A Mix-Integer Programming Based Deep Reinforcement Learning Framework for Optimal Dispatch of Energy Storage System in Distribution Networks

Hou, Shengren; Salazar, Edgar Mauricio; Palensky, Peter; Chen, Qixin; Vergara, Pedro P.

DOI
10.35833/MPCE.2024.000391

Publication date
2025

Document Version
Final published version

Published in
Journal of Modern Power Systems and Clean Energy

Citation (APA)
Hou, S., Salazar, E. M., Palensky, P., Chen, Q., & Vergara, P. P. (2025). A Mix-Integer Programming Based Deep Reinforcement Learning Framework for Optimal Dispatch of Energy Storage System in Distribution Networks. *Journal of Modern Power Systems and Clean Energy*, *13*(2), 597-608. https://doi.org/10.35833/MPCE.2024.000391

A Mix-integer Programming Based Deep Reinforcement Learning Framework for Optimal Dispatch of Energy Storage System in Distribution Networks

Shengren Hou, *Student Member*, *IEEE*, Edgar Mauricio Salazar, *Member*, *IEEE*, Peter Palensky, *Senior Member*, *IEEE*, Qixin Chen, *Senior Member*, *IEEE*, and Pedro P. Vergara, *Senior Member*, *IEEE*

***Abstract*—The optimal dispatch of energy storage systems (ESSs) in distribution networks poses significant challenges, primarily due to uncertainties of dynamic pricing, fluctuating demand, and the variability inherent in renewable energy sources. By exploiting the generalization capabilities of deep neural networks (DNNs), the deep reinforcement learning (DRL) algorithms can learn good-quality control models that adapt to the stochastic nature of distribution networks. Nevertheless, the practical deployment of DRL algorithms is often hampered by their limited capacity for satisfying operational constraints in real time, which is a crucial requirement for ensuring the reliability and feasibility of control actions during online operations. This paper introduces an innovative framework, named mixed-integer programming based deep reinforcement learning (MIP-DRL), to overcome these limitations. The proposed MIP-DRL framework can rigorously enforce operational constraints for the optimal dispatch of ESSs during the online execution. This framework involves training a *Q*-function with DNNs, which is subsequently represented in a mixed-integer programming (MIP) formulation. This unique combination allows for the seamless integration of operational constraints into the decision-making process. The effectiveness of the proposed MIP-DRL framework is validated through numerical simulations, demonstrating its superior capability to enforce all operational constraints and achieve high-quality dispatch decisions and showing its advantage over existing DRL algorithms.**

***Index Terms*—Voltage regulation, optimal dispatch, distribution network, mixed-integer programming, deep reinforcement learning (DRL), energy management.**

NOMENCLATURE

A. Sets and Indices

$\mathcal{A}$	Set of actions
$\mathcal{B}$	Set of nodes with energy storage systems (ESSs)
$\mathcal{L}$	Set of lines in distribution network
m, n	Indices of nodes
$\mathcal{N}$	Set of nodes in distribution network
t	Index of time steps
i	Index used for summations over layers and units
j	Index of units
k	Index of layers in deep neural network (DNN)
K	Total number of layers (excluding the input layer) in DNN
$\mathcal{P}$	State transition function
$\mathcal{R}$	Reward function
$\mathcal{S}$	Set of states
$\mathcal{T}$	Set of time steps
U_k	Total number of units in layer k

B. Parameters

$\eta^B_{m,c}, \eta^B_{m,d}$	Charging and discharging efficiencies of ESSs
λ	Discount factor
ω	Parameter of trained policy
π_ω	Policy network
ρ_t	Electricity price at time step t
σ	Penalty factor
θ	Parameter of trained critic networks consisting of weights and biases
$\nabla\theta, \nabla\omega$	Gradients for updating policy parameters
b^k_j	Bias of unit j in layer k
c^k_j	Objective function cost of unit j in layer k
d^k_j	Objective function cost for binary activation variable of unit j in layer k

Manuscript received: April 12, 2024; revised: June 23, 2024; accepted: August 9, 2024. Date of CrossCheck: August 9, 2024. Date of online publication: September 19, 2024.

This work was supported by the DATALESs project (No. 482.20.602) jointly financed by the Netherlands Organization for Scientific Research (NWO) and the National Natural Science Foundation of China.

S. Hou, P. Palensky, and P. P. Vergara (corresponding author) are with the Intelligent Electrical Power Grids (IEPG) Group, Delft University of Technology, Delft 2628CD, The Netherlands (e-mail: h.shengren@tudeflt.nl; P.Palensky@tudeflt.nl; P.P.VergaraBarrios@tudeflt.nl).

M. Salazar is with the Electrical Energy Systems (EES) Group, Eindhoven University of Technology, Eindhoven, The Netherlands (e-mail: e.m.salazar.duque@tue.nl).

Q. Chen is with the State Key Laboratory of Power System Operation and Control, Department of Electrical Engineering, Tsinghua University, Beijing 100084, China (e-mail: qxchen@tsinghua.edu.cn).

DOI: 10.35833/MPCE.2024.000391

JOURNAL OF MODERN POWER SYSTEMS AND CLEAN ENERGY

$h(\cdot)$	Activation function, specifically ReLU function for DNN
$\overline{I}^2_{mn}$	The maximum squared value of current magnitude for line *mn*
$\overline{P}^B_m, \underline{P}^B_m$	The maximum and minimum charging/discharging power of ESS connected to node *m*
Q_θ	Critic network
R_{mn}, X_{mn}	Resistance and reactance of line *mn*
$\overline{SOC}^B_m, \underline{SOC}^B_m$	The maximum and minimum states of charge (SOCs) of ESS connected to node *m*
$\overline{V}^2, \underline{V}^2$	The maximum and minimum squared values of voltage magnitudes

C. Continuous Variables

$\overline{E}^B_m$	Capacity of ESS connected to node *m*
$P_{mn,t}, Q_{mn,t}, I_{mn,t}$	Active power, reactive power, and current of line *mn* at time step *t*
$P^{PV}_{m,t}$	Active power generation of photovoltaic (PV) system at node *m* at time step *t*
$P^N_{m,t}$	Net power of node *m* at time step *t*
$P^B_{m,t}$	Charging/discharging power of ESS connected to node *m* at time step *t*
$P^D_{m,t}, Q^D_{m,t}$	Active and reactive power demands of node *m* at time step *t*
$P^S_{m,t}, Q^S_{m,t}$	Active and reactive power from slack node at time step *t*
$SOC^B_{m,t}$	SOC of ESS connected to node *m* at time step *t*
$V_{m,t}$	Voltage of node *m* at time step *t*
s^k_j	Slack variable associated with ReLU function for unit *j* in layer *k*
x^k_j	Output of unit *j* in layer *k*

D. Matrices and Vectors

$\boldsymbol{b}^{k-1}$	Matrix of biases for layer $k-1$
$\boldsymbol{W}^{k-1}$	Matrix of weights for layer $k-1$
$\boldsymbol{x}^k$	Output vector of layer *k*

I. Introduction

THE proliferation of distributed energy resources (DERs) poses various challenges in the control and operation of electrical distribution networks [1]. Voltage issues can be observed in networks with high photovoltaic (PV) penetration and peak loads. To overcome this problem, the energy storage systems (ESSs) are being increasingly deployed, offering ancillary services such as voltage magnitude regulation to the distribution system operators (DSOs). These ancillary services can be provided by exploiting the flexibility of ESSs in response to a dynamic electricity price throughout the day, which can be obtained by solving an optimal dispatch problem of ESSs. From the view of DSO, the defined dispatch of ESSs should minimize the operational costs while ensuring the voltage magnitude constraints of the distribution network. Nevertheless, such a dispatch problem is inherently challenging due to the stochastic and uncertain nature of the dynamic electricity prices, the demand consumption, and the renewable generation, e.g., PV generation [2].

Traditional research, e.g., [3], in the optimal dispatch of ESSs has predominantly focused on developing accurate models and approximated formulations that make the problem amenable for commercial solvers, collectively known as model-based approaches. Nevertheless, these model-based approaches struggle with real-time solution quality due to the increased complexity and uncertainty introduced by DERs [4]. To overcome these shortcomings, the model-free approaches have been proposed as an alternative. These model-free approaches model the optimal dispatch problem of ESSs as a Markov decision process (MDP) and leverage reinforcement learning (RL) algorithms to define the optimal sequential decisions [5], [6]. By exploiting the good generalization capabilities of deep neural networks (DNNs), the deep reinforcement learning (DRL) algorithms can perform sequential interpretations of data, learning good-quality control models that can adapt to the stochastic nature of an environment [7].

Implementing DRL algorithms in a real system typically follows a two-stage process: ① an offline initial training stage utilizing a simulator, and ② an online execution of the trained algorithm into the real system [8]. This allows refining and rigorously testing DRL algorithms before their exposure to the real system. As for the optimal dispatch problem of ESSs, the most crucial aspect is ensuring the feasibility and safety during the online execution of DRL algorithms [9]. Nevertheless, after training, the standard DRL algorithms cannot provide the feasibility for defined actions during the online execution, impeding the implementation of DRL algorithms in the dispatch problems of ESSs.

Several approaches have been developed to improve the constraint enforcement capabilities of DRL algorithms [10]. The enforcement of soft constraints is currently the most widely used approach [11]. In this approach, a large and fixed penalty term is incorporated into the reward function when training the parameters of the control policy [12]. This enables the DRL algorithm to avoid actions that result in unfeasible operations. For instance, in [13], the problem of dispatching PV inverters has been addressed by a decentralized framework that penalizes RL agents when their actions lead to voltage magnitude violations. Although these strategies may enforce operational constraints during training, they cannot guarantee the feasibility of the defined operating schedule in real time, especially during peak periods of consumption and renewable generation [14].

