Parallel Hybrid Deep Reinforcement Learning for Real-time Energy Management of Microgrid

Jianquan Zhu, *Senior Member*, *IEEE*, Dongying Li, Yixi Chen, *Graduate Student Member*, *IEEE*, Jiajun Chen, and Yuhao Luo

***Abstract*—This paper proposes a novel parallel hybrid deep reinforcement learning (DRL) approach to address the real-time energy management problem for microgrid (MG). As the proposed approach can directly approximate a discrete-continuous hybrid policy, it does not require the discretization of continuous actions like regular DRL approaches, which avoids accuracy degradation and the curse of dimensionality. In addition, a novel experience-sharing-based parallel technique is further developed for the proposed approach to accelerate the training speed and enhance the training robustness. Finally, a safety projection technique is introduced and incorporated into the proposed approach to improve the decision feasibility. Comparative numerical simulations with several existing MG real-time energy management approaches (i.e., myopic policy, model predictive control, and regular DRL approaches) demonstrate the effectiveness and superiority of the proposed approach.**

***Index Terms*—Deep reinforcement learning, real-time energy management, microgrid, hybrid policy, experience-sharing-based parallel technique, safety projection.**

I. Introduction

DUE to the detrimental effects of fossil fuels on the environment and the decreasing costs of renewable energy sources (RESs), RES deployment has witnessed a significant upswing worldwide [1]. Microgrid (MG) has emerged as a powerful technology for harnessing distributed RESs, enabling the effective integration of diverse energy sources and loads to achieve regional power self-balancing [2]. However, due to the inherent uncertainty and uncontrollability of RESs, the reliable, economical, and intelligent operation of MG has become a major challenge [3]. As a vital tool for optimizing MG operations, the MG real-time energy management (REM) problem has been extensively studied, leading to the development of various approaches [4].

As a classic approach, the myopic policy [5] can provide optimal real-time decisions with rapid computation. However, it is concerned only with gains during the current period, which leads to less satisfactory optimal decisions for the long-term operation of MG [6]. As an improved approach to the myopic policy [7], [8], model predictive control (MPC) enables decision-making for the current period while considering future implications by incorporating near-future forecasting information [9], [10]. However, MPC performance can be affected by the accuracy of forecasting information and the length of look-ahead time horizon [11].

In recent years, the Markov decision process (MDP)-based approaches have emerged as superior and promising alternative solutions to the REM problem of MG [12]. Unlike MPC, MDP-based approaches mitigate dependence on forecasting data, inherently accommodate the stochastic properties of environmental variables, and optimize long-term decisions by maximizing the expected cumulative rewards [13]. In general, MDP-based approaches encompass two major branches, i. e., approximate dynamic programming (ADP) and deep reinforcement learning (DRL). ADP approaches can obtain near-optimal online decisions based on the current system state and well-trained value functions [7], [8], [14]. However, ADP approaches are performed under a model-based paradigm, which makes their performance highly dependent on the modeling accuracy and uncertainty characterization method. In addition, the necessity of these model-based approaches to resolving the complex optimization problem at each time slot incurs substantial computational costs, which greatly impedes real-time decision-making [15].

To address the inherent limitations of model-based ADP approaches, a growing trend toward the application of model-free DRL approaches in the REM of MG has emerged [16]. DRL approaches do not rely on explicit models, which makes them suitable for complex and uncertain environments [17]. Unlike model-based approaches, DRL can rapidly derive real-time scheduling decisions on millisecond timescales [15]. Research on DRL-based REM solutions of MG has generally been categorized into the following two types [16].

1) Value-based approaches. These approaches learn the state or state-action values and choose the action with the highest value in the state. In [18], a value-based approach known as a deep Q-network (DQN) was first utilized in the MG REM problem, which represented the start of a new research area. In [19], a DQN was applied to a more complex

Manuscript received: June 24, 2024; revised: July 29, 2024; accepted: August 22, 2024. Date of CrossCheck: August 22, 2024. Date of online publication: September 5, 2024.

This work was supported in part by the National Natural Science Foundation of China (No. 51977081) and the Natural Science Foundation of Guangdong Province (No. 2022A1515011193).

J. Zhu (corresponding author), D. Li, Y. Chen, J. Chen, and Y. Luo are with the School of Electric Power, South China University of Technology, Guangzhou, China (e-mail: zhujianquan@scut. edu. cn; dongyingli107@foxmail. com; 1402574623@qq.com; epchenjiajun@mail.scut.edu.cn; 782551134@qq.com).

DOI: 10.35833/MPCE.2024.000662

MG model that considered uncertainties in loads, RESs, and electricity prices. In [20], a variant of the DQN, namely the branching dueling *Q*-network (BDQ) algorithm, was proposed for the REM problem of MG with distributed battery energy storage systems (ESSs). The BDQ algorithm is highly scalable and allows the outputs of the neural network to increase linearly with the number of battery ESSs. Recently, a novel NoisyNet-dueling double DQN algorithm was introduced in [21] for the power allocation of various components within a hydrogen gas station MG, where the NoisyNet can aid efficient exploration and the dueling network can generalize learning across actions. However, these studies that use value-based approaches cannot handle continuous actions, hindering their ability to finely schedule actions such as the charging and discharging power of ESSs.

2) Policy-based approaches. These approaches directly learn the policy function that maps the state and action, allowing them to adapt to the continuous action space problem through either a deterministic or stochastic policy form. As a representative deterministic policy algorithm, a deep deterministic policy gradient (DDPG) was utilized in [22] to determine the optimal control strategy for a battery in an MG. In [23], a novel finite-horizon DDPG algorithm was developed for the REM problem of a smart isolated MG to address the instability problem of DRL and the unique characteristics of the finite-horizon model. Unlike deterministic policy algorithms that output a single value, stochastic policy algorithms offer probabilistic policies that allow for more diverse and exploratory decision-making processes. Representative algorithms in this category include the proximal policy optimization (PPO) and asynchronous advantage actor-critic (A3C). In [24], the PPO algorithm was used to address the REM problem of MG, demonstrating superior performance in terms of accuracy and computational stability compared with the DQN and DDPG algorithms. In [25], an improved A3C algorithm integrating experience replay and a semi-deterministic training phase was proposed to tackle the multi-task REM problem of MG with multiple sources of flexibility.

Although existing research has encouraged the application of DRL techniques in the MG REM, these approaches have the following limitations. ① Existing DRL approaches are limited to handling either discrete or continuous actions. This necessitates the discretization of continuous actions when confronted with the problems involving a hybrid action space [26], e.g., on/off decisions of dispatchable generators (DGs) are discrete actions, while the output power of DGs is continuous. However, this discretization not only degrades the accuracy of results, but may also lead to the curse of dimensionality. ② Existing DRL approaches often require a relatively long training period, which becomes more pronounced when confronted with a significant increase in the action space size [27]. ③ Existing DRL-based MG REM solutions often ignore network power flow constraints to simplify the problem, which may lead to safety issues in real-world applications. In addition, regular DRL approaches incorporate only constraint violations as penalty terms in the reward function [28], [29], making it difficult to ensure the safety of decisions.

To address these limitations, this paper applies a novel parallel hybrid PPO (PH-PPO) algorithm in the MG REM problem with a hybrid action space. The main contributions of this paper are summarized as follows.

1) A novel hybrid actor-critic (H-AC) architecture is developed using the PH-PPO algorithm. Unlike existing DRL approaches that require the discretization of continuous actions when confronted with a discrete-continuous hybrid action space, the proposed approach adopts the H-AC architecture to deal directly and simultaneously with discrete and continuous actions, leading to faster convergence toward a superior solution.

2) An experience-sharing-based parallel technique is developed for the PH-PPO algorithm, which allows multiple agents to explore different environments simultaneously and share their collected experiences. The experience-sharing-based parallel technique fully utilizes the computational resources of multicore central processing unit (CPU) and graphics processing unit (GPU), resulting in accelerated training speed as well as improved training robustness.

3) A safety projection technique is introduced and incorporated into the PH-PPO algorithm, which utilizes the prior-domain knowledge of the MG REM to restrict the output actions within a feasible range, and greatly enhances the decision feasibility.

The remainder of this paper is organized as follows. Section II introduces the mathematical formulation of the MG REM problem. Section III reformulates the MDP. Section IV presents the PH-PPO algorithm in detail. Section V describes case studies. Finally, Section VI concludes this paper.

II. Mathematical Formulation of MG REM Problem

We first formulate a mathematical model of MG REM problem as a mixed-integer nonlinear programming (MINLP) problem. A representative MG configuration is considered comprising DGs such as micro-gas turbines (MTs) and diesel generators (DEs), non-dispatchable generators (NGs) such as wind turbines (WTs) and photovoltaic (PV) panels, ESSs, electrical loads, and an energy management system (EMS). The MG is interconnected to the utility grid, thereby engaging in bidirectional power exchange with the utility grid.

