CONTRIBUTED PAPER

Deep Reinforcement Learning for Smart Grid Operations: Algorithms, Applications, and Prospects

This article provides a detailed and well-organized overview of deep reinforcement learning (DRL) methodologies, which encompasses fundamental concepts and theoretical DRL principles, as well as the most sophisticated DRL techniques applied to power system operations.

By Yuanzheng Li, *Senior Member IEEE*, Chaofan Yu, Mohammad Shahidehpour, *Life Fellow IEEE*, Tao Yang, *Senior Member IEEE*, Zhigang Zeng, *Fellow IEEE*, and Tianyou Chai, *Life Fellow IEEE*

ABSTRACT | With the increasing penetration of renewable energy and flexible loads in smart grids, a more complicated power system with high uncertainty is gradually formed, which brings about great challenges to smart grid operations. Traditional optimization methods usually require accurate mathematical models and parameters and cannot deal well with the growing complexity and uncertainty. Fortunately, the widespread popularity of advanced meters makes it possible for smart grid to collect massive data, which offers opportunities for data-driven artificial intelligence methods to address the optimal operation and control issues. Therein, deep reinforcement learning (DRL) has attracted extensive attention for its excellent performance in operation problems with high uncertainty. To this end, this article presents a comprehensive literature survey on DRL and its applications in smart grid operations. First, a detailed overview of DRL, from fundamental concepts to advanced models, is conducted in this article. Afterward, we review various DRL techniques as well as their extensions developed to cope with emerging issues in the smart grid, including optimal dispatch, operational control, electricity market, and other emerging areas. In addition, an application-oriented survey of DRL in smart grid is presented to identify difficulties for future research. Finally, essential challenges, potential solutions, and future research directions concerning the DRL applications in smart grid are also discussed.

KEYWORDS | Deep reinforcement learning (DRL); electricity market; operational control; optimal dispatch; smart grid (SG).

Manuscript received 15 July 2022; revised 15 June 2023; accepted 1 August 2023. Date of publication 5 September 2023; date of current version 15 September 2023. This work was supported in part by the National Key R&D Program of China under Grant 2021ZD0201300, in part by the National Natural Science Foundation of China under Grant 62073148, in part by the Key Project of National Natural Science Foundation of China under Grant 62233006, in part by the Smart Grid Joint Key Project of National Natural Science Foundation of China and the State Grid Corporation of China under Grant U2066202, in part by the Major Program of National Natural Science Foundation of China under Grant 61991400, and in part by the 2020 Science and Technology Major Project of Liaoning Province under Grant 2020JH1/10100008. *(Corresponding author: Zhigang Zeng.)*

Yuanzheng Li and **Zhigang Zeng** are with the School of Artificial Intelligence and Automation, Autonomous Intelligent Unmanned System Engineering Research Center, Key Laboratory of Image Processing and Intelligence Control, Ministry of Education of China, and the Hubei Key Laboratory of Brain-Inspired Intelligent Systems and the Belt and Road Joint Laboratory on Measurement and Control Technology, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: Yuanzheng_Li@hust.edu.cn; zgzeng@hust.edu.cn).

Chaofan Yu is with the China-EU Institute for Clean and Renewable Energy, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: ycffish@hust.edu.cn).

Mohammad Shahidehpour is with the Robert W. Galvin Center for Electricity Innovation, Illinois Institute of Technology, Chicago, IL 60616 USA (e-mail: ms@iit.edu).

Tao Yang and **Tianyou Chai** are with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, China (e-mail: yangtao@mail.neu.edu.cn; tychai@mail.neu.edu.cn).

Digital Object Identifier 10.1109/JPROC.2023.3303358

NOMENCLATURE

Notations

$\mathcal{A}, a$	Set of actions and action.
$\mathcal{S}, s$	Set of all states and state.
P	Transition probability.
$\mathcal{R}$	Set of all possible rewards.

Ω	Set of observations.
$\mathcal{O}$	Observation probabilities.
$\mathcal{L}$	Loss error between prediction and target.
R_t	Cumulative reward at time step t.
π, π^*	Policy (decision-making rule) and optimal policy.
$\pi(a\|s)$	Probability of taking action a in state s under stochastic policy π.
$\mu(s)$	Action taken in state s under deterministic policy μ.
$o, b(s)$	Current observation and its belief about state s.
$p(s'\|s,a)$	Probability of transitioning to state s', from state s taking action a.
$r(s,a)$	Expected immediate reward from state s after action a.
$\mathcal{V}^\pi(s)$	Value of state s under policy π (expected return).
$\mathcal{V}^*(s)$	Value of state s under the optimal policy.
$\mathcal{Q}^\pi(s,a)$	Value of taking action a in state s under policy π.
$\mathcal{Q}^*(s,a)$	Value of taking action a in state s under the optimal policy.
$\mathscr{V}(s)$	State-value function.
$\mathscr{A}(a)$	State-dependent action advantage function.
δ_t	Temporal difference error at time step t.
$\boldsymbol{\theta}$	Parameter vector of target policy.
$\boldsymbol{\psi}$	Parameter vector of the critic.
$\pi_{\boldsymbol{\theta}}$	Policy corresponding to parameter $\boldsymbol{\theta}$.
$J(\boldsymbol{\theta})$	Performance measure for the policy $\pi_{\boldsymbol{\theta}}$.
ϵ	Probability of taking a random action in an ϵ-greedy policy.
α, β	Step-size parameters.
γ	Discount-rate parameter.
λ	Lagrange multiplier.
t	Discrete time step.
τ	Trajectory of state–action pairs.

