DOI: 10.19783/j.cnki.pspc.250406

配电网中基于混合 DRL 的任务卸载与多资源协同调度优化方法

周 雅[1,2]，王 乾[2]，方如举[1]

(1.许昌学院，河南 许昌 461000；2.华北水利水电大学，河南 郑州 450045)

摘要：针对配电网在数字化、分布式和智能化演进过程中面临的“计算-通信-能源”多资源协同调度与任务卸载导致的时延-能耗联合最优化问题，构建了涵盖本地终端、边缘服务器与云端的数据驱动三层协同计算模型。该模型以加权时延-能耗-公平指标函数为优化目标，综合刻画无线信道条件、传输速率和 CPU 频率等关键因素，从而量化多资源协同对系统性能的影响。为应对离散卸载决策与连续带宽/计算/能量分配构成的混合动作空间挑战，提出混合深度强化学习(hybrid deep reinforcement learning, HDRL)框架。上层采用双重深度 Q 网络(double deep Q-network, DDQN)进行卸载动作选择，下层利用深度确定性策略梯度(deep deterministic policy gradient, DDPG)实现连续资源调度，并设计改进优先级经验回放机制(improved prioritized experience replay, IPER)提高样本利用率与收敛速度。仿真结果表明，与纯本地计算、纯边缘计算、随机卸载、遗传算法(genetic algorithms, GA)和不含 IPER 的 DDQN+DDPG 方法相比，所提 HDRL 算法在多场景下显著降低了系统平均时延与总能耗，同时，能在用户规模扩大时依旧能维持高公平性，表现出最佳的扩展鲁棒性，提升了任务完成率与算法稳健性，为配电网多资源协同优化提供了可行、高效的解决方案。

关键词：边缘计算；任务卸载；资源分配；配电网；深度强化学习

Hybrid DRL-based task offloading and multi-resource coordinated scheduling optimization method in distribution networks

ZHOU Ya[1,2], WANG Qian[2], FANG Ruju[1]

(1. Xuchang University, Xuchang 461000, China; 2. North China University of Water Resources and Electric Power, Zhengzhou 450045, China)

Abstract: To address the joint latency-energy optimization problem arising from task offloading and coordinated scheduling of “computation-communication-energy” multiple resources during the digitalization, decentralization, and intelligent evolution of distribution networks, a data-driven three-layer collaborative computing model encompassing local terminals, edge servers, and the cloud is developed. With a weighted delay-energy-fairness objective function, the model comprehensively characterizes key factors such as wireless channel conditions, transmission rates, and CPU frequencies, thereby quantifying the impact of multi-resource coordination on system performance. To tackle the challenge of a hybrid action space composed of discrete offloading decisions and continuous bandwidth, computation, and energy allocation, a hybrid deep reinforcement learning (HDRL) framework is proposed. In this framework, a double deep Q-network (DDQN) is employed at the upper layer to select offloading actions, while a deep deterministic policy gradient (DDPG) algorithm is used at the lower layer for continuous resource scheduling. An improved prioritized experience replay (IPER) mechanism is further designed to enhance sample utilization efficiency and convergence speed. Simulation results demonstrate that, compared with pure local computing, pure edge computing, random offloading, genetic algorithms (GA), and the DDQN+DDPG method without IPER, the proposed HDRL approach significantly reduces average system delay and total energy consumption across multiple scenarios. Moreover, it maintains high fairness as the number of users increases, exhibiting superior scalability robustness, improved task completion rates, and enhanced algorithm stability. The proposed method thus provides a feasible and efficient solution for multi-resource coordinated optimization in distribution networks.

This work is supported by the National Natural Science Foundation of China (No. 62103349).

Key words: edge computing; task offloading; resource allocation; distribution network; deep reinforcement learning

基金项目：国家自然科学基金项目资助(62103349)；河南省科技攻关项目资助(232102210104)；河南省研究生联合培养基地项目资助(YJS2024JD38)

0 引言

随着电力系统向数字化、分布式化和智能化方向转型，可再生能源、储能装置和电动汽车等新型负荷的大规模接入，使得配电网数据量骤增、调度复杂性加剧以及节点电压波动风险提升[1-2]。为满足对实时性和可靠性的双重要求，多接入边缘计算(multi-access edge computing, MEC)。因其就近处理、低时延和高带宽利用率等优势，正被广泛应用于配电网的故障检测、负荷预测和应急响应等场景[3-4]。通过在配电台区或用户侧部署边缘节点，可在源头对海量数据进行预处理与初步分析，显著降低通信开销并缓解中心站压力，同时在网络异常或突发事件时，实现快速决策和局部自治，提升系统鲁棒性[5]。

目前，针对配电网环境下的计算任务卸载策略，研究者主要集中于离散卸载或连续资源分配的单一优化。文献[6-9]提出了基于边缘节点的卸载方案，在用户侧显著降低了处理延迟；文献[10-12]针对配电网故障定位与处理等典型应用，提出了基于边缘计算的系统架构与优化方法；文献[13-14]分别利用长短期记忆(long-short term memory, LSTM)网络和多目标进化算法，研究了云−边协同环境下的任务预测与调度策略，提高了系统的处理效率和调度灵活性。然而，上述工作多将卸载决策与资源调度割裂地建模，缺乏对离散卸载动作与连续资源分配耦合优化的系统化方案，难以兼顾时延与能耗的联合目标。

深度强化学习(deep reinforcement learning, DRL)凭借其无需精确环境模型且具备在线学习能力，已成为智能卸载策略的新兴方向[15-19]。现有研究多采用深度 Q 网络(deep Q-network, DQN)或深度确定性策略梯度(deep deterministic policy gradient, DDPG)实现离散卸载或连续调度，并在仿真平台上取得了初步成果[20-25]；部分工作又通过引入优先经验回放(prioritized experience replay, PER)机制，提升了学习效率和收敛速度[26-28]。但针对同时包含离散卸载与连续资源配置的混合动作空间，尚缺乏高效的 DRL 算法；且传统 PER 机制仅基于时序差分(temporal difference, TD)误差评估样本重要性，未充分融合历史收益与动作特征，易陷入局部最优。

针对上述问题，本文提出了一种融合双重深度 Q 网络(double deep Q-network, DDQN)与 DDPG 的混合深度强化学习(hybrid deep reinforcement learning, HDRL)框架，并设计了改进型优先级经验回放(improved prioritized experience replay, IPER)策略。主要贡献如下。

1) 协同决策框架。构建了本地、边缘与云端三层协同计算模型，将任务卸载与资源调度建模为离散与连续混合动作问题，实现两类决策的协同优化。

2) 改进经验回放机制。在传统 PER 基础上引入历史更新因子与动作收益等多维指标，并结合重要性采样权重与扰动项，增强样本多样性与学习稳定性。

3) 系统仿真验证。在多用户－多边缘－云典型场景下开展仿真实验，结果表明 HDRL 在满足系统约束的前提下，显著提升时延与能耗优化性能，展现出优越的收敛速度和鲁棒性。