A. Objective Function

The objective of the MG REM problem is to minimize the total operational cost of the MG by efficiently coordinating diverse energy resources and demands within the system while considering the dynamic nature of RESs and load demands. Mathematically, the objective can be expressed as:

$$\min_{x_t} \sum_{t=0}^{T} \Bigg[\sum_{g \in G} (C_g^{DG}(P_{g,t}^{DG}) + C_g^{SUP}(o_{g,t}^{DG})) + C^{EX}(P_t^{EX}) + \sum_{e \in E} C_e^{ESS}(P_{e,t}^{ESS}) \Bigg] \tag{1}$$

$$C_g^{DG}(P_{g,t}^{DG}) = (a_g (P_{g,t}^{DG})^2 + b_g P_{g,t}^{DG} + c_g)\Delta t \tag{2}$$

$$C_g^{SUP}(o_{g,t}^{DG}) = l_g^{SUP} o_{g,t}^{DG}(1 - o_{g,t-\Delta t}^{DG}) \tag{3}$$

$$C^{EX}(P_t^{EX}) = p_t P_t^{EX} \Delta t \tag{4}$$

$$C_e^{ESS}(P_{e,t}^{ESS}) = l_e^{ESS} |P_{e,t}^{ESS}| \Delta t \tag{5}$$

where x_t is the decision variable; T is the scheduling period; t is the index of time; G is the set of DGs; E is the set of ESSs; Δt is the time interval; C_g^{DG} is the fuel cost of DGs and is formulated as a quadratic function of the active output power of dispatchable units $P_{g,t}^{DG}$, as shown in (2); a_g, b_g, and c_g are the fuel cost coefficients; C_g^{SUP} is the start-up cost of DGs and can be calculated by (3); $o_{g,t}^{DG}$ is the on/off status of DGs (1 for operation and 0 for shutdown); l_g^{SUP} is the start-up cost of generator g; C^{EX} is the power exchange cost with the utility grid, which settles the trading power P_t^{EX} by real-time price p_t, as shown in (4); p_t represents both the electricity purchasing price and feed-in tariff of the MG and is similar to those in [21] and [29]; C_e^{ESS} is the operational cost of ESSs and is proportional to the output power of ESSs $P_{e,t}^{ESS}$, as shown in (5); and l_e^{ESS} is the operational cost coefficient.

B. Constraints

The MG system is governed by the following constraints.

1) Capacity Constraints

$$P_{g,\min}^{DG} o_{g,t}^{DG} \le P_{g,t}^{DG} \le P_{g,\max}^{DG} o_{g,t}^{DG} \quad \forall g \in G \tag{6}$$

where $P_{g,\max}^{DG}$ and $P_{g,\min}^{DG}$ are the upper and lower boundaries of the active power generated by the DGs, respectively.

2) Ramping Rate Constraints

$$R_{g,down}^{DG} \Delta t \le P_{g,t}^{DG} - P_{g,t-\Delta t}^{DG} \le R_{g,up}^{DG} \Delta t \quad \forall g \in G \tag{7}$$

where $R_{g,up}^{DG}$ and $R_{g,down}^{DG}$ are the maximum upward and downward ramping rates of the DGs, respectively.

3) Minimum On/off Time Constraints

$$\begin{cases} (o_{g,t-\Delta t}^{DG} - o_{g,t}^{DG})(S_{g,t-\Delta t}^{on} - T_{g,on}) \ge 0 \\ (o_{g,t}^{DG} - o_{g,t-\Delta t}^{DG})(S_{g,t-\Delta t}^{off} - T_{g,off}) \ge 0 \end{cases} \quad \forall g \in G \tag{8}$$

where $S_{g,t-\Delta t}^{on}$ and $S_{g,t-\Delta t}^{off}$ are the on and off time counters of the unit g until time $t-\Delta t$, respectively; and $T_{g,on}$ and $T_{g,off}$ are the minimum on and off time, respectively.

4) Power Exchange Constraints

$$P_{\min}^{EX} \le P_t^{EX} \le P_{\max}^{EX} \tag{9}$$

where $P_{\min}^{EX}$ and $P_{\max}^{EX}$ are the minimum and maximum power exchanges between the MG and utility grid, respectively.

5) Bus Voltage and Phase Angle Constraints

$$U_{i,\min} \le U_{i,t} \le U_{i,\max} \quad \forall i \in I \tag{10}$$

$$-\pi \le \delta_{i,t} \le \pi \quad \forall i \in I \tag{11}$$

where $U_{i,t}$ and $\delta_{i,t}$ are the voltage magnitude and phase angle of bus i, respectively; $U_{i,\min}$ and $U_{i,\max}$ are the minimum and maximum allowable voltage magnitudes, respectively; and I is the set of buses.

6) Power Flow Constraints

$$\begin{cases} \sum_{s \in S} M_{i,s} P_{s,t}^{IE} - P_{i,t}^{D} = U_{i,t} \sum_{j \in I} U_{j,t} (G_{ij} \cos \delta_{ij,t} + B_{ij} \sin \delta_{ij,t}) \\ \sum_{s \in S} M_{i,s} Q_{s,t}^{IE} - Q_{i,t}^{D} = U_{i,t} \sum_{j \in I} U_{j,t} (G_{ij} \sin \delta_{ij,t} + B_{ij} \cos \delta_{ij,t}) \end{cases} \quad \forall i \in I \tag{12}$$

where $S = \{DG, NG, ESS, EX\}$ is the set of injected elements including DGs, NGs, ESSs, and power exchanges; $M_{i,s}$ is the element in the generator-bus incidence matrix (equal to 1 when generator s is connected to bus i); $P_{i,t}^{D}$ and $Q_{i,t}^{D}$ are the active and reactive loads at bus i, respectively; $P_{s,t}^{IE}$ and $Q_{s,t}^{IE}$ are the active and reactive output power of the injected element s, respectively; G_{ij} and B_{ij} are the real and imaginary parts of row i and column j of the bus admittance matrix, respectively; and δ_{ij} is the phase angle difference between buses i and j.

7) Transmission Line Capacity Constraints

$$P_{ij,t} = g_{ij} U_{i,t}^2 - U_{i,t} U_{j,t} (g_{ij} \cos \delta_{ij,t} - b_{ij} \sin \delta_{ij,t}) \tag{13}$$

$$P_{ij,\min} \le P_{ij,t} \le P_{ij,\max} \quad \forall i,j \in I \tag{14}$$

where g_{ij} and b_{ij} are the conductance and susceptance of the line between buses i and j, respectively; and $P_{ij,\max}$ and $P_{ij,\min}$ are the upper and lower limits of the line transmission power $P_{ij,t}$ between buses i and j, respectively.

8) ESS Constraints

Two binary variables, $u_{e,t}^{ch}$ and $u_{e,t}^{dis}$, are employed to represent the charging and discharging states of the ESS, respectively. $u_{e,t}^{ch} = 1$ and $u_{e,t}^{dis} = 0$ indicate the charging mode, whereas $u_{e,t}^{ch} = 0$ and $u_{e,t}^{dis} = 1$ indicate the discharging mode. Let us denote the maximum allowed charging and discharging power as $P_{e,\max}^{ch}$ and $P_{e,\max}^{dis}$, respectively. We then have:

$$\begin{cases} 0 \le P_{e,t}^{ch} \le u_{e,t}^{ch} P_{e,\max}^{ch} \\ 0 \le P_{e,t}^{dis} \le u_{e,t}^{dis} P_{e,\max}^{dis} \end{cases} \quad e \in E \tag{15}$$

$$u_{e,t}^{dis} + u_{e,t}^{ch} \le 1 \quad e \in E \tag{16}$$

$$P_{e,t}^{ESS} = u_{e,t}^{dis} P_{e,t}^{dis} - u_{e,t}^{ch} P_{e,t}^{ch} \quad e \in E \tag{17}$$

where $P_{e,t}^{ch}$ and $P_{e,t}^{dis}$ are the charging and discharging power of ESSs, respectively. Let us denote the energy amount currently stored in ESSs as $E_{e,t}^{ESS}$. The dynamics of $E_{e,t}^{ESS}$ are described as:

$$E_{e,t}^{ESS} = E_{e,t-\Delta t}^{ESS} + \eta_e^{ch} P_{e,t}^{ch} \Delta t - P_{e,t}^{dis} \Delta t / \eta_e^{dis} \quad e \in E \tag{18}$$

$$E_{e,\min}^{ESS} \le E_{e,t}^{ESS} \le E_{e,\max}^{ESS} \quad e \in E \tag{19}$$

where η_e^{ch} and η_e^{dis} are the charging and discharging efficiencies, respectively; and $E_{e,\min}^{ESS}$ and $E_{e,\max}^{ESS}$ are the minimum and maximum energy limits, respectively. Ultimately, the REM problem of MG is mathematically formulated as an MINLP problem, where the objective function is expressed as (1), the constraints are expressed in (6) - (19), and the decision variables are defined by:

$$\begin{aligned} x_t = \{ & P_{g,t}^{DG}, Q_{g,t}^{DG}, o_{g,t}^{DG}, P_t^{EX}, U_{i,t}, \delta_{i,t}, \\ & P_{ij,t}, P_{e,t}^{ch}, P_{e,t}^{dis}, u_{e,t}^{dis}, u_{e,t}^{ch}, P_{e,t}^{ESS}, E_{e,t}^{ESS} \} \end{aligned} \tag{20}$$

It can be observed that this problem is a highly nonconvex nonlinear problem with mixed decision variables. Addressing this problem on a real-time scale can be extremely challenging, particularly when accounting for uncertainties. A DRL approach is next proposed to address this problem.

III. MDP Reformulation

We next map the mathematical model of the MG REM problem to an MDP, which is the mathematical foundation and modeling tool for DRL. The purpose of the MDP is to

provide a framework for the agent to collaboratively find a policy to maximize its total accumulated reward. To achieve this, we describe the components of the MDP to ensure that its outcome also corresponds to the solution to the MG REM problem given in (1)-(19).