Instead, safe DRL algorithms are implemented to directly handle constraints in distribution network operations without adding penalty terms in the reward function. In [15], a safe DRL algorithm is introduced to define a fast-charging strategy for lithium-ion batteries to enhance the efficiency of EV charging without compromising battery safety. Utilizing the soft actor-critic (SAC) based Lagrange DRL in a cyber-physical system, the charging speeds are optimized by leveraging an electro-thermal model, outperforming existing deep deterministic policy gradient (DDPG) and SAC based DRL algorithms in terms of optimality. To ensure that the updated policy stays within a feasible set, a cumulative constraint violation index is maintained below a predetermined threshold in [16] and [17]. This approach is also used in [18] and [19], where the constraint violation index is designed to reflect the voltage and current magnitude violation levels due to the defined dispatch of ESSs. Nevertheless, enforcing constraints via cumulative indices can only provide a probabilistic notion of safety, failing to enforce voltage and current magnitude constraints in real time due to their instantaneous nature. Alternatively, a projection operator can be developed to project actions defined by the DRL algorithm into a feasible set [20], [21]. For instance, the projection operator proposed in [22] uses the action defined by the DRL algorithm as a starting point to solve a mathematical programming formulation, thus ensuring compliance with the constraints. A similar approach is implemented in [23] to regulate the voltage magnitude of distribution networks via the control of smart transformers. However, implementing such projection operators can degrade the performance of DRL algorithm, as discussed in [24].

A summary of different constraint enforcement approaches used by safe DRL algorithms in various operational problems of energy system is presented in Table I [14], [15], [18], [19], [22], [25]-[38].

TABLE I
SUMMARY OF CONSTRAINT ENFORCEMENT APPROACHES USED BY SAFE DRL ALGORITHMS IN VARIOUS OPERATIONAL PROBLEMS OF ENERGY SYSTEM

Reference	Operational problem	Constraint enforcement approach	Is open-accessed?
[26]	Microgrid operation	Penalty function	No
[27]	Voltage regulation		Yes
[28]	Optimal power flow		No
[29], [30]	Energy dispatch		No
[14]	Optimal energy system dispatch		Yes
[31]	Home energy management	Primal-dual DDPG	No
[32]	Electric vehicle (EV) in microgrid	Primal-dual SAC	No
[33]	Microgrid energy management	Constrained policy optimization	No
[34]	Cooling system control	Gaussian process	No
[15]	EV charging/ discharging operation	Lagrange SAC	No
[35]	Distribution network operation	Safe layer	No
[36]	Voltage regulation	Safe layer	Yes
[25]	Microgrid operation	*Q*-network formulated MIP	Yes
[22]	Energy management	Safe layer	No
[37]	Energy hub trading	Gaussian process or safe layer	No
[38]	Microgrid operation	Action projection	No
[18]	Distribution network operation	Constrained policy optimization	No
[19]	EV management		

The optimal dispatch of ESSs mandates strict operational constraints so that the safety and feasibility can be guaranteed, especially during the online execution [25]. Although the safe DRL algorithms presented in Table I notably enhance the constraint enforcement capabilities and mitigate the violations significantly during the training, a significant challenge persists: these algorithms cannot provide control decisions with a theoretical guarantee of constraint enforcement during the online execution. This limitation poses a substantial barrier to the widespread implementation of DRL algorithms for the optimal dispatch of ESSs. It is paramount to ensure the action feasibility in real-time applications, not only for the operation reliability of ESSs but also for the broader adoption and trust in DRL solutions within this field.

In our previous work [25], a value-based safe DRL algorithm is proposed to address the microgrid operation problem with strict constraint enforcement of power balance equality. This work integrates the optimization techniques with DRL theory, representing the trained *Q*-network as a mixed-integer programming (MIP) formulation. Leveraging this innovative approach, we now broaden the scope of our research to conceptualize and develop a more versatile and comprehensive framework that strictly enables state-of-the-art (SOTA) actor-critic DRL algorithms to enforce the operational constraints. This framework is called MIP-DRL. Distinct from our earlier contribution, the proposed MIP-DRL framework is not confined to a specific algorithm but is envisioned as a general framework that can empower many standard actor-critic DRL algorithms to enforce the operational constraints. Our contributions are systematically structured to highlight the innovation and applicability of the proposed MIP-DRL framework, as follows.

1) We propose the MIP-DRL framework to enforce operational constraints with strict adherence during the online operations. Utilizing the robust constraint enforcement capabilities of MIP, the proposed MIP-DRL framework ensures compliance with operational constraints, guaranteeing zero constraint violations during the online execution. This innovation extends the theoretical underpinnings of DRL applicability and enables the feasibility of its real-time applications.

2) The proposed MIP-DRL framework broadens its utility across diverse DRL algorithms that employ DNNs for *Q*-function approximation. We implement and test the proposed MIP-DRL framework with SOTA standard DRL algorithms such as DDPG and SAC, demonstrating the capability to strictly enforce the operational constraints.

3) Demonstrating its practical efficacy, the proposed MIP-DRL framework is used to address the complex challenge of the optimal dispatch problem for ESSs in distribution networks. The results illustrate the performance superiority of the proposed MIP-DRL framework over existing standard or safe DRL algorithms to improve the performance and ensure action feasibility, even in unseen scenarios.

II. MATHEMATICAL FORMULATION FOR OPTIMAL DISPATCH PROBLEM OF ESSS

The optimal dispatch of ESSs in a distribution network can be modeled using the nonlinear programming (NLP) formulation given by (1) - (11). The objective function in (1) aims to minimize the total operational cost over the time horizon $\mathcal{T}$, comprising the cost of importing power from the main grid. The operational cost at time step t is settled according to the day-ahead electricity prices ρ_t in €/kWh.

$$\min_{P^B_{m,t},\forall m\in\mathcal{B},\forall t\in\mathcal{T}} \left\{\sum_{t\in\mathcal{T}}\left[\rho_t \sum_{m\in\mathcal{N}}\left(P^D_{m,t}+P^B_{m,t}-P^{PV}_{m,t}\right)\Delta t\right]\right\} \tag{1}$$

s.t.

$$\sum_{nm\in\mathcal{L}} P_{nm,t}-\sum_{mn\in\mathcal{L}}\left(P_{mn,t}+R_{mn}I^2_{mn,t}\right)+P^B_{m,t}+P^{PV}_{m,t}+P^S_{m,t}= P^D_{m,t} \quad \forall m\in\mathcal{N},\ \forall t\in\mathcal{T} \tag{2}$$

$$\sum_{nm\in\mathcal{L}} Q_{nm,t}-\sum_{mn\in\mathcal{L}}\left(Q_{mn,t}+X_{mn}I^2_{mn,t}\right)+Q^S_{m,t}=Q^D_{m,t} \quad \forall m\in\mathcal{N},\ \forall t\in\mathcal{T} \tag{3}$$

$$V^2_{m,t}-V^2_{n,t}=2\left(R_{mn}P_{mn,t}+X_{mn}Q_{mn,t}\right)+\left(R^2_{mn}+X^2_{mn}\right)I^2_{mn,t} \quad \forall m,n\in\mathcal{N},\ \forall t\in\mathcal{T} \tag{4}$$

$$V^2_{m,t}I^2_{mn,t}=P^2_{mn,t}+Q^2_{mn,t} \quad \forall m,n\in\mathcal{N},\ \forall t\in\mathcal{T} \tag{5}$$

$$SOC^B_{m,t}=SOC^B_{m,t-1}+\begin{cases}\dfrac{\eta^B_{m,c}P^B_{m,t}\Delta t}{\overline{E}^B_m} & P^B_{m,t}>0,\forall m\in\mathcal{B},\forall t\in\mathcal{T}\\[2ex] \dfrac{P^B_{m,t}\Delta t}{\eta^B_{m,d}\overline{E}^B_m} & P^B_{m,t}<0,\forall m\in\mathcal{B},\forall t\in\mathcal{T}\end{cases} \tag{6}$$

$$\underline{SOC}^B_m\le SOC^B_{m,t}\le \overline{SOC}^B_m \quad \forall m\in\mathcal{B},\forall t\in\mathcal{T} \tag{7}$$

$$\underline{P}^B_m\le P^B_{m,t}\le \overline{P}^B_m \quad \forall m\in\mathcal{B},\forall t\in\mathcal{T} \tag{8}$$

$$\underline{V}^2\le V^2_{m,t}\le \overline{V}^2 \quad \forall m\in\mathcal{N},\ \forall t\in\mathcal{T} \tag{9}$$

$$0\le I^2_{mn,t}\le \overline{I}^2_{mn} \quad \forall mn\in\mathcal{L},\forall t\in\mathcal{T} \tag{10}$$

$$P^S_{m,t}=Q^S_{m,t}=0 \quad \forall m\in\mathcal{N}\setminus\{1\},\forall t\in\mathcal{T} \tag{11}$$

The steady-state operation of distribution network is modeled by the load flow sweep method, as shown in (2)-(5), in terms of the active power $P_{mn,t}$, reactive power $Q_{mn,t}$, and current magnitude $I_{mn,t}$ of line mn at time step t, and the voltage magnitude of node m at time step t $V_{m,t}$. Formula (6) models the state of charge (SOC) dynamics of ESSs in set $\mathcal{B}$, while (7) enforces the SOC limits, and $\mathcal{B}\subseteq\mathcal{N}$. Finally, (8) enforces the discharging/charging limits of ESSs, (9) and (10) enforce the voltage magnitude and line current limits, respectively, while (11) enforces that only one node is connected to the substation. Notice that to solve the above NLP formulation, all long-term operational data (e. g., expected PV generation and consumption) must be collected to properly define the dispatch decisions of ESSs, while the power flow formulation must also be considered to enforce the voltage and current magnitude limits.