1 系统模型

1.1 系统描述

在配电网系统中，随着配电网和物联网技术的发展，任务卸载、计算资源的优化分配及无线宽带的高效利用，成为系统性能提升的关键因素。为应对这些需求，本文提出了一种面向配电网的任务卸载和资源分配系统架构。该架构基于边缘计算，通过合理配置计算资源和带宽资源，在多个 MEC 服务器、终端设备和云服务器之间实现高效的任务卸载与资源调度。

如图 1 所示，该系统模型由终端层、边缘计算层(MEC 层)和云层组成。终端层进行实时监测与基础控制，对可调度电力设备(如分布式电源、储能系统、电动汽车充电桩、智能用电终端等)通过传感器、智能电表或内置控制模块，实时采集电压、电流、温度、相位及设备运行状态等多维度信息，并执行初步的数据校验或过滤。这些终端设备将监测数据和可调度资源状态上报至部署在配电房附近的边缘计算节点。边缘计算层在汇集所有设备信息后，通过自有或与云端协同的分析算法，快速计算出调度与控制策略，并将其直接下发给相关的可调度设备，执行包括开启/关闭分布式电源、调节储能充放电功率或控制电动汽车充电负载等动作。对于需要大规模协同或依赖复杂模型的调度场景，边缘计算层会

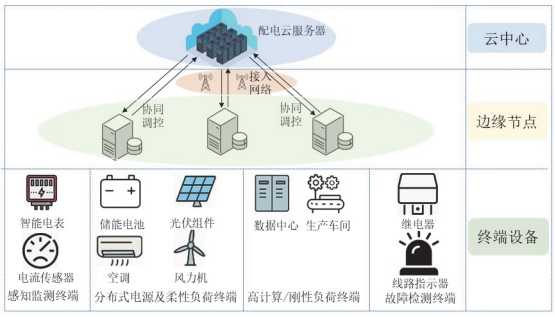

图 1 面向配电网的边缘计算系统模型

Fig. 1 Edge computing model for distribution networks

将关键数据上传至云端，由云端利用大数据分析或深度学习模型进行全局预测和统一规划，将最终的优化结果再下发给各个边缘节点进行分解执行。

在该系统中共有 $\mathcal{N}$ 个终端设备，记终端集合为 $\mathcal{N}=\{1,2,\cdots,N\}$，$\mathcal{M}$ 个边缘服务器，记集合为 $\mathcal{M}=\{1,2,\cdots,M\}$，另设云服务器 C。由于任务不可分，每个终端产生的任务必须在单一地点完成处理。

1.2 任务卸载与处理模型

第 i 个终端产生的任务用 $\langle D_i,C_i\rangle$ 描述，其中 D_i 为数据量(Mbit)，C_i 为每比特计算周期(cycles/bit)。为描述卸载决策，引入离散变量

$$a_i^{\mathrm{DDQN}}=[a_{i,0}^{\mathrm{DDQN}},a_{i,1}^{\mathrm{DDQN}},\cdots,a_{i,M}^{\mathrm{DDQN}},a_{i,\mathrm{C}}^{\mathrm{DDQN}}]$$

其中

$$\begin{cases}a_{i,0}^{\mathrm{DDQN}}=1 & \Rightarrow \text{任务在终端本地处理}\\ a_{i,j}^{\mathrm{DDQN}}=1 & \Rightarrow \text{任务整体卸载至第}j\text{个边缘服务器}(1\leqslant j\leqslant M)\\ a_{i,\mathrm{C}}^{\mathrm{DDQN}}=1 & \Rightarrow \text{任务卸载至云端}\end{cases}$$

针对终端 i 还需分配无线带宽与计算资源，引入连续动作变量

$$a_i^{\mathrm{DDPG}}=[a_i^{\mathrm{band}},a_i^{\mathrm{cpu}}] \tag{1}$$

式中：a_i^{band} 为终端 i 分配的带宽份额，影响传输速率；a_i^{cpu} 为终端 i 在边缘或云侧所获得的 CPU 资源比例，影响计算时延。

对于不同的处理方式，其延时和能耗计算如下。

1.2.1 终端处理

任务在终端处理的时延 T_i^{local} 和能耗 E_i^{local} 分别为

$$T_i^{\mathrm{local}}=\frac{D_i\cdot C_i}{f_i^{\mathrm{local}}} \tag{2}$$

$$E_i^{\mathrm{local}}=D_i\cdot C_i\cdot\epsilon_i^{\mathrm{local}} \tag{3}$$

式中：f_i^{local} 为终端的计算频率；$\epsilon_i^{\mathrm{local}}$ 为终端设备单位计算周期能耗。

1.2.2 边缘处理

当终端 i 将任务卸载至第 j 个边缘服务器时，其总时延 $T_{i,j}^{\mathrm{edge}}$ 与总能耗 $E_{i,j}^{\mathrm{edge}}$ 由上行数据传输和边缘侧计算两部分构成。鉴于计算结果的数据量远小于原始任务输入，下行返回链路的时延与能耗可忽略。于是，上行传输时延 $T_{i,j}^{\mathrm{trans}}$ 与能耗 $E_{i,j}^{\mathrm{trans}}$ 分别为

$$T_{i,j}^{\mathrm{trans}}=\frac{D_i}{r_{i,j}^{\mathrm{edge}}} \tag{4}$$

$$E_{i,j}^{\mathrm{trans}}=p_i\cdot T_{i,j}^{\mathrm{trans}} \tag{5}$$

式中：$r_{i,j}^{\mathrm{edge}}$ 为根据信道条件和带宽分配确定的传输速率；p_i 为终端的发射功率。

边缘服务器计算时延 $T_{i,j}^{\mathrm{comp}}$ 和能耗 $E_{i,j}^{\mathrm{comp}}$ 分别如式(6)和式(7)所示。

$$T_{i,j}^{\mathrm{comp}}=\frac{D_i\cdot C_i}{f_{i,j}^{\mathrm{edge}}} \tag{6}$$

$$E_{i,j}^{\mathrm{edge}}=D_i\cdot C_i\cdot\epsilon_j^{\mathrm{edge}} \tag{7}$$

式中：$f_{i,j}^{\mathrm{edge}}$ 为分配给第 i 个终端设备的计算频率；$\epsilon_j^{\mathrm{edge}}$ 为边缘节点单位计算能耗。