An MDP problem consists of a quintuple $\langle S, A, P, r, \gamma \rangle$, where S and A are the state space and action space, respectively; P is the state transition function; r is the reward function; and γ is the discount factor. In each step of an MDP, the agent observes a state $\boldsymbol{s}_t$ from the environment. Based on $\boldsymbol{s}_t \in S$, the agent selects and executes an action $\boldsymbol{a}_t \in A$. Then, the environment transitions to the next state $\boldsymbol{s}_{t+1}$ according to the state transition function $p(\boldsymbol{s}_{t+1}|\boldsymbol{s}_t, \boldsymbol{a}_t)$. The environment then returns a reward $r_t(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1})$ to the agent. This process continues through subsequent time steps until the required state or a predetermined termination condition is reached. These elements are defined as follows.

1) State. The following critical variables are used to form the state space:

$$\boldsymbol{s}_t = [\boldsymbol{o}_{t-1}^{DG}, \boldsymbol{P}_{t-1}^{IE}, \boldsymbol{Q}_{t-1}^{IE}, \boldsymbol{P}_t^{NG}, \boldsymbol{Q}_t^{NG}, \boldsymbol{P}_t^{D}, \boldsymbol{Q}_t^{D}, \boldsymbol{U}_{t-1}, \boldsymbol{P}_{t-1}, \boldsymbol{E}_t, p_t] \quad (21)$$

where $\boldsymbol{o}_{t-1}^{DG}$, $\boldsymbol{P}_{t-1}^{IE}$, $\boldsymbol{Q}_{t-1}^{IE}$, $\boldsymbol{P}_t^{NG}$, $\boldsymbol{Q}_t^{NG}$, $\boldsymbol{P}_t^{D}$, $\boldsymbol{Q}_t^{D}$, $\boldsymbol{U}_{t-1}$, $\boldsymbol{P}_{t-1}$, and $\boldsymbol{E}_t$ are the vectors consisting of $o_{g,t-1}^{DG}$, $P_{s,t-1}^{IE}$, $Q_{s,t-1}^{IE}$, $P_{i,t}^{NG}$, $Q_{i,t}^{NG}$, $P_{i,t}^{D}$, $Q_{i,t}^{D}$, $U_{i,t-1}$, $P_{ij,t-1}$, and $E_{e,t}$, respectively, and $P_{i,t}^{NG}$ and $Q_{i,t}^{NG}$ are the output power of NGs.

2) Action. Given the sequential coupling characteristics exhibited by the output power of the DGs across various time periods, this paper adopts the output power increment as the action variable to decouple the output power of the DGs. Therefore, the action space can be represented by:

$$\boldsymbol{a}_t = [\boldsymbol{dP}_t^{DG}, \boldsymbol{U}_t^{DG}, \boldsymbol{o}_t^{DG}, \boldsymbol{P}_t^{ESS}] \quad (22)$$

where $\boldsymbol{dP}_t^{DG}$ is the active output power increment vector of the DGs; $\boldsymbol{U}_t^{DG}$ is the terminal voltage vector of the DGs; and $\boldsymbol{P}_t^{ESS}$ is the vector consisting of $P_{e,t}^{ESS}$.

3) State transition function. In a real-world MG, state transitions occur spontaneously. However, in simulation scenarios, these transitions should be effectively characterized using the following formulations: in the next state $\boldsymbol{s}_{t+1}$, $P_{g,t}^{IE}$ can be computed according to (23); $\boldsymbol{P}_t$ and $\boldsymbol{E}_{t+1}$ can be determined based on (13) and (18); $\boldsymbol{o}_t^{DG}$, $P_{NG,t}^{IE}$, $Q_{NG,t}^{IE}$, $P_{ESS,t}^{IE}$, $Q_{ESS,t}^{IE}$, $\boldsymbol{P}_{t+1}^{D}$, $\boldsymbol{Q}_{t+1}^{D}$, $U_{g,t}$, and p_t are known states; and the remaining states can be calculated through power flow computation in accordance with (12). In the power flow computation, we choose buses connected to DGs ($P_{g,t}^{IE} \neq 0$) as PV buses, buses connected to the utility grid as slack buses, and the remaining buses within the network framework as PQ buses. The power flow distribution within the power grid is then computed using the Newton-Raphson method as:

$$P_{g,t}^{IE} = dP_{g,t}^{DG} + P_{g,t-1}^{IE} \quad \forall g \in G \quad (23)$$

4) Reward function. The total cost C_t of the MDP problem is defined by (24). To maximize the satisfaction of the inequality constraints within the MINLP problem, we introduce a penalty term D_t to penalize violations, as expressed in (25). The reward function r_t for the MDP problem is then formulated according to the cost and overlimit penalties, as expressed in (26).

$$C_t = \sum_{g \in G} (C_g^{DG} + C_g^{SUP}) + C^{EX} + \sum_{e \in E} C_e^{ESS} \quad (24)$$

$$D_t = \begin{cases} |d_t - d_{\min}| & d_t < d_{\min} \\ 0 & d_{\min} \le d \le d_{\max} \\ |d_t - d_{\max}| & d_t > d_{\max} \end{cases} \quad (25)$$

$$r_t(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1}) = -f_c C_t - f_d D_t - f_f F_t \quad (26)$$

where d_t is the variable constrained by the inequality constraint; $d_{\min}$ and $d_{\max}$ are the lower and upper limits of the inequality constraint, respectively; F_t is a binary variable that equals 1 when the power flow calculation does not converge and 0 when it does; and f_c, f_d, and f_f are the cost factor, constraint penalty factor, and power flow penalty factor, respectively, and f_f is a large constant.

Thus, the REM problem of MG is redefined as an MDP with a hybrid action space, which can be solved using regular DRL approaches. However, the following limitations may be encountered: ① inability to directly handle the hybrid action space; ② slow training speed; and ③ suboptimal feasibility of results. To overcome these limitations, the PH-PPO algorithm is applied.

IV. PH-PPO Algorithm

This section describes the PH-PPO algorithm in detail, including an H-AC architecture, an experience-sharing-based parallel technique, and a safety projection technique that helps overcome the three aforementioned limitations.

A. H-AC Architecture

Conventional DRL approaches can only address either a continuous or discrete action space. For the aforementioned MDP problem with a hybrid action space, a conventional DRL approach must first discretize the continuous actions, which may lead to decreased accuracy and the curse of dimensionality. For example, if all the continuous actions are discretized into Z levels, the action space would consist of $Z^{N_{DG}} \times Z^{N_{DG}} \times 2^{N_{DG}} \times Z^{N_{ESS}}$ distinct choices (corresponding to actions $\boldsymbol{dP}_t^{DG}$, $\boldsymbol{U}_t^{DG}$, $\boldsymbol{o}_t^{DG}$, and $\boldsymbol{P}_t^{ESS}$, respectively), where N_{DG} and N_{ESS} are the numbers of DGs and ESSs, respectively. In this type of paradigm, the solution accuracy depends on the level of discrete granularity. However, an overly fine-grained discretization may lead to the curse of dimensionality, and thus hinder practical applications. To overcome these limitations, an H-AC architecture is developed as follows.

The H-AC architecture is grounded in the actor-critic architecture, which is widely employed in DRL approaches. The actor-critic architecture consists of two main components: an actor network that selects actions based on the policy, and a critic network that estimates the value function to compute the gradient of the parameters of the actor network. However, the H-AC architecture, which is tailored to address the hybrid action space problem, differs from the traditional actor-critic architecture which incorporates two actor networks. Figure 1 shows the H-AC architecture, where the discrete actor network is designed to learn a stochastic policy π^d to select discrete actions $\boldsymbol{a}_t^d$, and the continuous actor network learns a stochastic policy π^c to choose continuous actions $\boldsymbol{a}_t^c$. The hybrid policy π represents the joint distribution of independent policy distributions π^d and π^c. The two actor networks share the same state information by sharing the

first few layers of the neural network. The critic network is used to estimate the state-value function, which is then used to compute the advantage function.

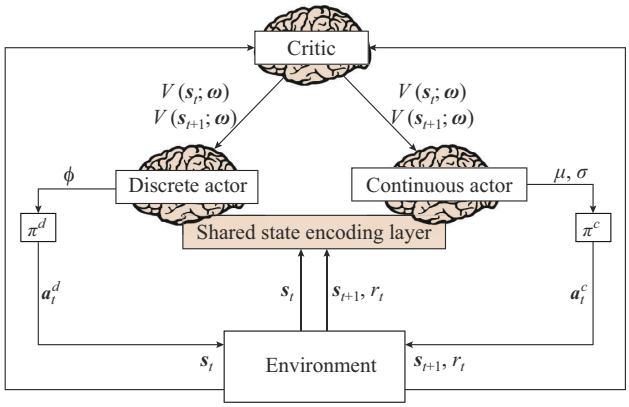

Fig. 1. H-AC architecture.

The detailed form of policy distributions π^d and π^c can be expressed as:

$$\begin{cases} \pi_i^d(a_{i,t}^d|\boldsymbol{s}_t;\boldsymbol{\theta}^d) = Cat(\phi_{i,1}(\boldsymbol{s}_t),\phi_{i,2}(\boldsymbol{s}_t),...,\phi_{i,k}(\boldsymbol{s}_t)) \\ \sum_{i=1}^{K_i}\phi_{i,k}(\boldsymbol{s}_t)=1 \quad i=1,2,...,D \end{cases} \tag{27}$$

$$\pi_i^c(a_{i,t}^c|\boldsymbol{s}_t;\boldsymbol{\theta}_c) = N(\mu_i(\boldsymbol{s}_t),\sigma_i(\boldsymbol{s}_t)) \quad i=1,2,...,C \tag{28}$$

where $a_{i,t}^d$ and $a_{i,t}^c$ are the i^{th} actions of the action vectors $\boldsymbol{a}_t^d$ and $\boldsymbol{a}_t^c$, respectively; $\boldsymbol{\theta}^d$ and $\boldsymbol{\theta}^c$ are the parameters of the two actor networks, respectively; π_i^d and π_i^c are the distributions of $a_{i,t}^d$ and $a_{i,t}^c$, respectively; *Cat* and *N* are the categorical and Gaussian distributions, respectively; K_i is the category count of $a_{i,t}^d$; $\phi_{i,k}$ is the probability that $a_{i,t}^d$ outputs $a_{i,k,t}^d$; μ_i and σ_i are the Gaussian distribution parameters of $a_{i,t}^c$; $\phi_{i,k}$, μ_i, and σ_i are the outputs of the actor network; and D and C are the lengths of $\boldsymbol{a}_t^d$ and $\boldsymbol{a}_t^c$, respectively.