Abbreviations

AC	Actor–critic.
A2C	Advantage actor–critic.
A3C	Asynchronous advantage actor–critic.
AGC	Automatic generation control.
AI	Artificial intelligence.
ANN	Artificial neural network.
ARS	Augmented random search.
AVC	Autonomous voltage control.
CHP	Combined heat and power.
CNN	Convolutional neural network.
CPO	Constrained policy optimization.
DDPG	Deep deterministic policy gradient.
DDQN	Double deep Q-network.
DER	Distributed energy resource.
DFF	Deep feedforward.
DFRL	Deep forest reinforcement learning.
DG	Diesel generator.
DIRL	Deep inverse reinforcement learning.
DL	Deep learning.
DN	Distribution network.
DNN	Deep neural network.
DNR	Distribution network reconfiguration.
DQN	Deep Q-network.
DRL	Deep reinforcement learning.
ESS	Energy storage system.
EV	Electric vehicle.
FDI	False data injection.
IES	Integrated energy system.
LFC	Load frequency control.
LSTM	Long short-term memory.
MAAC	Multi-actor-attention-critic.
MCDRL	Monte Carlo DRL.
MDP	Markov decision process.
P2P	Peer-to-peer.
PARS	Parallel ARS.
PPO	Proximal policy optimization.
RL	Reinforcement learning.
RNN	Recurrent neural network.
SAC	Soft actor–critic.
SARSA	State-action-reward-state-action.
SG	Smart grid.
SSA	Security situational awareness.
TD	Temporal difference.
TD3	Twin delay DDPG.
TDAC	Three-network double-delay actor–critic.
TRPO	Trust region policy optimization.

I. INTRODUCTION

Energy signifies an important basis for the development and the survival and human societies, where social developments are accompanied by a continuous increase in energy demand. However, the massive use of fossil energy in developed societies brings about a series of global problems pertaining to environmental pollution, ecological destruction, and global warming. In order to mitigate the expanding environment concerns, attractive SG technologies have been developed in various parts of the world [1], [2], [3], [4], [5]. The essence of using SGs along with conventional power systems is itemized as follows.

1) The power generation profile is shifting from controllable and continuous supply of coal-fired power generation to renewable energy (RE) with high uncertainty and weak controllability.
2) The load characteristic is shifting from the traditional rigid and purely captive to flexible and active type that combines both production and consumption of energy.
3) The power grid is transitioning from a traditional one-way flow to an energy Internet with a two-way flow that includes large-scale hybrid ac/dc subsystems, microgrids, and adjustable loads.
4) The power grid foundation is shifting from a traditional mechanical–electromagnetic system of synchronous generators with high inertia to a hybrid system dominated by low-inertia power electronic devices.

These characteristics increase the uncertainty and complexity, which brings great challenges to the secure and economic operations of SG [6]. To solve these problems, various approaches have been proposed for the optimal operations of SG. However, conventional optimization methods require accurate mathematical models and parameters, which makes it difficult to apply them to increasing complex and distributed systems with multiple uncertain subsystems. Consequently, the applications of traditional methods are limited in practice, which calls for a more intelligent and efficient solution.

With the wide application of advanced sensors, smart meters, and monitoring systems, SG is producing massive data with mutual correlations [7], [8], [9], which also offers the data basis for data-driven AI method, for instance, RL. Indeed, RL is one of the most important research topics of AI over the last two decades, due to its excellent ability of self-directed learning, adaptive adjustment, and optimal decision. Specifically, RL is a learning process that allows the agent to periodically make decisions, observe the results, and then automatically adjust its action to achieve the optimal policy. For instance, as one of the pioneering works in the application of RL to renewable power system operations, Liao et al. [10] proposed a multiobjective optimization algorithm based on learning automata for economic emission dispatching and voltage stability enhancement in SGs. Simulation results have demonstrated that the proposed method achieves accurate solutions and adapts effectively to dynamic fluctuations in wind power and load demand.

Despite all these advantages, RL is still unsuitable and inapplicable to complicated large-scale problem environments as it has to explore and gain knowledge from the entire system, which takes much time to obtain the best policy. In this situation, the applicability of the RL has encountered serious challenges in the real world. Recently, the rapid development of DL has aroused great interests in industry and academia [11], [12]. The deep RL architectures result in a better data processing and representation learning capability, which provide a potential solution to overcome the RL limitations, that is, the combination of RL and DL has led to a breakthrough technique, named DRL, which integrates the decision-making capacity of RL and the DL perception capability [13]. More precisely, DRL improves the learning speed and performance of conventional RL, in virtue of the advantages of DNNs in the training process. Therefore, DRL has been introduced in various applications and achieved phenomenal success, such as games, robotics, natural language processing, computer vision, and SG operations [14].

In the field of SG, DRL has been intensively adopted to undertake various tasks, which stem from developing the optimal policy. As mentioned above, SG is one of the largest artificial systems, which is well known for their highly uncertain and nonlinear operating characteristics. Although several approaches have been developed for SG, they still suffer from great computational complexity and strong randomness. To this end, DRL has been regarded as an alternative solution to overcome these challenges. General speaking, the DRL methods provide the following advantages.

1) DRL can achieve the optimal solution of sophisticated grid optimization without using complete and accurate network information.
2) DRL allows grid entities to learn and build knowledge about the environment on the basis of historical data.
3) DRL offers autonomous decision-making within minimum information exchange, which not only reduces computational burden but also improves the SG security and robustness.
4) DRL significantly enhances the learning capability in comparison to the traditional RL, especially in problems with numerous states and action spaces.