1.2.3 云端处理

同理，当终端 i 选择将任务上传至云数据中心时，总时延 $T_{i,\mathrm{C}}$ 与总能耗 $E_{i,\mathrm{C}}$ 由长距离上行传输和云侧计算两部分组成。由于结果数据量远小于输入数据量，下行返回可忽略，于是上行传输时延 $T_{i,\mathrm{C}}^{\mathrm{trans}}$ 与能耗 $E_{i,\mathrm{C}}^{\mathrm{trans}}$ 分别为

$$T_{i,\mathrm{C}}^{\mathrm{trans}}=\frac{D_i}{r_{i,\mathrm{C}}} \tag{8}$$

$$E_{i,\mathrm{C}}^{\mathrm{trans}}=p_i\cdot T_{i,\mathrm{C}}^{\mathrm{trans}} \tag{9}$$

式中：$r_{i,\mathrm{C}}$ 为云端传输速率。

云端计算时延 $T_{i,\mathrm{C}}^{\mathrm{comp}}$ 与能耗 $E_{i,\mathrm{C}}^{\mathrm{comp}}$ 分别为

$$T_{i,\mathrm{C}}^{\mathrm{comp}}=\frac{D_i\cdot C_i}{f_{i,\mathrm{C}}} \tag{10}$$

$$E_{i,\mathrm{C}}^{\mathrm{comp}}=D_i\cdot C_i\cdot\epsilon_{\mathrm{C}} \tag{11}$$

式中：$f_{i,\mathrm{C}}$ 为云端分配给第 i 个终端设备的计算频率；ϵ_{C} 为云服务器功耗。

1.3 公平性

在多终端场景中，若仅最小化总体时延与能耗，可能导致部分终端长时间处于资源饥饿状态而资源分配极不均衡。为衡量各终端时延分布的均衡程度，引入 Jain 公平指数 F_{Jain}，其定义为

$$F_{\mathrm{Jain}}=\frac{\left(\sum_{i=1}^{N}T_i\right)^2}{N\sum_{i=1}^{N}(T_i)^2} \tag{12}$$

式中：T_i 为第 i 个终端卸载的总体执行时延。当所有终端时延完全相同时 $F_{\mathrm{Jain}}=1$；若存在严重不均则趋近 0。

为了在优化目标中显式惩罚不公平分配，定义公平性惩罚项 P_f 为

$$P_f=\lambda(1-F_{\mathrm{Jain}}) \tag{13}$$

式中：$\lambda>0$ 为惩罚项权重系数。

1.4 信道模型与传输速率

在配电网中，终端与边缘服务器或云数据中心之间通常通过无线或混合通信网络进行数据传输。

为描述实际通信环境，本系统采用如式(14)所示的信道模型。

$$g(d)=\frac{1}{d^{\alpha_{\mathrm{PL}}}}\cdot h \tag{14}$$

式中：d 为终端与处理单元之间的距离；α_{PL} 为路径损耗指数；h 为描述多径衰落效应的随机变量。终端以发射功率 p_i 发送数据，其传输链路信噪比 R_{SNR} 的计算公式为

$$R_{\mathrm{SNR}}=\frac{p_i\cdot g(d)}{\sigma^2} \tag{15}$$

式中：σ^2 为噪声功率。基于香农定理，分配给终端的传输速率 r_{b} 的计算公式为

$$r_{\mathrm{b}}=a_i^{\mathrm{band}}\cdot W\cdot\log_2(1+R_{\mathrm{SNR}}) \tag{16}$$

式中：a_i^{band} 为终端 i 分配到的带宽比例；W 为总的带宽资源。

1.5 优化目标

综合上述模型，系统总体目标在于选择最佳的处理地点和资源分配方案，使得整体成本最小化。设延时、能耗及公平性惩罚的权重分别为 α、β 和 λ，则系统成本 C_{sys} 为

$$C_{\mathrm{sys}}=-[\alpha\cdot\bar{T}+\beta\cdot\bar{E}+\lambda(1-F_{\mathrm{Jain}})] \tag{17}$$

式中：$\bar{T}=\sum_{i=1}^{N}T_i$ 为系统时延；$\bar{E}=\sum_{i=1}^{N}E_i$ 为系统能耗。任务不可分的特点要求每个终端仅作出单一离散决策(本地、边缘或云端)，该决策与无线资源和计算资源的分配协同作用，共同决定系统总体性能。

2 问题建模

综合上述模型与约束，本文的研究目标是在保证各终端任务可完成且满足带宽、计算资源等限制的前提下，联合优化任务卸载决策、带宽分配以及计算资源分配，使得系统整体时延与能耗最小。可将本研究的问题形式转化为如式(18)所示的优化问题。

$$\begin{cases}\min\limits_{a_i^{\mathrm{DDQN}},a_i^{\mathrm{band}},f_{i,j}^{\mathrm{edge}},f_{i,\mathrm{C}}} \alpha\cdot\bar{T}+\beta\cdot\bar{E}+\gamma(1-F_{\mathrm{Jain}})\\ \text{s.t.}\quad \mathrm{C1}: a_{i,0}^{\mathrm{DDQN}}+\sum\limits_{j=1}^{M}a_{i,j}^{\mathrm{DDQN}}+a_{i,\mathrm{C}}^{\mathrm{DDQN}}=1,\forall i\\ \quad\ \mathrm{C2}: 0\leqslant a_i^{\mathrm{band}}\leqslant 1,\forall i\\ \quad\ \mathrm{C3}: 0\leqslant a_i^{\mathrm{cpu}}\leqslant 1,\forall i\\ \quad\ \mathrm{C4}: \sum\limits_{i=1}^{N}f_{i,j}^{\mathrm{edge}}\leqslant f_j^{\max},f_{i,j}^{\mathrm{edge}}\geqslant 0,\forall i\\ \quad\ \mathrm{C5}: \sum\limits_{i=1}^{N}f_{i,\mathrm{C}}\leqslant f_{\mathrm{C}}^{\max},f_{i,\mathrm{C}}\geqslant 0,\forall i\end{cases} \tag{18}$$

本文优化的变量包括任务卸载决策、带宽资源分配、服务器 CPU 频率分配。约束式(C1)为任务的卸载决策约束；约束式(C2)为分配给任务的带宽资源约束；约束式(C3)为分配任务的计算资源约束；约束式(C4)为服务器分配频率，不能超过最大总频率；约束式(C5)为云服务器分配频率，不能超过最大总频率。

3 算法设计

由于上述问题是一个混合整数的非凸问题，因此，本文引入强化学习技术，实现在线优化，并提出一种基于分层强化学习的联合决策方案，以同时解决处理地点离散选择与卸载任务资源连续分配的问题。具体而言，该方案构建了两级决策结构：上层利用 DDQN 对任务卸载决策进行离散建模，下层采用 DDPG 对卸载任务资源进行连续优化；两级之间通过信息交互与联合训练，协同实现系统整体成本的最小化。

3.1 分层决策架构

1) 离散决策模块。上层模块采用 DDQN 对每个终端的卸载决策进行建模。根据终端任务的基本信息，DDQN 为每个终端选择最优的处理地点，即确定 a_i^{DDQN} 的取值。

