL agent dynamically learns resource allocation strategies by interacting with the environment, flexible scheduling of idle resources to high-demand services, and significant improvement in resource utilization efficiency. In the original distillation strategy (Vanilla Distill), due to its inability to fully inherit the optimization strategies of the teacher model, the student model's utilization is marginally lower than that of its teacher (DDPG), suggesting some performance degradation during model compression.

In comparison, the proposed model overcomes the performance degradation issue in traditional knowledge distillation and surpasses the teacher model in specific scenarios. By introducing stringent security constraints, this loss function restricts the student model's exploration of ineffective or non-compliant action spaces, thereby focusing its learning process on efficient and compliant strategies. This approach enhances decision reliability and overall performance.

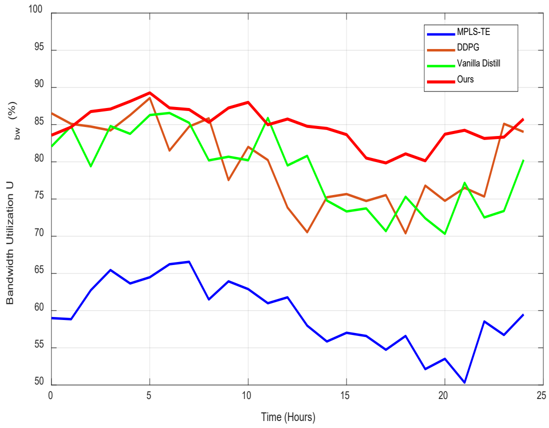

Figure 4. Bandwidth Utilization Over Time (24 h)

Regarding security, the experiment analyzed the CSVR of different schemes over 24 h, with the results shown in Figure 5, indicating that the MPLS-TE scheme has a relatively low violation rate; however, its security stems from a conservative over-provisioning strategy rather than from intelligent decision-making. When the network load exceeds its static capacity planning, the violation rate rises continuously. In the DDPG scheme, to compensate for the lack of exploration in DRL, the agent enhances overall performance by sacrificing a small portion of critical services. Therefore, this approach carries fatal risks and fails to meet the requirements of real-world power production control scenarios. In the original distillation scheme (Vanilla Distill), the inability of the student model to mitigate unsafe behaviors from the teacher model gave rise to security risks. In contrast, our proposed method employs hard constraint losses to eliminate actions violating predetermined rules, while the introduced soft constraint losses further prevent high-risk operations near the constraint boundaries.

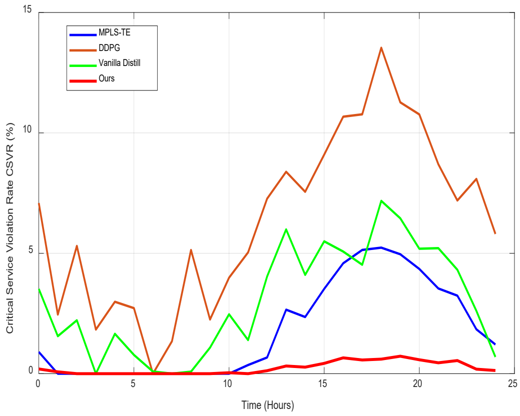

Figure 5. Constraint Violation Rate Over Time (24 h)

Figure 6 displays the inference latency results. The substantial volume and high computational demands of DDPG models result in extremely high inference latency (>18 milliseconds). Consequently, reliance on cloud resources is necessary, yet this approach fails to meet the real-time, localized processing requirements of power services and is unsuitable for edge deployment. Whilst the original distillation method (Vanilla Distill) achieves edge deployment (model size ~2.3 MB, latency ~4.2 ms), its performance and security remain influenced by the teacher model. In contrast, our approach features extremely low inference latency and a small parameter size, enabling direct deployment on edge devices such as the Huawei Atlas 500. This framework eliminates cloud dependency, fully meeting the stringent real-time requirements of power services.