Ideologically, the H-AC architecture shares essential similarities with a fully cooperative multiagent mechanism. It employs two actor networks to handle discrete and continuous actions separately while sharing the observation space, state-encoding layer, and critic network to update the parameters of the actor network. This enables direct adaptation to the hybrid action space and avoids the negative effects of the discretization operation.

B. Hybrid PPO (H-PPO) Algorithm

The H-AC architecture serves only as a foundational framework and requires the selection of appropriate policy optimization algorithms such as trust region policy optimization [26], PPO, and A3C, during concrete implementation. PPO is one of the most state-of-the-art (SOTA) actor-critic architecture algorithms in the field of DRL and is known for its strong stability and versatility. In addition, PPO has the advantage of being easily extended to parallel versions [30]. Therefore, we employ the PPO algorithm as the policy optimization method for both its discrete policy π^d and continuous policy π^c within the PH-AC architecture, resulting in an H-PPO algorithm. The architecture of the PH-PPO algorithm is illustrated in Fig. 2.

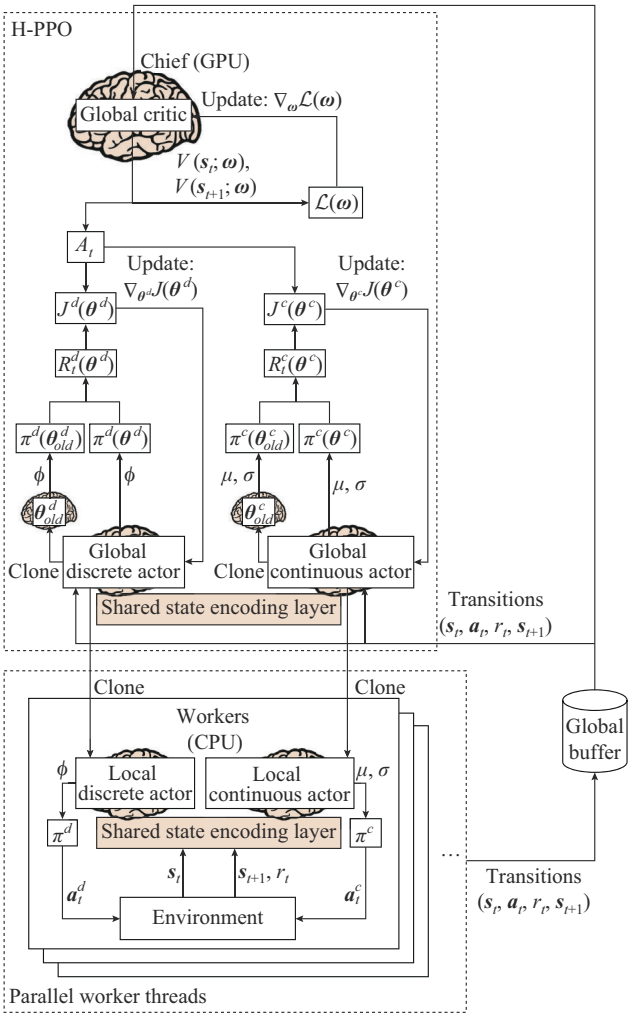

Fig. 2. Architecture of PH-PPO algorithm.

In PPO, the actor and critic networks have different loss functions and update methods. The parameters of the critic network $\boldsymbol{\omega}$ are updated through the optimization of the mean-square error loss function $\mathcal{L}(\boldsymbol{\omega})$:

$$\mathcal{L}(\boldsymbol{\omega}) = \frac{1}{2}(V_{target}(\boldsymbol{s}_t;\boldsymbol{\omega}) - V(\boldsymbol{s}_t;\boldsymbol{\omega}))^2 \tag{29}$$

$$V_{target}(\boldsymbol{s}_t;\boldsymbol{\omega}) = r_t + \gamma V(\boldsymbol{s}_{t+1};\boldsymbol{\omega}) \tag{30}$$

$$\boldsymbol{\omega} \leftarrow \boldsymbol{\omega} - \tau_{critic}\nabla_{\boldsymbol{\omega}}\mathcal{L}(\boldsymbol{\omega}) \tag{31}$$

where $V(\boldsymbol{s}_t;\boldsymbol{\omega})$ is the value of the current state $\boldsymbol{s}_t$ estimated by the critic network; $V_{target}(\boldsymbol{s}_t;\boldsymbol{\omega})$ is the temporal difference (TD) target; and τ_{critic} is the learning rate of the critic network.

The parameters of the actor network $\boldsymbol{\theta}$ are updated through the optimization of the objective function $\mathcal{L}(\boldsymbol{\theta})$:

$$\mathcal{L}(\boldsymbol{\theta}) = E_{(\boldsymbol{s}_t,\boldsymbol{a}_t)\sim\pi(\cdot;\boldsymbol{\theta}_{old})}(\min(R_t(\boldsymbol{\theta})A_t, clip(R_t(\boldsymbol{\theta}),1-\epsilon,1+\epsilon)A_t) \tag{32}$$

$$R_t(\boldsymbol{\theta}) = \frac{\pi(\boldsymbol{a}_t|\boldsymbol{s}_t;\boldsymbol{\theta})}{\pi(\boldsymbol{a}_t|\boldsymbol{s}_t;\boldsymbol{\theta}_{old})} \tag{33}$$

where $\boldsymbol{\theta}_{old}$ is the parameter of the actor network under the old policy; $R_t(\boldsymbol{\theta})$ is the probability ratio, which serves as a metric for assessing the similarity between the new policy and old policies; the *clip* function constrains $R_t(\boldsymbol{\theta})$ within

$1-\epsilon$ and $1+\epsilon$, which restricts the magnitude of updates to the new policy; ϵ is a hyperparameter that controls the degree of clipping; and A_t is the advantage function. PPO exhibits the characteristics of a small deviation and large variance. However, in DRL, deviation can lead to local optima, whereas variance can result in low data utilization. Therefore, this paper introduces a generalized advantage estimation (GAE) technique to estimate the advantage function and strike a balance between deviation and variance [31]:

$$A_t=(1-\lambda)\left(\frac{\delta_t}{1-\lambda}+\frac{\gamma\lambda\delta_{t+1}}{1-\lambda}+\frac{(\gamma\lambda)^2\delta_{t+2}}{1-\lambda}+...\right) \tag{34}$$

$$\delta_t=V_{target}(\boldsymbol{s}_t;\boldsymbol{\omega})-V(\boldsymbol{s}_t;\boldsymbol{\omega}) \tag{35}$$

where $\lambda\in[0,1]$ is an additional GAE hyperparameter; and δ_t is the TD error. At this juncture, $\boldsymbol{\theta}$ can be updated using the gradient ascent as:

$$\boldsymbol{\theta}\leftarrow\boldsymbol{\theta}+\tau_{actor}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}) \tag{36}$$

where τ_{actor} is the learning rate of the actor network. In the H-PPO algorithm, both discrete and continuous policies have their own loss functions, which are indicated in (32). In their own loss functions, the probability ratio $R_t^d(\boldsymbol{\theta}^d)$ considers only the discrete policy, whereas $R_t^d(\boldsymbol{\theta}^c)$ considers only the continuous policy.

C. Experience-sharing-based Parallel Technique

In DRL approaches, offline training must sample substantial amounts of data by interacting with the MG REM simulator, which often requires significant CPU time consumption. To mitigate this limitation, we propose an experience-sharing-based parallel technique for the purpose of developing a parallel version of the H-PPO algorithm, which we refer to as the PH-PPO algorithm.

In the PH-PPO algorithm shown in Fig. 2, the chief thread located in the GPU consists of a global continuous actor network, a global discrete actor network, and a global critic network inherited from the H-PPO algorithm. In addition, the PH-PPO algorithm sets up a set of parallel worker threads in multicore CPUs, where each worker thread encompasses a local continuous actor network and a local discrete actor network. During training, multiple worker threads with different random seeds collect data in diverse environments and push them into a global buffer located in the chief thread. These worker threads are solely responsible for data collection and do not engage in gradient calculations or transmit gradients to the chief thread. When the global buffer reaches a cumulative data-quantity threshold, the global networks in the chief thread update themselves by reading the data. At this point, the worker threads are frozen. After the global networks have been updated, they replicate their network parameters onto local networks, and the global buffer is cleared, thus preparing for the subsequent rounds of data acquisition and network updates.

The experience-sharing-based parallel technique allocates sampling tasks to multicore CPUs and assigns a high-density gradient computational task to the GPU, thereby realizing a rational distribution of computational resources and accelerating the training speed. The experience-sharing-based parallel technique also allows multiple agents to explore different environment simultaneously and to share their individual experiences, which helps alleviate the sensitivity of the algorithm to random seeds and contributes to better training robustness.