2) 连续调度模块。对于选择卸载至边缘或云端的任务，系统需要进一步在目标处理单元上进行资源分配。下层模块采用 DDPG 方法，根据全局状态输出连续资源分配决策，其输出决定了各任务在无线带宽和服务器计算资源上获得的比例。对于在终端本地处理的任务，资源分配动作可直接设置为默认值。

两层模块之间通过信息交互实现协同优化。下层 DDPG 模块输出的资源分配信息可作为 DDQN 模块的扩充状态输入，帮助上层更准确地评估不同卸载策略的长期收益；而上层确定的卸载决策则为下层资源调度提供明确约束。

3.2 离散决策模块设计

针对任务整体卸载的离散决策，DDQN 模块设计主要包括状态构造部分。

对于每个终端，其状态向量包括任务数据量 D_i、计算需求 C_i 和当前无线信道质量。

DDQN 的核心思想是用两套结构相同、参数独立的 Q 网络将动作选择(在线网络)与价值评估(目标网络)分离，从而缓解单网络 Q-learning 的过估计偏差。具体而言，Q 值表示在状态 s 下采取动作 a 时未来折扣累计奖励的期望，其定义为

$$Q(s_i,a_i^{\mathrm{DDQN}})=\mathbb{E}\left[\sum_{j=0}^{\infty}\gamma^j r_{i+j}\mid s_0=s_i,a_0=a_i^{\mathrm{DDQN}}\right] \tag{19}$$

式中：r 为时间步 t 的即时奖励；γ 为折扣因子。

DDQN 通过神经网络近似 Q 值函数，网络输入为状态 s_i，输出为每个动作的 Q 值，即

$$Q(s_i,a_i^{\mathrm{DDQN}};\theta)\approx\hat{Q}(s_i,a_i^{\mathrm{DDQN}}) \tag{20}$$

式中：θ 为神经网络的参数。通过最小化 Q 值误差来更新神经网络的参数，采用的损失函数为

$$L(\theta)=\mathbb{E}[(y_i-Q(s_i,a_i^{\mathrm{DDQN}};\theta))^2] \tag{21}$$

式中：y_i 为目标 Q 值，其计算公式为

$$y_i=r_i+\gamma Q(s_{i+1},\arg\max_{a'}Q(s_{i+1},a';\theta);\theta^-) \tag{22}$$

式中：θ^- 为目标网络的参数，更新时采用软更新方法。

$$\theta^-\leftarrow\tau\theta+(1-\tau)\theta^- \tag{23}$$

式中：τ 为目标网络更新步长。DDQN 采用 $\varepsilon-$贪心策略来平衡探索和利用。代理以概率 ε 选择随机动作(探索)，以概率 $1-\varepsilon$ 选择当前 Q 值最大的动作。随着训练的进行，逐渐减小 ε，使得策略从探索转向利用。

3.3 连续调度模块设计

DDPG 模块用于卸载任务的资源分配问题。DDPG 基于 Actor-Critic 框架，通过连续动作输出优化资源分配。其目标是根据当前状态输出最优的资源分配策略，最大化长期回报。

DDPG 的状态空间包括全局系统状态、任务卸载决策(DDQN 模块给出)、以及当前的资源使用情况。对于每个终端，状态 s_i 可以表示为

$$s_i^{\mathrm{DDPG}}=[o_i,a_{i-1}^{\mathrm{DDQN}},a_{i-1}^{\mathrm{DDPG}}] \tag{24}$$

式中：o_i 汇集实时观测量，如剩余带宽、CPU 利用率、信道增益；a_i^{DDQN} 为 DDQN 模块输出的卸载决策状态；a_{i-1}^{DDPG} 为下层 DDPG 模块的资源分配动作。动作空间是一个连续的资源分配空间，用带宽和计算资源的分配比例表示。对于每个终端，资源分配动作 $a_i^{\mathrm{DDPG}}=[a_i^{\mathrm{band}},a_i^{\mathrm{cpu}}]$，表示带宽和计算资源的分配比例。

Actor 网络接收全局状态 s_i 作为输入、输出连续的资源分配动作 a_i^{DDPG}，如式(25)所示。

$$a_i^{\mathrm{DDPG}}=\pi(s_i;\theta^\pi) \tag{25}$$

式中：θ^π 为网络的参数。

Critic 网络评估状态−动作对的价值，计算 Q 值。

$$Q(s_i,a_i^{\mathrm{DDPG}};\theta^Q)\approx\hat{Q}(s_i,a_i^{\mathrm{DDPG}}) \tag{26}$$

式中：θ^Q 为 Critic 网络的参数。

通过最小化 Critic 的损失函数来更新 Critic 训练网络参数，损失函数为

$$\mathcal{L}(\theta^Q)=\mathbb{E}[(r_i+\gamma Q(s_{i+1},\pi_-(s_{i+1};\theta_-^\pi);\theta_-^Q)-Q(s_i,a_i^{\mathrm{DDPG}};\theta^Q))^2] \tag{27}$$

式中：θ_-^π、θ_-^Q 分别为目标 Actor、目标 Critic 的参数。

Actor 训练网络则通过梯度上升方法来更新，目标是最大化 Critic 网络的 Q 值。

$$\nabla_{\theta^\pi}J(\theta^\pi)=\mathbb{E}[\nabla_{a_i^{\mathrm{DDPG}}}Q(s_i,a_i^{\mathrm{DDPG}};\theta^Q)\nabla_{\theta^\pi}\pi(s_i;\theta^\pi)] \tag{28}$$

Actor 目标网络和 Critic 目标网络同样采用软更新方式进行，如式(29)和式(30)所示。

$$\theta_-^\pi\leftarrow\tau\theta^\pi+(1-\tau)\theta_-^\pi \tag{29}$$

$$\theta_-^Q\leftarrow\tau\theta^Q+(1-\tau)\theta_-^Q \tag{30}$$

3.4 改进的优先经验采样

为提高学习效率并加速收敛，本文提出了一种 IPER 机制。与传统的经验采样和 PER 方法不同，IPER 引入了历史经验的平滑误差和动态优先级更新策略，从而避免过度关注不重要的经验，确保模型在训练过程中充分探索关键经验。

IPER 对每个经验的优先级进行计算，优先级不仅基于当前的 TD 误差，还结合历史误差和动态调整因子，以确保模型在训练过程中能够充分利用重要经验。

先对每条经验的原始 TD 误差 δ_i 做指数移动平均(EMA)平滑，得到 $\tilde{\delta}_i$ 如式(31)所示。

$$\tilde{\delta}_i\leftarrow\gamma\tilde{\delta}_i+(1-S)|\delta_i|,0\leqslant S<1 \tag{31}$$