In the formulated problem, we assume that only PV panels and ESSs are installed in the distribution networks. The active power flexibility provided by the dispatch of ESSs is used to provide economic benefits and ensure safe voltage magnitude levels for the distribution network. It should be mentioned that the ESS model can be further refized, including a detailed physical dynamic model, e. g., efficiency curves, temperature, and degradation. However, since this paper aims to assess the performance of the proposed MIP-DRL framework, the ESS dynamics are simplified using the linear model [12].

III. MDP Formulation for Dispatch Problem of ESSs

The above mathematical formulation can be modeled as a finite MDP, represented by a 5-tuple $(\mathcal{S},\mathcal{A},\mathcal{P},\mathcal{R},\gamma)$. The decision of which action a_t is chosen in a particular state s_t is governed by a policy $\pi(a_t|s_t)$. In a standard RL algorithm, an RL agent employs the policy $\pi(a_t|s_t)$ to interact with the formulated MDP, which defines a trajectory of states, actions, and rewards: $\tau=(s_0,a_0,s_1,a_1,\dots)$. Here, the goal of RL agent is to estimate a policy that maximizes the expected discounted return $J(\pi)=\mathbb{E}_{\tau\sim\pi}\left(\sum_{t\in\mathcal{T}}\gamma^t r_t\right)$, where $\mathbb{E}_{\tau\sim\pi}$ is the expectation of the trajectory distribution under the current policy; and $\sum_{t\in\mathcal{T}}\gamma^t r_t$ is the cumulative return.

Different from the standard RL algorithm, in a constrained MDP, the RL agent aims to estimate a policy π confined in a feasible set $\varPi_C=\{\pi: J_{C_i}(\pi)\le 0, i=1,2,\dots\}$, where $J_{C_i}(\pi)=\mathbb{E}_{\tau\sim\pi}\left(\sum_{t\in\mathcal{T}}\gamma^t C_{i,t}\right)$ is a cost-based constraint function induced by the constraint violation functions $C_{i,t}(\cdot), i=1,2,\dots$; and $\sum_{t\in\mathcal{T}}\gamma^t C_{i,t}$ is the cumulative constraint violation. Based on these definitions, a constrained MDP can be formulated as a constrained optimization problem:

$$\begin{cases}\max\limits_{\pi} J(\pi)=\mathbb{E}_{\tau\sim\pi}\left(\sum\limits_{t\in\mathcal{T}}\gamma^t r_t\right)\\ \text{s.t. } J_{C_i}(\pi)\le 0 \quad \forall i=1,2,\dots\end{cases} \tag{12}$$

A more detailed MDP description for the optimal dispatch problem of ESSs is presented below.

The state $s_t=(P^N_{m,t}|_{m\in\mathcal{N}},\rho_t,SOC^B_{m,t}|_{m\in\mathcal{B}},t)$ denotes the operating status of the distribution network that the agent can observe. The PV generation $P^{PV}_{m,t}$ and consumption $P^D_{m,t}$, day-ahead electricity price ρ_t, and current time step t belong to endogenous features, which are independent of the agent actions, while $SOC^B_{m,t}$ belongs to exogenous features, which depends on the agent action and previous state s_{t-1}.

The action $a_t=(P^B_{m,t}|_{m\in\mathcal{B}})$, which refers to the charging/discharging dispatch for the ESS connected to node m in the distribution network. $a_t\in\mathcal{A}$, and $\mathcal{A}$ is a continuous space.

Given the state s_t and action a_t, the system transiting to the next state s_{t+1} is defined by the transition probability:

$$\mathcal{P}(S_{t+1},R_t|S_t,A_t)=\Pr\{S_{t+1}=s_{t+1},R_t=r_t|S_t=s_t,A_t=a_t\} \tag{13}$$

The transition probability function $\mathcal{P}$ models the endogenous distribution network and ESS dynamics, determined by the physical model of the distribution network and ESSs, and the exogenous uncertainty caused by the PV generation, demand consumption, and day-ahead electricity price dynamics. In practice, it is not possible to build an accurate mathematical model for such a transition probability function. Nevertheless, the model-free RL algorithms do not require prior knowledge of function $\mathcal{P}$ as it can be implicitly learned by interacting with the environment.

RL algorithms can learn representative operation strategies from interactions with the environment. To achieve this goal, the environment must provide a reward r_t to quantify the goodness of any action taken during the interaction process. In this case, the raw reward is defined as the negative value of the operational cost for the distribution network, i.e.,

$$R_t(s_t,a_t)=r_t=-\rho_t\left[\sum_{m\in\mathcal{N}}(P^D_{m,t}+P^B_{m,t}-P^{PV}_{m,t})\right]\Delta t \tag{14}$$

DRL algorithms optimize the operational costs while adhering to the operational constraints of ESSs and the distribution network. These constraints include the SOC limit (7), the maximum discharging/charging limit (8), and voltage magnitude constraint (9). While constraints on action spaces ((7) and (8)) are straightforward to enforce through action boundaries, the voltage magnitude constraint (9) requires addressing the physical dynamics of the distribution network. To manage these limits, the constraint violation functions $C_{m,t}$ are integrated into the reward function (14) as penalties, converting the constrained optimization problem (12) into an unconstrained one, redefined as:

$$r_t=-\rho_t\left[\sum_{m\in\mathcal{N}}(P^D_{m,t}+P^B_{m,t}-P^{PV}_{m,t})\right]\Delta t-\sigma\sum_{m\in\mathcal{B}}C_{m,t}(V_{m,t}) \tag{15}$$

where σ balances the operational costs against penalties for constraint violations. The constraint violation functions $C_{m,t}$ in (15) can be modeled using different functions, e. g., L_2 function, which is defined as [12]:

$$C_{m,t}=\min\left\{0,\frac{\overline{V}-\underline{V}}{2}-\left|V_0-V_{m,t}\right|\right\}\quad\forall m\in\mathcal{B} \tag{16}$$

Nevertheless, it is critical to notice that enforcing operational constraints by only adding a penalty term into the reward function during the training might lead to infeasible operational states during the online execution, as observed in [14]. To address this, we propose the MIP-DRL framework, leveraging constraint enforcement capabilities of MIP to ensure feasible solutions during the online execution.

IV. Constraint Enforcement of Proposed MIP-DRL Framework

The proposed MIP-DRL framework is defined through two main procedures: ① training, where the Q-function is approximated, and ② deployment, which is executed during the online decision-making. Both of these procedures are explained in detail below [39]-[46].

A. Step-by-step Training

The step-by-step training for the proposed MIP-DRL framework integrates concepts from actor-critic DRL algorithms, including DDPG [39], twin delayed deep deterministic policy gradient (TD3) [40], and SAC [44], within a unified training procedure. Figure 1 illustrates the interaction of the actor $\pi_\omega(\cdot)$ (also known as policy) and critic $Q_\theta(\cdot)$ (also known as Q-function) models with the environment (distribution network). Initially, the parameters of actor $\pi_\omega(\cdot)$ and critic $Q_\theta(\cdot)$ are randomly initialized. The training progresses through interaction with the environment: actions a_t are sampled from the actor model, prompting the environment to transition to new states and generate rewards, as shown in Fig. 1(a). These state transitions and rewards inform the storage of transition tuples (s_t,a_t,r_t,s_{t+1}) in a replay buffer R. Subsequently, the subsets of these tuples are used to iteratively update the actor and critic models, enhancing the performance of policy and accuracy of Q-function estimation.

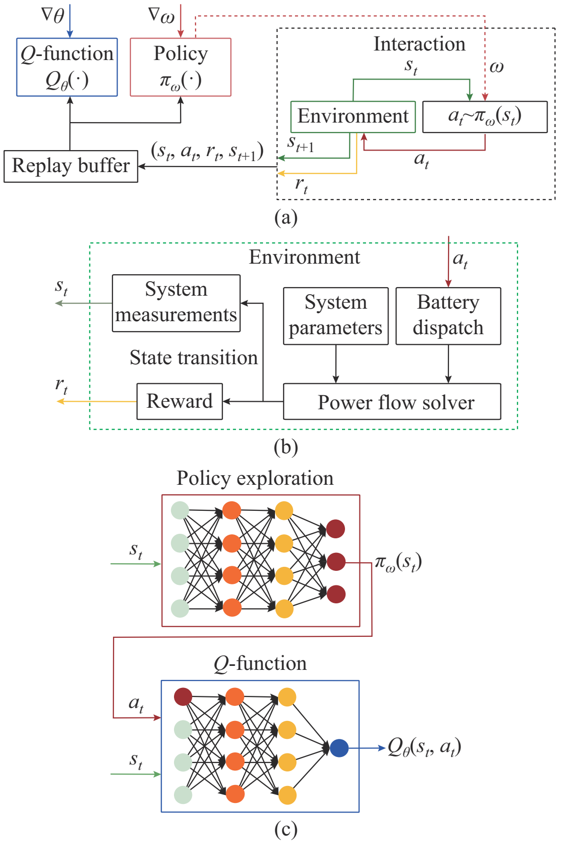

Fig. 1. Training of proposed MIP-DRL framework. (a) Interaction with environment. (b) Environment (distribution network). (c) Policy network $\pi_\omega(s_t)$.