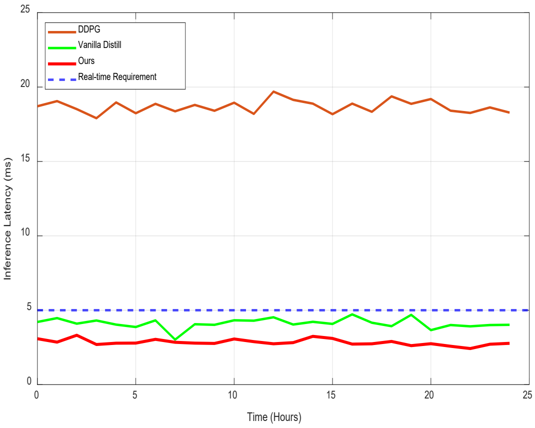

Figure 6. Inference Latency Over Time (24 h)

To comprehensively evaluate each method's performance under extreme conditions, our experiments analyzed over 4,000 latency measurements (1,000 per method), comparing central tendency and tail latency. The experimental results are illustrated in the block diagram of Figure 7. The findings indicate that the DDPG algorithm performed most poorly, exhibiting a median latency of approximately 5.1 milliseconds—exceeding the security threshold—and demonstrating exceptionally high latency under high-load scenarios. In the Vanilla Distill approach, despite the smaller student model, it fully replicated the decision behavior of the teacher model (DDPG), including its unstable strategies. It fundamentally fails to address the issue of high tail latency. The MPLS-TE performance, configured in fixed mode, is unaffected by dynamic network changes, resulting in exceptional stability. However, its static rules prevent fine-grained optimization, leading to suboptimal baseline latency. The proposed constraint distillation method compresses the model and reshapes its strategy. The hard constraint loss eliminates non-compliant actions that could lead to high latency, while the soft constraint loss guides the model to learn smooth, efficient scheduling strategies. This results in fast, highly stable, and reliable decisions, fundamentally eliminating high-tail latency.

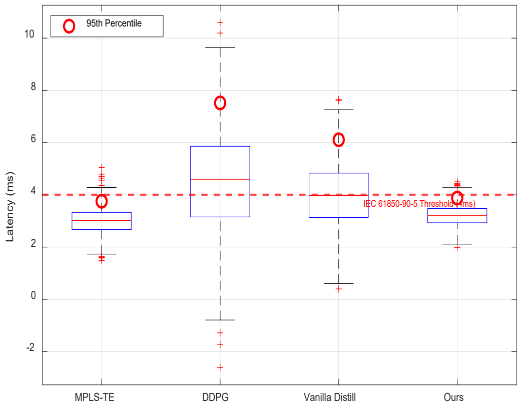

Figure 7. Distribution and 95th Percentile of Latency

4.3.2 Ablation Study

1) Distillation Strategy Ablation Study

A distillation strategy ablation study was designed to evaluate each core component's contribution in the proposed constraint policy distillation framework. Four experimental groups were constructed to keep the teacher model, student model structure, and all training hyperparameters constant while varying only the combination of loss functions used in the strategy distillation, as detailed in Table 2.

In this ablation study, two sets of experiments were designed to compare and analyze two performance metrics: bandwidth utilization and CSVR. The experimental results are shown in Figure 8.

Table 2. Ablation Study Design

Model Variant	Loss Function	Design Purpose
KL only	$L_{total}=L_{KL}$	As a baseline, validate the effectiveness of traditional knowledge distillation methods in this field.
KL + Hard	$L_{total}=L_{KL}+\beta L_{hard}$	Quantify the individual contribution of hard constraint loss in injecting physical security rules and enforcing a reduction in violation rates.
KL + Soft	$L_{total}=L_{KL}+\gamma L_{soft}$	Quantify the individual contribution of soft constraint loss in mimicking the teacher's priority strategy and improving resource utilization.
Ours (Full)	$L_{total}=\alpha L_{KL}+\beta L_{hard}+\gamma L_{soft}$	The complete loss function proposed in this paper is used to verify the overall performance and the synergistic effects of all components combined.