式中：S 为平滑系数；每当样本被回放并重新计算 TD 误差后执行一次更新。

为避免 TD 误差为 0 的样本永远得不到采样，引入微小常数 ε。

$$P_i=\frac{(\tilde{\delta}_i+\varepsilon)^\alpha}{\sum_{j=1}^{N}(\tilde{\delta}_j+\varepsilon)^\alpha},0\leqslant\alpha\leqslant1 \tag{32}$$

式中：α 控制优先级 P_i 对 TD 误差幅度的敏感度；$\varepsilon\approx10^{-6}$。

在训练早期可适当降低优先级的影响，随后逐渐转向完全优先采样，如式(33)所示。

$$p_i(t)=\frac{P_i(t)^{\nu(t)}}{\sum_{j=1}^{N}P_j(t)^{\nu(t)}},0\leqslant\nu(t)\leqslant1 \tag{33}$$

式中：ν 为动态调整因子，通常从 0 线性(或分段)递增到 1，使采样过程由近似均匀逐步过渡到完全按优先级。

为抵消非均匀采样带来的估计偏差，引入 IS 权重 w_i 为

$$w_i=\left(\frac{1}{N}\cdot\frac{1}{p_i(t)}\right)^{\beta(t)},0\leqslant\beta(t)\leqslant1 \tag{34}$$

式中：$\beta(t)$ 为动量参数，训练时从初始值 β_0 线性递

增到 1，保证后期梯度估计无偏；实际更新梯度时使用归一化后的权重 $\hat{w}_i = w_i / \max_k w_k$ 。

3.5 联合算法训练实施过程

本方案采用分层联合训练方式，依次更新离散决策模块和连续调度模块，实现整体目标函数的优化。具体训练流程如表 1 所示。

表 1 算法训练过程

Table 1 Model training process

HDRL 算法
输入：系统环境参数 HDRL 参数
输出：HDRL 网络参数
1) 初始化网络参数，经验池回放池
2) for episode = 1 to $P_{\max}$ do
3) 初始化环境、状态、奖励值
4) for $t=1$ to T do
5) 通过 DDQN 模块根据式(22)选择任务卸载决策
6) 将 a_i^{DDQN} 传递到 DDPG 模块作为资源分配的基础信息。对于卸载至边缘或云端的任务，DDPG 通过 Actor 网络输出资源分配的连续动作。
7) 执行 a_i^{DDQN} 和 a_i^{DDPG}，并获取新的状态 s_{t+1} 和奖励 r_i
8) 将经验 (s_t, a_t, r_t, s_{t+1}) 存储在经验池 $\mathcal{D}$ 中
9) 利用 IPER 机制从经验回放池 $\mathcal{D}$ 中采样
10) 根据式(17)更新 DDQN 网络，使用 Adam 优化器来更新 DDQN 网络的参数
11) 根据式(26)、式(27)更新 Critic 训练网络参数和 Actor 训练网络参数
12) 根据式(29)、式(30)更新 Critic 目标网络参数和 Actor 目标网络参数
13) end for
14) end for

4 仿真实验与结果分析

4.1 仿真环境

本文采用数值模拟的方式进行仿真实验，实验环境基于 Python 平台自定义搭建。综合考虑时延开销和能耗开销，并使用表 2 和表 3 中的参数设置仿真环境。

4.2 仿真结果

为验证所提 HDRL 方法的有效性，本文选取以下 5 种基线算法进行对比。LOCAL 的所有任务均在终端本地处理；MEC 的所有任务全部卸载至距离最近的边缘服务器；RANDOM 对每个任务随机选择本地、任一边缘或云端执行；GA 采用经典遗传算法；DDQN + DDPG 与 HDRL 共享同一分层动作空间、网络结构和超参数，但回放机制为均匀经验采样而非 IPER。

表 2 参数设置

Table 2 Parameter setting

参数	数值
D_i/Mbit	[2,7]
C_i/(cycles/bit)	[300,600]
f_i^{local}/GHz	0.9
$\epsilon_i^{\mathrm{local}}$/(J/cycle)	1.6×10^{-10}
f_j^{edge}/GHz	10
$\epsilon_j^{\mathrm{edge}}$/(J/cycle)	5×10^{-10}
$f_C^{\max}$/GHz	20
ϵ_C/(J/cycle)	1×10^{-9}
$W_j^{\max}$/MHz	10
$W_C^{\max}$/MHz	20
M/台	3
N/台	[5,10]
α	0.5
β	0.5
λ	0.1

表 3 超参数设置

Table 3 Hyperparameter settings

参数	值
最大训练回合数	1000
批量大小	256
ε −贪婪策略的初始探索概率	1
ε −贪婪策略的最小探索概率	0.05
动作噪声的缩放因子	0.1

由图 2—图 4 验证了系统中任务大小变化对系统性能的影响，LOCAL(全本地计算)由于所有任务都在终端本地进行计算，不存在排队和传输等待，因此其时延能耗主要由终端本地计算速度和任务数据量大小决定，整体上呈线性增长。RANDOM(随机

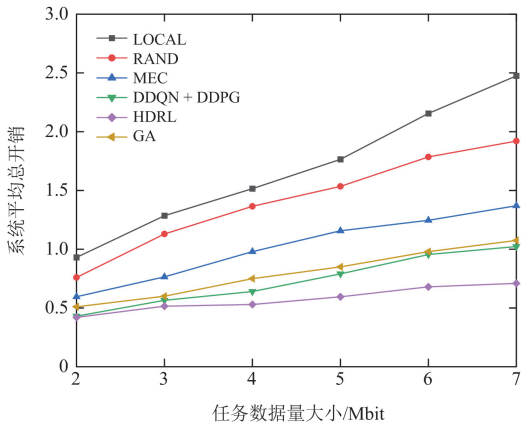

图 2 任务数据量大小对平均开销的影响

Fig. 2 Task data volume vs. average overhead

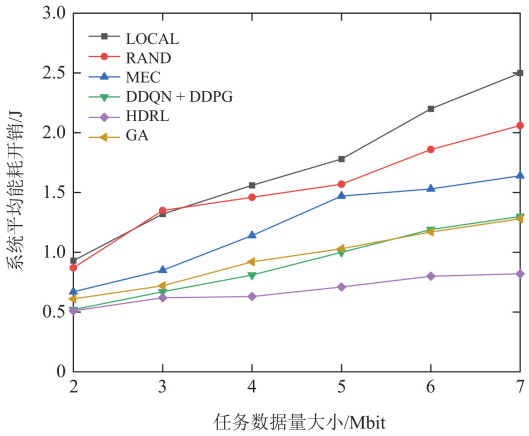

图 3 任务数据量大小对平均能耗的影响

Fig. 3 Task data volume vs. average energy consumption

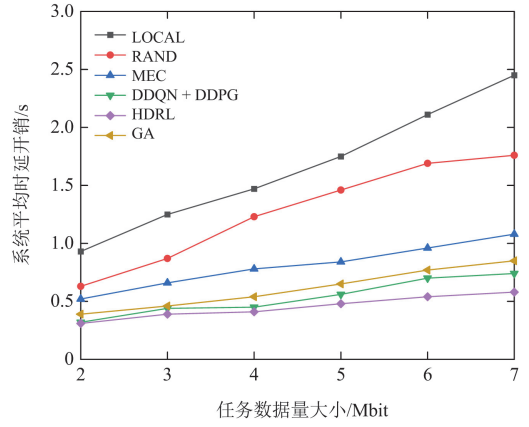

图 4 任务数据量大小对平均时延的影响

Fig. 4 Task data volume vs. average latency

计算)随机地在本地或边缘进行任务计算，缺乏对网络状态和资源利用情况的有效感知，导致其在大多数情况下无法充分降低排队和传输等待时间。因而，时延能耗随数据量增大而波动，整体效果并不理想。