Although there exist some RL reviews, detailed discussions on DRL applications in SG are still lacking. Specifically, existing surveys have focused on the DRL applications to the Internet of Things, natural language processing, and computer vision [15], [16], [17], [18]. Indeed, there are some excellent reviews on RL applications to energy systems [19], [20], [21], [22], [23], [24], [25], [26]. However, they mainly concentrate on conventional RL methods or power system, rather than presenting state-of-the-art DRL approaches to SG applications. A detailed comparison between our work and related surveys is presented in Table 1 to identify the unique aspects and novel perspectives that distinguish our work. It could be observed that there exist some reviews on DRL-based decision-making in conventional power system and modern SG. For instance, Chen et al. [19] provided a comprehensive review of various DRL techniques and their potential applications in power systems, with a focus on three key applications: frequency regulation, voltage control, and energy management.

However, emerging energy solutions for improving the SG efficiency and ensuring its secure operations are not covered in [19]. Zhang et al. [20] and Glavic [21] covered multiple aspects of power system operations, which includes optimal dispatch, operational control, electricity market, and others. Although Zhang et al. [20] and Glavic [21] provided summaries of typical DRL algorithms such as DQN, DDPG, and AC, they do not cover recently developed state-of-the-art DRL methods. To this end, Cao et al. [22] provided a comprehensive summary of advanced DRL algorithms and their SG applications, including value-based, policy-based, and AC solution methods. Although summaries of DRL algorithms are detailed, Cao et al. [22] did not convey emerging SG areas, including P2P trading markets and privacy preservation issues.

Yang et al. [23], Perera and Kamalaruban [24], and Yu et al. [25] classified DRL papers in the literature into seven categories according to their application fields. Their study reveals that about half of the publications use Q-learning, whereas some of the state-of-the-art DRL methods are

Table 1 Comparison Between Our Work and Related Surveys

Literature	Main Focus	Scope											Methodology						
		Optimal dispatch			Operational control			Electricity Market			Emerging areas		Value-based	Policy-based		Actor-critic			
		Distribution	Microgrid	IES	AGC	AVC	LFC	Bidding	Pricing	P2P	Security	Privacy	DQN	TRPO	PPO	A2C/A3C	SAC	DDPG	TD3
X. Chen *et al.* [19]	Power system	✓	✓	✓	✓	✓	✓	✓					✓	✓	✓	✓	✓	✓	✓
Z. Zhang *et al.* [20]	Power system	✓	✓		✓	✓	✓	✓	✓		✓		✓			✓		✓	
M. Glavic *et al.* [21]	Power system	✓	✓		✓		✓				✓		✓			✓		✓	
D. Cao *et al.* [22]	Power system	✓	✓	✓	✓	✓		✓	✓				✓	✓	✓	✓	✓	✓	✓
T. Yang [23]	Energy system		✓	✓	✓		✓	✓	✓		✓		✓	✓	✓	✓		✓	
A.T.D. Perera *et al.* [24]	Energy system	✓	✓			✓	✓	✓	✓				✓	✓	✓		✓	✓	✓
L. Yu *et al.* [25]	Energy system		✓	✓									✓	✓	✓	✓	✓	✓	
D. Zhang *et al.* [26]	Smart grid		✓		✓						✓		✓			✓		✓	
Our work	Smart grid	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

not utilized in power system applications. Although Zhang et al. [26] attempted to provide an in-depth analysis of DRL application algorithms in SGs, it lacks a summary of advanced algorithms applied to power systems. With the significant AI applications in power systems, the number of publications that use DRL has also grown rapidly, where many state-of-the-art DRL algorithms have been proposed for SG operations. This major development calls for a more comprehensive analysis of potential DRL applications to SGs. Accordingly, this article is dedicated to presenting a relatively holistic overview of various DRL-based methods applied to SG operations.

The main contributions of this article are listed as follows.

1) A detailed and well-organized overview of DRL methodologies is provided, which encompasses fundamental concepts and theoretical DRL principles, as well as the most sophisticated DRL techniques applied to power system operations.
2) The SG operation issues are divided into four critical categories to illustrate DRL applications to modeling, design, solution, and numerical experimentation.
3) An in-depth understanding of challenges, potential solutions, and future directions for deploying DRL to SG problems is discussed and the outlook for additional developments is presented.

Different from the excellent prior works about RL, this article attempts to conduct a relatively exhaustive review of DRL applications to SG operations, especially for the last few years. These reviews will encompass emerging topics such as optimal economic dispatch, distributed control, and electricity markets. First, with the increasing penetration of RE, the optimal dispatch of SG resources is confronted with unprecedented challenges, including massive operational uncertainty, lower system inertia and new dynamics phenomena, and highly nonlinear and complex power systems that cannot be effectively represented and constructed by existing mathematical tools [27]. Second, operational control is a critical SG task that involves device control and coordination, including generators, transformers, and capacitors. Traditional mathematical methods may be based on simplified models and linear control strategies, which may struggle to address the complexities of nonlinear and dynamic SG characteristics.

Third, electricity market operation is a complex optimization problem that involves multiple participants, variables, and uncertainties. Conventional mathematical methods may rely on simplified models and assumptions, which may not fully capture the complexity of the prevailing assumptions and may not fully comprehend the complexity of real-world scenarios. Finally, widespread deployments of Internet-connected SG devices have significantly increased the vulnerability of power systems to cyberattacks. Cyberattack dynamics and complexities necessitate the implementation of responsive, adaptive, and scalable protection mechanisms in SGs. These requirements are difficult to achieve by typical operation methods that would rely on static security measures [28]. More importantly, through summarizing, highlighting, and analyzing the DRL characteristics and their SG applications, this survey article would highlight specific potential research directions for interested parties. The rest part of this article is organized as follows. Section II introduces the evolution of DRL and discusses its state-of-the-art techniques as well as their extensions. In Section III, the detailed DRL applications in SG are presented. After that, Section IV discusses the prospects and challenges of DRL in SG operations in the future. Finally, the conclusion of this article is drawn in Section V.

