Multi-Agent Deep Reinforcement Learning for Voltage Control With Coordinated Active and Reactive Power Optimization

Daner Hu, *Student Member, IEEE*, Zhenhui Ye, *Student Member, IEEE*, Yuanqi Gao, *Member, IEEE*, Zuzhao Ye, *Graduate Student Member, IEEE*, Yonggang Peng, *Member, IEEE*, and Nanpeng Yu, *Senior Member, IEEE*

***Abstract*—The increasing penetration of distributed renewable energy resources causes voltage fluctuations in distribution networks. The controllable active and reactive power resources such as energy storage (ES) systems and electric vehicles (EVs) in active distribution networks play an important role in mitigating the voltage excursions. This paper proposes a two-timescale hybrid voltage control strategy based on a mixed-integer optimization method and multi-agent reinforcement learning (MARL) to reduce power loss and mitigate voltage violations. In the slow-timescale, the active and reactive power optimization problem involving capacitor banks (CBs), on-load tap changers (OLTC), and ES systems is formulated as a mixed-integer second-order cone programming problem. In the fast-timescale, the reactive power of smart inverters connected to solar photovoltaic systems and active power of EVs are adjusted to mitigate short-term voltage fluctuations with a MARL algorithm. Specifically, we propose an experience augmented multi-agent actor-critic (EA-MAAC) algorithm with an attention mechanism to learn high-quality control policies. The control policies are executed online in a decentralized manner. The proposed hybrid voltage control strategy is validated on an IEEE testing distribution feeder. The numerical results show that our proposed control strategy is not only sample-efficient and robust but also effective in mitigating voltage fluctuations.**

***Index Terms*—Deep reinforcement learning, experience augmentation, multi-agent, soft actor-critic, voltage control.**

NOMENCLATURE

Acronyms

CB	Capacitor bank
CS	Charging station
CTDE	Centralized training and decentralized execution
CVV	Cumulative voltage violations
DG	Distributed generation
DRL	Deep reinforcement learning
EA	Experience augmentation
ES	Energy storage
EV	Electric vehicles
MAAC	Multi-agent actor-critic
MADDPG	Multi-agent deep deterministic policy gradient
MARL	Multi-agent reinforcement learning
MGP	Markov Game Process
MINLP	Mixed-integer nonlinear programming
MPC	Model predictive control
OLTC	On-load tap changers
PV	Photovoltaics
SAC	Soft actor-critic
SoC	State of charge
SOCR	second-order cone relaxation
VVC	Volt-VAR control.

Parameters

B_n	Batch size
C_{loss}	The cost of power loss
C_β	The penalty coefficient for voltage violation
C_{grid}	Electricity prices
$\bar{K}$	Maximum tap position of transformer tap positions
$l_{ij,t}$	Square of the current magnitude from bus i to j
N	Number of agents
$\mathcal{N}_{PV}$, $\mathcal{N}_{CS}$	Set of solar PV systems and CSs.
$OLTC^{max}$	Maximum number of daily tap position changes for the OLTC
$P_{i,t}^{load}$, $Q_{i,t}^{load}$	Active/Reactive power output of load at bus i
P^{loss}	Total line loss
P^k	Permutation matrix
r_{ij}	Resistance of branch ij
$S_{i,t}^{PV}$	Apparent power of the solar PV at bus i
$\mathbf{s}_{\setminus i}$ and $\mathbf{a}_{\setminus i}$	Set of states and actions in group except agent i
$SoC_i^{min/max}$	Minimum/maximum SoC value of ES system

Manuscript received 9 December 2021; revised 13 April 2022; accepted 16 June 2022. Date of publication 24 June 2022; date of current version 21 October 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2020YFB0906000 and Grant 2020YFB0906002, and in part by the University of California under Award L22CR4556. Paper no. TSG-01956-2021. *(Corresponding author: Nanpeng Yu.)*

Daner Hu, Zhenhui Ye, and Yonggang Peng are with the Department of Electrical Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: pengyg@zju.edu.cn).

Yuanqi Gao, Zuzhao Ye, and Nanpeng Yu are with the Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA 92521 USA (e-mail: nyu@ece.ucr.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSG.2022.3185975.

Digital Object Identifier 10.1109/TSG.2022.3185975

$v_{i,t}$	Primary voltage of the transformer
$\underline{V}$, $\overline{V}$	Maximum and minimum voltage value.

Variables

$a_{i,m,t}^{CB}$	The on/off status of the mth CB at bus i for hour t
b_t^{TAP}	Integer variable indicating the tap position of OLTC
$P_{i,t}^{PV}$,$Q_{i,t}^{PV}$	Active/Reactive power output of the solar PV system
$P_{i,t}^{ES}$,$Q_{i,t}^{ES}$	Active/Reactive power output of ES system for hour t
$P_{Cv,\tau}$	Power set points of CSs
$P_{ij,t}$, $Q_{ij,t}$	Active/Reactive power through branch from bus i to j
$S_{i,t}^{SoC}$	ES system's SoC
$V_{i,t}^2$	Square of voltage
$Q_{Cj,\tau}$	Reactive power set points of solar PV systems.

I. Introduction

THE INCREASING penetration of distributed generation (DG) such as solar photovoltaics (PV) systems has brought more frequent voltage violations and higher losses to the distribution network [1], [2]. To mitigate the issues, Volt-VAR control (VVC) has been adopted to improve voltage quality and reduce network loss in power distribution systems [3].

In conventional VVC implementations, voltage regulating devices such as on-load tap changer (OLTC), capacitor banks (CBs), and voltage regulators are leveraged to mitigate voltage violations and reduce network losses. The inverters connected to DGs are able to absorb and produce reactive power in real-time operations, and thus they are ideal voltage regulating devices [4]. Combining the conventional devices and inverters, many researchers and engineers developed two-timescale VVC [5]. In the slow-timescale, conventional voltage regulating devices are often controlled on an hourly basis. In the fast-timescale, the smart inverters connected to DGs are controlled on a minute-by-minute basis. The above-mentioned voltage controls are primarily reactive power-based approaches. It has been shown that active power controls could also provide voltage regulation services [6].

By adjusting the charging power output, charging stations (CSs) can provide services to both electric vehicles (EVs) and active distribution grids. A model predictive control-based algorithm is developed to minimize charging cost and the impact of EV charging on power grid [7]. An offline coordinated discrete charging model is developed to reshape the net load profile while satisfying the distribution transformer capacity constraint [8]. A model-free approach is proposed to schedule the home charging of EVs considering the time-varying electricity price in distribution systems [9]. On top of adjusting the reactive power set points of solar PV connected inverters, we propose to schedule the charging and discharging of EV batteries in CSs to provide voltage regulation service to the distribution grid in real-time operations.

Conventionally, the active and reactive power optimization problem in distribution grids is formulated as a mixed-integer nonlinear programming (MINLP) problem based on optimal power flow (OPF) [10]. Because the formulated problem is non-convex and NP-hard, existing algorithms cannot guarantee convergence to the global optimal solution [11]. The widely used approaches to handle the non-convexity of the MINLP problem are convex relaxation techniques, such as second-order cone relaxation (SOCR) [12], [13]. Although the physical model-based control approaches achieve good performance in simulation settings, they rely upon exact distributing system models, which may not be available in practice. Furthermore, the computation time of the physical model-based methods usually grows exponentially with the size of the distribution network and number of DGs [14].

To tackle these challenges, reinforcement learning (RL) has been adopted to control active power distribution grids [15]–[19]. A deep Q-network (DQN)-based algorithm was developed for an unbalanced distribution system to reduce power loss and regulate voltage [15]. In [16], a VVC policy that minimizes the total system operation costs is learned through two policy gradient methods. A safe off-policy deep reinforcement learning (DRL) algorithm named constrained soft actor-critic is proposed to solve the VVC problem with discrete action space in [17]. To improve robustness of control algorithms and learning efficiency, the centralized training and decentralized execution (CTDE) framework of multi-agent reinforcement learning (MARL) has been explored in the power system domain. The autonomous voltage control problem was solved by the multi-agent deep deterministic policy gradient (MADDPG) method in [18]. A multi-agent constrained soft actor-critic (MACSAC) algorithm is used to perform VVC in an online environment [19]. A novel consensus multi-agent RL algorithm is proposed to learn distributed VVC policies from historical operational data in a communication efficient manner [20]. However, the low sample efficiency and high operational data collection cost make it difficult to train MARL algorithms in practice for power distribution systems.

Training data augmentation is an effective technique to enhance sample efficiency for RL algorithms. Data augmentation is initially designed for the single-agent RL setup to generate useful samples with low cost [21]. Another approach to generated augmented data for RL algorithm is to scale the amplitude of the original data set [22]. In the field of power distribution systems, a Gaussian process model is developed to produce synthetic training data for RL-based distribution control problem [23]. A surrogate model is developed to provide augmented data for a RL-based voltage control strategy for unbalanced three-phase power distribution systems [24]. A physical model-based training data augmentation technique is proposed for VVC problem in [25]. An RL agent is trained using a learned environment model to improve the sample efficiency for VVC problem [26]. Nevertheless, the topic of training data augmentation for multi-agent RL-based VVC has not been explored.

To fill the research gaps, this paper introduces a novel data augmentation technique called Experience Augmentation

(EA) for MARL-based VVC problem. The proposed algorithm not only provides unbiased synthetic data but also accelerates training data processing by shuffling agents. The generated synthetic operational experience significantly improves the VVC performance. In addition, we propose an innovative MARL algorithm called experience augmented multi-agent actor-critic (EA-MAAC) to solve the fast-timescale voltage control problem by coordinating the operations of solar PV systems and CSs. Furthermore, we theoretically prove the convergence of EA-based soft policy iteration. The main contributions of this paper are listed below:

- A two-timescale hybrid control strategy is proposed to regulate the voltage in active distribution grids. The slow-timescale control decisions of OLTC, CBs, and ES systems are obtained by solving a mixed-integer second-order cone programming (MISOCP) problem, while the fast-timescale operation of smart inverters follows a MARL-based algorithm to mitigate voltage fluctuations.
- We propose an *Experience Augmentation* method to improve the sample efficiency and convergence speed of the MARL-based VVC algorithm. Numerical study results on the IEEE test circuit shows that the Experience Augmentation technique significantly accelerates the training process of RL agents, which consist of smart inverters and CSs in power distribution grids.
- By synergistically combining the Experience Augmentation technique and the CTDE framework, we propose a MARL algorithm named EA-MAAC, which has centralized critics with an attention mechanism. Compared to the state-of-the-art multi-agent policy gradient algorithm such as MADDPG [27], our proposed method not only yields higher sample efficiency and lower operational cost for fast-timescale VVC problems, but it also has a much shorter computation time than model-based algorithms.

The remainder of this paper is organized as follows. Section II introduces the proposed two-timescale active and reactive power optimization framework. The mathematical models and the solution method of voltage control in the slow-timescale are provided in Section III. Section IV formulates the multi-agent voltage control problem under the fast-timescale and presents the novel EA-MAAC algorithm to solve the problem. Section V demonstrates the performance of the proposed hybrid VVC algorithm. Section VI gives the conclusion.