In general, the main objective of actor-critic algorithms is to approximate a good policy network $\pi_\omega(\cdot)$ while the Q-function is used during exploration to improve the quality of the policy network. After training, the Q-function $Q_\theta(\cdot)$ is discarded. Different from this procedure, the proposed MIP-DRL framework follows the actor-expert definition [45], which aims to get an optimal action based on the optimal Q-function $Q_\theta(\cdot)$. Thus, during the training, the policy network $\pi_\omega(\cdot)$ is only used to explore and exploit new states and actions to improve the quality of Q-function $Q_\theta(\cdot)$, while the policy network π_ω is discarded. Once a good-quality representation of $Q^*_\pi(\cdot)$ is obtained via the optimal Q-function $\hat{Q}(\cdot)$, the state s_t and optimal actions a_t can be sampled from the optimal policy, i.e., $a_t{\sim}\pi^*(s_t)$, which is obtained as:

$$\pi^*(s_t)=\max_{a\in\mathcal{A}}\hat{Q}(s_t,a) \tag{17}$$

As a result, the training procedure for the MIP-DRL algorithms, i. e., MIP-DDPG, MIP-TD3, and MIP-SAC, resembles that of their corresponding standard DRL algorithms. Nevertheless, the actions defined using only such a Q-function $Q_\theta(\cdot)$ cannot strictly enforce the operational constraints during the online execution. To overcome this, the proposed MIP-DRL framework leverages the MIP formulation of the trained Q-function $Q_\theta(\cdot)$ to enforce operational constraints during the online execution.

B. Constraint Enforcement During Online Execution

The trained Q-function $Q_\theta(\cdot)$ obtained from MIP-DRL algorithms with fixed parameters θ can be transformed into an MIP model, facilitating the operational constraint enforcement during the online execution. This transformation enables the incorporation of system constraints directly into the action decision process, as detailed in [25].

Based on the definitions in [25], the Q-function $Q_\theta(\cdot)$ obtained from trained MIP-DRL algorithms with fixed parameters θ can be modeled as a valid MIP problem [46]:

$$\max_{x_j^k, s_j^k, z_j^k} \left\{ \sum_{k=0}^{K} \sum_{j=1}^{U_k} c_j^k x_j^k + \sum_{k=1}^{K} \sum_{j=1}^{U_k} d_j^k z_j^k \right\} \tag{18}$$

s.t.

$$\begin{cases} \sum_{i=1}^{U_k} w_{ij}^{k-1} x_i^{k-1} + b_j^{k-1} = x_j^k - s_j^k \\ x_j^k, s_j^k \geq 0 \\ z_j^k \in \{0, 1\} \\ z_j^k = 1 \rightarrow x_j^k \leq 0 \\ z_j^k = 0 \rightarrow s_j^k \leq 0 \end{cases} \quad \forall k, \forall j \tag{19}$$

$$lb_j^0 \leq x_j^0 \leq ub_j^0 \quad j \in l_0 \tag{20}$$

$$\begin{cases} lb_j^k \leq x_j^k \leq ub_j^k \\ \overline{lb}_j^k \leq s_j^k \leq \overline{ub}_j^k \end{cases} \quad \forall k, \forall j \tag{21}$$

Each layer $k \in \{0, 1, \dots, K\}$ in DNN-formulated Q-function has U_k units, with j being the unit index in layer k. We denote the output vector of layer k as $\boldsymbol{x}^k = [x_j^k]$, $j = 1, 2, \dots, U_k$. The weights $w_{i,j}^{k-1}$ and biases b_j^k are fixed (constant) parameters, and the same holds for the objective function costs c_j^k and d_j^k. The activation function output for each unit is defined by (19), while (20) and (21) define the lower and upper bounds for the x and s variables. For the input layer ($k = 0$), the input x^0 is the same as the inputs of Q-function $Q_\theta(\cdot)$, i. e., state s_t and action a_t, while the defined bounds have physical meanings (the same limits as the inputs of $Q_\theta(\cdot)$). For $k \geq 1$, the bounds are defined based on the fixed parameters, as explained in [25].

Then, the max-Q problem for Q-function $Q_\theta(\cdot)$ in (17) is equivalent to solving (18)-(21) [43]. In this case, as the decision variables are the actions a_t (corresponding to the charging/discharging dispatch of ESSs), the SOC limit (7), the charging/discharging limit (8), the voltage magnitude constraint (9) can all be added on top of (18)-(21). As a result, the optimal actions obtained by solving this MIP formulation strictly enforce all the actions and operational constraints of environment. A general mathematical proof of the optimality for the proposed MIP-DRL framework is presented in [25]. This integrated MIP formulation can be represented as:

$$\begin{cases} \max\limits_{a \in \mathcal{A}, x_j^k, s_j^k, z_j^k} \left\{ \sum_{k=0}^{K} \sum_{j=1}^{U_k} c_j^k x_j^k + \sum_{k=1}^{K} \sum_{j=1}^{U_k} d_j^k z_j^k \right\} \\ \text{s.t. } (19)\text{-}(21), (7)\text{-}(9) \end{cases} \tag{22}$$

To better understand the MIP formulation, Fig. 2 shows a visual representation, where a_1 and a_2 are the action values in two dimensions. Such formulation defines the linear space within the blue line, whose boundaries are formed by the hyperplanes defined by the activation functions derived from the deconstructed DNN $Q_\theta(s, \cdot)$ [44]. In Fig. 2, the blue point represents the optimal solution of (17), denoted as $\hat{a}$. Note that $\hat{a}$ also corresponds to the solution of the MIP formulation in (18)-(21). Similarly, the set of constraints (7)-(9) forms the linear space, as represented within the dashed grey line. Therefore, solving the MIP formulation in (22) provides solution a^*, which represents the optimal solution of (17) that simultaneously enforces the operational constraints defined by (7)-(9).

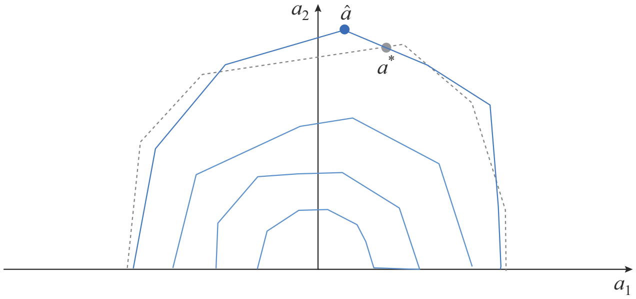

Fig. 2. Visualization representation of MIP formulation.

The online execution for the MIP-DRL algorithms, i. e., MIP-DDPG, MIP-TD3, and MIP-SAC, as shown in Algorithm 1, takes this MIP representation, incorporating not only the structure of Q_θ but also system-specific constraints, e.g., voltage magnitude constraint. By solving the MIP formulation in (22), we obtain action a_t that maximizes the expected reward while strictly adhering to operational constraints, thus ensuring the feasibility and optimality of the decisions made by the proposed MIP-DRL framework.

Algorithm 1: online execution for MIP-DDPG, MIP-TD3, and MIP-SAC

```
Extract trained parameters $\theta$ from $Q_\theta(\cdot)$
Formulate $Q_\theta(\cdot)$ as an MIP formulation according to (18)-(21) and add
  the operational constraints (7)-(9)
Extract initial state $s^0$ based on real-time data
for $t \in \mathcal{T}$ do
    Get the optimal action by solving (22) using commercial MIP solvers
      for state $s_t$
end for
```

V. Simulation Results and Discussions

A. Simulation Setup

1) Environment Data and Framework Implementation

To demonstrate the effectiveness of the proposed MIP-DRL framework, a modified IEEE 34-node test system is used, as shown in Fig. 3. ESSs are placed at nodes 12, 16, 27, 30, and 34 due to their higher chances of over- and under-voltage issues. The training data used corresponds to historical day-ahead electricity prices in the Dutch market, while load and PV generation measurements with a 15-min resolution are provided by a DSO. The original one-year dataset is divided into two additional datasets: training and testing datasets. The training dataset contains the first three weeks of data in each month, while the testing dataset contains the remaining data. This allows the DRL algorithm to learn any seasonal and weekly PV generation and load consumption data [25].

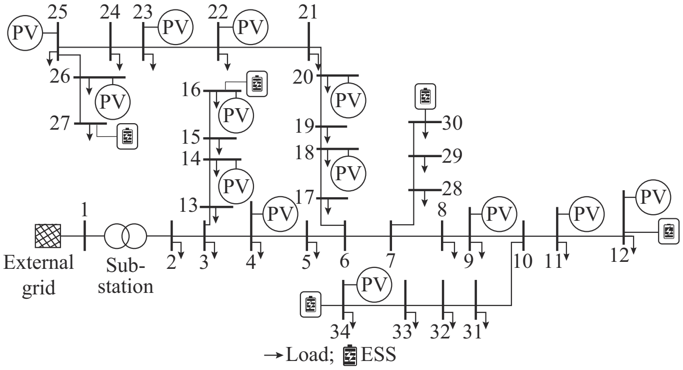

Fig. 3. Modified IEEE 34-node test system with distributed PV generation and ESSs.