D. Safety Projection Technique

In regular DRL approaches, violations of the operational constraints in the MG are often integrated as penalty terms into the reward function within the MDP framework [28], as shown in (25) and (26). However, this setting cannot fully guarantee the feasibility of the obtained decisions, hindering their real-world application in REM scenarios of MG with stringent security requirements. To mitigate this limitation, we introduce a safety projection technique into the PH-PPO algorithm that involves policy representation reconstruction and action mask (AM) configuration.

1) Policy Representation Reconstruction

Regular policy-based DRL typically employs a Gaussian distribution as the probability distribution for continuous actions. However, the unbounded nature of the Gaussian distribution can cause actions to fall into infeasible areas during the online execution stage. To address this issue, the probability distribution corresponding to specific actions $\boldsymbol{U}_t^{DG}$, $\boldsymbol{dP}_t^{DG}$, and $\boldsymbol{P}_t^{ESS}$ is reconstructed as a bounded Beta distribution. Consequently, (28) is superseded by (37), and the outputs of the continuous actor network as shown in Figs. 1 and 2 now correspond to parameters α and β instead of μ and σ, respectively. The use of the Beta distribution helps restrict these actions to a feasible bounded interval, which guarantees that the corresponding constraints in (7), (10) (DG buses), and (15)-(17) are completely satisfied.

$$\pi_i^c(\boldsymbol{a}_t^c|\boldsymbol{s}_t;\boldsymbol{\theta}_c)=B(\alpha_i(\boldsymbol{s}_t),\beta_i(\boldsymbol{s}_t))\quad i=1,2,...,C \tag{37}$$

where B is the Beta distribution; and α_i and β_i are the Beta distribution parameters of $a_{i,t}^c$.

2) AM Configuration

In regular policy-based DRL, even invalid or unsafe actions are assigned a nonzero probability. When random policies are used, these invalid or unsafe actions can potentially be sampled during the online execution stage, leading to undesirable system behaviors or even system crashes. In addition, sampling invalid or unsafe actions can impede policy training because the collected experiences related to invalid actions are meaningless and can mislead the direction of policy updates [32]. To address these issues, we adopt the AM configuration, which is designed to enhance the decision feasibility of agent by identifying and masking invalid and unsafe actions that violate either the actual physical constraints or predetermined physical rules based on prior physical knowledge.

In this paper, the proposed AM is presented in (38), where the "if" statement signifies the physical rule utilized to identify the invalid or unsafe action, and the "then" statement represents the mask that masks out the invalid or unsafe action. AM_1 and AM_2 are generated using (19) under the consideration that the output power of the ESSs does not cause the stored energy to exceed the upper and lower limits. AM_3-AM_7 are based on (6), which takes into account that the on/off decision action and power increment action of the DGs are to be coordinated. Specifically, AM_3 ensures that the power increment does not cause the output power to exceed

its upper and lower limits when DGs are continuously on; AM_4 ensures the maximum upward ramping rate limits of power increment when DGs are start-up; AM_5 and AM_6 ensure the maximum downward ramping rate limits of power increment when DGs are turned off; and AM_7 ensures that the power increment action is masked when DGs are continuously off. When the AM configuration is utilized, the corresponding constraints in (6) and (19) can be guaranteed to be fully satisfied.

$$\begin{cases} AM_1: & \text{if } P_{e,t}^{ESS}>0, \text{ then} \\ & P_{e,t}^{ESS}=(E_{e,t}^{ESS}-clip(E_{e,t}^{ESS}-P_{e,t}^{ESS}/\eta_e^{dis},E_{e,\min}^{ESS},E_{e,\max}^{ESS}))\eta_e^{dis} \\ AM_2: & \text{if } P_{e,t}^{ESS}<0, \text{ then} \\ & P_{e,t}^{ESS}=(E_{e,t}^{ESS}-clip(E_{e,t}^{ESS}-P_{e,t}^{ESS}\eta_e^{ch},E_{e,\min}^{ESS},E_{e,\max}^{ESS}))/\eta_e^{ch} \\ AM_3: & \text{if } o_{g,t-1}^{DG}=1 \text{ and } o_{g,t}^{DG}=1, \text{ then} \\ & dP_{g,t}^{DG}=clip(dP_{g,t}^{DG}+P_{g,t-1}^{IE},P_{g,\min}^{IE},P_{g,\max}^{IE})-P_{g,t-1}^{IE} \\ AM_4: & \text{if } o_{g,t-1}^{DG}=0 \text{ and } o_{g,t}^{DG}=1, \text{ then} \\ & dP_{g,t}^{DG}=clip(dP_{g,t}^{DG},P_{g,\min}^{IE},R_{g,up}^{DG}\Delta t) \\ AM_5: & \text{if } o_{g,t-1}^{DG}=1, o_{g,t}^{DG}=0, \text{ and } P_{g,t-1}^{IE}\le -R_{g,down}^{DG}, \text{ then} \\ & dP_{g,t}^{DG}=-P_{g,t-1}^{IE} \\ AM_6: & \text{if } o_{g,t-1}^{DG}=1, o_{g,t}^{DG}=0, \text{ and } P_{g,t-1}^{IE}-R_{g,down}^{DG}, \text{ then} \\ & o_{g,t}^{DG}=1, dP_{g,t}^{DG}=P_{g,\min}^{IE}-P_{g,t-1}^{IE} \\ AM_7: & \text{if } o_{g,t-1}^{DG}=0 \text{ and } o_{g,t}^{DG}=0, \text{ then} \\ & dP_{g,t}^{DG}=0 \end{cases} \tag{38}$$

The safety projection technique restricts the output action within a feasible range, which ensures that the associated inequality constraints in the MINLP problem are fully satisfied, thereby enhancing the decision feasibility. This technique also avoids exploration in the infeasible action intervals, thereby improving exploration efficiency.

V. Case Study

We first introduce the parameter settings used to implement and test the proposed approach. Simulation results and comparisons with other SOTA approaches are then presented to demonstrate the effectiveness and superiority of the proposed approach.

A. Parameter Settings

The training and testing are conducted using a typical 15-bus MG, as illustrated in Fig. 3. In this MG, the injected elements include MT, DE, WT, PV, ESS, and utility grid. Tables I-III list the parameters of the DGs and ESS. Both the resistance and reactance of the MG lines are set to be 0.09 Ω/km [33]. Table IV lists the transmission distances of the MG lines. The wind, solar, and load data used in the simulations are sourced from historical datasets originating from the Grand Est region of France in 2019 [34]. A training dataset consisting of 36-day data is created by randomly selecting 3 days from each month of the year, and a test dataset is constructed by randomly drawing a sample of 30 days from the dataset of 2019. The dynamic electricity price of a Southern California residential area is also adopted [11], as shown in Table V. Similar to [10], [23], and [24], the optimization horizon T is standardized to be 24 hours, and the time interval Δt is set to be 1 hour. Table VI lists the hyperparameters of the H-PPO algorithm. All simulations are conducted on a personal computer with an Intel(R) Core(TM) CPU Model i7-13700 @ 2.10 GHz with RAM of 16.0 GB and NVIDIA GeForce GPU Model RTX 3060 @ 12 GB. For the PH-PPO algorithm, the codes are written using the Python programming language (version 3.9.7) and the Pytorch package (version 1.13.0).

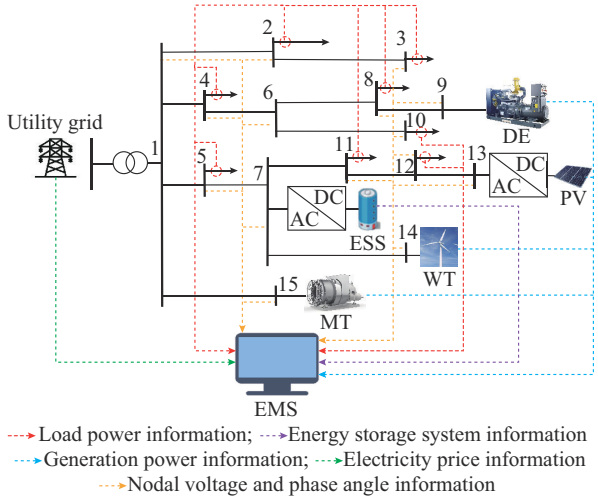

Fig. 3. Typical 15-bus MG.

TABLE I
PARAMETERS OF DGS

DG	$P_{\max}^{DG}$ (kW)	$P_{\min}^{DG}$ (kW)	l^{SUP} ($)	T_{on} (hour)	T_{off} (hour)	R_{up}^{DG} (kW/h)	R_{down}^{DG} (kW/h)
MT	900	50	26	1	1	900	–900
DE	1200	80	30	1	1	1200	–1200

TABLE II
FUEL COST COEFFICIENTS OF DGS

DG	a ($/((kW)^{2}h))	b ($/kWh)	c ($)
MT	3.472×10^{-5}	0.025002	48
DG	3.086×10^{-5}	0.016680	56

TABLE III
PARAMETERS OF ESS

Parameter	Value	Parameter	Value
$P_{\max}^{ch}$ (kW)	400	l^{ESS} ($/kWh)	0.049
$P_{\max}^{dis}$ (kW)	–400	η^{ch}	0.9
$E_{\max}^{ESS}$ (kWh)	1800	η^{dis}	0.9
$E_{\min}^{ESS}$ (kWh)	400		

B. Comparison Studies

A series of case studies are conducted to assess the effectiveness of the proposed approach for the MG REM problem and to showcase its superiority over several SOTA approaches. The performance of the proposed approach is evaluated comprehensively, encompassing both the training and test phases.