II. Proposed Two-Timescale Active and Reactive Power Optimization Framework

The proposed two-timescale voltage control strategy manages both active and reactive power of energy resources to maintain nodal voltages within an appropriate range. Specifically, a two-timescale hybrid voltage control framework is proposed as shown in Fig. 1.

In the slow-timescale, the CBs, OLTC and ES systems are controlled on an hourly basis according to the MISCOP-based day-ahead OPF results. In the fast-timescale, an active and reactive power coordination strategy is designed to mitigate the voltage deviations. An EA-MAAC based control

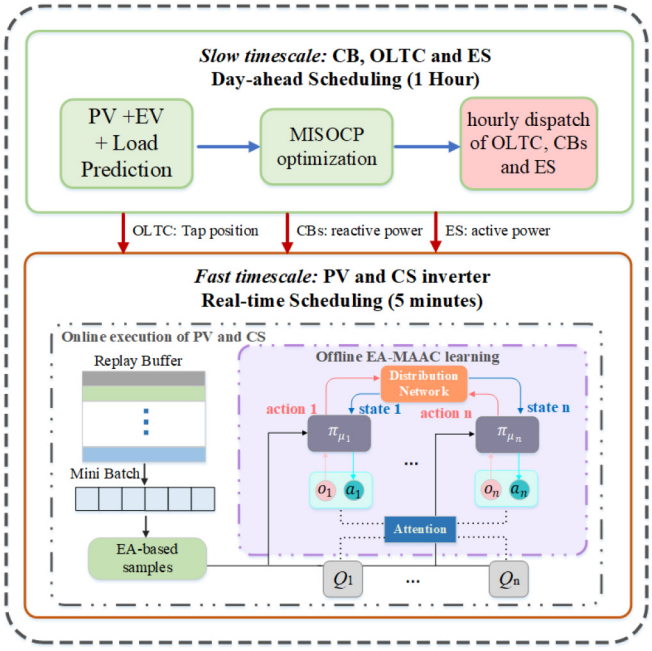


Fig. 1. The architecture of the proposed two-timescale voltage control with coordinated active and reactive power optimization.

algorithm determines the real and reactive power set points of the smart inverters connected to the solar PV systems and CSs on a 5-minute basis. The proposed EA-MAAC algorithm is a DRL-based algorithm, which consists of an offline centralized learning phase and an online decentralized execution phase. In the offline training process, the system states and rewards are stored in a replay buffer, from which the DRL algorithm samples experiences to update the parameters of its deep neural networks. In the online execution phase, the distributed smart inverters make decisions based on the learned control policies and the current system state.

III. Slow-Timescale: Day-Ahead Optimization Model for Voltage Control

A. Problem Formulation

In the slow-timescale, the proposed controller aims at minimizing both the active power loss and the power purchase cost from the transmission grid by controlling OLTC, CBs and ES systems. The control problem is solved by a model predictive control (MPC) strategy with predictions of solar PV generation, CS power consumption and other electric loads.

The objective function of the MPC can be formulated as:

$$\min_{\substack{b_t^{TAP},a_{i,m,t}^{CB}\\ Q_{i,t}^{PV},P_{i,t}^{ES}}} \sum_{t\in T}\left\{C_{grid}(t)P_t^{Grid}+\sum_{(i,j)\in\mathcal{L}}C_{loss}P_{ij,t}^{loss}\right\}, \quad (1)$$

where C_{loss} represents the cost of network loss and $C_{grid}(t)$ denotes the power purchase cost from the transmission grid. T represents the set of operating hours in a day. The binary variable $a_{i,m,t}^{CB}$ denotes the on/off status of the mth CB at bus i for hour t. b_t^{TAP} is an integer variable indicating the tap position of OLTC for hour t. $Q_{i,t}^{PV}$ and $P_{i,t}^{ES}$ indicate the reactive power

output of the solar PV system and the active power output of ES system for hour t, respectively. Note that only a subset of the day-ahead decision variables $b_t^{TAP}, a_{i,m,t}^{CB}$ and $P_{i,t}^{ES}$ are implemented in the next day. The dispatch of reactive power $Q_{i,t}^{PV}$ are re-optimized in the intra-day operation process.

The network loss of a distribution branch is calculated as:

$$P_{ij,t}^{loss} = r_{ij} l_{ij,t}, \quad \forall ij, \ \forall t \tag{2}$$

where r_{ij} is the resistance of branch ij and l_{ij} is the square of the current magnitude from bus i to j. The amount of real power purchase from the transmission grid P_t^{Grid} can be calculated as:

$$P_t^{Grid} = \sum_{j \in Ne(i)} P_{ij,t}, \quad \forall t \tag{3}$$

where i denotes the point of coupling to the transmission grid and Ne is a set of buses of the distribution network connecting to it. Note that this set of buses does not include the high voltage side of the transmission system bus bar.

The operational constraints of the MPC-based controller are listed below. (4) and (5) represent the switching constraints of CBs and OLTC, respectively. $CB_{i,m}^{max}$ denotes the maximum number of daily switching operations for the mth CB. $OLTC^{max}$ denotes the maximum number of daily tap position changes of the OLTC. b_t^{TAP} is an integer variable denoting the tap position within the range of [-5, 5]. (6) calculates the reactive power output of the CBs at bus i. The downstream voltage of the OLTC $v_{1,t}$ can be calculated by (7), where the distribution feeder head voltage is V_0 and $\Delta V_{TAP} = 0.01$ is the step size of the voltage regulator.

$$\sum_{t \in T} \left| a_{i,m,t+1}^{CB} - a_{i,m,t}^{CB} \right| \le CB_{i,m}^{max}, \ a_{i,m,t}^{CB} \in \{0, 1\}, \tag{4}$$

$$\sum_{t \in T} |b_{t+1}^{TAP} - b_t^{TAP}| \le OLTC^{max}, \tag{5}$$

$$Q_{i,t}^{CB} = \sum_m a_{i,m,t}^{CB} Q_{i,m,t}^{CB}, \quad \forall i, t \tag{6}$$

$$V_{1,t} = V_0 + b_t^{TAP} \Delta V_{TAP}, \quad \forall t \tag{7}$$

To handle the absolute values of the form $|X - Y|$ in (4) and (5), an auxiliary variable Z can be added. This helps transform the nonlinear formulation into the linear constraints of $Z \ge X - Y$ and $Z \ge Y - X$ [28].

$$\begin{cases} \omega_t^+ + \omega_t^- = 1 \\ 0 \le P_{i,t}^{ES+} \le P_{i,t,max}^{ES+} \omega_t^+, \ 0 \le P_{i,t}^{ES-} \le P_{i,t,max}^{ES-} \omega_t^- \\ {P_{i,t}^{ES+}}^2 + Q_{i,t}^{ES^2} \le S_{i,t}^{ES^2}, \ {P_{i,t}^{ES-}}^2 + Q_{i,t}^{ES^2} \le S_{i,t}^{ES^2} \\ S_{i,t+1}^{SoC} = S_{i,t}^{SoC} + \frac{P_{i,t}^{ES+} \eta_+ \Delta t}{E_{rate}} - \frac{P_{i,t}^{ES-} \eta_- \Delta t}{E_{rate}} \\ SoC_i^{min} \le S_{i,t}^{SoC} \le SoC_i^{max} \quad \forall t, i \end{cases} \tag{8}$$

The operational constraints of the ES system are shown in (8). At each time step t, the ES system is assumed to be either in the charging or discharging state, which are denoted by the binary variables ω_t^+ and ω_t^-. The charging $P_{i,t}^{ES+}$ and discharging power $P_{i,t}^{ES-}$ of the ES system located at bus i are limited by the maximum charging $P_{i,t,max}^{ES+}$ and discharging power rates $P_{i,t,max}^{ES-}$. The combination of real and reactive power outputs $Q_{i,t}^{ES}$ of the ES system are limited by the maximum apparent power $S_{i,t}^{ES}$. The state of charge (SoC) of the ES system must satisfy the temporal constraints, where E_{rate} represents the rated capacity of the ES system, η_+ and η_- are the charging and discharging efficiency. The ES system's SoC $S_{i,t}^{SoC}$ should be limited by its minimum value SoC_i^{min} and maximum value SoC_i^{max}.

$$\begin{cases} P_{i,t}^{PV} + P_{i,t}^{ES-} - P_{i,t}^{ES+} - P_{i,t}^{load} - P_{i,t}^{CS} = \sum_{g \in G(i)} P_{ig,t} \\ \quad - \sum_{j \in I(i)} \left(P_{ij,t} - r_{ij} l_{ij,t} \right) \quad \forall i, t \\ Q_{i,t}^{CB} + Q_{i,t}^{PV} - Q_{i,t}^{load} = \sum_{g \in G(i)} Q_{ig,t} \\ \quad - \sum_{j \in I(i)} \left(Q_{ij,t} - r_{ij} l_{ij,t} \right) \quad \forall i, t \end{cases} \tag{9}$$

$$V_{j,t}^2 = V_{i,t}^2 - 2\left(r_{ij} P_{ij,t} + x_{ij} Q_{ij,t}\right) + \left(r_{ij}^2 + x_{ij}^2\right) l_{ij,t}, \quad \forall ij, t \tag{10}$$

$$l_{ij,t} = \left(P_{ij,t}^2 + Q_{ij,t}^2\right) / \ V_{i,t}^2, \quad \forall ij, t \tag{11}$$

$$\left|Q_{i,t}^{PV}\right| \le \sqrt{\left(S_{i,t}^{PV}\right)^2 - \left(P_{i,t}^{PV}\right)^2}, \quad \forall i, t \tag{12}$$

$$V_i^{min} \le V_{i,t} \le V_i^{max}. \quad \forall i, t \tag{13}$$

The operational constraints of the active distribution network are summarized in (9)–(13). Equations (9)–(11) are the nodal power balance equations in Distflow format [29]. The electric load is modeled by the active $P_{i,t}^{load}$ and reactive $Q_{i,t}^{load}$ load. The CS power consumption is given by $P_{i,t}^{CS}$ in (9). Equation (12) shows that the reactive power set point of the smart inverter of each solar PV system is limited by the corresponding apparent power $S_{i,t}^{PV}$ and real power output $P_{i,t}^{PV}$. Equation (13) indicates the lower V_i^{min} and upper V_i^{max} bounds of the allowed voltage range at node i.