II. DRL: AN OVERVIEW

In this section, the fundamental knowledge of MDP, RL, and DL techniques, which are crucial components of DRL, is introduced first. Then, the combination of DL and RL is presented, which results in the formation of DRL. Finally, advanced DRL models as well as their state-of-the-art extensions are reviewed.

A. Markov Decision Process

In mathematics, MDP is a discrete-time stochastic control progress, which assumes that the future state is only related to the present state and is independent of the past states [29]. MDP provides a useful framework for modeling the decision-making problems in situations where the solutions are deemed to be partly random and uncontrollable. MDPs are popular for studying optimization problems solved by dynamic programming and RL approaches [30]. Generally, an MDP is defined as a tuple $(\mathcal{S}, \mathcal{A}, P, \mathcal{R})$, where $\mathcal{S}$ is the set of finite states named the state space, $\mathcal{A}$ represents the set of actions called action space, P is the transition probability from state s to state s' after action a is executed, and $\mathcal{R}$ denotes the immediate reward received after state transition from state s to state s', due to the performance of action a.

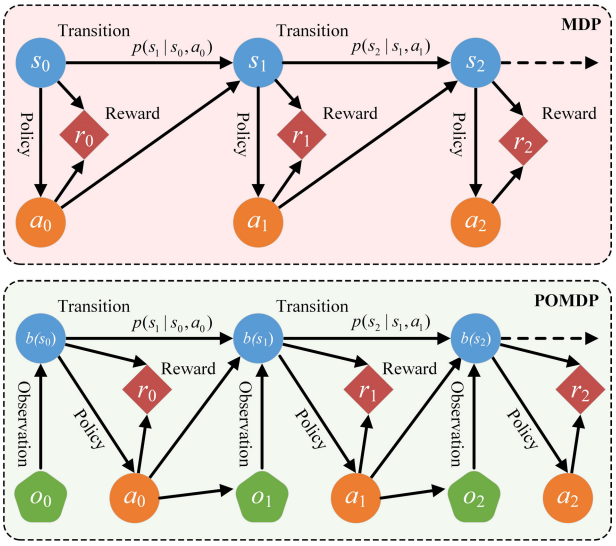


Fig. 1. *Illustration diagrams of completely observable and partially observable MDP.*

The reward function serves as a guide for DRL agent, shaping its behavior and influencing its learning process. While there is no one-size-fits-all approach to design the reward function, there are certain principles that can be followed to enhance DRL effectiveness. One important principle is to ensure that the reward function effectively reflects the control goals of the DRL problem. Rewards should be aligned with the desired behavior or outcome that the agent is expected to achieve. This requires careful consideration of specific tasks and the objectives that need to be optimized. A policy function π is defined as the mapping from state space $\mathcal{S}$ to the action space $\mathcal{A}$, which determines how the decision-maker selects actions. The illustration of MDP is shown in Fig. 1. In each epoch, the decision-maker chooses an action a_i based on its policy about the current state s_i, i.e., $\pi(s_i)$. Then, the current state s_i transfers to the next state s_{i+1} with a probability of $p(s_{i+1}|s_i, a_i)$ and obtains the immediate reward $r(s_i, a_i, s_{i+1})$.

1) Partially Observable Markov Decision Process: It is assumed that the system state is completely observable by the agent in conventional MDP. However, the agent can only observe a part of the system state in many cases, and thus, a partially observable Markov decision process (POMDP) is proposed to establish the decision-making model while considering the uncertainty introduced by the partial observation. Actually, POMDP is a mathematical framework for modeling the decision-making situations where the decision-maker only has partial information about the state of system. POMDP is an extension of MDP, which accounts for cases when some state data are missing or considered uncertain. In POMDP, the decision-maker receives an observation of the system's state, rather than the true state itself.

A typical POMDP is defined as a six-tuple $(\mathcal{S}, \mathcal{A}, P, \mathcal{R}, \Omega, \mathcal{O})$, where $(\mathcal{S}, \mathcal{A}, P, \mathcal{R})$ are denoted in the same way as in MDPs. Ω and $\mathcal{O}$ refer to the set of observations and their corresponding observation probabilities, respectively. The illustration of POMDP is presented in Fig. 1. At each time period, the agent chooses an action $a_t \in \mathcal{A}$ according to the popular belief on current state $s_t \in \mathcal{S}$, i.e., $b(s_t)$. Then, the current state s_t transfers to the next state s_{t+1} with a probability of $p(s_{t+1}|s_t, a_t)$. What distinguishes a POMDP from the completely observable MDP is that the agent now perceives an observation $o_{t+1} \in \Omega$ of the next state s_{t+1}, rather than the true state itself. The probability of observing which observation depends on the next state s_{t+1} as well as its action a_t in state s_t, which is drawn according to the observation function, i.e., $\mathcal{O}(o_{t+1}|s_t, a_t, s_{t+1})$. Finally, the agent receives an immediate reward $r_t(s_t, a_t) \in R$ and repeats the above process.