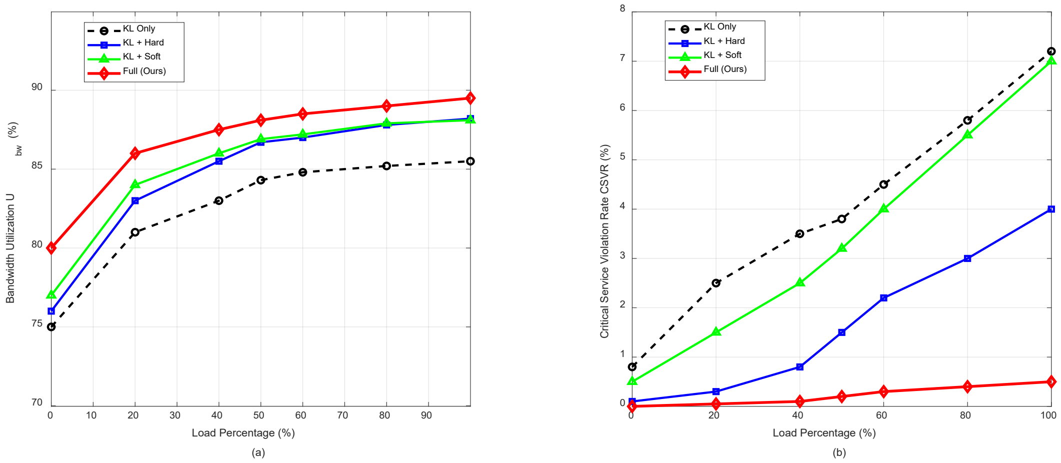

Figure 8. Load Percentage and Bandwidth Utilization Trends Under Varying Loads

In terms of bandwidth utilization, as shown in Figure 6(a), the KL Only model exhibits lower utilization across the whole load range than other configurations, indicating that traditional distillation methods inevitably degrade performance during model compression and fail to retain the teacher model's optimization capabilities. The KL+Hard model shows improved performance, but the gain is limited. While hard constraints prevent unsafe actions, they do not actively guide the model toward learning better allocation strategies; performance improvements mainly stem from avoiding invalid exploration. The KL+Soft model demonstrates significant performance enhancement, indicating that soft constraints effectively guide the model to optimize resource allocation by mimicking the teacher model's priority scheduling strategy, proving its critical role in improving efficiency. The complete method maintains the highest utilization under all load conditions, achieving a peak rate of 89.5%. Its advantage is particularly pronounced in high-load scenarios, suggesting a synergistic effect between hard and soft constraints: hard constraints ensure exploration within a safe space. In contrast, soft constraints guide the model toward an optimal solution within that space.

Regarding the CSVR, as shown in Figure 6(b), the KL Only scheme exhibits the highest violation rate, sharply increasing after exceeding 50% critical load. This result demonstrates that simply mimicking the teacher model's output distribution fully inherits its security flaws, resulting in an unsafe distilled model. The KL+Soft scheme shows a slight reduction in the violation rate, but it remains high. The KL+Hard method significantly improves security, reducing the violation rate to below 4.0%, clearly demonstrating the core role of hard-constraint loss in preventing actions that violate predefined service rules (e.g., minimum bandwidth), thereby providing a foundational guarantee for model security. The complete method consistently controls the violation rate below 0.5% across the whole load range, with its performance curve closely hugging the horizontal axis and remaining extremely stable. This finding indicates that adding soft constraints further optimizes the hard-constraint strategy, avoiding oscillations near constraint boundaries and ensuring a plan that is both safe and smooth.

2) Security Verification Mechanism Ablation Study

The purpose of this experiment is to quantitatively evaluate the independent contribution of the security verification mechanism in the final deployed system through controlled comparisons. Two schemes were designed: Group A deployed a lightweight student model without a security verification mechanism, with all output actions directly sent to the actuator. Group B deployed a complete system equipped with a security verification mechanism, in which the student model's outputs were verified and potentially replaced. To ensure fairness, the same student model trained in Section 3.3 was used for both groups. Over a 24-h period, both Group A and Group B systems were subjected to identical traffic loads and disturbances, and the actual number of critical service violations was recorded, with each group performing 86.4 million decision cycles. Normal fluctuating traffic was injected to simulate fundamental network uncertainty, while abnormal surge traffic was used to simulate sudden failures, network attacks, or extreme scenarios unseen by the model. The experimental results, including the numbers of actual and catastrophic violations and the average inference latency for both groups, are shown in Table 3.