Table II summarizes the key parameters for the MIP-DDPG, MIP-TD3, and MIP-SAC. This includes the discount factor γ, optimizer type, learning rate, batch size, and replay buffer size for each algorithm. Additionally, the specific parameters for the entropy in the MIP-SAC, the reward function, and the operational limits for ESSs are listed. The voltage magnitude limits are defined as $\overline{V}=1.05$ and $\underline{V}=0.95$ p. u.. PyTorch and OMLT [45] packages have been used to implement the proposed MIP-DRL framework. Default settings shown in Table II are used for all the implemented MIP-DRL algorithms. The MIP-DRL algorithms are solved with Gurobi [46]. All implemented algorithms and the environment are open-accessed in [47] and [48].

2) Validation and Benchmarks for Comparison

To demonstrate the superior performance of the MIP-DRL algorithms (MIP-DDPG, MIP-TD3, and MIP-SAC), we compare their dispatch outcomes with those of standard DRL algorithms (DDPG, TD3, and SAC) and a safe DRL algorithm (safe DDPG). The hyperparameters of DDPG, TD3, and SAC are aligned with those of MIP-DDPG, MIP-TD3, and MIP-SAC, respectively. For safe DDPG, we adopt a linear safe layer and follow the default parameter settings as described in [20]. The comparison relies on two key metrics: ① operational cost, which reflects the economic efficiency of the schedules, and ② cumulative penalty for voltage magnitude violations, indicating the ability to enforce constraints during the online execution of these algorithms. Furthermore, we also use the optimal global solution based on a perfect forecast for the next 24 hours as the benchmark. This optimal solution is obtained by solving the NLP formulation in Section III, implemented using Pyomo and IPOPT solver.

TABLE II
KEY PARAMETERS FOR DIFFERENT ALGORITHMS AND ENVIRONMENT

Algorithm or environment	Parameter
MIP-DDPG	$\gamma=0.995$ Optimizer type: Adam Learning rate: 6×10^{-4} Batch size: 512 Replay buffer size: 4×10^{5}
MIP-TD3	$\gamma=0.995$ Optimizer type: Adam Learning rate: 6×10^{-4} Batch size: 512 Replay buffer size: 4×10^{5}
MIP-SAC	$\gamma=0.995$ Optimizer type: Adam Learning rate: 6×10^{-4} Batch size: 512 Replay buffer size: 4×10^{5} Entropy: fixed
Reward	$\sigma=400$
ESSs	$\overline{P}^{B}=100$ kW, $\underline{P}^{B}=-100$ kW, $\overline{SOC}^{B}=0.8$, $\underline{SOC}^{B}=0.2$, $\eta_{c}^{B}=0.98$, $\eta_{d}^{B}=0.98$

B. Performance of MIP-DRL Algorithms on Training Set

Figure 4 displays the average total reward (15), operational cost (the first term in (15)), and the cumulative penalty of voltage magnitude violations (the second term in (15)) during the training process for the MIP-DRL algorithms. The results shown in Fig. 4 are the average of over five executions. The average total reward increases rapidly during the training, while simultaneously, the cumulative penalty of voltage magnitude violations decreases. This is a typical training trajectory of the penalty-based DRL algorithms. At the beginning of the training process, the parameters of DNN are randomly initialized, and as a consequence, the actions defined cause a high penalty of voltage magnitude violations. Throughout the training, introducing a large penalty term in the reward definition in (14) leads to updating the parameters of DNN, resulting in higher quality of actions. It primarily learns to reduce the voltage magnitude violations, and later on improves the general performance. All three MIP-DRL algorithms converge at around 1000 episodes. The total rewards of MIP-TD3 and MIP-DDPG converge at 2.01 ± 0.02 and 1.94 ± 0.02, respectively, and that of MIP-SAC converges at a low value of 1.57 ± 0.01, indicating that MIP-SAC has a lower quality of actions. Notice that for MIP-DDPG and MIP-TD3, the operation costs significantly increase during the training process, while MIP-SAC does not improve after 400 episodes.

After the last training episode, the cumulative penalty of voltage magnitude violations of MIP-TD3 is around 1. In contrast, a higher cumulative penalty of voltage magnitude violations for the MIP-DDPG and MIP-SAC is observed at around 2. This result shows that MIP-DRL algorithms can effectively learn from interactions, reducing the cumulative penalty of voltage magnitude violations while minimizing the total operation cost by learning to dispatch the ESSs correctly. However, these trained policies cannot strictly enforce voltage magnitude constraints. If such algorithms are used directly during the online execution, they might lead to infeasible operation, causing voltage violations.

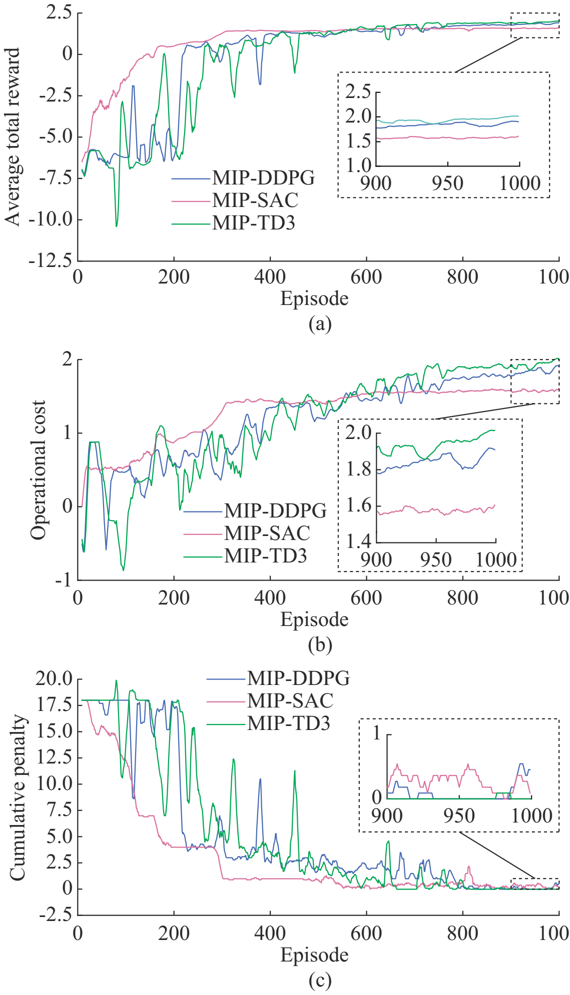

Fig. 4. Results during training process for MIP-DRL algorithms. (a) Average total reward. (b) Operational cost. (c) Cumulative penalty of voltage magnitude violations.

C. Constraint Enforcement Capabilities and Performance

Figure 5 displays the voltage magnitudes of the nodes to which the ESSs are connected, the SOC of ESSs, and day-ahead electricity price during a typical day in the test dataset. The results in Fig. 5(c)-(h) are obtained after using the dispatch decisions provided by the MIP-DDPG, MIP-TD3, and MIP-SAC. As can be observed in Fig. 5(a), if the operation of ESSs is disregarded, the voltage magnitude at node 27 faces under-voltage problems during 14:00-16:00 and 18:00-20:30. Thus, a proper dispatch of the available ESSs must enforce that such voltage magnitude constraints are met. As all the MIP-DRL algorithms dispatch the ESS connected to node 27 in the discharging mode during 14:00-16:00 and 18:00-20:30, all under-voltage issues are solved. In terms of dispatch decisions, all the MIP-DRL algorithms first learn to discharge all ESSs to the minimum SOC during 00:00-06:00, as observed in Fig. 5(b), (f), and (h). Then, all ESSs are dispatched in the charging mode during 10:00-17:00 when the electricity price is low, and then operate in the discharging mode during 16:00-22:00. This operational dispatch during the peak consumption period reduces the amount of power consumption while simultaneously solves the under-voltage issues. Compared with the MIP-DDPG, the MIP-TD3 and MIP-SAC provide more conservative dispatch decisions, leading to higher operational costs. The operational cost resulting from the dispatch defined by the MIP-DDPG is 13.87 k€, which is 3.1% and 7.5% lower than the dispatch defined by MIP-TD3 and MIP-SAC, respectively.

D. Performance Comparison with Benchmarks

Figure 6 displays the charging/discharging decisions and SOC changes of the ESS connected to node 27 provided by the MIP-DDPG, safe DDPG, and standard DDPG as well as the optimal solution provided by solving the NLP formulation. Compared with the optimal solution provided by solving the NLP formulation, the dispatch decisions provided by the MIP-DDPG in Fig. 6(c) show a similar operation pattern, especially in the afternoon when the electricity price changes dynamically, as shown in Fig. 5(b). As expected, when the electricity price is low during 10:00-14:30, the MIP-DDPG dispatches the ESS in the charging mode, while when the electricity price is high during 17:00-22:00, the ESS is dispatched in the discharging mode. In this sense, the standard DDPG and safe DDPG capture and exploit such arbitrage opportunities. Although the MIP-DDPG fails to capture such behavior for this specific ESS, the decisions defined for the remaining ESSs ensure a maximization of profits without voltage magnitude violations. In this case, the costs of dispatch decisions provided by the standard DDPG and safe DDPG are 22.3% and 27.3% higher than that by the MIP-DDPG, respectively. This shows that the standard DDPG and safe DDPG fail to fully leverage and coordinate all ESSs connected to the distribution network.