B. Second-Order Conic Relaxation

Note that the MPC-based optimization problem cannot be solved directly because of the nonconvex branch flow equations (9) - (11). Instead of directly solving the mixed integer nonlinear nonconvex programming problem, we apply the second-order conic relaxation (SOCR) method to relax the nonconvex branch flow constraints [12], [13]. Specifically, we use $v_{i,t}$ to substitute the square of voltage $V_{i,t}^2$, then (11) could be rewritten as:

$$l_{ij,t} = \left(P_{ij,t}^2 + Q_{ij,t}^2\right) / \ v_{i,t}, \quad \forall ij, t \tag{14}$$

By following the SOCR technique, (14) could be transformed to the standard second-order conic quadratic inequality constraint as follows:

$$\left\| \begin{matrix} 2P_{ij,t} \\ 2Q_{ij,t} \\ l_{ij,t} - v_{i,t} \end{matrix} \right\|_2 \le l_{ij,t} + v_{i,t}, \quad \forall ij, t \tag{15}$$

Note that under certain conditions, SOC relaxation is exact which means the set of inequalities (15) remain equal at the optimum [30], [31].

Similarly, constraints (10) and (13) can be rewritten as:

$$v_{j,t} = v_{i,t} - 2\left(r_{ij} P_{ij,t} + x_{ij} Q_{ij,t}\right) + \left(r_{ij}^2 + x_{ij}^2\right) l_{ij,t}, \quad \forall ij, t \tag{16}$$

$$\left(V_i^{min}\right)^2 \le v_{i,t} \le \left(V_i^{max}\right)^2, \quad \forall i, t \tag{17}$$

Using $v_{i,t}$ the primary voltage of the transformer at the slack bus, (7) can be computed as:

$$v_{1,t} = \left(V_0 + b_t^{TAP}\Delta V_{tap}\right)^2, \quad \forall t \tag{18}$$

Note that b_t^{TAP} in the above equation is an integer variable and should be replaced by binary variables $\sigma_{t,k}$ as:

$$b_t^{TAP} = \sum_{k=0}^{2\bar{K}} (k - \bar{K})\sigma_{t,k}, \quad \forall t \tag{19}$$

$$\sum_{k=0}^{2\bar{K}} \sigma_{t,k} = 1, \forall \sigma_{t,k} \in \{0, 1\}, \quad \forall t \tag{20}$$

where $\bar{K}$ is the maximum tap position and $2\bar{K}$ is the total number of transformer tap positions.

Finally, the primary voltage of the transformer at the slack bus (7) can be written as:

$$v_{1,t} = \sum_{k=0}^{2\bar{K}} \left[\left(V_0 + \left(k - \bar{K}\right)\Delta V_{tap}\right)^2 \sigma_{t,k} \right]. \quad \forall t \tag{21}$$

In summary, the MPC-based voltage control problem in the slow-timescale is formulated as an MISOCP problem: (1)-(5), (8)-(9), (12) and (14) - (21), which can be handled directly by commercial solvers.

IV. Fast-Timescale: Multi-Agent DRL-Based Real-Time Voltage Control Strategy

In this fast-timescale, the reactive power set points of inverters connected to solar PV systems and the active power set points of CSs are controlled to mitigate voltage fluctuations on a 5-minute basis. We treat each solar PV system and CS as an intelligent agent. In other words, each smart inverter and CS has its own local voltage controller, and the agents cooperate to obtain a good voltage profile in the active distribution system. The trained DRL models called EA-MAAC help the agents make decentralized decisions in the online environment.

A. Formulate Fast-Timescale Voltage Control Problem as a Markov Game Process

In this subsection, the multi-agent voltage control problem is formulated as a Markov Game Process (MGP). In a multi-agent system, agents are not only affected by the environment, but also by other agents. An MGP is often represented by tuples $\langle \mathcal{N}, \mathcal{S}, \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_N, \mathcal{T}, \mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_N, \gamma \rangle$. They are defined by a set of states, $\mathcal{S}$, actions for $\mathcal{N}$ agents, $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_N$, a state transition function, $\mathcal{T} : \mathcal{S} \times \mathcal{A}_1, \mathcal{A}_2, \ldots, A_N \times \mathcal{S} \rightarrow [0, 1]$, and reward for each agent depending on the global states and actions of all agents: $\mathcal{R}_i : \mathcal{S} \times \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_N \rightarrow \mathcal{R}$. Each agent learns a policy, $\pi_i : \mathcal{S}_i \rightarrow \mathcal{P}(\mathcal{A}_i)$, which maps the agent's observation of the environment to the probability of taking each possible action.

In the fast-timescale voltage control problem, the distribution grid is treated as the environment. The smart inverters of the solar PV systems and CSs are the RL agents. The state space, action set and reward functions of the MGP are defined as follows:

(1) *State Space:* The local information $s_\tau \in \mathcal{S}_\tau$ for each agent at time step τ in each episode is defined as $s_\tau = (\mathbf{P}_\tau, \mathbf{Q}_\tau, \mathbf{V}_\tau)$, which consists of the vector of nodal active power injections $\mathbf{P}_\tau$, reactive power injections $\mathbf{Q}_\tau$ of the distribution grid, and a vector of voltage magnitudes $\mathbf{V}_\tau$. The ith element of the voltage vector $V_{i,\tau}$ is the voltage magnitude of bus i.

(2) *Action Set:* For each agent i, the action space $\mathcal{A}_\tau$ at time step τ is designed separately for different types of controllable devices. For smart inverters of the solar PV systems, the action space $\mathcal{A}_\tau$ is defined as the reactive power set points $Q_{Ci,\tau}$. For CSs, the action space $\mathcal{A}_\tau$ is defined as the active power set points $P_{Cv,\tau}$. That is, $\mathcal{A}_\tau = \{Q_{Ci,\tau}, P_{Cv,\tau}\}, i \in \mathcal{N}_{PV}, v \in \mathcal{N}_{CS}$, where $\mathcal{N}_{PV}$ and $\mathcal{N}_{CS}$ denote the set of solar PV systems and CSs, respectively.

(3) *Reward Function:* We define $r_\tau \in \mathcal{R}_\tau$ as the reward for each agent i at time step τ. In the offline training phase, each local agent observes voltages and line currents in the neighborhood and calculates a local reward, which will be sent to a central learning module to derive the global reward. In the online execution phase, the local agents will observe their own state and calculate the global reward in a collaborative manner. The reward function r_τ has two components and is defined as follows:

$$r_\tau = R_{LL}(\tau) + C_\beta R_{VV}(\tau) = -C_{loss}P_\tau^{loss} + C_\beta R_{VV}(\tau), \tag{22}$$

$$R_{VV}(\tau) = -\sum_{i \in \mathcal{N}_i} \left[\left[\underline{V} - V_i(\tau)\right]_+ + \left[V_i(\tau) - \overline{V}\right]_+ \right]. \tag{23}$$

The first component of the reward $R_{LL}(\tau)$ corresponds to the cost of total line loss P_τ^{loss} in the distribution grid. The second component of the reward $R_{VV}(\tau)$ corresponds to the cumulative magnitude of voltage violations in the distribution grid where $\mathcal{N}_i$ denotes the set of total nodes. C_β is the penalty coefficient for voltage violation, and the function $[\cdot]_+$ is defined as $[x]_+ = \max(0, x)$. The reward function is designed in the above-mentioned manner to mitigate voltage violations and reduce line losses.

B. Multi-Agent Actor-Critic Method

Actor-critic algorithms combine the learning of policy and value functions and they are the foundation for single-agent RL algorithms. In multi-agent systems, however, the environment is non-stationary for each individual agent. Hence, simply adopting traditional single-agent RL algorithms for each agent cannot guarantee the convergence in general. The architecture of multi-agent RL (MARL) was first proposed in the MADDPG algorithm [27], which follows the CTDE framework. Under this framework, the critics are granted global observations and actions to estimate the discounted future reward.

Multi-agent RL with CTDE framework has been studied in recent years. In [32], a multi-agent actor-critic (MAAC) method with an attention-based centralized critic was developed. This attention mechanism optimizes the scale of the

critic from $O(N)$ to $O(1)$, where N is the number of agents. Such algorithms achieve more effective and scalable learning in cooperative multi-agent settings. In the distribution grid environment, we develop a fast converging, maximum entropy, multi-agent RL algorithm on top of the existing MAAC architecture to solve the fast-timescale voltage control problem. The detailed descriptions are provided in the following subsections.

C. Soft Actor-Critic Method

There are two major challenges to develop and train model-free deep RL algorithms. First, the high sample complexity makes the learning process slow. Second, a large number of hyperparameters such as learning rates and exploration parameters need to be set carefully, otherwise it is difficult to obtain stable results. To address these challenges, soft actor-critic (SAC), an off-policy actor-critic RL algorithm based on the maximum entropy framework, was developed by Haarnoja in [33]. This entropy term will guide the agents to explore and exploit in a balanced way. Compared with DDPG [34], SAC instead combines off-policy actor-critic training with a stochastic actor and provides both sample-efficient learning and stability in the training process.

Standard RL maximizes the expectation of rewards $\sum_\tau \mathbb{E}_{(s_\tau,a_\tau)\sim\pi}[R(s_\tau,a_\tau)]$. SAC, on the other hand, maximizes the trade-off between the expected return and the policy's entropy:

$$J(\pi)=\sum_{\tau=0}^{T}\mathbb{E}_{(s_\tau,a_\tau)\sim\pi}[r(s_\tau,a_\tau)+\alpha\mathcal{H}(\pi(\cdot|s_\tau))], \quad (24)$$

where $\mathcal{H}(\pi(\cdot|s_\tau))=-\sum_a \pi(a|s_\tau)\log\pi(a|s_\tau)$ is the entropy function and the hyperparameter α in (24) is called temperature parameter, which controls the stochasticity of the optimal policy.

The entropy-regularized value function is formulated as

$$V^\pi(s)=\mathbb{E}_\pi\left[\sum_{\tau=0}^{\infty}\gamma^\tau(r_\tau+\alpha\mathcal{H}(\pi(\cdot|s_\tau)))|s_0=s\right], \quad (25)$$

$$Q^\pi(s,a)=\mathbb{E}_\pi\left[\sum_{\tau=0}^{\infty}\gamma^\tau(r_\tau+\alpha\mathcal{H}(\pi(\cdot|s_\tau))) |s_0=s,a_0=a\right], \quad (26)$$

The relationship between V^π and $Q^\pi(s,a)$ is defined as

$$V^\pi(s)=\mathbb{E}_{a\sim\pi}\left[Q^\pi(s,a)+\alpha\mathcal{H}(\pi(\cdot|s_\tau))\right] \quad (27)$$

Hence, the corresponding entropy-regularized Bellman equation is written as:

$$\begin{aligned}Q^\pi(s,a)&=\mathop{\mathbb{E}}_{\substack{s_{\tau+1}\sim P\\ a_{\tau+1}\sim\pi}}\left[r_\tau+\gamma\left(Q^\pi(s_{\tau+1},a_{\tau+1})\right)+\alpha\mathcal{H}(\pi(\cdot|s_{\tau+1}))\right]\\ &=\mathop{\mathbb{E}}_{s_{\tau+1}\sim P}\left[r_\tau+\gamma V^\pi(s_{\tau+1})\right]. \end{aligned}\quad (28)$$

During the policy improvement step, the policy is updated towards the exponential Q-function: $\pi(\cdot|s)=\text{argmax}_\pi \mathop{\mathbb{E}}_{a\sim\pi}[Q(s,a)+\alpha\mathcal{H}(\pi)]$. The closed-form solution is given by $\pi(\cdot|s)=\frac{\exp(Q^\pi(s,\cdot)/\alpha)}{Z_\pi(s)}$, where $Z_\pi(s)$ is the partition function that normalizes the numerator to a probability distribution.