MEC(全边缘计算)将所有任务卸载到边缘侧，初始时由于终端侧计算负载大幅减轻，展现出性能较好。然而，随着任务数据量增大，边缘服务器出现排队等待，导致时延逐步增加。特别在高负载情况下，集中式边缘计算难以提供理想的低性能保障。

当任务大小在 3 Mbit 以下时，GA、HDR 和 DDQN + DDPG 的 3 条曲线几乎重合。此时负载偏轻，搜索空间有限，GA 依托全局进化机制即可快速收敛，而 HDRL 与 DDQN+DDPG 也能在少量训练步内得到近似最优策略，因而三者差异不显。然而当任务规模增至 5 Mbit 时，性能分化迅速出现：HDRL 曲线仍保持较缓斜率，而 GA 与 DDQN + DDPG 的时延与能耗开始显著上扬。其根本原因在于 GA 的局限-任务规模放大后，染色体长度与交叉/变异搜索空间呈指数增长，导致收敛代数显著增加。而 HDRL 的 IPER 机制在任务规模变化时，能够更高效、全面地搜索最优的任务卸载与资源配置方案。在策略决策中，HDRL 同步考虑了时延和能耗，并充分挖掘边缘计算资源的潜力，使其在较宽范围的任务数据量变化下依旧保持较低时延。由于 HDRL 在任务卸载与资源调度的策略上更具针对性，时延的增长幅度得以有效抑制，与其他算法相比始终处于更佳水平。

由图 5—图 7 中可以看到，当终端设备数量不断上升时，系统在平均时延、平均能耗以及二者加权之后的总开销上均随之增大，主要原因在于要同时处理的任务增多，但可用的计算和通信资源却具有上限。虽然边缘服务器具有较高的计算能力，但同时也会带来较大的能耗消耗。

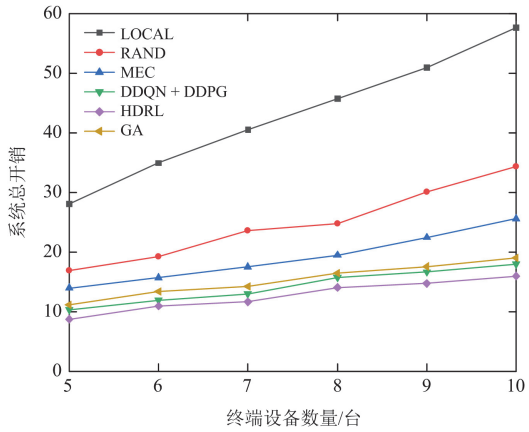

图 5 终端设备数量对总开销的影响

Fig. 5 Number of terminal devices vs. total overhead

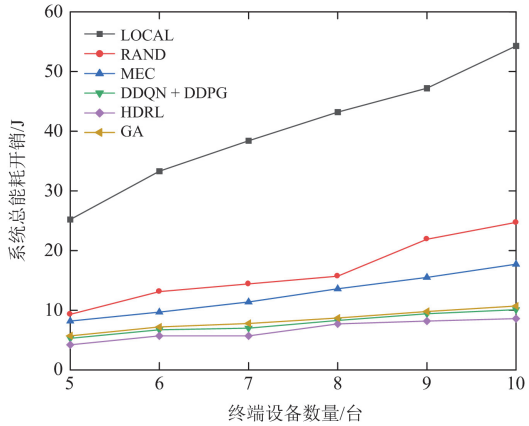

图 6 终端设备数量对总能耗开销的影响

Fig. 6 Number of terminal devices vs. total energy consumption

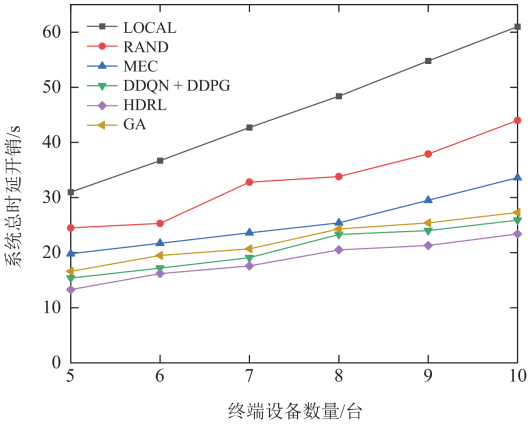

图 7 终端设备数量对总时延开销的影响

Fig. 7 Number of terminal devices vs. total latency

当终端设备数目进一步增加时，全边缘计算可能面临带宽和算力不足的问题，无法及时处理所有未被卸载至 MEC 服务器的任务。在这种高负载情境下，合理的资源调度至关重要，不仅要尽可能减少系统的总体开销，还需保障各终端设备的服务质量。为此，本文提出的 HDRL 算法，能够在资源有限的条件下提升系统整体性能，并通过动态地依据用户需求进行资源分配，最大程度满足各类用户的服务需求。

由图 8 可以看出，终端设备数量由 5 台递增至 10 台后，各算法的公平指数呈现明显分化：LOCAL 始终保持在 0.95 以上并略有提升，HDRL 稳定在 0.91~0.94 之间几乎不受规模扩大影响，而 DQN + DDPG 由约 0.93 小幅下降至 0.88。相比之下，MEC 从 0.94 跌至 0.83 左右，GA 则从接近 0.90 骤降至不足 0.40，RAND 在 7 台设备时最低不到 0.20 后小幅回升至 0.33 左右。总体而言，分层强化学习策略