Based on current belief $b(s_t)$ and its observation o_{t+1}, the agent updates its belief about the next (unobserved) state s_{t+1}, i.e., $b_{s_{t+1}}$, which is stated as follows:

$$b(s_{t+1}) = \frac{\mathcal{O}(o_{t+1}|s_t, a_t, s_{t+1}) \sum_{s_t \in \mathcal{S}} p(s_{t+1}|s_t, a_t) b(s_t)}{\sum_{s_{t+1} \in \mathcal{S}} \mathcal{O}(o_{t+1}|s_t, a_t, s_{t+1}) \sum_{s_t \in \mathcal{S}} p(s_{t+1}|s_t, a_t) b(s_t)} \tag{1}$$

where $\mathcal{O}(o_{t+1}|s_t, a_t, s_{t+1})$ represents the probability that agent perceives observation o_{t+1} after action a_t is executed in state s_t, and then, it moves to the next state s_{t+1} with probability $p(s_{t+1}|s_t, a_t)$. Similar to a fully observed MDP, the goal of POMDP agent is also devoted to finding an optimal policy π^*, in order to maximize the cumulative discounted reward $\sum_{i=0}^{\infty} \gamma r_t(s_t, \pi^*(s_t))$.

2) Multiagent Markov Decision Process: A single-agent MDP may suffer from limited exploration, in which the agent fails to explore the entire state space and can get stuck in suboptimal solutions. To this end, multiagent Markov decision process (MMDP) is developed to deal with complex tasks that cannot be accomplished by a single agent. Specifically, MMDP generalizes the classical MDP modeling framework with the notion of multiple agents, each with its own state and action space, interacting in a shared environment to achieve a common goal. In general, a multiagent Markov decision process is defined by a five-tuple $(\mathcal{I}, \mathcal{S}, \{\mathcal{A}^i\}_{i \in \mathcal{I}}, P, R)$, where $\mathcal{I} \triangleq \{1, 2, \ldots, i, \ldots, I\}$ represents the finite set of agents and $\mathcal{S} \triangleq \{\mathcal{S}^1, \ldots, \mathcal{S}^i, \ldots, \mathcal{S}^I\}$ refers to the global state space of all agents with $\mathcal{S}^i$ denoting the state space of agent i. $\{\mathcal{A}^i\}_{i \in \mathcal{I}}$ indicate sets of joint action spaces, while $\mathcal{A}^i$ is the action space of agent i. $P \triangleq \mathcal{S} \times \mathcal{A}^1 \times \cdots \times \mathcal{A}^{\mathcal{I}} \to [0, 1]$ represents the joint transition probability function of the whole system and $R \triangleq \mathcal{S} \times \mathcal{A}^1 \times \cdots \times \mathcal{A}^{\mathcal{I}} \to \mathbb{R}$ denotes the joint reward function.

Intuitively, all MMDP agents try to find their individual optimal policies to maximize their own cumulative expected rewards, i.e., $\sum_{t=0}^{\infty}\gamma_t r_t^i(s_t,\pi_i^*(s_t))\forall i$. The joint policy π^* induced by the set of individual policies $\{\pi_i^*\}_{i\in\mathcal{I}}$ maps states to joint actions. In this way, an MMDP could be regarded as a single-agent MDP where the agent takes joint actions. On this basis, the MMDP goal is to find a policy that maximizes the expected total reward for all agents by taking their interactions into account. The MDP objective is to find a good policy that maximizes the future reward function, which could be expressed by the following cumulative discounted reward:

$$\begin{aligned} R_t &= r_{t+1}+\gamma r_{t+2}+\gamma^2 r_{t+3}+\cdots=\sum_{k=0}^{\infty}\gamma^k r_{t+k+1}\\ &= r_{t+1}+\gamma(r_{t+2}+\gamma r_{t+3}+\gamma^2 r_{t+4}+\cdots)\\ &= r_{t+1}+\gamma R_{t+1} \end{aligned} \tag{2}$$

where R_t denotes the cumulative reward at time step t and $\gamma\in[0,1]$ represents the discount factor. Here, γ determines the importance of future rewards compared with the current one. If γ approaches one, it means that the decision-maker regards the long-term reward as important. On the contrary, the decision-maker prefers to maximize the current reward, while the discount factor γ approaches zero.

In order to find an optimal policy $\pi^*:\mathcal{S}\to\mathcal{A}$ for the agent to maximize the long-term reward, the state-value function $\mathcal{V}^\pi:\mathcal{S}\to\mathcal{R}$ is first defined in the RL that denotes the expected value of current state s under policy π. The state-value function $\mathcal{V}$ for the following policy π measures the quality of this policy through the discounted MDP, which could be shown as follows:

$$\mathcal{V}^\pi(s)=\mathbb{E}_\pi\left[R_t|s_t=s\right]=\mathbb{E}_\pi\left[\sum_{k=0}^{\infty}\gamma^k r_{t+k+1}\mid s_t=s\right] \tag{3}$$

where the state value is the expected reward for the following policy π from state s.

Similarly, the value of taking action a in state s under policy π, i.e., action-value function $Q^\pi(s,a)$, is defined as

$$\begin{aligned} Q^\pi(s,a) &= \mathbb{E}\left[R_t\mid s_t=s,a_t=a\right]\\ &= \mathbb{E}\left[\sum_{k=0}^{\infty}\gamma^k r_{t+k+1}\mid s_t=s,a_t=a\right]. \end{aligned} \tag{4}$$

Since the purpose of RL is to find the optimal policy that achieves the largest cumulative reward in long-term, we define the optimal policy π^* as

$$\pi^*=\underset{\pi}{\operatorname{argmax}}\,\mathcal{V}^\pi(s). \tag{5}$$

In this situation, the optimal state-value function $\mathcal{V}^*(s)$ and the action-value function $Q^*(s,a)$ could be obtained as

$$\begin{aligned} \mathcal{V}^*(s) &= \max_\pi \mathcal{V}^\pi(s)\\ Q^*(s,a) &= \max_\pi Q^\pi(s,a). \end{aligned} \tag{6}$$