Table 3. Results of the Security Verification Mechanism Ablation Study

Experimental Group	Number of Actual Violations	Number of Catastrophic Violations	Average Inference Latency (ms)
A	17,281	42	2.8
B	0	0	2.888

The experimental results show that, under identical model and testing conditions, removing the security verification mechanism (Group A) resulted in 17,281 actual violations, averaging 720 violations per hour. This outcome demonstrates that even a highly accurate, lightweight student model incurs non-negligible probabilistic risks in its outputs when faced with real-world complexity and uncertainty. Additionally, this includes 42 catastrophic violations that could trigger cascading failures in the power system. In contrast, our solution mitigates such risks by introducing a security verification mechanism. Notably, this mechanism introduces only 0.08 milliseconds of additional latency, indicating it imposes no significant performance overhead.

Experiments demonstrate that under identical models and testing conditions, disabling the security verification mechanism (Group A) resulted in 17,281 actual violations, averaging 720 violations per hour. This indicates that even high-precision, lightweight student models exhibit non-negligible risks in their outputs when confronted with the complexity and uncertainty of the real world. Furthermore, this included 42 catastrophic violations, each potentially triggering cascading failures in the power system, resulting in significant economic losses and societal impacts. In contrast, by introducing the security verification mechanism, our approach eliminated such risks, reducing violations to zero. Moreover, the security verification mechanism introduced only 0.08 milliseconds of additional latency, indicating negligible delay overhead.

5. Conclusion

By offering a novel framework based on Constraint Policy Distillation, this study tackles the fundamental problem of balancing lightweight deployment with robust security guarantees in power communication network resource allocation. First, the multimodal state encoder based on GNN and Bi-LSTM provides an accurate data foundation for intelligent decision-making by precisely representing grid topology, service timing, and resource states. Second, the teacher model trained with the SAC algorithm learns near-optimal strategies under complex conditions to generate a high-quality decision trajectory library. By integrating KL divergence loss, hard-constrained squared loss, and soft-constrained L2 loss, the constrained distillation loss function resolves the trade-off between security and performance inherent in traditional distillation methods. Finally, the edge-deployed security verification mechanism provides reliable safeguards for the system. Experimental results demonstrate that our approach significantly outperforms mainstream benchmarks in bandwidth utilization, violation rate, and inference latency, proving its comprehensive advantages.

In the future, we will explore distillation architectures, online learning of dynamic rule bases, and the deep integration of digital twin systems to further accelerate the rapid adaptation of this technology to industrial application scenarios.

Author Contribution

Conceptualization, Y.Z., Z.L., and S.C.; methodology, Y.Z., Z.L., and G.H.; software, Y.Z. and S.C.; validation, G.H., P.L., and R.W.; formal analysis, Y.Z., Z.L., and S.C.; investigation, Y.Z. and Z.L.; resources, G.H.; data curation, S.C.; writing—original draft preparation, Y.Z., Z.L. and S.C.; writing—review and editing, G.H., P.L., and R.W.; visualization, S.C.; supervision, G.H. and P.L.; project administration, R.W.; funding acquisition, R.W. All authors have read and agreed to the published version of this manuscript.

Acknowledgements

This research was funded by the Science and Technology Project of Inner Mongolia Power (Group) Co., Ltd. (No.2025-4-12).

Data Availability Statement

The data used in this study are available from the corresponding author upon reasonable request.

Ethical Approval

Not applicable.

Competing Interest

The authors have no relevant financial or non-financial interests to disclose.

References

[1] He H, Meng X, Wang Y, et al. Deep reinforcement learning based energy management strategies for electrified vehicles: Recent advances and perspectives. Renewable and Sustainable Energy Reviews. 2024; 192: 114248.

[2] Zhao H, Sun W, Ni Y, et al. Deep deterministic policy gradient-based rate maximization for RIS-UAV-assisted vehicular communication networks. IEEE Transactions on Intelligent Transportation Systems. 2024; 25(11):15732-15744.