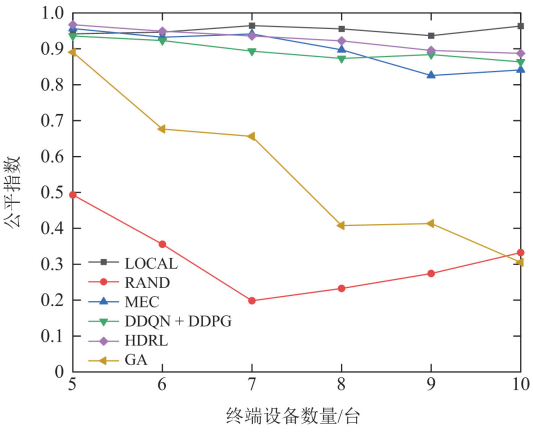

图 8 终端设备数量对公平指数的影响

Fig. 8 Influence of the number of terminal devices on the fairness index

(HDRL)在用户规模扩大时依旧能维持高公平性，表现出最佳的扩展鲁棒性；LOCAL 得益于本地任务处理同样保持高公平度，而集中式或启发式方法(MEC、GA)及随机策略在高并发场景下公平性明显恶化，已难以满足多终端资源均衡分配需求。

图 9 对比了 HDRL 算法与 DDQN + DDPG 算法在训练过程中的收敛性能。图 9 中，横轴为训练回合数，纵轴为平均成本，曲线变化反映了算法在优化过程中策略质量的演化趋势。从结果可以明显看出，HDRL 在整个训练过程中表现出更快的收敛速度和更高的稳定性。具体而言，HDRL 在前期训练阶段平均成本下降更迅速，能够在更少的回合数内达到较优的策略表现；而 DDQN + DDPG 则在前期存在较大波动，稳定性较差。此外，尽管两种算法最终均趋于收敛，HDRL 在后期的平均成本表现更低，且曲线波动幅度更小，体现出更强的训练稳定性和策略鲁棒性。综上所述，HDRL 在收敛速度、稳定性及最终性能方面均优于 DDQN + DDPG，验证了其在复杂任务中的有效性和优势。

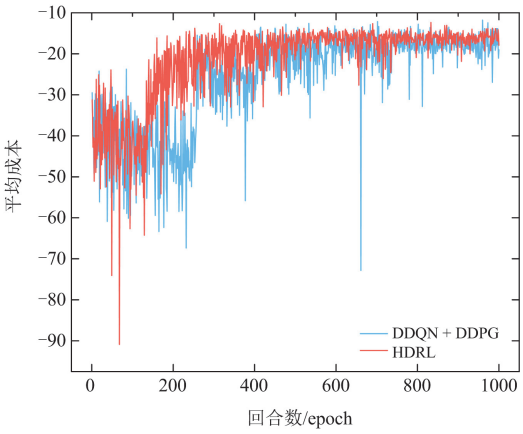

图 9 收敛对比图

Fig. 9 Convergence comparison diagram

表 4 给出了各算法在单个决策时隙内的平均计算时间，可以直观看出 3 类方法在“实时性”维度的差异：GA 在单时隙推理中平均耗时 21.17 s，远高于 DDQN + DDPG 与 HDRL；这主要因为 GA 每次都需重新进化完整种群，缺乏历史经验迁移，而基于深度网络的两类 DRL 方法仅需一次前向传播即可给出决策。尽管 HDRL 引入双 Q 网络和 IPER 机制，相比 DDQN + DDPG 多出约 2 ms 的运算开销，但其总时耗仍处于毫秒级，远低于典型配电网 3 s 以内的实时控制要求，因而可满足在线调度场景的响应速度。

表 4 算法用时对比

Table 4 Runtime comparison of different algorithms

算法	计算时间/s
GA	21.169
DDQN + DDPG	0.016
HDRL	0.018

5 结论

本文所提基于HDRL的任务卸载与资源分配方法，能够快速计算出最优的调度与控制策略，并将其直接下发给相关可调度设备。仿真实验结果表明，与传统方法相比，HDRL 在系统时延与能耗方面均有显著降低，尤其在高负载场景下展现出较强的自适应能力和资源调度效率。该方法充分利用了边缘计算与云端协同的优势，对于需要实时处理海量数据、并且对系统可靠性与灵活性要求较高的配电网场景，尤其在高负载场景下，该算法展现出更强的自适应能力和资源调度效率。

参考文献

[1] 张瑶, 王傲寒, 张宏. 中国智能电网发展综述[J]. 电力系统保护与控制, 2021, 49(5): 180-187.
ZHANG Yao, WANG Aohan, ZHANG Hong. Overview of smart grid development in China[J]. Power System Protection and Control, 2021, 49(5): 180-187.

[2] 赵琛, 叶金池, 和萍, 等. 考虑源荷多重不确定性的园区综合能源系统优化策略[J]. 电力系统保护与控制, 2025, 53(4): 148-164.
ZHAO Chen, YE Jinchi, HE Ping, et al. A PIES optimization strategy considering multiple uncertainties in source and load[J]. Power System Protection and Control, 2025, 53(4): 148-164.

[3] 赵文卓, 唐宏, 冯琛, 等. 一种基于权重网络的约束多目标任务卸载优化方法[J]. 计算机应用研究, 2025, 42(5): 1447-1452.
ZHAO Wenzhuo, TANG Hong, FENG Chen, et al. Constrained multi-objective task offloading optimization method based on weight networks[J]. Application Research of Computers, 2025, 42(5): 1447-1452.

[4] 习伟, 李鹏, 李鹏, 等. 面向配电网边缘计算装置的两阶段 PMU 数据压缩方法[J]. 电网技术, 2023, 47(8): 3184-3192.
XI Wei, LI Peng, LI Peng, et al. Two-stage PMU data compression for edge computing devices of distribution networks[J]. Power System Technology, 2023, 47(8): 3184-3192.

[5] 秦强, 刘文泽, 谭炜豪, 等. 面向软件定义网络的配电边缘计算终端优化部署方法[J]. 电力建设, 2023, 44(1): 82-90.
QIN Qiang, LIU Wenze, TAN Weihao, et al. An optimal deployment method of distribution edge computing terminals for software defined network[J]. Electric Power Construction, 2023, 44(1): 82-90.

[6] 阎帅, 卢志刚, 王婧, 等. 基于分布式边缘计算的智能配电网规划策略研究[J]. 计算机仿真, 2024, 41(10): 301-305, 332.
YAN Shuai, LU Zhigang, WANG Jing, et al. Research on smart distribution network planning strategy based on distributed edge computing[J]. Computer Simulation, 2024, 41(10): 301-305, 332.

[7] 张文柱, 余静华. 移动边缘计算中基于云边端协同的任务卸载策略[J]. 计算机研究与发展, 2023, 60(2): 371-385.
ZHANG Wenzhu, YU Jinghua. Task offloading strategy in mobile edge computing based on cloud-edge-end cooperation[J]. Journal of Computer Research and Development, 2023, 60(2): 371-385.

[8] 孙畅, 吴牧野. 物联网技术在配电网故障检测中的应用[J]. 互联网周刊, 2024(20): 34-36.

[9] 杨文迪, 黄泽湘, 孙振东, 等. 云边协同计算与任务分配优化研究[J]. 中国新技术新产品, 2025(4): 5-7.