As for the state–action pair (s,a), it is observed that the optimal action-value function gives the expected return for taking action a in state s and thereafter follows an optimal policy. Therefore, $Q^*(s,a)$ could also be written in terms of optimal state-value function $\mathcal{V}^*(s)$, which is expressed as

$$\begin{aligned} Q^*(s,a) &= \mathbb{E}\left[R_t\mid s_t=s,a_t=a\right]\\ &= \mathbb{E}\left[r_{t+1}+\gamma R_{t+1}\mid s_t=s,a_t=a\right]\\ &= \mathbb{E}\left[r_{t+1}+\gamma\mathcal{V}^*(s_{t+1})\mid s_t=s,a_t=a\right]. \end{aligned} \tag{7}$$

Since the action is selected by the policy, an optimal action at each state is found through the optimal policy as well as the optimal state-value function. In this way, the optimal state-value function is rewritten as follows:

$$\begin{aligned} \mathcal{V}^*(s) &= \max_\pi \mathcal{V}^\pi(s)=\max_a \mathbb{E}_{\pi^*}\left[R_t\mid s_t=s,a_t=a\right]\\ &= \max_a \mathbb{E}_{\pi^*}\left[r_{t+1}(s,a)+\gamma\mathcal{V}^*(s_{t+1})\mid s_t=s,a_t=a\right]\\ &= \max_a Q^*(s,a). \end{aligned} \tag{8}$$

Taking the expression of optimal action-value function into account, the problem of optimal sate value is simplified to the optimal values of action function, i.e., $Q^*(s,a)$. Intuitively, (8) indicates that the value of a state under an optimal policy should be equal to the expected reward for the best action from that state, which is denoted by the Bellman optimality equation in MDP [31]. With the definition of optimal value functions and policies, the rest of the work would be to update the value function and achieve the optimal policy, which can be accomplished by RL approaches.

B. Reinforcement Learning

As one of the machine learning paradigms, RL is concerned with a decision-maker's action for maximizing the notion of cumulative reward R_t [32]. In RL, the decision-making process is executed by the agent, which learns the optimal policy by interacting with the environment. Here, the agent first observes the current state and then performs an action in the environment, which is based on its policy. After that, the environment feeds its immediate reward back into the agent and updates its new state at the same time. The typical RL interactions between environment and agent are shown in Fig. 2. The agent will constantly adjust its policy according to the observed information, i.e., the received immediate reward and updated state. This adjustment process will be repeated until the policy of agent approaches its optimum.

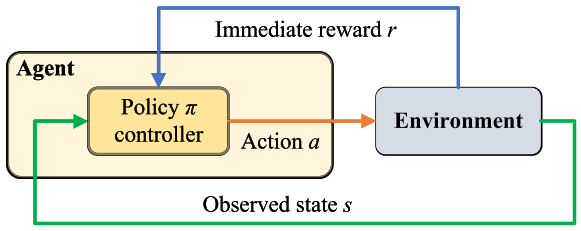


Fig. 2. *Interaction of RL between the environment and agent.*

Generally, RL methods are divided into on-policy and off-policy categories, which indicate whether the training samples are collected by following the target policy. The policy employed to generate samples by interacting with the environment is referred to as the behavior policy, whereas the policy that agents aim to learn and improve upon, based on the collected samples, is known as the behavior policy, whereas the policy that agents aim to learn and improve upon, based on the collected samples, is known as the target policy. As for on-policy methods, target and behavior policies are the same. This means that agents learn a policy (i.e., target policy) and implement it to generate samples for the algorithm training. On the contrary, off-policy methods improve a policy based on samples collected from a different policy (i.e., behavior policy). In the rest part of this section, we introduce the two classical RL methods, including Q-learning (off-policy) and SARSA (on-policy) algorithm, which are also the most effective and widely used methods in the real world. Indeed, these two algorithms are categorized as tabular RL methods due to the fact that they use a tabular representation of Q values.

1) Q-Learning Algorithm: Q-learning is regarded as one of the early breakthroughs in RL, which finds an optimal policy in the sense of maximizing the expected value of the total reward over all successive steps [33]. In particular, the action-value function $Q(s,a)$ is updated by the weighted average of the old value and the new information as

$$\begin{aligned} &Q_{t+1}(s,a) \\ &\quad = Q_t(s,a) \\ &\qquad + \alpha\left[r_{t+1} + \gamma \max_{a'} Q_{t+1}\left(s',a'\right) - Q_t(s,a)\right] \end{aligned} \tag{9}$$

where r_{t+1} is the reward obtained when moving from the state s to the state s' and α represents the learning rate. The core idea behind this update is to find the TD between the estimated action value $r_{t+1} + \gamma \max_{a'} Q_{t+1}(s',a')$ and its old value $Q(s,a)$. In (9), the learning rate α denotes the extent to which the newly acquired information overrides the old one. If the value of learning rate approaches one, it means that the agent considers more recent information and ignores prior knowledge to explore possibilities. In contrast, a zero value makes the agent learn nothing from the current information, which exclusively exploits the prior knowledge. Usually, the learning rate is selected as a constant value in a deterministic environment; otherwise, it may be dynamically adjusted during the learning process for stochastic problems. The detailed framework of Q-learning algorithm is presented in Algorithm 1. Before learning starts, $Q(s,a)$ is initialized to an available arbitrary value. Then, in each episode t, the agent selects an action a according to the policy π and observes a reward r. Subsequently, the agent enters a new state s', which may depend on both the previous state s and the selected action a. After that, the value of Q-learning table is iteratively updated by (9) until the state s reaches the terminal. In summary, Q-learning is a model-free RL algorithm that learns the value of an action in a particular state. It does not require the mathematical model of the environment and has the capacity to handle problems with stochastic transitions and rewards. However, the standard Q-learning algorithm using Q-table would only be applied to discrete action and limited state spaces, which is mainly due to the curse of dimensionality. In other words, this method falters with an increasing number of states/actions since the maintenance of this tremendous table is time-consuming and inefficient.