Figure 7 displays the voltage magnitude of node 27 to which an ESS is connected. Without the dispatch of ESSs, node 27 suffers serious under-voltage conditions during 14:30-15:30 and 17:30-21:00 due to overloading. As expected, the dispatch decisions provided by the MIP-DDPG strictly enforce the voltage magnitude constraints due to the feasibility guarantee. In contrast, although the dispatch decisions provided by the standard DDPG and safe DDPG can significantly alleviate the under-voltage condition, they fail to enforce voltage magnitude constraints during several time periods such as 18:30-19:30 and 20:00-21:00. These results indicate that the constraint enforcement capabilities of both standard DDPG and safe DDPG are not capable of handling complex stochastic environments (such a distribution network), and even the projection layer deployed by the safe DDPG fails to map the relationship between actions and constraints accurately, ultimately deploying unfeasible actions.

Comparing the optimal solution provided by solving the NLP formulation, it can be observed that the MIP-DRL algorithms dispatch the ESSs following a more conservative approach (see charging/discharging behavior in Fig. 5(d)). The MIP-DRL algorithms generally avoid charging all ESSs to the maximum SOC when the electricity price is low.

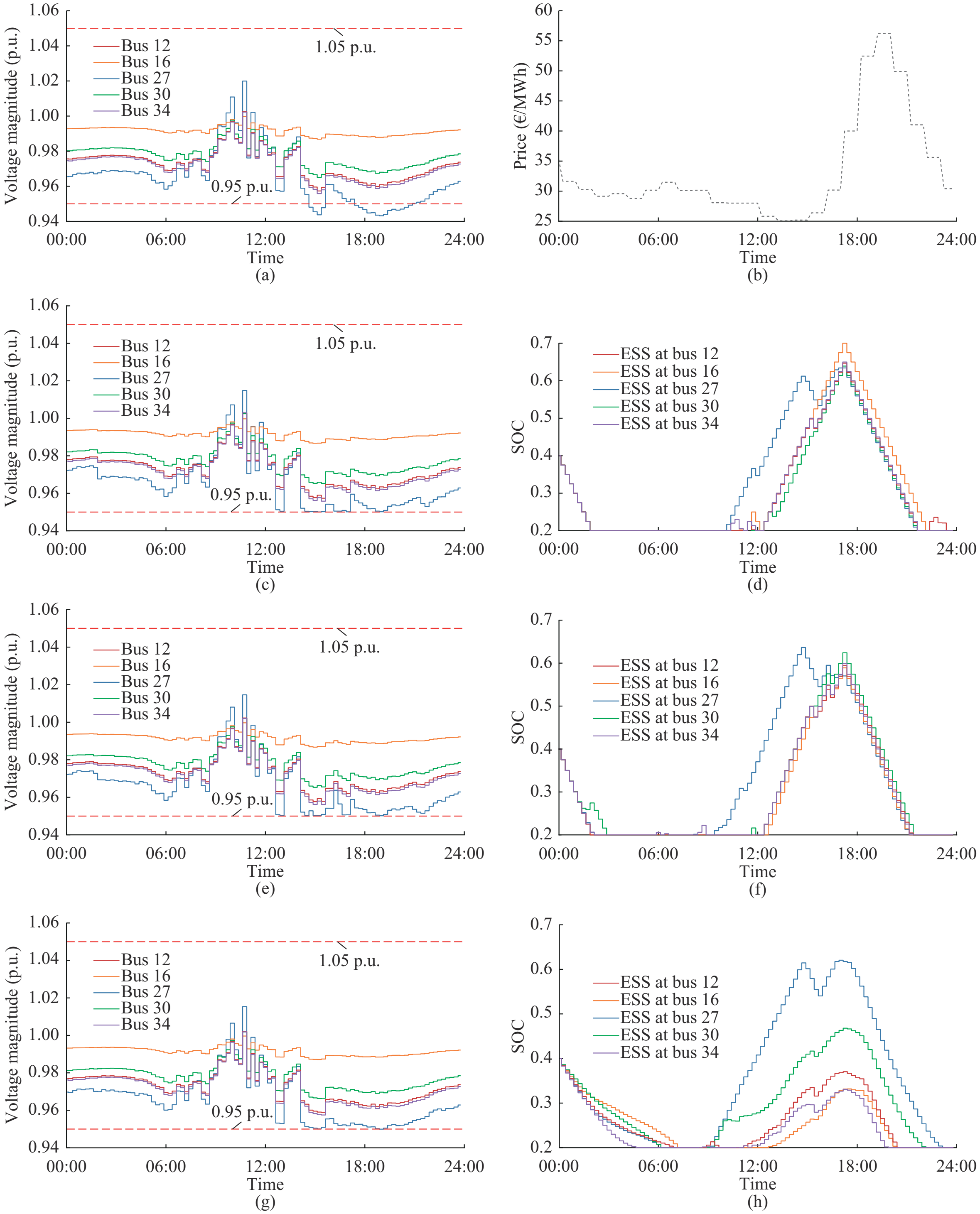

Fig. 5. Voltage magnitude of nodes to which ESSs are connected, SOC of ESSs, and day-ahead electricity price. (a) Voltage magnitude of nodes (without operation of ESSs). (b) Day-ahead electricity price. (c) Voltage magnitude of nodes (MIP-DDPG). (d) SOC of ESSs (MIP-DDPG). (e) Voltage magnitude of nodes (MIP-TD3). (f) SOC of ESSs (MIP-TD3). (g) Voltage magnitude of nodes (MIP-SAC). (h) SOC of ESSs (MIP-SAC).

This can be considered a sub-optimal decision. In this case, the operational cost resulting from the dispatch decisions provided by MIP-DDPG, MIP-TD3, and MIP-SAC are 9.5%, 12.9%, and 18.4% higher, respectively, than the optimal solution provided by the NLP formulation. The difference in this dispatch decision can be due to the estimated Q-function, which might not be good enough to represent the true Q-function. As the MIP-DRL algorithms choose actions that maximize the Q-value estimation, the largest Q-value might not represent the best action for this specific state-action pair. Nevertheless, even in executing a sub-optimal decision, the MIP-DRL algorithms enforce all voltage magnitude constraints, guaranteeing the operational feasibility. On the other hand, the safe DRL algorithm, i.e., safe DDPG, fails to enforce voltage magnitude constraints strictly, as the safe layer cannot track the dynamics of complex environments.

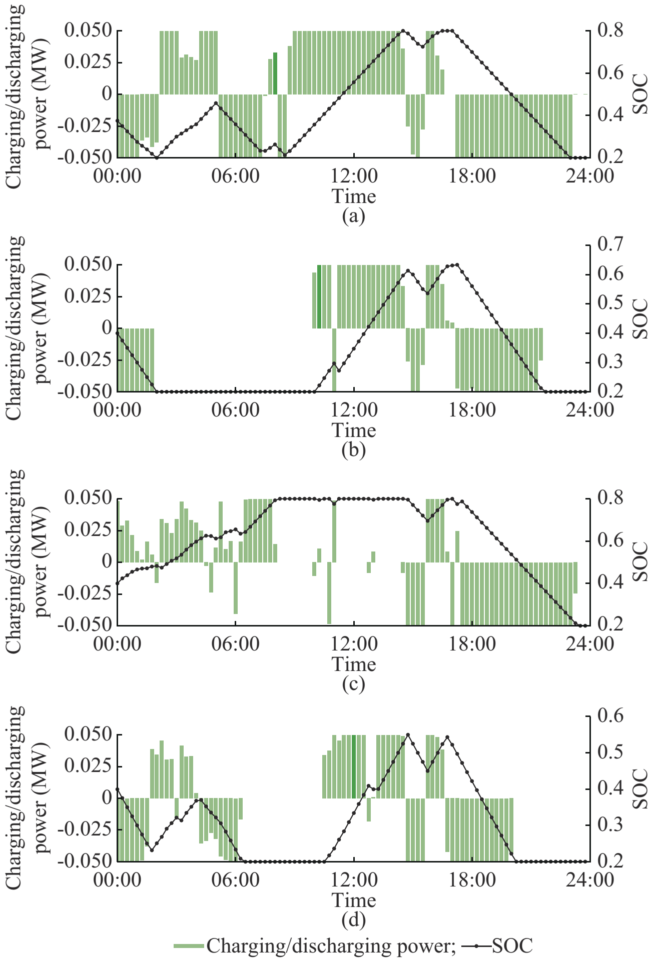

Fig. 6. Charging/discharging decisions and SOC changes of ESS connected to node 27 provided by different algorithms. (a) NLP formulation. (b) MIP-DDPG. (c) DDPG. (d) Safe DDPG.

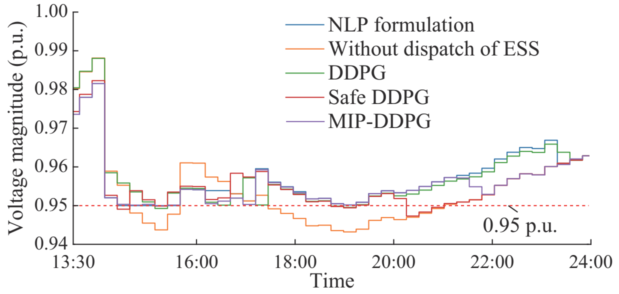

Fig. 7. Voltage magnitude of node 27 to which an ESS is connected.