TABLE IV
TRANSMISSION DISTANCES OF MG LINES

Line	From bus	To bus	Distance (km)	Line	From bus	To bus	Distance (km)
L1	1	2	1.6	L8	1	5	1.6
L2	2	3	2.8	L9	5	7	1.9
L3	1	4	0.1	L10	7	11	0.3
L4	4	6	3.4	L11	7	14	0.9
L5	6	8	0.3	L12	11	12	1.2
L6	6	10	0.8	L13	12	13	0.2
L7	8	9	1.2	L14	1	15	0.1

TABLE V
ELECTRICITY PRICE

Time period	Price ($/kWh)	Time period	Price ($/kWh)
08:00-14:00	0.14	20:00-22:00	0.14
14:00-20:00	0.24	22:00-08:00	0.06

TABLE VI
HYPERPARAMETERS OF H-PPO ALGORITHM

Parameter	Value	Parameter	Value
Actor learning rate μ_{critic}	1×10^{-5}	GAE hyperparameter λ	0.9
Critic learning rate μ_{actor}	5×10^{-4}	Clipping threshold ϵ	0.2
Discount factor γ	0.96		

1) Effective Validation of H-AC Architecture

To verify the effectiveness of the H-AC architecture, the training process of the H-PPO algorithm is compared with that of the existing PPO algorithm. Notably, if we directly apply the PPO algorithm by discretizing all continuous actions into five levels, the action space is discretized to a size of 125000, making it impossible for the PPO algorithm to explore and converge efficiently in this REM problem. Thus, to facilitate a comparison with the PPO algorithm, we choose to set the voltage of the PV bus where the DGs are located at a fixed value of 1, which simplifies the AC power flow equation, as in [35], [36]. After simplification, the size of action space of the PPO algorithm is reduced to 500.

Figure 4 shows the training curves of H-PPO and PPO algorithms. The curves are averaged over five random seeds, where the shaded region shows the standard deviation. Initially, when the agent has no knowledge of the environment, the selection of actions tends to be random, leading to significant variations in rewards. Following multiple interaction episodes, experiences are accumulated, and the network parameters are optimized accordingly. As the agent learns a better policy, the reward increases gradually until convergence is achieved. The figure shows that the H-PPO algorithm converges after approximately 2500 episodes, whereas the PPO algorithm converges after approximately 4000 episodes. The H-PPO algorithm exhibits a faster learning speed and higher reward as compared with the PPO algorithm. In fact, the PPO algorithm has difficulty in rapidly exploring a satisfactory solution due to the large scale of the action space. Even in the most ideal situation, a suboptimal solution can be approximated with accuracy depending on the granularity of the discretization.

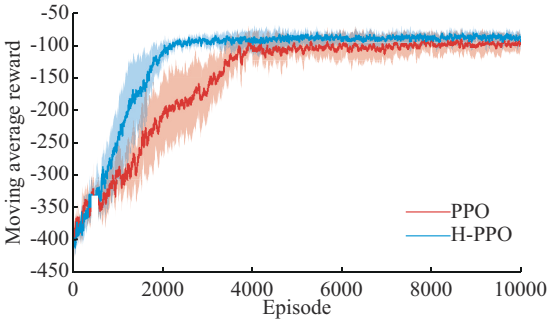

Fig. 4. Comparison of training curves of H-PPO and PPO algorithms.

These findings show that for the PPO algorithm, a small granularity of discretization can result in the curse of dimensionality. By contrast, addressing the dimensionality curse by increasing the granularity may degrade the accuracy. Achieving a satisfactory trade-off between the two poses a significant challenge for the PPO algorithm. Unlike the PPO algorithm, the H-PPO algorithm can handle the hybrid action space directly, effectively avoiding the adverse effects of discretization.

2) Effective Validation of Experience-sharing-based Parallel Technique

To demonstrate the effectiveness of the experience-sharing-based parallel technique, the training process of the PH-PPO algorithm with varying numbers of workers ($n=1,4,8,12$) is investigated. Because different workers must use different random seeds to ensure the diversity of the collected samples, each experiment requires that a random seed cluster is set up. To test the robustness of the proposed approach, experiments are repeated using five random seed clusters. Notably, when the PH-PPO algorithm employs one worker, it is equivalent to the H-PPO algorithm.

Figure 5 shows the training curves. The curves are averaged over five random seed clusters, where the shaded region shows the standard deviation. We can observe that as the number of workers increases, the training speed of the PH-PPO algorithm also increases noticeably, leading to a significant reduction in the time required to reach convergence. This occurs because the experience-sharing-based parallel technique can fully utilize the advantages of multicore CPUs to parallelize the sampling process, thus increasing the efficiency at which samples are collected within a limited period. The figure also shows that the difference in the convergence reward between different numbers of workers is negligible (we utilize an agent trained by eight workers in the test phase). Experimental results confirm that the experience-sharing-based parallel technique can effectively improve the training speed without sacrificing accuracy.

With the exception of the speed advantage, we find that as the number of workers increases, the shaded region of the training curve of the PH-PPO algorithm decreases. This can be explained by the ability of the experience-sharing-based parallel technique to increase sample diversity, as it can integrate all samples related to each random seed within the ran-

dom seed cluster to achieve a more comprehensive and unbiased evaluation. Therefore, once an outlier is sampled by a local actor dominated by a specific random seed, the samples collected by other local actors can help diminish its effects, thus effectively improving the overall training robustness.

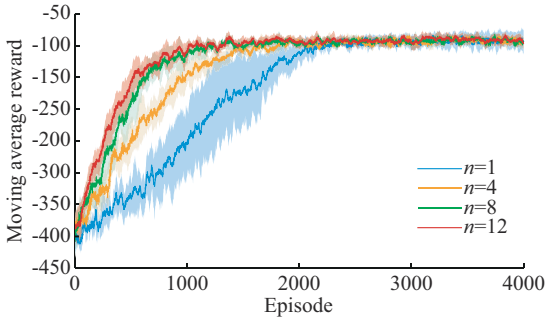

Fig. 5. Training curves of PH-PPO algorithm using different numbers of workers.

3) Effective Validation of Safety Projection Technique

To verify the effectiveness of the safety projection technique, a comparative study is conducted between the complete PH-PPO algorithm and a version that excludes the safety projection technique. For ease of assessment, we introduce the notion of a safe action [37], which is defined as an action that does not violate system constraints during operation.

We use the 30-day test dataset to calculate the safety action ratio of the two versions of the PH-PPO algorithm, as shown in Table VII. The version without the safety projection technique achieves a safety action ratio of only 92.64%, which may be attributed to the agent not having encountered scenarios from the test dataset during training. By contrast, the version with the safety projection technique can achieve a safety action ratio of 99.17%, which may be attributed to the use of prior domain knowledge in the safety projection technique, ensuring strict adherence to certain inequality constraints. Therefore, we can reasonably conclude that the safety projection technique can help improve the feasibility of agent decision-making in unseen scenarios.

TABLE VII
SAFETY ACTION RATIOS UNDER TEST DATASET

Algorithm	Safety action ratio (%)
With safety projection technique	99.17
Without safety projection technique	92.64

4) Comparative Results with Other Approaches

To verify the superiority of the proposed approach, it is compared with other SOTA real-time optimization approaches in terms of test results. The SOTA approaches include the aforementioned PPO algorithm, myopic policy, and MPC. To simulate the effects of sampling errors under these four approaches, random numbers following a Gaussian distribution $N(0,\sigma_s^2)$ are superimposed when sampling the power of the RESs and loads in real time, where σ_s is set to be 1% of the actual value. In the PH-PPO algorithm, the aforementioned three techniques that have been proven to be effective are considered. In the MPC approach, forecasting data for the power of RESs and loads are generated by adding a deviation to the actual values. This deviation is sampled from a Gaussian distribution $N(0,\sigma_p^2)$ in which the standard deviation σ_p is set to be 10% of the actual value. The look-ahead time window for the MPC approach is set to be four hours. The PH-PPO algorithm is also compared with the perfect information optimum (PIO) approach [26], [38], which is considered an ideal day-ahead benchmark experiment. In the PIO approach, we assume that the power of the RESs and loads can be perfectly predicted one day in advance. This allows us to formulate the REM problem of MG as a deterministic optimization problem that can be solved using the LINDO solver. To facilitate a more convenient comparison of the approaches, we introduce the concept of relative cost, which is defined as:

$$C_{rel}=\left(\sum_{t=0}^{T}C_t^{oth}-\sum_{t=0}^{T}C_t^{PIO}\right)\Big/\sum_{t=0}^{T}C_t^{PIO}\times 100\% \tag{39}$$

where C_t^{PIO} and C_t^{oth} are the operating costs obtained by the PIO and other approaches for a specific day, respectively.