[10] 徐策, 王俊江, 钟浩, 等. 基于物联网边缘计算的配电网故障定位方法[J]. 计算机应用与软件, 2024, 41(10): 95-103, 109.
XU Ce, WANG Junjiang, ZHONG Hao, et al. Design and application of distribution network fault location method based on IOT edge computing[J]. Computer Applications and Software, 2024, 41(10): 95-103, 109.

[11] 张卫红, 陈小龙, 万顺, 等. 基于边缘计算的分布式配电故障处理系统[J]. 供用电, 2019, 36(9): 28-33, 58.
ZHANG Weihong, CHEN Xiaolong, WAN Shun, et al. A distributed distribution network fault processing system based on edge computing technology[J]. Distribution & Utilization, 2019, 36(9): 28-33, 58.

[12] 朱莹, 徐正宏, 常颢, 等. 基于配电网全景诊断分析的网架项目设计及优选方法[J]. 光源与照明, 2025(1): 192-194.

[13] ZHOU Ya, JIAO Xiaobo. Intelligent analysis system for signal processing tasks based on LSTM recurrent neural network algorithm[J]. Neural Computing and Applications, 2022, 34: 12257-12269.

[14] ZHOU Ya, JIAO Xiaobo. Knowledge-driven multi-objective evolutionary scheduling algorithm for cloud workflows[J]. IEEE Access, 2022, 10: 2952-2962.

[15] LIU Chuan, CHEN Lei, GAO Wei, et al. DRL-based online task offloading and energy resource aggregation for edge-computing-empowered smart grid networks[J]. IEEE Internet of Things Journal, 2024, 11(24): 41008-41020.

[16] NIMKAR S, KHANAPURKAR M M. Design of a

Q-learning based smart grid and smart water scheduling model based on heterogeneous task specific offloading process[C] // 2022 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON), December 23-25, 2022, Bangalore, India: 1-9.

[17] SHU Chang, ZHAO Zhiwei, MIN Geyong, et al. Deploying network functions for multiaccess Edge-IoT with deep reinforcement learning[J]. IEEE Internet of Things Journal, 2020, 7(10): 9507-9516.

[18] 全欢, 彭显刚, 刘涵予, 等. 基于深度强化学习的配电网实时电压优化控制方法[J]. 电网技术, 2023, 47(5): 2029-2038.
QUAN Huan, PENG Xiangang, LIU Hanyu, et al. Voltage optimal control of distribution network based on deep reinforcement learning[J]. Power System Technology, 2023, 47(5): 2029-2038.

[19] 罗欣儿, 杜进桥, 田杰, 等. 基于深度强化学习的主动配电网高恢复力决策方法[J]. 南方电网技术, 2022, 16(1): 67-74.
LUO Xiner, DU Jinqiao, TIAN Jie, et al. High resilience decision-making method of active distribution network based on deep reinforcement learning[J]. Southern Power System Technology, 2022,16(1): 67-74.

[20] 蒲天骄, 杜帅, 李烨, 等. 面向隐私保护基于联邦强化学习的分布式电源协同优化策略[J]. 电力系统自动化, 2023, 47(8): 62-70.
PU Tianjiao, DU Shuai, LI Ye, et al. Collaborative optimization strategy of distributed generators based on federated reinforcement learning for privacy preservation[J]. Automation of Electric Power Systems, 2023, 47(8): 62-70.

[21] 齐英, 许潇, 殷科, 等. 基于深度强化学习的含储能有源配电网电压联合调控技术[J]. 电力建设, 2023, 44(11): 64-74.
QI Yunying, XU Xiao, YIN Ke, et al. Voltage co-regulation technology of active distribution network with energy storage based on deep reinforcement learning[J]. Electric Power Construction, 2023, 44(11): 64-74.

[22] 王光华, 李晓影, 宋秉睿, 等. 基于深度强化学习的配电网负荷转供控制方法[J]. 电力自动化设备, 2022, 42(7): 253-260.
WANG Guanghua, LI Xiaoying, SONG Bingrui, et al. Load transfer control method of distribution network based on deep reinforcement learning[J]. Electric Power Automation Equipment, 2022, 42(7): 253-260.

[23] 李宏川, 赵宇, 李彬, 等. 配电物联网边缘计算场景下基于改进ANFIS的电缆通道综合评估及智能预警方法研究[J]. 电力系统保护与控制, 2024, 52(12): 94-103.
LI Hongchuan, ZHAO Yu, LI Bin, et al. Comprehensive assessment and intelligent early warning of cable passages based on improved ANFIS in the edge computing scenario of PDIoT[J]. Power System Protection and Control, 2024, 52(12): 94-103.

[24] 邓芳明, 单运, 解忠鑫, 等. 基于博弈论和强化学习的无人机电力巡检卸载策略[J]. 电网技术, 2021, 45(9): 3649-3657.
DENG Fangming, SHAN Yun, XIE Zhongxin, et al. Power inspection and unloading strategy of UAV based on game theory and reinforcement learning[J]. Power System Technology, 2021, 45(9): 3649-3657.

[25] 李士丹, 李航, 李国杰, 等. 考虑分区与模仿学习的深度强化学习配电网电压优化策略[J]. 电力系统保护与控制, 2024, 52(22): 1-11.
LI Shidan, LI Hang, LI Guojie, et al. Voltage optimization strategy for a distribution network based on deep reinforcement learning considering regionalization and imitation learning[J]. Power System Protection and Control, 2024, 52(22): 1-11.

[26] 高昂, 董志明, 李亮, 等. MADDPG 算法并行优先经验回放机制[J]. 系统工程与电子技术, 2021, 43(2): 420-433.
GAO Ang, DONG Zhiming, LI Liang, et al. Parallel priority experience replay mechanism of MADDPG algorithm[J]. Systems Engineering and Electronics, 2021, 43(2): 420-433.

[27] ZHOU Shiyang, CHENG Yufan, LEI Xia, et al. Deep deterministic policy gradient with prioritized sampling for power control[J]. IEEE Access, 2020, 8: 194240-194250.

[28] 萧文聪, 陈俊斌, 余涛, 等. 基于专家知识嵌入强化学习的配电系统灾后恢复决策方法[J]. 电力系统自动化, 2025, 49(12): 91-100.
XIAO Wencong, CHEN Junbin, YU Tao, et al. Decision-making method for post-disaster distribution system restoration based on expert knowledge-embedded reinforcement learning[J]. Automation of Electric Power Systems, 2025, 49(12): 91-100.

收稿日期：2025-04-25； 修回日期：2025-08-01

作者简介：

周 雅(1981—)，女，通信作者，硕士，教授，研究方向为智能电网控制与信息处理； E-mail: 12004042@xcu.edu.com

王 乾(1999—)，男，硕士，研究方向为智能电网控制与信息处理； E-mail: m17513364071@163.com

方如举(1975—)，男，博士，教授，研究方向为电气智能控制。E-mail: fangruju@163.com

(编辑 石晋美)