SAC has been further enhanced by other researchers with double Q-networks [35], auto-tuned α [36], and the delayed update of value functions [33]. Interested readers are referred to the cited references for detailed description of the methods.

Algorithm 1: Experience Augmentation

Input: a sampled transition $(\mathbf{s},\mathbf{a},\mathbf{r},\mathbf{s}')$ and the index of agent i

Output: a generated EA-based sample for training agent i $\left(\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{r}},\hat{\mathbf{s}}'\right)$

1 Find the feasible permutation matrices set $\mathcal{P}$ in which each matrix shuffles the permutable agents in a specific order;
2 Randomly select a permutation matrix P^k from the feasible permutation matrices $\mathcal{P}$;
3 Derive the permutation matrix P^k on the original transition: $\left(\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{r}},\hat{\mathbf{s}}'\right)\leftarrow(P^k\cdot\mathbf{s},P^k\cdot\mathbf{a},P^k\cdot\mathbf{r},P^k\cdot\mathbf{s}')$;
4 Return $\left(\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{r}}_i,\hat{\mathbf{s}}'\right)$;

D. Improve Algorithm Convergence and Sample Efficiency With Experience Augmentation

In the multi-agent RL domain, exploration of high-dimensional state-action space is one of the biggest challenges. The applicability of MARL algorithms in practice is limited by the low model training efficiency and high data collection costs. To mitigate these issues, we design a new technique called *Experience Augmentation*, which exploits the underlying symmetry of the experiences collected by different agents in the distribution system environment and provides an efficient and fast-converging learning algorithm.

As is implemented in typical MARL setup [37]–[39], the reward of agent i can be written as (29), where $f_{g(i)}$ denotes the reward function of agent i in group $g(i)$. We represent the set of all agents *except* i as $\backslash i$:

$$r_i=f_{g(i)}\left(s_i,a_i,\mathbf{s}_{\backslash i},\mathbf{a}_{\backslash i}\right),\ \forall i=1,\ldots,N, \quad (29)$$

where $\mathbf{s}_{\backslash i}$ and $\mathbf{a}_{\backslash i}$ are the set of states and actions in group $g(i)$ except agent i. We refer to this sample augmentation method as *Experience Augmentation* (EA). The principle and the detailed process is depicted in Fig. 2 and the detailed description of EA is shown in Algorithm 1.

Note that agents in the same group are homogeneous: they have the same structure, size, and reward function. Therefore, exchanging the transition tuple $(\mathbf{s},\mathbf{a},\mathbf{r},\mathbf{s}')$ among homogeneous agents will not change the learning objective. This is because homogeneous agents have the same impact on the environment given the same state and action. In other words, we can train an agent using the experience collected by another, homogeneous, agent. Hence, we can fully exploit experiences collected by the multi-agent system by permuting them among members of homogeneous groups. The following Lemma 1 guarantees the generated sample belongs to the ground truth sample space.

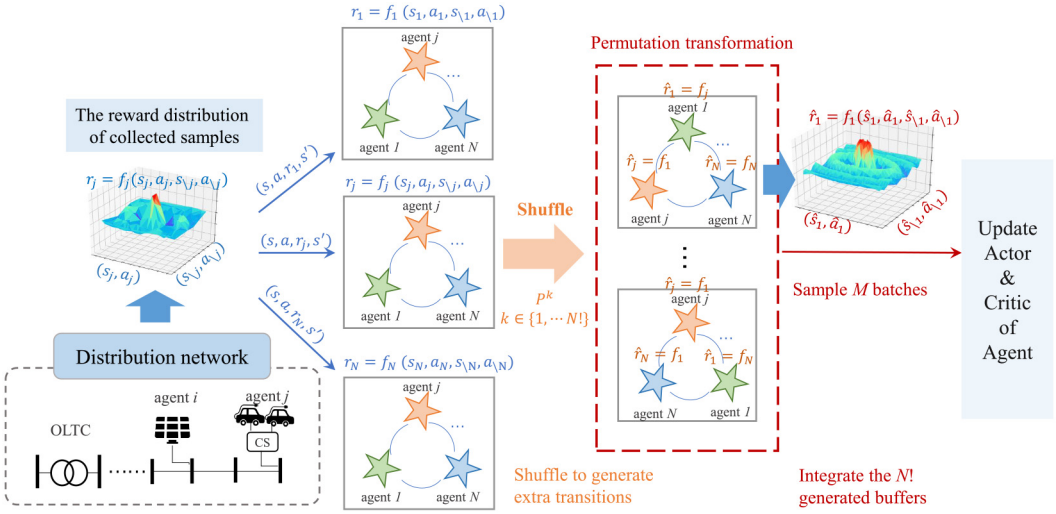


Fig. 2. Detailed process of Experience Augmentation for N homogeneous and cooperative agents. **Black:** the reward distribution of collected samples. **Blue:** sample (s, a, r, s') inserted into replay buffers. **Orange:** shuffle the permutable agents to generate $N!$ replay buffers. **Red:** integrate the $N!$ buffers and sample M batches. Each buffer is reused for $B_n \times \frac{E}{\mathcal{T} \times (N!)}$ times.

Lemma 1 (EA-generated Sample): Let $\mathcal{P}$ be the permutation matrices and let S_{truth} be the ground truth sample space, define the EA-generated sample $(\hat{s}, \hat{a}, \hat{r}, \hat{s}')$ as follows:

$$\left(\hat{s}, \hat{a}, \hat{r}, \hat{s}'\right) \triangleq P^k\left(s, a, r, s'\right)$$

Then $(\hat{s}, \hat{a}, \hat{r}, \hat{s}') \in S_{truth}$ for any permutation transformation matrix $P^k \in \mathcal{P}$, assuming the agents are homogeneous.

The value function can be found by the following iterative process when given policy π:

Lemma 2: Consider the soft Bellman backup operator $\mathcal{T}^\pi$ in eq.(28) and initial $Q_i^0 : \mathcal{S} \times \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_N \rightarrow \mathcal{R}$ with $|\mathcal{A}_i| < \infty$, and define $Q_i^{k+1} = \mathcal{T}^\pi Q_i^k$. Then, the sequence Q_i^k for agent i will converge to the entropy-regularized Q-value of π as $k \rightarrow \infty$.

Lemma 3: For agent i, given policy $\pi, \pi' \in \Pi$, where π' is defined as follows:

$$\pi'(\cdot|s) = \arg\min_{\pi' \in \Pi} D_{\mathrm{KL}}\left(\tilde{\pi}(\cdot|s) || \frac{\exp\left(Q_i^\pi(s, \cdot)/\alpha\right)}{Z_\pi(s)}\right) \quad (30)$$

Then $Q_i^{\pi'}(s, a) \geq Q_i^\pi(s, a)$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_N$, with $|\mathcal{A}_i| < \infty$.

Combining Lemma 1, Lemma 2 and Lemma 3, we can establish the following EA-based soft policy iteration theorem:

Theorem 1 (EA-Based Soft Policy Iteration Theorem): Any policy $\pi_i \in \Pi$, $i \in \mathcal{N}$, will converge to a policy π^* with repeated application of EA, policy evaluation and improvement, such that $Q_i^{\pi_i *}(\mathbf{s}, \mathbf{a}) \geq Q_i^{\pi_i}(\mathbf{s}, \mathbf{a})$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_N$ and EA-generated samples $(\hat{s}, \hat{a})$.

All proofs can be found in the Appendix.

Theorem 1 establishes the theoretical foundation for finding the optimal EA-based soft policy. Given the data augmentation strategy which provides a fast, thorough, and symmetric exploration of observation and action space, EA could effectively expand the original dataset and improve the sample efficiency. In the next subsection, we will introduce the detailed implementation of EA.

E. Experience Augmentation Implementation

Experience Replay plays an important role in off-policy RL where we can store the agents' experiences at each time-step over many episodes into a replay memory. Samples from the replay memory are used to train the RL agent. Replaying past experiences can dramatically reduce the unstable learning problem caused by auto-correlated samples.

When designing the replay memory, there is an important trade-off between the amount of data to keep and the agent's updating frequency, which has a direct impact on the balance between learning speed and model performance. The common experience replay mechanism maintains a sliding window to store the most recent $\mathcal{D}$ transition memories while the agent is interacting with the environment. The number of transitions being collected at each step is defined as buffer acquisition speed $\mathcal{V}_c$. For example, if one transition data is collected at each step, then $\mathcal{V}_c = 1$. The training model will be updated every $\mathcal{T}$ step by sampling n_s batches of data. Hence, the update frequency can be calculated as $n_s/\mathcal{T}$.

The expected number of training samples in the replay buffer to be collected could be calculated as $\mathbb{E}[N_{sample}] = \frac{n_s \cdot B_n}{\mathcal{V}_c \mathcal{T}}$, where B_n is the batch size. It can be seen that there is a trade off between the learning speed and model performance. The training process will converge faster with a larger updating frequency, while training models might be overfitting if $\mathbb{E}[N_{sample}]$ becomes too large. In order to prevent the overfitting in experience replay, we need to keep $\mathbb{E}[N_{sample}]$ unchanged while increasing update frequency $n_s/\mathcal{T}$. Since the capacity of batch B_n should be large enough to perform gradient descent, a suitable solution is to increase the buffer collection speed $\mathcal{V}_c$. This enhancement will benefit the experience replay mechanism and improve the training process.

The EA is implemented in two steps. First, we enumerate the set of permutation matrices $\mathcal{P}$. Each permutation matrix shuffles every agent. Second, we randomly select a

Algorithm 2: Proposed EA-MAAC

for *agent* $i = 1, 2, \ldots N$ **do**
  Initialize parameters for critic, target critic, actor and target actor, ϕ_i, $\bar{\phi}_i$, μ_i, $\bar{\mu}_i$
end
Set global time step $T \leftarrow 0$
for *episode* = *1,2,...M* **do**
  for *time step* $t = 1, 2, \ldots$ *episode length* **do**
    Interact with environment and obtain a transition $(\mathbf{s}, \mathbf{a}, \mathbf{r}, \mathbf{s}')$
    Store the transition into replay buffer $\mathcal{D}$
  end
  Update global time step $T \leftarrow T + 1$
  for *agent* $i = 1, 2, \ldots N$ **do**
    Sample a random mini-batch B of transition $(\mathbf{s}, \mathbf{a}, \mathbf{r}, \mathbf{s}')$ from $\mathcal{D}$
    Perform EA to generate extra samples $(P^k \cdot \mathbf{s}, P^k \cdot \mathbf{a}, P^k \cdot \mathbf{r}, P^k \cdot \mathbf{s}')$
    for $j = 1 \rightarrow N_c$ **do**
      Update the parameters ϕ of critic with generated transitions
    end
    for $j = 1 \rightarrow N_a$ **do**
      Update the parameters μ of actor with generated transitions
    end
    Soft-update the parameters of target critic network and target actor network
  end
end

permutation matrix P^k from $\mathcal{P}$ and perform the transformation on the original experience in a specific order. The transformation can be expressed as:

$$\left(\mathbf{s}, \mathbf{a}, \mathbf{r}, \mathbf{s}'\right) \rightarrow \left(P^k \cdot \mathbf{s}, P^k \cdot \mathbf{a}, P^k \cdot \mathbf{r}, P^k \cdot \mathbf{s}^{'}\right), \tag{31}$$

The details of the EA technique is summarized in Algorithm 1. Consider N_S agents that are divided into N_g groups. In each group w, N_S^w agents are permutable with each other. Using the technique of EA, the original data set could be expanded up to $\prod_{w=1}^{N_g}(N_S^w!)$ times.