[3] Luo L, Yan X. Scheduling of stochastic distributed hybrid flow-shop by hybrid estimation of distribution algorithm and proximal policy optimization. Expert Systems with Applications. 2025; 271: 126523.

[4] Rahmani A M, Haider A, Moghaddasi K, et al. Self-learning adaptive power management scheme for energy-efficient IoT-MEC systems using soft actor-critic algorithm. Internet of Things, 2025; 31: 101587.

[5] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531, 2015.

[6] Vrahatis AG, Lazaros K, Kotsiantis S. Graph attention networks: a comprehensive review of methods and applications. Future Internet, 2024; 16(9): 318.

[7] Rathi M, Gomathy C. Smart agriculture resource allocation and energy optimization using bidirectional long short-term memory with ant colony optimization (Bi-LSTM–ACO). Frontiers in Communications and Networks, 2025; 6: 1587402.

[8] Zhang X, Peng M, Yan S, et al. Deep-reinforcement-learning-based mode selection and resource allocation for cellular V2X communications. IEEE Internet of Things Journal. 2019; 7(7): 6380-6391.

[9] Liu S, Yu G, Wen D, et al. Communication and energy efficient decentralized learning over D2D networks. IEEE Transactions on Wireless Communications. 2023; 22(12): 9549-9563.

[10] Ji M, Wu Q, Fan P, et al. Graph neural networks and deep reinforcement learning based resource allocation for v2x communications. IEEE Internet of Things Journal. 2024; 12(4): 3613-3628.

[11] Van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double q-learning. Proceedings of the AAAI conference on artificial intelligence, Phoenix, AZ, USA, 12–17 February 2016; pp. 2094-2100

[12] Safavinejad R, Chang H, Liu L. Deep reinforcement learning for dynamic spectrum access: Convergence analysis and system design. IEEE Transactions on Wireless Communications. 2024; 23(12): 18888-18902.

[13] Iqbal A, Tham M L, Chang Y C. Double deep Q-network-based energy-efficient resource allocation in cloud radio access network. IEEE access, 2021; 9: 20440-20449.

[14] Rashid HU, Jeong SH. Resource allocation in multi-cell networks: A deep reinforcement learning approach. 2023 International Conference on Information and Communication Technology Convergence. IEEE, Jeju Island, Korea, 11-13 October 2023; pp. 793-795.

[15] Hussain F, Hassan S A, Hussain R, et al. Machine learning for resource management in cellular and IoT networks: Potentials, current solutions, and open challenges. IEEE communications surveys & tutorials. 2020; 22(2): 1251-1275.

[16] Ayepah-Mensah D, Sun G, Owusu Boateng G, et al. Federated Policy Distillation for Digital Twin-Enabled Intelligent Resource Trading in 5G Network Slicing. IEEE Transactions on Network and Service Management. 2025; 22(1): 361-379.

[17] Mensah D A, Sun G, Boateng G O, et al. Federated Policy Distillation for Digital Twin-Enabled Intelligent Resource Trading in 5G Network Slicing. IEEE Transactions on Network and Service Management. 2025; 22(1):361-379.

[18] Ma L, Cheng N, Wang X, et al. Distilling knowledge from resource management algorithms to neural networks: A unified training assistance approach. 2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall), IEEE, Hong Kong, 10-13 October 2023; pp. 1-5.

[19] Zhang Q, Wang J, Shen Y, et al. Privilege-guided knowledge distillation for edge deployment in excavator activity recognition. Automation in Construction, 2024; 166: 105688.

[20] Chen Y, Wang Z, Cai H, et al. Federated Knowledge Distillation using Hierarchical Reinforcement Learning in Resource-Constrained IoT Edge-Cloud Computing Environments. IEEE Transactions on Mobile Computing, 2025; 7:1-15.

[21] Mao T, Zhu J, Zhang M, et al. A Decentralized Actor–Critic Algorithm With Entropy Regularization and Its Finite-Time Analysis. IEEE Transactions on Neural Networks and Learning Systems. 2025; 36(10):19423-19436.

[22] Ihle F, Menth M. MPLS Network Actions: Technological overview and P4-based implementation on a high-speed switching ASIC. IEEE Open Journal of the Communications Society. 2025; 6:3480-3501.