E. Error Assessment and Computational Performance

Table III presents the average error (with respect to optimal solution provided by solving the NLP formulation) of the operational cost, the average number of voltage magnitude violations, and the total average computational time of the MIP-DRL algorithms as well as their benchmark DRL algorithms over 30 test days. As can be seen in Table III, the MIP-TD3, MIP-DDPG, and MIP-SAC can strictly enforce the voltage constraints. Among all these MIP-DRL algorithms, MIP-DDPG has the lowest average error, i.e., 10.4%. In contrast, their standard counterparts, such as DDPG, show poor performance, with an error of 34.3%, and the voltage magnitude constraint violations in around 45 time steps. As expected, the computational time required to execute the MIP-DRL algorithms is higher than standard DRL algorithms. This increase in the computational time is due to the MIP formulation to be solved to enforce all the operational constraints (see (22)). Nevertheless, in this case, the MIP-DRL algorithms can still be used for real-time operation as they only require less than 60 s for one-day (96 time steps) execution.

TABLE III
PERFORMANCE COMPARISON OF DIFFERENT DRL ALGORITHMS

Algorithm	Error of operational cost (%)	Number of voltage magnitude violations	Computational time (s)
MIP-TD3	13.2±0.5	0	57±6.7
MIP-DDPG	10.4±0.7	0	43±5.1
MIP-SAC	19.3±1.5	0	57±6.3
TD3	28.5±0.4	33±2	16±0.1
DDPG	34.3±0.7	45±11	16±0.1
SAC	32.2±0.5	44±17	16±0.1
Safe-DDPG	39.7±0.8	41±1	37±0.1

F. Scalability Analysis

Table IV presents the performance of MIP-DDPG in distribution networks with different sizes. Table IV includes the training time, computational time, number of voltage magnitude violations, and error of operational cost for networks with 34, 69, and 123 nodes. The training time increases with the size of distribution network, as expected. This increase is primarily due to the time required to solve the power flow equations during the training process. As the size of distribution network grows, the complexity of solving these equations increases, leading to longer training time. The error of operational cost remains consistent across different sizes, ranging from 10.1% to 11.3%. This suggests that the size of distribution network does not significantly impact the performance of MIP-DDPG. Moreover, the MIP-DDPG successfully enforces voltage constraints in all the tested networks, as evidenced by the absence of voltage magnitude violations. Finally, the computational time does not increase significantly with the size of distribution network. This is because the computational time is primarily influenced by the size of *Q*-network used in the MIP formulation. Once the *Q*-network is trained, the execution phase involves solving the MIP, which only depends on the complexity of *Q*-network.

TABLE IV
PERFORMANCE OF MIP-DDPG IN DISTRIBUTION NETWORKS WITH DIFFERENT SIZES

Node number	Training time (hour)	Computational time (s)	Number of voltage magnitude violations	Error of operational cost (%)
34	4.0	43±5.1	0	10.4±0.7
69	4.7	49±6.9	0	10.1±0.9
123	6.5	53±3.4	0	11.3±0.7

VI. Discussion

We have successfully combined deep learning and optimization theory to bring constraint enforcement to DRL algorithms. By using the trained *Q*-network as the surrogate function of the optimal operational decisions, we have guaranteed the optimality of the action from the *Q*-network through the MIP formulation. Moreover, by integrating the voltage constraints into the MIP formulation, the feasibility of the action is enforced. However, the performance of MIP-DRL algorithms is determined by the approximation quality of the *Q*-network obtained after the training process. During this training process, the *Q*-iteration faces the exploration v.s. exploitation dilemma, which can impact the approximation quality. For instance, the MIP-DDPG outperforms the MIP-TD3, while the MIP-SAC performs poorly. This discrepancy may be caused by the divergence between the exploration policies, leading to different exploration efficiencies and *Q*-network update rules. The conservative performance of the MIP-SAC might be caused by the soft *Q*-network update rule, which introduces more assumptions, impacting the estimation for accurate approximation.

Formulating a trained *Q*-network as an MIP problem introduces extra computational time due to the maximization of the *Q*-value function. In this case, such an MIP formulation is considered to be a nondeterministic polynomial (NP) complete problem. The worst-case computational time grows exponentially with the number of integer variables, which is proportional to the total number of ReLU activation functions used. However, the computational time can be greatly reduced by various techniques like improved branch-and-bound, and customized ReLU function algorithms [46]. Previous research shows that only 0.8 s are needed for solving an MIP problem formulated by a network with 300 ReLU units [49]. In our experiments, the MIP-DRL algorithms required less than 60 s for execution, supporting the applicability of MIP-DRL algorithms in real systems. In summary, the MIP-DRL algorithms can provide good quality dispatch decisions while strictly enforcing all voltage magnitude constraints, leading to high-quality feasible decisions. Compared to standard DRL algorithms, this superiority is achieved by directly transforming the *Q*-network (after training) as an MIP formulation, defining the optimal solution instead of leveraging an approximated policy. The operational constraints are added on top of the obtained MIP formulation, guaranteeing feasibility.

VII. Conclusion

This paper proposes an MIP-DRL framework to define high-quality dispatch decisions (in terms of the total operational cost) for ESSs in a distribution network, while ensuring their technical feasibility (related to enforcing voltage magnitude constraints). The proposed MIP-DRL framework consists of a *Q*-iteration and deployment procedure. During the *Q*-iteration procedure, a DNN is trained to represent the accurate state-action value function. Then, during the deployment procedure, this *Q*-function DNN is transformed into an MIP formulation that can be solved by commercial solvers. Results show that the dispatch decisions defined by MIP-DRL algorithms can ensure zero voltage magnitude violations while standard DRL algorithms fail to meet such constraints in uncertain scenarios. Additionally, the MIP-DRL algorithms show less errors compared with the optimal solution obtained with a perfect forecast of the stochastic variables.

Reference

[1] Y. Li, Y. Gu, G. He *et al*., "Optimal dispatch of battery energy storage in distribution network considering electrothermal-aging coupling," *IEEE Transactions on Smart Grid*, vol. 14, no. 5, pp. 3744-3758, Sept. 2023.

[2] A. Marot, A. Kelly, M. Naglic *et al*., "Perspectives on future power system control centers for energy transition," *Journal of Modern Power Systems and Clean Energy*, vol. 10, no. 2, pp. 328-344, Mar. 2022.

[3] C. Li, K. Zheng, H. Guo *et al*., "Intra-day optimal power flow considering flexible workload scheduling of IDCs," *Energy Reports*, vol. 9, pp. 1149-1159, Sept. 2023.

[4] P. P. Vergara, J. C. López, M. J. Rider *et al*., "Optimal operation of unbalanced three-phase islanded droop-based microgrids," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 928-940, Jan. 2019.

[5] 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.

[6] Z. Yin, S. Wang, and Q. Zhao, "Sequential reconfiguration of unbalanced distribution network with soft open points based on deep reinforcement learning," *Journal of Modern Power Systems and Clean Energy*, vol. 11, no. 1, pp. 107-119, Jan. 2023.

[7] C. Huang, H. Zhang, L. Wang *et al*., "Mixed deep reinforcement learning considering discrete-continuous hybrid action space for smart home energy management," *Journal of Modern Power Systems and Clean Energy*, vol. 10, no. 3, pp. 743-754, May 2022.

[8] J. Degrave, F. Felici, J. Buchli *et al*., "Magnetic control of tokamak plasmas through deep reinforcement learning," *Nature*, vol. 602, no. 7897, pp. 414-419, 2022.

[9] Y. Du, F. Li, K. Kurte *et al*., "Demonstration of intelligent HVAC load management with deep reinforcement learning: real-world experience of machine learning in demand control," *IEEE Power and Energy Magazine*, vol. 20, no. 3, pp. 42-53, May 2022.

[10] A. Ray, J. Achiam, and D. Amodei. (2019, Oct.). Benchmarking safe exploration in deep reinforcement learning. [Online]. Available: https://arxiv.org/abs/1910.01708

[11] H. Ding, Y. Xu, B. Chew *et al*., "A safe reinforcement learning approach for multi-energy management of smart home," *Electric Power Systems Research*, vol. 210, p. 108120, Sept. 2022.

[12] E. M. S. Duque, J. S. Giraldo, P. P. Vergara *et al*., "Community energy storage operation via reinforcement learning with eligibility traces," *Electric Power Systems Research*, vol. 212, p. 108515, Nov. 2022.

[13] P. P. Vergara, M. Salazar, J. S. Giraldo *et al*., "Optimal dispatch of PV inverters in unbalanced distribution systems using reinforcement learning," *International Journal of Electrical Power & Energy Systems*, vol. 136, p. 107628, Mar. 2022.

[14] S. Hou, E. M. Salazar, P. P. Vergara *et al*., "Performance comparison of deep RL algorithms for energy systems optimal scheduling," in *Proceedings of 2022 IEEE PES Innovative Smart Grid Technologies Conference Europe* (*ISGT-Europe*), Novi Sad, Serbia, Oct. 2022, pp. 1-6.

[15] X. Yang, H. He, Z. Wei *et al*., "Enabling safety-enhanced fast charging of electric vehicles via soft actor critic-Lagrange DRL algorithm in a cyber-physical system," *Applied Energy*, vol. 329, p. 120272, Jan. 2023.

[16] H. Cui, Y. Ye, J. Hu *et al*., "Online preventive control for transmission overload relief using safe reinforcement learning with enhanced spatial-temporal awareness," *IEEE Transactions on Power Systems*, vol. 39, no. 1, pp. 517-532, Jan. 2024.

[17] J. Achiam, D. Held, A. Tamar *et al*., "Constrained policy optimization," in *Proceedings of International Conference on Machine Learning*, Sydney, Australia, Aug. 2017, pp. 22-31.