Algorithm 1 Q-Learning Algorithm

Input: Initialize $Q(s,a)$, for all $s \in \mathcal{S}, a \in \mathcal{A}$, arbitrarily except that $Q(terminal,\cdot) = 0$

for *each episode* t **do**
- From the current state–action pair (s,a), execute action a and receive the immediate reward r and the new state s'.
- Select an action a' with maximal Q-value from the new state s' and then update the table entry for $Q(s,a)$ as follows:

$$\begin{aligned} &Q_{t+1}(s,a) = Q_t(s,a) \\ &\qquad + \alpha\left[r_{t+1} + \gamma \max_{a'} Q_{t+1}(s',a') - Q_t(s,a)\right] \end{aligned}$$

- Replace $s \leftarrow s'$.

end

Output: $\pi^*(s) = \arg\max_a Q^*(s,a)$

2) SARSA Algorithm: Even though Q-learning can find the optimal policy without the need of prior knowledge about the environment, it is an off-policy RL algorithm that obtains the optimal policy only after all Q values have converged. Thus, an alternative on-policy RL method, named SARSA, is introduced in this section to provide an on-policy learning pattern for the agent to approach the optimal policy. Different from the off-policy Q-learning algorithm, SARSA allows the agent to grasp the optimal policy and use the same one to act. As for the Q-

Algorithm 2 SARSA Algorithm

Input: Initialize $Q(s,a)$, for all $s \in \mathcal{S}, a \in \mathcal{A}$, arbitrarily except that $Q(terminal,\cdot)=0$

for *each episode* t **do**

From the current state-action pair (s,a), execute action a and receive the immediate reward r and the new state s'.

Select an action a' from the new state s' using the same policy and then update the table entry for $Q(s,a)$ as follows:

$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha\left[r_{t+1} + \gamma Q_{t+1}(s',a') - Q_t(s,a)\right]$$

Replace $s \leftarrow s'; a \leftarrow a'$.

end

Output: $\pi^*(s) = \arg\max_a Q^*(s,a)$

learning algorithm, the target policy is updated according to the maximal reward of available actions rather than the behavior policy used for choosing actions, i.e., off-policy learning. On the contrary, the SARSA algorithm uses the same policy to update the Q values and select actions, i.e., on-policy learning. The details of the SARSA algorithm are provided in Algorithm 2, which illustrates that the SARSA agent interacts with the environment and updates the policy based on actions taken. Hence, it is regarded as an on-policy RL algorithm. In particular, the action-value function $Q(s,a)$ is updated by the Q value of the next state s' and the current policy's action a' as

$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha\left[r_{t+1} + \gamma Q_{t+1}(s',a') - Q_t(s,a)\right]. \quad (10)$$

In conclusion, the state and action spaces in tabular RL methods are small enough to allow the Q values to be represented as a table. This is feasible when the number of states and actions is small. However, state and action spaces in many real-world applications are excessively large, which makes it impossible to represent the Q values in a table. In such cases, function approximation methods, such as DNNs, are used to approximate the Q values or policy, which is introduced in the following.

C. Deep Learning

As mentioned before, RL is not suitable for handling complicated problems with large-scale environments and high uncertainty, which limits its application in SG operations. To this end, DL is introduced to assist RL in dealing with these challenges. To be specific, DL is a subset of machine learning based on DNNs, which attempts to simulate the human brain behavior and extract the important features from massive raw data. The adjective *deep* in DL refers to the use of multiple layers in a neural network, which enhances the perception capacity of DNN.

Fig. 3 shows the DFF neural network, which is considered the simplest type of DNN. It is observed that a DFF network contains multiple layers of interconnected nodes, i.e., artificial neurons, which are analogous to biological neurons in brain. Each connection between neurons transmits a signal to other ones, and the receiving neuron processes this signal. Then, the receiving neuron activates downstream connected neurons. The signal within a connection is usually represented by a real number between 0 and 1, and the output of each neuron is computed by the weighted summation of its inputs as well as a nonlinear transformation through the activation function. This computation process from the neural network is named the forward propagation, which achieves the data processing during the computation from the input to the output. Typically, neurons are aggregated into layers, and different layers may perform different transformations on their inputs. It should be noted that signals travel from the first layer named input layer to the last one, i.e., the output layer, possibly after traversing deeply hidden layers multiple times.

The DFF neural network shown in Fig. 3 is the simplest among DNNs. There are two classical DNN models, including the CNN [34] and RNN [35]. CNN is distinguished from other DNNs by its superior computer vision performance, which comprises three main layers shown in Fig. 4, i.e., convolutional, pooling, and fully connected. The name CNN stems from the convolution operation that occurred in the convolutional layer, which converts the raw input data to numerical values and allows CNN to interpret and extract relevant features. Similar to the convolutional layer, the pooling layer derives its name from the pooling operation, and it conducts dimension reduction to decrease complexity. The pooling layers can improve the efficiency and reduce the risk of overfitting. Furthermore, the fully connected layer performs the task of classification based on the features learned through previous conventional and pooling layers, which map the extracted features back to the final output.