F. Algorithm Design for the Proposed EA-MAAC

To accelerate the algorithm convergence and improve the sample efficiency, we propose EA-MAAC, a novel sample-efficient MARL algorithm. Below we introduce the technique of improving the updating frequency of the critic and the learning process of both actor and critic. The detailed algorithm is summarized in Algorithm 2.

1) Improved Updating Frequency for the Critic Network: To utilize the generated data by EA, we analyzed the updating buffer acquisition frequency in Section IV-E. To reduce the variance of the centralized value function and improve the performance of the MARL model, the updating ratio of critic and actor $N_c : N_a$ is set as $v_s : 1$ where $v_s > 1$. The centralized value function is iteratively updated using the TD-error based method δ_e rather than learning an estimation. The proposed EA-MAAC utilizes the centralized state-action function to approximate eq.(32). The following (32) shows that the TD-error could accumulate at each update. Thus the variance of the estimated Q-value grows quickly with the number of look ahead steps. Increasing the number of update steps for the critic will reduce the TD-error and yield a smaller variance.

$$\begin{aligned} Q_i(s_\tau, a_\tau) &= r_\tau + \gamma \cdot \mathbb{E}\left[Q_i(s_{\tau+1}, a_{\tau+1})\right] - \delta_{e,\tau} \\ &= r_\tau + \gamma \cdot \mathbb{E}\left[r_{\tau+1} + \gamma \cdot \mathbb{E}\left[Q_i(s_{\tau+2}, a_{\tau+2}) - \delta_{e,\tau+1}\right]\right] - \delta_{e,\tau} \\ &= E_{s_\tau, a_\tau}\left[\sum_{i=\tau}^{T} \gamma^{i-\tau}\left(r_i - \delta_{e,i}\right)\right]. \end{aligned} \tag{32}$$

2) Learning the Actor and Critic: To achieve a more effective and stable learning process, this paper proposes EA-MAAC algorithm based on an attention mechanism, which is trained under the CTDE framework. In the training stage, each agent will learn an actor and a critic. The critic receives global observation to guide the actor; the actor takes action according to the local observation. In addition, the agents perform EA by shuffling the experiences shared from their homogeneous peers. The centralized critic network can be trained from both permutable agents and in-permutable agents' experiences because of the permuted global states.

During the training process, an attention mechanism is included in our centralized critic to facilitate the cooperation between multiple agents, which is shown in Fig. 3. Each agent considers other agents as part of their observations before taking actions, after which the agent updates its value function accordingly. The Q function of agent i, $Q_i^\pi(s, a)$, is parameterized by a two-layer critic network:

$$Q_i^\pi(s, a) = f_i(h_i(s_i, a_i), x_i), \tag{33}$$

where $h_i(\cdot)$ is the embedding function with one-layer neural network; $f_i(\cdot)$ is a two-layer critic network. x_i is given by:

$$x_i = \sum_{j \neq i} \alpha_j z_j, \tag{34}$$

where $z_j = \mathcal{V} h_j(s_j, a_j)$ and $\mathcal{V}$ is a shared linear transformation matrix. j aggregates all agents except agent i. α_j represents the attention weights agent i pays for agent j, which can be expressed as follow:

$$\alpha_j \propto \exp\left(z_j^T W_k^T W_q h_i(s_i, a_i)\right), \tag{35}$$

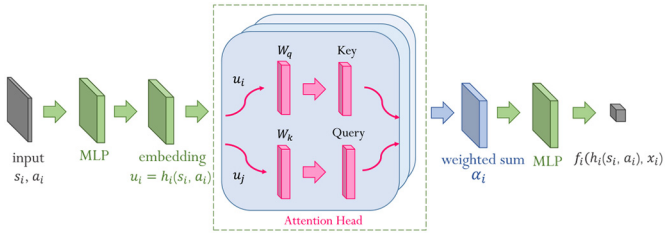


Fig. 3. The **proposed attention mechanism** to calculate Q-value function for agent i including two multi-layer perception (MLP).

From the equation above, the attention weight α_j can be obtained by performing a linear transformation of the embedding functions of agent i and j followed by a softmax. W_q and W_k are the transformation matrices. The parameters of attention are shared among all agents. Hence, all critics are updated together using a joint regression loss function:

$$\mathcal{L}_Q(\phi) = \sum_{i=1}^{N} \mathbb{E}_{(s,a,r,s')\sim D}\left[\left(Q_i^{\phi}(s,a) - y_i\right)^2\right], \tag{36}$$

$$y_i = r_i + \gamma \mathbb{E}_{a'\sim\pi_{\bar{\mu}_i}(s')}\left[Q_i^{\bar{\phi}}(s',a') - \alpha \log\left(\pi_{\bar{\mu}_i}(a_i'|s_i')\right)\right], \tag{37}$$

where the target actor network and the target critic networks are parameterized by $\bar{\mu}_i$ and $\bar{\phi}$, respectively.

The parameters of the actor is updated using the gradient ascent method. The policy gradient is given by:

$$\nabla_{\mu_i}\mathcal{J}(\pi_\mu) = \mathbb{E}_{s\sim D, a\sim\pi}\Big[\nabla_{\mu_i}\log\pi_{\mu_i}(a_i|s_i)\big(-\alpha\log\pi_{\mu_i}(a_i|s_i)\big) + Q_i^{\phi}(s,a) - v\big(s,a_{\setminus i}\big)\Big]. \tag{38}$$

where μ_i is the parameter of the actor neural network.

V. Numerical Studies

In this section, numerical studies are conducted to evaluate the proposed hybrid voltage control strategy. We start by presenting the experimental data and the algorithm setup in Section V-A. The slow-time voltage control decisions are provided in Section V-B. The optimality, robustness, and computation efficiency of the proposed algorithm and benchmark algorithms are shown in Sections V-C–V-E.

A. Setup for Testing System and Algorithms

Our proposed two-timescale hybrid voltage control strategy is tested on a modified IEEE 33-bus [40] and IEEE 123-bus distribution network [41]. In the modified IEEE 33-bus testing distribution circuit, two 4-stage CBs are placed at node 17 and node 33 (4×100 kVAr). The maximum number of daily operations for the CBs are set at 4. An ES system with capacity of 500kWh is located at node 7. In the modified IEEE 123-bus distribution network, four 4-stage CBs are placed at node 17, 51, 62, and 68 (4 × 100 kVAr). The maximum number of daily operations for the CBs are set at 4. Two ES systems with capacity of 500kWh are located at node 12 and node 103. For both ES systems, an OLTC is located between node 1 and 2 with 11 tap positions, which correspond to turns ratios ranging from 0.95 to 1.05. The feasible SoC range of the ES system is [20%, 80%]. The parameters of the testing IEEE 33-bus system and electricity prices C_{grid} are listed in Table I and the specific parameters of the IEEE 123-bus test feeder are shown in Table II.

The power consumption profiles of three different charging stations and the total remaining load are shown in Fig. 4, where CS1-3 represent home charging, workplace charging, and charging at recreational facilities. The electric load is allocated to various nodes according to the spatial load distribution in the original IEEE test feeder. The solar PV systems

TABLE I
Parameters of Voltage Regulation Devices and Electricity Price in IEEE 33-Bus System

Device	Parameters	Capacity/Operation Limits	Node
OLTC	±5 × 1%	6	1
CB1-2	4×100 kVar	4	17, 33
ES	500 kW	0.2-0.8	7
PV1-3	500 kW	-	10, 14, 28
CS1-3	home,work,recreation	450 kW	22, 25, 30

Time-of-Use Periods	Electric Price $/kWh
1:00-2:00, 14:00-16:00, 22:00-24:00	0.12
9:00-13:00, 17:00-21:00	0.15
3:00-8:00	0.07

TABLE II
Parameters of Voltage Regulation Devices in IEEE 123-Bus System

Device	Parameters	Capacity/Operation Limits	Node
OLTC	±5 × 1%	6	1
CB1-4	4×100 kVar	4	17, 51, 62, 68
ES	500 kW	0.2-0.8	12, 103
PV1-5	400 kW	-	25,45,82,101,110
CS1-6	home,work,recreation home,work,recreation	450 kW	11, 16, 61, 71, 89, 104

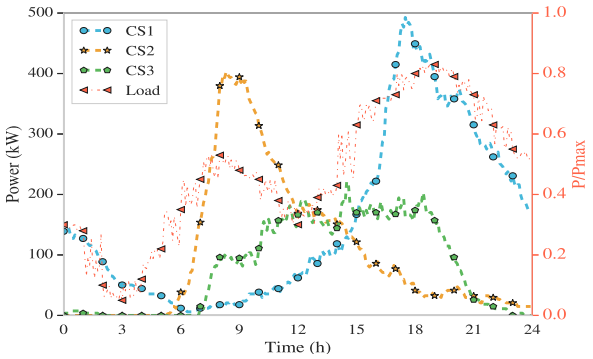


Fig. 4. Power consumption data of different charging stations.

generation data are taken from the Renewable ninja Web platform [42]. The resolution of electric loads, CS power consumption, and solar PV generation is 5 minutes. Random Gaussian noises with a standard deviation of 5% are added to these three types of time series to model the uncertainties in the fast-timescale voltage control problem. The data set spanning over 30 days is separated into the training and test sets. 20-days' of data are used to train the proposed RL algorithm in the fast-timescale. The rest of the data is used for testing purposes. The cost of power loss C_{loss} is assumed to be \$48/MWh. The penalty coefficient for voltage violation C_β is set to be \$80/p.u.. The hyperparameters of the proposed DRL algorithm and two baseline RL algorithms are provided in Table III. The two parameters from left to right in each of the curly brackets are for 33- and 123-bus distribution network, respectively.