After the training process is completed, a well-trained agent is applied to the test dataset. Using the 30-day test dataset, we calculate the daily operation costs of the REM problem of MG under various approaches, where the statistical results are presented in Table VIII. Based on the daily operating costs, the daily relative cost distribution of the various REM approaches is calculated, as illustrated in Fig. 6. The white dots indicate the median obtained by each approach. Based on the statistical indicators, the results demonstrate that the average daily operating cost achieved by the proposed approach is 13.4%, 6.3%, and 4.7% better than those of the myopic policy, MPC, and PPO algorithm, respectively, and only 3.8% worse than that of the PIO approach. The data distribution shows that the daily relative cost of the proposed approach is significantly less than that of the other REM approaches. Notably, although the PIO approach demonstrates the best performance, it is not realistically achievable due to the inherent uncertainties involved. Therefore, this approach serves only as a benchmark experiment for evaluating the performance of the different approaches. By contrast, the myopic policy performs the worst. This could be anticipated because the myopic policy focuses on immediate cost reduction without considering the potential long-term effects of current decisions. Although MPC considers long-term returns, its overall relative cost is higher than those of DRL approaches. This is explained by the deviations between the predicted and actual values of the RESs and loads and by the short time window considered by the MPC, which may affect the accuracy of the control decisions made by the MPC. In addition to considering long-term cumulative rewards, DRL approaches have the advantage of learning policies from real-world historical data that explicitly capture uncertainty characteristics, thereby increasing the likelihood of achieving lower relative costs as compared with other approaches. In DRL approaches, the PH-

PPO algorithm performs better than the PPO algorithm, achieving a lower relative cost and smaller relative cost variation in the test scenarios. The reason for this performance advantage is that the PPO algorithm requires the discretization of actions, which means that the accuracy of the approximate optimal solution depends on the granularity of the discretization. By contrast, the proposed approach can directly handle the REM problem with a hybrid action space, enabling it to achieve a lower relative cost. In addition, we find that DRL approaches can generalize well to the unseen scenarios in the test dataset, which means that they require only a simple neural network mapping time (e.g., 0.003 s) for single-time-step decision-making under real-time application. This advantage makes DRL approaches superior choices for real-time applications.

TABLE VIII
DAILY OPERATING COSTS OF VARIOUS APPROACHES UNDER TEST DATASET

Approach classification	Approach name	Mean cost ($)	The maximum cost ($)	The minimum cost ($)
Day-ahead benchmark	PIO	856.90	1035.50	680.49
REM approach	Myopic	1008.94	1188.43	832.73
	MPC	945.69	1122.00	781.31
	PPO	931.64	1111.50	754.71
	PH-PPO	889.85	1069.98	712.93

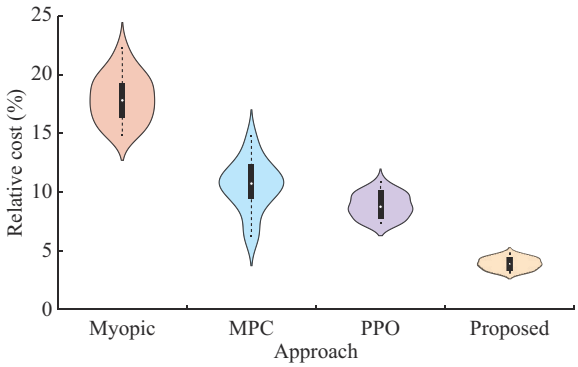

Fig. 6. Violin plot of relative costs of various approaches.

Figures 7 and 8 further present the REM details of the proposed approach for a specific scenario randomly selected from the test dataset. Specifically, Fig. 8 illustrates the on/off decisions of DGs, the output power of various injected elements, the load power, and the energy currently stored in the ESS E_t. The figure clearly shows that when the electricity price is low, the DGs (i.e., MT and DE) are in a shutdown state. During this period, the MG relies on purchasing electricity from the utility grid to fulfill the load demand. When the electricity prices are high, the DGs increase their output power to meet the load demand at a comparatively lower operating cost. This allows the MG to sell the excess power back to the utility grid and generate profits. The agent has also learned to charge the ESS when the price is low and discharge it when the price is high. This strategy helps to reduce the cost of power purchase. This analysis shows that the overall logic of the obtained scheduling results is reasonable, further validating the effectiveness of the proposed approach.

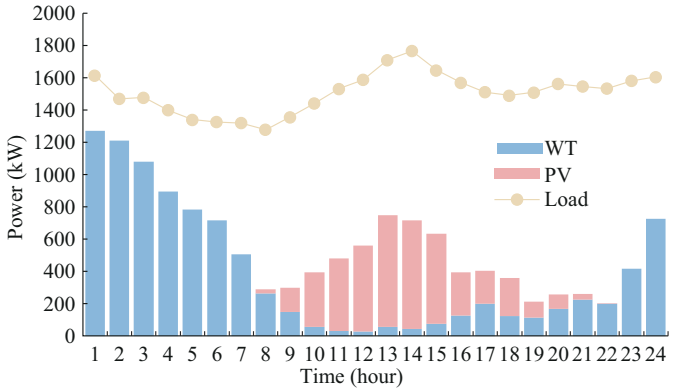

Fig. 7. Power curves of WT, PV, and total load for 24 hours in a given scenario.

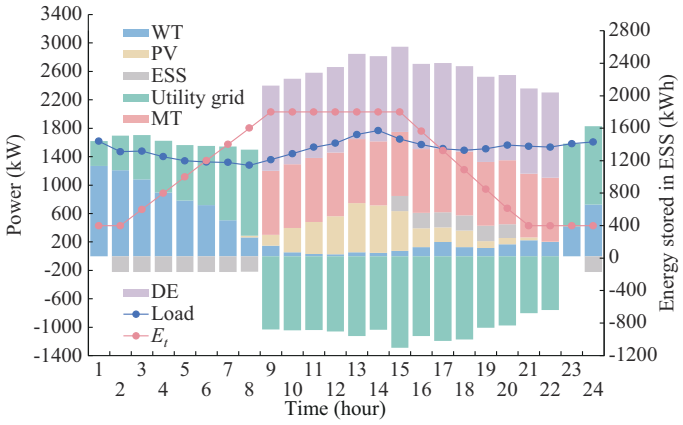

Fig. 8. REM details of proposed approach.

5) Scalability Validation of PH-PPO Algorithm

Similar to [39]-[42], simulations are conducted on a modified IEEE 33-bus MG system composed of four DGs, three PVs, three WTs, and two ESSs to validate the scalability of the proposed approach on larger-scale MG systems. The topology and line parameters of the MG system can be found in the "case33.m" file of MATPOWER. The parameters of the DGs, PVs, and WTs refer to the settings of the aforementioned modified 15-bus MG. The ESS parameters can be found in [40].

The PH-PPO algorithm is compared with the approaches described earlier (i.e., PPO, myopic policy, MPC, and PIO), and the test results are presented in Table IX. The action space of the modified IEEE 33-bus MG system becomes excessively large after discretization, making it impossible for the PPO algorithm to explore and converge efficiently during training. In addition, the table shows that the average daily operation cost achieved by the PH-PPO algorithm is 12.1% and 6.1% better than those of the myopic policy and MPC, respectively, and close to that of the PIO approach, which serves as the ideal benchmark, with only a difference of 4.5%. This means that the proposed approach achieves the best test results among all the REM approaches, demonstrating its scalability for larger-scale MG.

TABLE IX
DAILY OPERATING COSTS UNDER VARIOUS APPROACHES ON MODIFIED IEEE 33-BUS MG SYSTEM

Approach classification	Approach name	Mean cost ($)	Maximum cost ($)	Minimum cost ($)
Day-ahead benchmark	PIO	1681.46	1900.32	1522.80
REM approach	Myopic	1972.04	2262.15	1802.30
	MPC	1867.88	2084.48	1714.18
	PPO			
	PH-PPO	1759.81	2005.94	1595.63

VI. CONCLUSION

In this paper, a novel parallel hybrid DRL approach is proposed for the REM problem of MG. The unit commitment, AC power flow, and uncertainties are considered. The conclusions are as follows.

1) The PH-PPO algorithm adopts an H-AC architecture to handle the hybrid action space directly, which leads to faster convergence toward a superior solution as compared with regular DRL approaches.

2) The PH-PPO algorithm adopts a novel experience-sharing-based parallel technique that can fully utilize the computational resources of multicore CPUs and GPU, thus contributing to an improved convergence speed and training robustness.

3) The PH-PPO algorithm adopts a safety projection technique that can utilize prior-domain knowledge to enhance the feasibility of agent decision-making outcomes, thereby increasing the safety action ratio by 6.53%.

4) The test results confirm that the PH-PPO algorithm offers obvious advantages in terms of accuracy as compared with traditional REM approaches such as the myopic policy and MPC, while ensuring superior generalization and real-time decision-making capabilities.

In a future work, more realistic and refined environmental simulators including finer energy-storage systems, higher temporal resolutions, and more realistic electricity price settings will be considered. In addition, the PH-PPO algorithm could be further extended to a multi-agent DRL framework, providing a solution to the energy management problem of multi-MG systems. Finally, investigating other SOTA DRL approaches (e.g., soft AC) as policy optimization methods to further improve the performance of the PH-PPO algorithm will also be considered.

REFERENCES

[1] Y. Zhuo, J. Zhu, J. Chen *et al*., "RSM-based approximate dynamic programming for stochastic energy management of power systems," *IEEE Transactions on Power Systems*, vol. 38, no. 6, pp. 5392-5405, Nov. 2023.

[2] S. Li, D. Cao, W. Hu *et al*., "Multi-energy management of interconnected multi-microgrid system using multi-agent deep reinforcement learning," *Journal of Modern Power Systems and Clean Energy*, vol. 11, no. 5, pp. 1606-1617, Sept. 2023.

[3] V. Murty and A. Kumar, "Optimal energy management and techno-economic analysis in microgrid with hybrid renewable energy sources," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 5, pp. 929-940, Sept. 2020.

[4] M. F. Zia, E. Elbouchikhi, and M. Benbouzid, "Microgrids energy management systems: a critical review on methods, solutions, and prospects," *Applied Energy*, vol. 222, pp. 1033-1055, Jul. 2018.

[5] W. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Hoboken: Wiley, 2007.