[18] H. Li and H. He, "Learning to operate distribution networks with safe deep reinforcement learning," *IEEE Transactions on Smart Grid*, vol. 13, no. 3, pp. 1860-1872, May 2022.

[19] H. Li, Z. Wan, and H. He, "Constrained EV charging scheduling

based on safe deep reinforcement learning," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2427-2439, May 2020.

[20] G. Dalal, K. Dvijotham, M. Vecerik *et al*. (2018, Jan.). Safe exploration in continuous action spaces. [Online]. Available: https://arxiv.org/abs/1801.08757

[21] G. Ceusters, M. A. Putratama, R. Franke *et al*., "An adaptive safety layer with hard constraints for safe reinforcement learning in multi-energy management systems," *Sustainable Energy*, *Grids and Networks*, vol. 36, p. 101202, Dec. 2023.

[22] M. Eichelbeck, H. Markgraf, and M. Althoff, "Contingency-constrained economic dispatch with safe reinforcement learning," in *Proceedings of 2022 21st IEEE International Conference on Machine Learning and Applications*, Nassau, Bahamas, Dec. 2022, pp. 597-602.

[23] P. Kou, D. Liang, C. Wang *et al*., "Safe deep reinforcement learning-based constrained optimal control scheme for active distribution networks," *Applied Energy*, vol. 264, p. 114772, Apr. 2020.

[24] S. Gros, M. Zanon, and A. Bemporad, "Safe reinforcement learning via projection on a safe set: how to achieve optimality?" *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 8076-8081, Apr. 2020.

[25] S. Hou, P. P. Vergara, E. M. S. Duque *et al*., "Optimal energy system scheduling using a constraint-aware reinforcement learning algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 152, p. 109230, Oct. 2023.

[26] 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.

[27] J. Wang, W. Xu, Y. Gu *et al*., "Multi-agent reinforcement learning for active voltage control on power distribution networks," *Advances in Neural Information Processing Systems*, vol. 34, pp. 3271-3284, Dec. 2021.

[28] Y. Zhou, B. Zhang, C. Xu *et al*., "A data-driven method for fast AC optimal power flow solutions via deep reinforcement learning," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 6, pp. 1128-1139, Nov. 2020.

[29] L. Liu, J. Zhu, J. Chen *et al*., "Deep reinforcement learning for stochastic dynamic microgrid energy management," in *Proceedings of 2021 IEEE 4th International Electrical and Energy Conference*, Wuhan, China, May 2021, pp. 1-6.

[30] Y. Ji, J. Wang, J. Xu *et al*., "Data-driven online energy scheduling of a microgrid based on deep reinforcement learning," *Energies*, vol. 14, no. 8, p. 2120, Apr. 2021.

[31] S. Zhang, R. Jia, H. Pan *et al*., "A safe reinforcement learning-based charging strategy for electric vehicles in residential microgrid," *Applied Energy*, vol. 348, p. 121490, Oct. 2023.

[32] Y. Ye, H. Wang, P. Chen *et al*., "Safe deep reinforcement learning for microgrid energy management in distribution networks with leveraged spatial-temporal perception," *IEEE Transactions on Smart Grid*, vol. 14, no. 5, pp. 3759-3775, Sept. 2023.

[33] P. Yu, H. Zhang, and Y. Song, "District cooling system control for providing regulation services based on safe reinforcement learning with barrier functions," *Applied Energy*, vol. 347, p. 121396, Oct. 2023.

[34] M. M. Hosseini and M. Parvania, "On the feasibility guarantees of deep reinforcement learning solutions for distribution system operation," *IEEE Transactions on Smart Grid*, vol. 14, no. 2, pp. 954-964, Mar. 2023.

[35] Y. Shi, G. Qu, S. Low *et al*., "Stability constrained reinforcement learning for real-time voltage control," in *Proceedings of 2022 American Control Conference*, Atlanta, USA, Jun. 2022, pp. 2715-2721.

[36] D. Qiu, Z. Dong, X. Zhang *et al*., "Safe reinforcement learning for real-time automatic control in a smart energy-hub," *Applied Energy*, vol. 309, p. 118403, Mar. 2022.

[37] H. Park, D. Min, J. H. Ryu *et al*., "DIP-QL: a novel reinforcement learning method for constrained industrial systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 11, pp. 7494-7503, Nov. 2022.

[38] L. H. Macedo, J. F. Franco, M. J. Rider *et al*., "Optimal operation of distribution networks considering energy storage devices," *IEEE Transactions on Smart Grid*, vol. 6, no. 6, pp. 2825-2836, Nov. 2015.

[39] T. P. Lillicrap, J. J. Hunt, A. Pritzel *et al*. (2015, Sept.). Continuous control with deep reinforcement learning. [Online]. Available: https://arxiv.org/abs/1509.02971

[40] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *Proceedings of International Conference on Machine Learning*, Stockholm, Sweden, Jul. 2018, pp. 1587-1596.

[41] T. Haarnoja, A. Zhou, P. Abbeel *et al*. (2018, Jan.). Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor. [Online]. Available: https://arxiv.org/abs/1801.01290

[42] S. Lim, A. Joseph, L. Le *et al*. (2018, Oct.). Actor-expert: a framework for using *Q*-learning in continuous action spaces. [Online]. Available: https://arxiv.org/abs/1810.09103

[43] M. Fischetti and J. Jo, "Deep neural networks and mixed integer linear optimization," *Constraints*, vol. 23, no. 3, pp. 296-309, Jul. 2018.

[44] G. F. Montufar, R. Pascanu, K. Cho *et al*. (2014, Dec.). On the number of linear regions of deep neural networks. [Online]. Available: https://proceedings. neurips. cc/paperfiles/paper/2014/file/ 109d2dd3608f669ca17920c511c2a41e-Paper.pdf

[45] F. Ceccon, J. Jalving, J. Haddad *et al*. (2022, Feb.). OMLT: optimization & machine learning toolkit. [Online]. Available: https://arxiv.org/abs/2202.02414

[46] Gurobi Optimization, LLC. (2022, Jun.). What's new – Gurobi 10.0. [Online]. Available: https://www.gurobi.com/whats-new-gurobi-10-0/

[47] S. Hou. (2022, Dec.). Energy management MIP deep reinforcement learning. [Online]. Available: https://github. com/ShengrenHou/Energy-management-MIP-Deep-Reinforcement-Learning

[48] P. Vergara. (2022, Dec.). MIP-DRL-framework. [Online]. Available: https://github.com/distributionnetworksTUDelft/MIP-DRL-Framework

[49] T. Wei and C. Liu, "Safe control with neural network dynamic models," in *Proceedings of Learning for Dynamics and Control Conference*, Hawaii, USA, Jul. 2022, pp. 739-750.

Shengren Hou received the B.S. degree in electric engineering from Northeast Electric Power University, Jilin, China, in 2018, and the M.S. degree in electric engineering from Guangxi University, Guangxi, China, in 2021. He is currently pursuing the Ph.D. degree at the Delft University of Technology, Delft, The Netherlands. His main research interests include active distribution network optimization control, short-term electricity market arbitrage, and reinforcement learning.

Edgar Mauricio Salazar Duque received the B.E. degree in electrical and electronic engineering from the Universidad de Los Andes, Bogotá, Colombia, in 2008, the M.Sc. degree (cum laude) in smart electrical grids and systems from the Kungliga Tekniska Högskolan (KTH), Stockholm, Sweden, and the Technical University of Eindhoven, Eindhoven, The Netherlands, in 2018. He is currently working towards a Ph.D. degree in the electrical energy systems group at the Technical University of Eindhoven. His main research interests include data analysis, and applications of machine learning techniques on power distribution grid for planning and operation.

Peter Palensky received the M.Sc., Ph.D., and Habilitation degrees from Vienna University of Technology, Vienna, Austria, in 1997, 2001, and 2015, respectively. He is currently a Full Professor of intelligent electric power grids and the Head of the Electrical Sustainable Energy Department, TU Delft, Delft, The Netherlands. His main research interests include energy automation network, smart grid, and modeling of intelligent energy system.

Qixin Chen received the Ph. D. degree from the Department of Electrical Engineering, Tsinghua University, Beijing, China, in 2010, where he is currently a Tenured Professor. His main research interests include electricity market, power system economics and optimization, low-carbon electricity, and data analytics in power system.

Pedro P. Vergara received the B.Sc. degree (with honors) in electronic engineering from the Universidad Industrial de Santander, Bucaramanga, Colombia, in 2012, the M.Sc. degree in electrical engineering from the University of Campinas, UNICAMP, Campinas, Brazil, in 2015, and the Ph.D. degree from the University of Campinas, UNICAMP, and the University of Southern Denmark, SDU, Denmark, funded by the Sao Paulo Research Foundation (FAPESP), in 2019. In 2019, he joined the Eindhoven University of Technology, TU/e, Eindhoven, The Netherlands, as a Postdoctoral Researcher. In 2020, he was appointed as Assistant Professor at the Intelligent Electrical Power Grids (IEPG) Group at Delft University of Technology, Delft, The Netherlands. He received the Best Presentation Award at the Summer Optimization School in 2018 organized by the Technical University of Denmark (DTU) and the Best Paper Award at the 3rd IEEE International Conference on Smart Energy Systems and Technologies (SEST), in Turkey, in 2020. His main research interests include development of algorithms for control, planning, and operation of electrical distribution system with high penetration of low-carbon energy resources (e. g., electric vehicle, PV system, electric heat pump) using optimization and machine learning approaches.