TABLE III
PARAMETER SETTING FOR DRL ALGORITHMS

Algorithm	Parameters	Values
EA-MAAC	α	$\{0.05, 0.05\}$
	learning rate for each agent	$\{1e-3, 1e-3\}$
	number of hidden units	$\{256, 250\}$
	batch size	$\{256, 512\}$
	number of permutable agents	$\{6, 11\}$
SAC	α	$\{0.03, 0.03\}$
	learning rate	$\{1e-3, 1e-3\}$
	number of hidden units	$\{256, 250\}$
	batch size	$\{256, 512\}$
MADDPG	learning rate for each agent	$\{1e-3, 1e-3\}$
	number of hidden units	$\{256, 250\}$
	batch size	$\{256, 512\}$
Shared	number of hidden layers	2
	replay buffer size	1,000,000
	discount factor	0.95
	delay factor ρ	5e-4
	hidden unit nonlinearity	Leaky ReLu

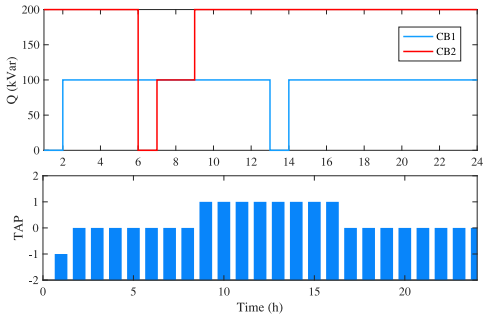


Fig. 5. Hourly tap positions of OLTC and switching schedules of CBs.

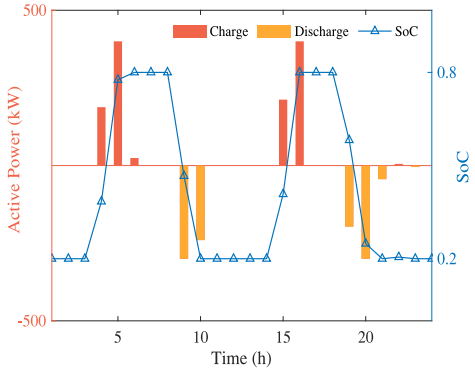


Fig. 6. Charging/discharging power and SoC of the ES system.

B. Slow-Timescale Voltage Control Decisions

The dispatch results of IEEE 33-bus testing for OLTC, CBs and ES systems from the slow-timescale voltage control are shown in this subsection. The hourly dispatch results for OLTC and BCs are illustrated in Fig. 5. The top subfigure shows the reactive power outputs of the two CBs. The bottom subfigure shows the hourly tap positions of the OLTC. Note that the numbers of switching operations of the OLTC, CB1 and CB2 in the testing day are all 3, which satisfy the operational constraints.

The charging/discharging power and SoC of the ES system are shown in Fig. 6. The ES system mostly discharges during the morning and evening peak hours when the electricity prices

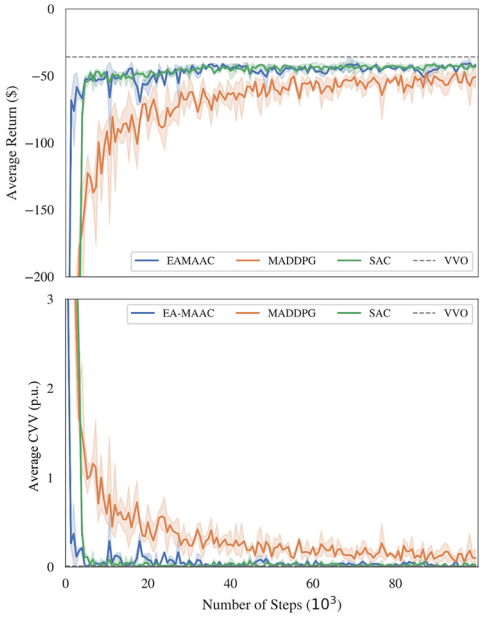


Fig. 7. Average return and cumulative voltage violation of the proposed RL algorithms and benchmarks in IEEE 33-bus distribution network.

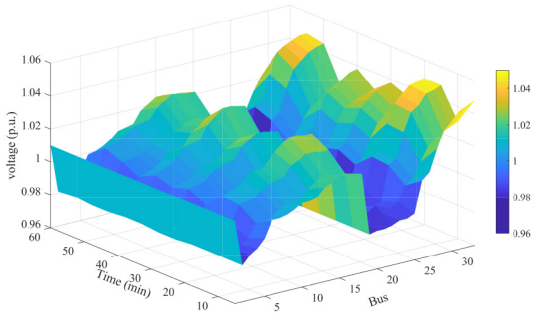


Fig. 8. Hourly bus voltages of the proposed EA-MAAC method.

are high. The charging activities mostly take place during off-peak and mid-peak hours. The ES system dispatch results satisfy charging/discharge power and SoC constraints.

C. RL-Based Fast-Timescale Voltage Control Performance

In this subsection, we investigate the fast-timescale voltage control performance of the proposed EA-MAAC algorithm and two state-of-the-art RL algorithms. The first baseline algorithm is MADDPG [27], which also performs centralized training and decentralized voltage control. The second baseline algorithm is SAC, which follows a single-agent centralized learning and control framework. Both baseline algorithms have been applied to perform data-driven volt-var controls. An optimization-based algorithm with SOCP relaxation [12] is applied in this voltage problem serving as a baseline of theoretically best solution. This model-based benchmark called VVO is marked by dashed grey curves in Fig. 7 and Fig. 9. All of the proposed and baseline RL-algorithms are evaluated in the same distribution system with the same data set.

The returns and average cumulative voltage violations (CVV) of the proposed and baseline RL algorithms in the training process are shown in Fig. 7 and Fig. 9. The solid

TABLE IV
ONLINE PERFORMANCE COMPARISON OF VOLTAGE CONTROL ALGORITHMS

Test system	Algorithm	AR ($)		ACVV (p.u.)	
		Mean	Std.	Mean	Std.
33-bus sys.	SAC	-4.01e+01	1.06e+00	2.35e-02	3.01e-02
	MADDPG	-5.78e+01	4.46e+00	1.02e-01	5.57e-02
	EA-MAAC	-3.96e+01	7.71e-01	8.30e-03	7.80e-03
	VVO	-3.58e+01	-	0e+00	-
123-bus sys.	SAC	-2.43e+01	1.52e+00	3.40e-03	4.50e-03
	MADDPG	-4.31e+01	8.43e+00	1.09e-01	1.02e-01
	EA-MAAC	-2.38e+01	7.61e-01	2.50e-03	3.30e-03
	VVO	-1.87e+01	-	0e+00	-

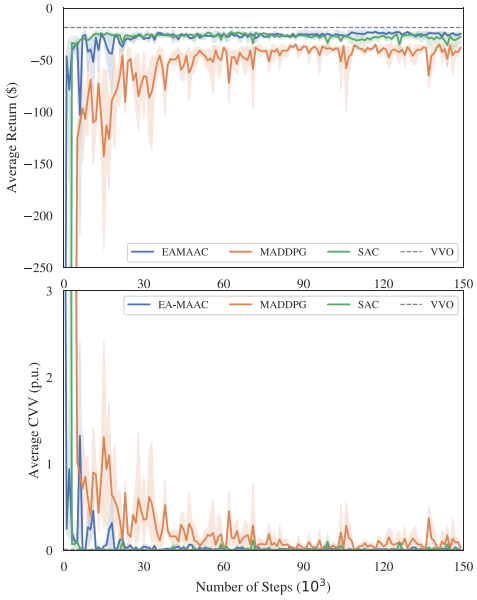


Fig. 9. Average return and cumulative voltage violation of the proposed and baseline RL algorithms in IEEE 123-bus distribution network.

curves denote the average performance of 5 experiments with different random seeds and the light-colored shadow areas show the error bounds. In the initial learning phase, the RL agents are incapable of concurrently keeping the nodal voltages within 0.95∼1.05 p.u. and minimizing the network loss at the same time. However, as training progresses, the RL agents gradually learn to reduce the cumulative voltage violations and network losses. In Table IV, we calculate the mean values (Mean) and standard deviations (Std.) with test data set for both average rewards (AR) and average cumulative voltage violations (ACVV).

The voltage profiles of the 33-bus case under the proposed EA-MAAC voltage control algorithm are shown in Fig. 8, from which we can see that the proposed algorithm is able to keep all nodal voltages within the appropriate range of $[\underline{V}, \overline{V}] = [0.95, 1.05]$ p.u. in all hours.

As shown in Fig. 7, Fig. 9 and Table IV, our proposed EA-MAAC outperforms the multi-agent RL baseline algorithm MADDPG in terms of average return and voltage violation mitigation. From Fig. 7 and Fig. 9, it can be seen that the convergence of EA-MAAC is relative fast and stable. Although

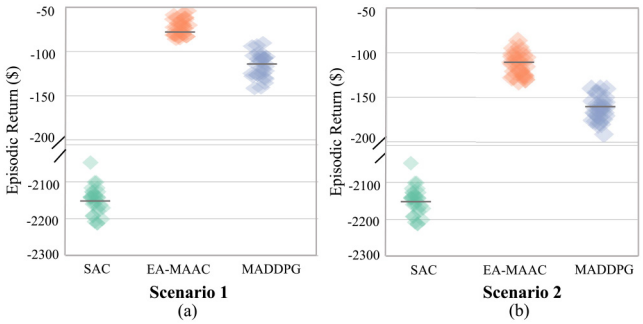


Fig. 10. The VVC performance of RL algorithms with communication failure.

the performance of SAC is on a par with EA-MAAC, the *Experience Augmentation* technique enables our proposed EA-MAAC algorithm to achieve higher sample efficiency and faster convergence speed. In addition, the performance of EA-MAAC is more stable as indicated by smaller standard deviations. The shared attention mechanism allows the RL agents in EA-MAAC to learn a more effective voltage control strategy than that of the deterministic policies in MADDPG.

D. Robustness Analysis With Communication Interruptions

1) Communication Failure of Agent(s): In real-world operations, the communication system may fail occasionally. We studied two cases with different communication failure scenarios:

- s.1 In this case scenario, a single agent loses communication in all three algorithms. Specifically, while the single-agent in SAC algorithm loses communication, RL agent 1 associated with PV1 at node 10 in the EA-MAAC algorithm and RL agent 1 in MADDPG algorithm both lose communication.
- s.2 In this case scenario, the single agent in SAC remains losing communication and two other agents lose their communication in EA-MAAC and MADDPG. Specifically, RL agent 1 and agent 5 associated with CS2 at node 25 in the EA-MAAC algorithm lose communication. Similarly, RL agents 1 and 5 in the MADDPG algorithm also lose their communication.

The average reward values of testing data sets are denoted by the black lines. As shown in Fig. 10 (a) and (b), our proposed EA-MAAC has the best robustness against communication system failure. Since SAC utilizes a central training and central execution control structure and single-based algorithm, once the communication system fails its control performance deteriorates remarkably.