[6] K. B. Gassi and M. Baysal, "Improving real-time energy decision-making model with an actor-critic agent in modern microgrids with energy storage devices," *Energy*, vol. 263, p. 126105, Jan. 2023.

[7] H. Shuai, J. Fang, X. Ai *et al.*, "Stochastic optimization of economic dispatch for microgrid based on approximate dynamic programming," *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 2440-2452, May 2019.

[8] H. Shuai, J. Fang, X. Ai *et al.*, "Optimal real-time operation strategy for microgrid: an ADP-based stochastic nonlinear optimization approach," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 2, pp. 931-942, Apr. 2019.

[9] J. Silvente, G. M. Kopanos, V. Dua *et al*., "A rolling horizon approach for optimal management of microgrids under stochastic uncertainty," *Chemical Engineering Research and Design*, vol. 131, pp. 293-317, Mar. 2018.

[10] Y. Zhang, F. Meng, R. Wang *et al.*, "Uncertainty-resistant stochastic MPC approach for optimal operation of CHP microgrid," *Energy*, vol. 179, pp. 1265-1278, Jul. 2019.

[11] H. Shuai and H. He, "Online scheduling of a residential microgrid via Monte-Carlo tree search and a learned model," *IEEE Transactions on Smart Grid*, vol. 12, no. 2, pp. 1073-1087, Mar. 2021.

[12] X. Liu, T. Zhao, H. Deng *et al.*, "Microgrid energy management with energy storage systems: a review," *CSEE Journal of Power and Energy Systems*, vol. 9, no. 2, pp. 483-504, Mar. 2023.

[13] M. L. Puterman, "Markov decision processes," *Handbooks in Operations Research and Management Science*, vol. 2, pp. 331-434, Jan. 1990.

[14] D. Liu, S. Xue, B. Zhao *et al.*, "Adaptive dynamic programming for control: a survey and recent advances," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 142-160, Jan. 2021.

[15] J. Hu, Y. Ye, Y. Tang *et al.*, "Towards risk-aware real-time security constrained economic dispatch: a tailored deep reinforcement learning approach," *IEEE Transactions on Power Systems*, vol. 39, no. 2, pp. 3972-3986, Mar. 2024.

[16] D. Cao, W. Hu, J. Zhao *et al.*, "Reinforcement learning and its applications in modern power and energy systems: a review," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 6, pp. 1029-1042, Nov. 2020.

[17] H. Zhang, D. Yue, C. Dou *et al.*, "Resilient optimal defensive strategy of TSK fuzzy-model-based microgrids system via a novel reinforcement learning approach," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 4, pp. 1921-1931, Apr. 2023.

[18] V. François-Lavet, D. Taralla, D. Ernst *et al.*（2016, Nov.). Deep reinforcement learning solutions for energy microgrids management. [Online]. Available: http://orbi. ulg. ac. be/bitstream/2268/203831/1/EWRL_Francois-Lavet_et_al.pdf

[19] Y. Ji, J. Wang, J. Xu *et al.*, "Real-time energy management of a microgrid using deep reinforcement learning," *Energies*, vol. 12, no. 12, p. 2291, Jun. 2019.

[20] H. Shuai, F. Li, H. Pulgar-Painemal *et al.*, "Branching dueling Q-network-based online scheduling of a microgrid with distributed energy storage systems," *IEEE Transactions on Smart Grid*, vol. 12, no. 6, pp. 5479-5482, Nov. 2021.

[21] Y. Qi, X. Xu, Y. Liu *et al.*, "Intelligent energy management for an on-grid hydrogen refueling station based on dueling double deep *Q* network algorithm with NoisyNet," *Renewable Energy*, vol. 222, p. 119885, Feb. 2024.

[22] P. Chen, M. Liu, C. Chen *et al.*, "A battery management strategy in microgrid for personalized customer requirements," *Energy*, vol. 189, p. 116245, Dec. 2019.

[23] L. Lei, Y. Tan, G. Dahlenburg *et al.*, "Dynamic energy dispatch based on deep reinforcement learning in IoT-driven smart isolated microgrids," *IEEE Internet of Things Journal*, vol. 8, no. 10, pp. 7938-7953, Dec. 2020.

[24] C. Guo, X. Wang, Y. Zheng *et al.*, "Real-time optimal energy management of microgrid with uncertainties based on deep reinforcement learning," *Energy*, vol. 238, p. 121873, Jan. 2022.

[25] T. Nakabi and P. Toivanen, "Deep reinforcement learning for energy management in a microgrid with flexible demand," *Sustainable Energy, Grids and Networks*, vol. 25, p. 100413, Mar. 2021.

[26] H. Li, Z. Wan, and H. He, "Real-time residential demand response," *IEEE Transactions on Smart Grid*, vol. 11, no. 5, pp. 4144-4154, Sept. 2020.

[27] Y. Chen, J. Zhu, Y. Liu *et al.*, "Distributed hierarchical deep reinforcement learning for large-scale grid emergency control," *IEEE Transactions on Power Systems*, vol. 39, no. 2, pp. 4446-4458, Mar. 2024.
[28] H. Li, Z. Wang, L. Li *et al.*, "Online microgrid energy management based on safe deep reinforcement learning," in *Proceedings of 2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, Orlando, USA, Dec. 2021, pp. 1-8.
[29] T. Lu, R. Hao, Q. Ai *et al.*, "Distributed online dispatch for microgrids using hierarchical reinforcement learning embedded with operation knowledge," *IEEE Transactions on Power Systems*, vol. 38, no. 4, pp. 2989-3002, Jul. 2023.
[30] N. Heess, T. B. Dhruva, S. Sriram *et al*. (2017, Jul.). Emergence of locomotion behaviours in rich environments. [Online]. Available: http://arxiv.org/abs/1707.02286
[31] J. Schulman, F. Wolski, P. Dhariwal *et al.* (2017, Aug.). Proximal policy optimization algorithms. [Online]. Available: http://arxiv. org/abs/1707.06347
[32] D. Chen, M. R. Hajidavalloo, Z. Li *et al.*, "Deep multi-agent reinforcement learning for highway on-ramp merging in mixed traffic," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 11, pp. 11623-11638, Nov. 2023.
[33] J. Zhu, Y. Zhuo, J. Chen *et al.*, "An expected-cost realization-probability optimization approach for the dynamic energy management of microgrid," *International Journal of Electrical Power & Energy Systems*, vol. 136, p. 107620, Mar. 2022.
[34] RTE. (2024, Jan.). éCO$_2$mix. [Online]. Available: https://www. rte-france.com/eco2mix.
[35] P. Tian, X. Xiao, K. Wang *et al.*, "A hierarchical energy management system based on hierarchical optimization for microgrid community economic operation," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2230-2241, Sept. 2016.
[36] X. Xue, X. Ai, J. Fang *et al.*, "Real-time schedule of microgrid for maximizing battery energy storage utilization," *IEEE Transactions on Sustainable Energy*, vol. 13, no. 3, pp. 1356-1369, Jul. 2022.
[37] M. Alshiekh, R. Bloem, R. Ehlers *et al.* (2018, Apr.). Safe reinforcement learning via shielding. [Online]. Available: https://arxiv. org/abs/1708.08611
[38] S. Gao, C. Xiang, M. Yu *et al.*, "Online optimal power scheduling of a microgrid via imitation learning," *IEEE Transactions on Smart Grid*, vol. 13, no. 2, pp. 861-876, Mar. 2022.
[39] N. Zografou-Barredo, C. Patsios, I. Sarantakos *et al.*, "Microgrid resilience-oriented scheduling: a robust misocp model," *IEEE Transactions on Smart Grid*, vol. 12, no. 3, pp. 1867-1879, May 2021.
[40] A. Gholami, T. Shekari, F. Aminifar *et al.*, "Microgrid scheduling with uncertainty: the quest for resilience," *IEEE Transactions on Smart Grid*, vol. 7, no. 6, pp. 2849-2858, Nov. 2016.
[41] S. Zeinal-Kheiri, A. M. Shotorbani, and B. Mohammadi-Ivatloo, "Real-time energy management of grid-connected microgrid with flexible and delay-tolerant loads," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 6, pp. 1196-1207, Nov. 2020.
[42] M. Yin, K. Li, and J. Yu, "A data-driven approach for microgrid distributed generation planning under uncertainties," *Applied Energy*, vol. 309, p. 118429, Jan. 2022.

Jianquan Zhu received the M. S. degree from Guangxi University, Nanning, China, in 2008, and the Ph. D. degree from Tsinghua University, Beijing, China, in 2012. He is currently working as a Professor in South China University of Technology, Guangzhou, China. His research interests include modeling and optimization of power systems.

Dongying Li received the B.S. degree from Guangdong University of Technology, Guangzhou, China, in 2022. She is currently pursuing the M. S. degree in South China University of Technology, Guangzhou, China. Her research interest includes application of deep reinforcement learning in power system optimization.

Yixi Chen received the B. S. degree from South China University of Technology, Guangzhou, China, in 2022. He is currently pursuing the Ph. D. degree in South China University of Technology. His current research interests include power system modeling and simulation, data-driven power system stability and control.

Jiajun Chen received the Ph. D. degree from South China University of Technology, Guangzhou, China, in 2024. He is currently working in South China University of Technology. His research interests include application of approximate dynamic programming and optimization in power systems.

Yuhao Luo received the B.S. degree in South China University of Technology, Guangzhou, China, in 2022. He is currently pursuing the Ph. D. degree in South China University of Technology. His research interests include application of approximate dynamic programming and computational intelligence in power systems.