2) Missing Observations: We evaluate the proposed and baseline RL algorithms' capability of dealing with missing observations. We set the observation drop rate to be between 0 and 1. The RL-based VVC algorithms' average reward and cumulative voltage violations are shown in Fig. 11. Note that the default values for missing observations of V, P, Q are set as 0.95, 0.1, 0.1, respectively. It can be seen from Fig. 11 that our proposed EA-MAAC significantly outperforms state-of-the-art RL-based baseline algorithms when the observation

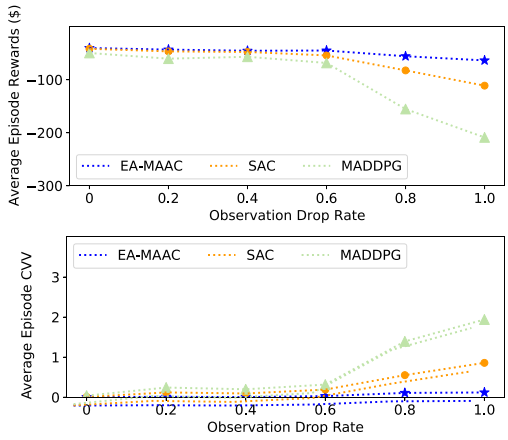


Fig. 11. The VVC performance of RL algorithms with missing observations.

TABLE V
TOTAL COMPUTATION TIME OF TESTING SYSTEM

		33-bus	123-bus
Testing (seconds)	MADDPG	8.79	14.78
	SAC	6.57	12.04
	EA-MAAC	7.13	14.53
	MISOCP	235.34	1563.42

drop rate is high. This is primarily benefiting from the shared attention mechanism in the proposed EA-MAAC algorithm.

To summarize, the robustness of EA-MAAC is verified by tests regarding communication failure as well as missing observations. Such a robustness could be potentially explained in this way: the *Experience Augmentation* helps neural networks obtain more training samples with permutation transformation, in which case, the neural networks successfully learn the symmetrical characteristic of samples. Therefore, even in the communication failure scenarios, our proposed EA-MAAC can still maintain better performance than MADDPG and SAC.

E. Computation Speed

In order to show the superior computation speed of RL-based control methods, we compare their computation time with the MPC-based control method. We implement the MISOCP [43] in MATLAB with YALMIP optimization modeling toolbox [44] and Gurobi optimization solver. The DRL-based algorithms are implemented with PyTorch 1.2.0 framework. All of the algorithms are tested on a desktop with an NVIDIA GTX 2080 Ti GPU and a 16-core Intel i5 2.9 GHz CPU. Since the training process of RL-based algorithms is conducted in an off-line manner, we focus on comparing the testing time of RL-based algorithms to the model-based algorithm. The testing time during a day of all methods is listed in Table V.

Table V shows that the testing computation time of RL-based algorithms for a day is at least two orders of magnitudes shorter than the model-based control algorithm MISOCP. The common superiority of RL-based methods comes from the fact that they only require simple forward passes of neural networks during testing, rather than complex branch-and-bound operations involved in solving the model-based algorithm. The advantage of the RL-based algorithms will become more pronounced when the size of the distribution test feeder increases.

VI. CONCLUSION

A two-timescale hybrid voltage control strategy with coordinated active and reactive power optimization is proposed for active power distribution grid. In the slow-timescale, the control problem of OLTC, CBs and ES systems is formulated as a MISOCP to reduce active power loss and the power purchase cost. In the fast-timescale, we developed a sample efficient and robust multi-agent deep reinforcement learning algorithm to select the set points of smart inverters connected to solar PV systems and CSs to reduce line losses and mitigate voltage deviations. The experience augmentation and shared attention modules of our proposed EA-MAAC algorithm greatly improves the sample efficiency and increases convergence speed. Numerical study results on the IEEE 33-bus and 123-bus test systems demonstrated that the proposed EA-MAAC algorithm outperforms the state-of-the-art RL-based algorithms in terms of optimality, sample efficiency, and robustness. The proposed EA-MAAC is also highly scalable and has significantly lower computation time than optimization-based methods.

A potential future research direction is to develop a fully decentralized DRL model, which requires information of only the neighboring agents, eliminating the difficulty of acquiring a global state of the power distribution feeder.

APPENDIX
PROOFS

A. Proof of Lemma 1

Consider the ground truth sample $(s_i, a_i, r_i, s_i') \in S_{truth}$ and the EA-generated sample $(\hat{s}_j, \hat{a}_j, \hat{r}_j, \hat{s}_j')$ for any two homogeneous agents i and j, we have

$$\left(\hat{s}_j, \hat{a}_j, \hat{r}_j, \hat{s}_j'\right) = P^k\left(s_i, a_i, r_i, s_i'\right)$$

According to $\mathcal{R}_i : \mathcal{S} \times \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_N \rightarrow \mathcal{R}$, the term can be rewritten as $r_i = f_i(s_i, a_i, \mathbf{s}_{\setminus i}, \mathbf{a}_{\setminus i})$, where f_i is the predefined reward function for agent i. Then we can get $\hat{r}_j = f_i(\hat{s}_j, \hat{a}_j, \hat{\mathbf{s}}_{\setminus j}, \hat{\mathbf{a}}_{\setminus j})$. By the homogeneity of agent i, j, we can obtain

$$\hat{r}_j = f_i\left(\hat{s}_j, \hat{a}_j, \hat{\mathbf{s}}_{\setminus j}, \hat{\mathbf{a}}_{\setminus j}\right) = f_j\left(\hat{s}_j, \hat{a}_j, \hat{\mathbf{s}}_{\setminus j}, \hat{\mathbf{a}}_{\setminus j}\right)$$

Hence, $\hat{r}_j$ still satisfies the predefined reward function of agent j. Since $s, a, s' \in \mathcal{S} \times \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_N$ $\forall$ agent $i \in \mathcal{N}$ in same environment,

$$\left(\hat{s}_j, \hat{a}_j, f_j\left(\hat{s}_j, \hat{a}_j, \hat{\mathbf{s}}_{\setminus j}, \hat{\mathbf{a}}_{\setminus j}\right), \hat{s}_j'\right) \in S_{truth}$$

The above proof is valid for any agent i, j. Therefore, the EA-generated sample $(\hat{s}_j, \hat{a}_j, \hat{r}_j, \hat{s}_j') = P^k(s_i, a_i, r_i, s_i') \in S_{truth}$.

B. Proof of Lemma 2

Define the entropy-regularized reward for agent i as $r_i^{\pi}(s,a) = r_i(s,a) + \mathbb{E}_{s\sim p}[\mathcal{H}(\pi(\cdot|s))]$ and the updated rule can be rewritten as

$$Q_i^{\pi}(s,a) \leftarrow r_i^{\pi}(s,a) + \gamma \mathbb{E}_{s_i\sim p, a_i\sim\pi}\left[Q_i^{\pi}\left(s',a'\right)\right]$$

This updated standard convergence result has no effect on convergence [45]. The assumption $|\mathcal{A}_i| < \infty$ is required to guarantee that the entropy-regularized reward is bounded.

C. Proof of Lemma 3

Let $\frac{\exp(Q^{\pi}(s,\cdot)/\alpha)}{Z_{\pi}(s)} = \pi^m(\cdot|s)$, then $\pi' = \arg\min_{\pi'\in\Pi} D_{\mathrm{KL}}(\tilde{\pi}(\cdot|s)||\pi^m(\cdot|s))$. Therefore, we have $D_{\mathrm{KL}}(\pi'(\cdot|s)||\pi^m(\cdot|s)) \leq D_{\mathrm{KL}}(\pi(\cdot|s)||\pi^m(\cdot|s))$. The KL-Divergence can be rewritten as

$$\begin{aligned}&\mathbb{E}_{a\sim\pi'}\left[\log\pi'(a|s) - Q_i^{\pi'}(s,a) + \log Z_{\pi}(s)\right]\\ &\quad\leq \mathbb{E}_{a\sim\pi}\left[\log\pi'(a|s) - Q_i^{\pi}(s,a) + \log Z_{\pi}(s)\right] \end{aligned} \tag{39}$$

where $\log Z_{\pi}(s)$ can be eliminated in the equation since Z_{π} depends only on the state. Hence, we can obtain $\mathbb{E}_{a\sim\pi'}[Q_i^{\pi'}(s,a) - \log\pi'(a|s)] \leq \mathbb{E}_{a\sim\pi}[Q_i^{\pi}(s,a) - \log\pi'(a|s)]$. From the entropy-regularized value iteration equation (27), the inequality of the state-value can be get as $V^{\pi}(s) \leq \mathbb{E}_{a\sim\pi'}[Q_i^{\pi}(s,a) - \log\pi'(a|s)]$. Consider the soft Bellman equation:

$$\begin{aligned}Q_i^{\pi}(s,a) &= r_i + \gamma\mathbb{E}_{s'\sim p}\left[V^{\pi}\left(s'\right)\right]\\ &\leq r_i + \gamma\mathbb{E}_{s'\sim p}\left[\mathbb{E}_{a'\sim\pi'}\left[Q_i^{\pi}\left(s',a'\right) - \log\pi'(a|s)\right]\right]\\ &\leq \cdots\\ &\leq Q^{\pi'}(s,a)\end{aligned} \tag{40}$$

Convergence to $Q^{\pi'}$ follows from Lemma 2.

D. Proof of Theorem 1

Consider the policy iteration sequence π^k based on Lemma 3, then the sequence $Q_i^{\pi^k}$ is monotonically increasing and bounded since the entropy-regularized reward is bounded. Hence π^k converges to some π^*. At convergence, it must be the case that π^* is a minimizer of the function $D_{\mathrm{KL}}(\pi(\cdot|s_i)||\frac{\exp(Q^{\pi^*}(s_i,\cdot)/\alpha)}{Z_{\pi^*}(s_i)})$ for all $\pi\in\Pi$. By Lemma 3, we get $Q_i^{\pi^*}(s,a) \geq Q_i^{\pi}(s,a)$ for all $(s,a)\in\mathcal{S}\times\mathcal{A}_1,\mathcal{A}_2,\ldots,\mathcal{A}_N$. According to the proof of Lemma 1, $(s,a,s')\in\mathcal{S}\times\mathcal{A}_1,\mathcal{A}_2,\ldots,\mathcal{A}_N$, $\hat{r}_i\in S_{truth}$, we can obtain $Q_i^{\pi^*}(\hat{s},\hat{a}) \geq Q_i^{\pi}(\hat{s},\hat{a})$ for EA-generated samples. Therefore, the Q-value of any other policy in Π lower than that of π^*, that is, π^* is indeed the optimal policy.

REFERENCES

[1] P. Jahangiri and D. C. Aliprantis, "Distributed volt/VAR control by PV inverters," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 3429–3439, Aug. 2013.

[2] R. Anilkumar, G. Devriese, and A. K. Srivastava, "Voltage and reactive power control to maximize the energy savings in power distribution system with wind energy," *IEEE Trans. Ind. Appl.*, vol. 54, no. 1, pp. 656–664, Jan./Feb. 2018.

[3] H. Ahmadi, J. R. Martí, and H. W. Dommel, "A framework for volt-VAR optimization in distribution systems," *IEEE Trans. Smart Grid*, vol. 6, no. 3, pp. 1473–1483, May 2015.

[4] M. H. K. Tushar and C. Assi, "Volt-VAR control through joint optimization of capacitor bank switching, renewable energy, and home appliances," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4077–4086, Sep. 2018.

[5] F. Kabir, Y. Gao, and N. Yu, "Reinforcement learning-based smart inverter control with polar action space in power distribution systems," in *Proc. IEEE Conf. Control Technol. Appl. (CCTA)*, 2021, pp. 315–322.

[6] R. Tonkoski, L. A. C. Lopes, and T. H. M. El-Fouly, "Coordinated active power curtailment of grid connected PV inverters for overvoltage prevention," *IEEE Trans. Sustain. Energy*, vol. 2, no. 2, pp. 139–147, Apr. 2011.

[7] W. Tang and Y. J. Zhang, "A model predictive control approach for low-complexity electric vehicle charging scheduling: Optimality and scalability," *IEEE Trans. Power Syst.*, vol. 32, no. 2, pp. 1050–1063, Mar. 2017.

[8] B. Sun, Z. Huang, X. Tan, and D. H. K. Tsang, "Optimal scheduling for electric vehicle charging with discrete charging levels in distribution grid," *IEEE Trans. Smart Grid*, vol. 9, no. 2, pp. 624–634, Mar. 2018.

[9] Z. Wan, H. Li, H. He, and D. Prokhorov, "Model-free real-time EV charging scheduling based on deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 10, no. 5, pp. 5246–5257, Sep. 2019.

[10] Z. Wang, J. Wang, B. Chen, M. M. Begovic, and Y. He, "MPC-based voltage/VAR optimization for distribution circuits with distributed generators and exponential load models," *IEEE Trans. Smart Grid*, vol. 5, no. 5, pp. 2412–2420, Sep. 2014.

[11] C. Zhang, Y. Xu, Z. Dong, and J. Ravishankar, "Three-stage robust inverter-based voltage/VAR control for distribution networks with high-level PV," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 782–793, Jan. 2019.

[12] Y. Chai, L. Guo, C. Wang, Z. Zhao, X. Du, and J. Pan, "Network partition and voltage coordination control for distribution networks with high penetration of distributed PV units," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3396–3407, May 2018.

[13] L. H. Macedo, J. F. Franco, M. J. Rider, and R. Romero, "Optimal operation of distribution networks considering energy storage devices," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2825–2836, Nov. 2015.

[14] Y. Gao and N. Yu, "Deep reinforcement learning in power distribution systems: Overview, challenges, and opportunities," in *Proc. IEEE Power Energy Soc. Innov. Smart Grid Technol. Conf. (ISGT)*, 2021, pp. 1–5.

[15] Y. Zhang, X. Wang, J. Wang, and Y. Zhang, "Deep reinforcement learning based volt-VAR optimization in smart distribution systems," *IEEE Trans. Smart Grid*, vol. 12, no. 1, pp. 361–371, Jan. 2021.

[16] W. Wang, N. Yu, J. Shi, and Y. Gao, "Volt-VAR control in power distribution systems with deep reinforcement learning," in *Proc. IEEE Int. Conf. Commun. Control Comput. Technol. Smart Grids (SmartGridComm)*, 2019, pp. 1–7.

[17] W. Wang, N. Yu, Y. Gao, and J. Shi, "Safe off-policy deep reinforcement learning algorithm for volt-VAR control in power distribution systems," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3008–3018, Jul. 2020.

[18] S. Wang *et al.*, "A data-driven multi-agent autonomous voltage control framework using deep reinforcement learning," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4644–4654, Nov. 2020.

[19] H. Liu and W. Wu, "Online multi-agent reinforcement learning for decentralized inverter-based volt-VAR control," *IEEE Trans. Smart Grid*, vol. 12, no. 4, pp. 2980–2990, Jul. 2021.

[20] Y. Gao, W. Wang, and N. Yu, "Consensus multi-agent reinforcement learning for volt-VAR control in power distribution networks," *IEEE Trans. Smart Grid*, vol. 12, no. 4, pp. 3594–3604, Jul. 2021.

[21] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[22] M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivas, "Reinforcement learning with augmented data," 2020, *arXiv:2004.14990*.

[23] Y. Gao, J. Shi, W. Wang, and N. Yu, "Dynamic distribution network reconfiguration using reinforcement learning," in *Proc. IEEE Int. Conf. Commun. Control Comput. Technol. Smart Grids (SmartGridComm)*, 2019, pp. 1–7.

[24] D. Cao *et al.*, "Model-free voltage regulation of unbalanced distribution network based on surrogate model and deep reinforcement learning," 2020, *arXiv:2006.13992*.

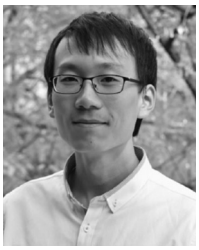
[25] H. Xu, A. D. Domínguez-García, and P. W. Sauer, "Optimal tap setting of voltage regulation transformers using batch reinforcement learning," *IEEE Trans. Power Syst.*, vol. 35, no. 3, pp. 1990–2001, May 2020.

[26] Y. Gao and N. Yu, "Model-augmented safe reinforcement learning for volt-VAR control in power distribution networks," *Appl. Energy*, vol. 313, May 2022, Art. no. 118762.

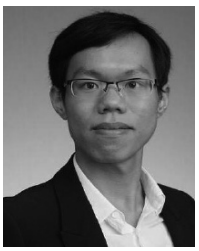
[27] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," 2017, *arXiv:1706.02275*.
[28] X. Sun and J. Qiu, "A customized voltage control strategy for electric vehicles in distribution networks with reinforcement learning method," *IEEE Trans. Ind. Informat.*, vol. 17, no. 10, pp. 6852–6863, Oct. 2021.
[29] M. Baran and F. F. Wu, "Optimal sizing of capacitors placed on a radial distribution system," *IEEE Trans. Power Del.*, vol. 4, no. 1, pp. 735–743, Jan. 1989.
[30] L. Gan, N. Li, U. Topcu, and S. H. Low, "Exact convex relaxation of optimal power flow in radial networks," *IEEE Trans. Autom. Control*, vol. 60, no. 1, pp. 72–87, Jan. 2015.
[31] R. A. Jabr, "Radial distribution load flow using conic programming," *IEEE Trans. Power Syst.*, vol. 21, no. 3, pp. 1458–1459, Aug. 2006.
[32] S. Iqbal and F. Sha, "Actor-attention-critic for multi-agent reinforcement learning," in *Proc. 36th Int. Conf. Mach. Learn.*, vol. 97, Jun. 2019, pp. 2961–2970.
[33] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. ICML*, vol. 80, Jul. 2018, pp. 1861–1870.
[34] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*.
[35] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1587–1596.
[36] Y. Wang and T. Ni, "Meta-SAC: Auto-tune the entropy temperature of soft actor-critic via metagradient," 2020, *arXiv:2007.01932*.
[37] I. Mordatch and P. Abbeel, "Emergence of grounded compositional language in multi-agent populations," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1495–1502.
[38] R. Vidal, O. Shakernia, H. J. Kim, D. H. Shim, and S. Sastry, "Probabilistic pursuit-evasion games: Theory, implementation, and experimental evaluation," *IEEE Trans. Robot. Autom.*, vol. 18, no. 5, pp. 662–669, Oct. 2002.
[39] J. K. Gupta, M. Egorov, and M. Kochenderfer, "Cooperative multi-agent control using deep reinforcement learning," in *Proc. Int. Conf. Auton. Agents Multiagent Syst.*, 2017, pp. 66–83.
[40] M. E. Baran and F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Power Eng. Rev.*, vol. 9, no. 4, pp. 101–102, Apr. 1989.
[41] W. H. Kersting, "Radial distribution test feeders," *IEEE Trans. Power Syst.*, vol. 6, no. 3, pp. 975–985, Aug. 1991.
[42] "Web Application-Renewable Ninja." [Online]. Available: https://www.renewables.ninja/# (Accessed: Jan. 24, 2019).
[43] H. Ji *et al.*, "A centralized-based method to determine the local voltage control strategies of distributed generator operation in active distribution networks," *Appl. Energy*, vol. 228, pp. 2024–2036, Oct. 2018.
[44] J. Lofberg, "YALMIP : A toolbox for modeling and optimization in MATLAB," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004, pp. 284–289.
[45] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

Daner Hu (Student Member, IEEE) is currently pursuing the Ph.D. degree in electrical engineering with the College of Electrical Engineering, Zhejiang University, Hangzhou, China. Her research interests include active distribution system operation and control, optimization of distribution network, and applications of machine learning especially reinforcement learning in smart grids.

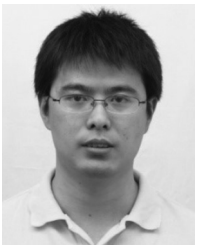
Zhenhui Ye (Student Member, IEEE) received the B.S. degree from Zhejiang University, China, in 2020, where he is currently pursuing the Ph.D. degree with the College of Computer Science and Technology. His research interests include practical reinforcement learning and deep learning in real-world applications.



Yuanqi Gao (Member, IEEE) received the B.E. degree in electrical engineering from Donghua University, Shanghai, China, in 2015, and the Ph.D. degree in electrical engineering from the University of California at Riverside, Riverside, USA in 2020, where he was a Postdoctoral Researcher, before joining Lucid Motors as a Senior Machine Learning Engineer. His research interest includes big data analytics in smart grids.



Zuzhao Ye (Graduate Student Member, IEEE) received the B.E. degree in thermal energy and power engineering from the University of Science and Technology of China, Hefei, China in 2015. He is currently pursuing the Ph.D. degree in electrical and computer engineering from the University of California at Riverside, Riverside, CA, USA. His research interests include big data analytics, machine learning, and optimization, particularly in their applications to the planning and operation of electric-vehicle charging infrastructure.



Yonggang Peng (Member, IEEE) received the B.S. degree in automation and the M.S. and Ph.D. degrees in control theory and control engineering from the College of Electrical Engineering, Zhejiang University, Hangzhou, China, in 2001, 2004, and 2008, respectively, where he is currently a Professor with the College of Electrical Engineering. His research interests include distributed generation, microgrid, and hybrid AC/DC power system.

Nanpeng Yu (Senior Member, IEEE) received the B.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2006, and the M.S. and Ph.D. degrees in electrical engineering from Iowa State University, Ames, IA, USA, in 2007 and 2010, respectively. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA, USA. His current research interests include machine learning theory, big data analytics in smart grid, electricity market design, and smart energy communities. He is an Associate Editor of the IEEE TRANSACTIONS ON SMART GRID and the IEEE TRANSACTIONS ON SUSTAINABLE ENERGY.