Unlike CNN, which assumes that inputs and outputs are independent, RNN extracts the information from prior

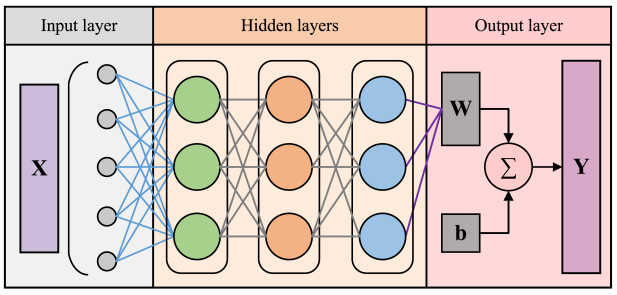


Fig. 3. *Structure diagram of typical DFF neural network.*

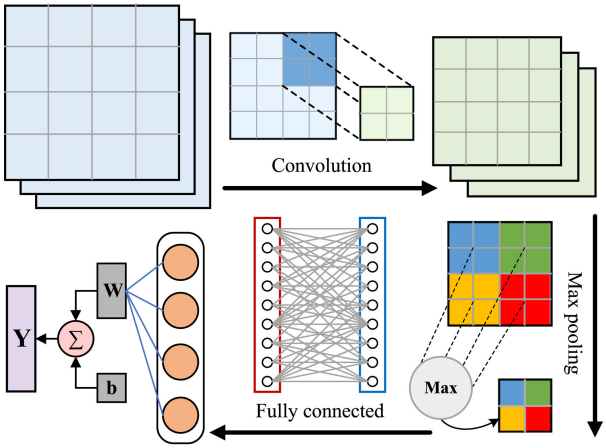


Fig. 4. *Structure diagram of CNN.*

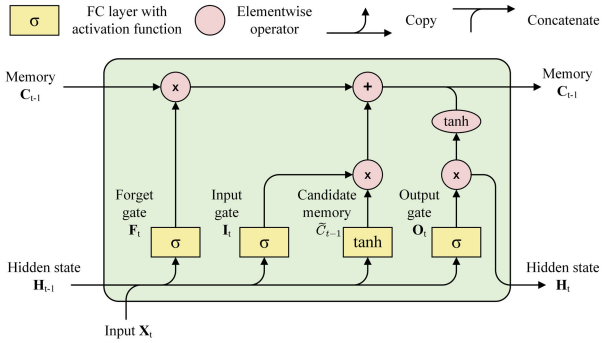


Fig. 6. *Illustration diagram of LSTM.*

inputs to determine the current input and output, as shown in Fig. 5. Here, the RNN output depends not only on its immediate inputs but also on the neural state of previous layers within the sequence. In this way, RNN could utilize its internal state to process variable-length sequences of inputs, which makes it applicable to tasks such as speech recognition.

As an RNN architecture, LSTM is employed extensively for DL. LSTM, shown in Fig. 6, stands out for its remarkable capability to capture and retain long-term dependencies by integrating memory cells and diverse gating mechanisms. The memory cells in LSTM allow the network to store and access information over long periods of time. Moreover, gating mechanisms, including the input gate, forget gate, and output gate, control the information flow and enable the network to selectively retain or forget the information based on its relevance.

After mapping the DNN results to the input data, another challenge is to optimize the results. To this end, backpropagation is proposed to adjust the network parameters in the reverse direction, using the network output deviation from its actual values [36]. Here, the backpropagation uses algorithms such as gradient descent to calculate prediction errors and then refine the activation function weights and biases for training the DNN by moving backward through the layers. Combined forward and backpropagations allow DNN to make predictions and adjust its parameters in accordance with errors. As the training continues, DNN outputs will gradually become more accurate. To conclude, the excellent perception capacity of DNN provides RL with an opportunity to address the existing limitations. First, the strong feature extraction capacity of DNN could help RL avoid the manual feature design process, which is usually difficult to be represented by hand. Second, the outstanding prediction capacity of DNN rescues RL from the curse of dimensionality, which allows it to cope with scenarios with high-dimensional and continuous state/action spaces. Therefore, the combination of RL and DL brings about the formulation of DRL, and it will be discussed in the next section.

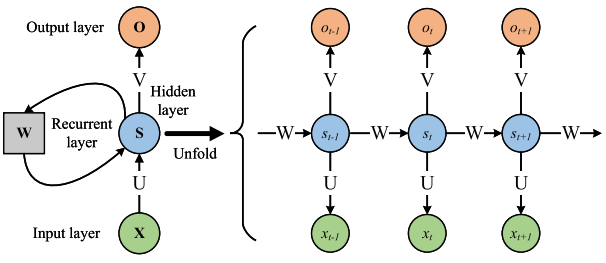


Fig. 5. *Structure diagram of typical RNN.*

D. Deep Reinforcement Learning

As mentioned before, the states of MDP are high-dimensional and difficult to design, which would limit the RL applications in practical decision-making problems. To this end, DRL is introduced to overcome this drawback, which incorporates the DL technique to address the dimensional curse of RL. The DRL value functions and policy are usually parameterized by the DNN variables, rather than the Q-table in RL. Using the excellent DL ability in feature extraction, DRL could complete complex tasks without any prior knowledge. A detailed taxonomy of DRL algorithms is shown in Fig. 7. According to the policy optimization, various DRL approaches could be divided into two categories of value- and policy-based algorithms. The value-based DRL methods usually imply the optimization over the action-value function and further derive the optimal policy. Consequently, the value-based algorithm possesses a relatively higher sampling efficiency and smaller estimation variance of value function and will not fall easily into a local optimum.

However, DRL methods cannot deal with continuous action space problems, which would limit the use of value-based methods in SG. As for the policy-based DRL approaches, they directly optimize the policy and iteratively update the policy to maximize the accumulative

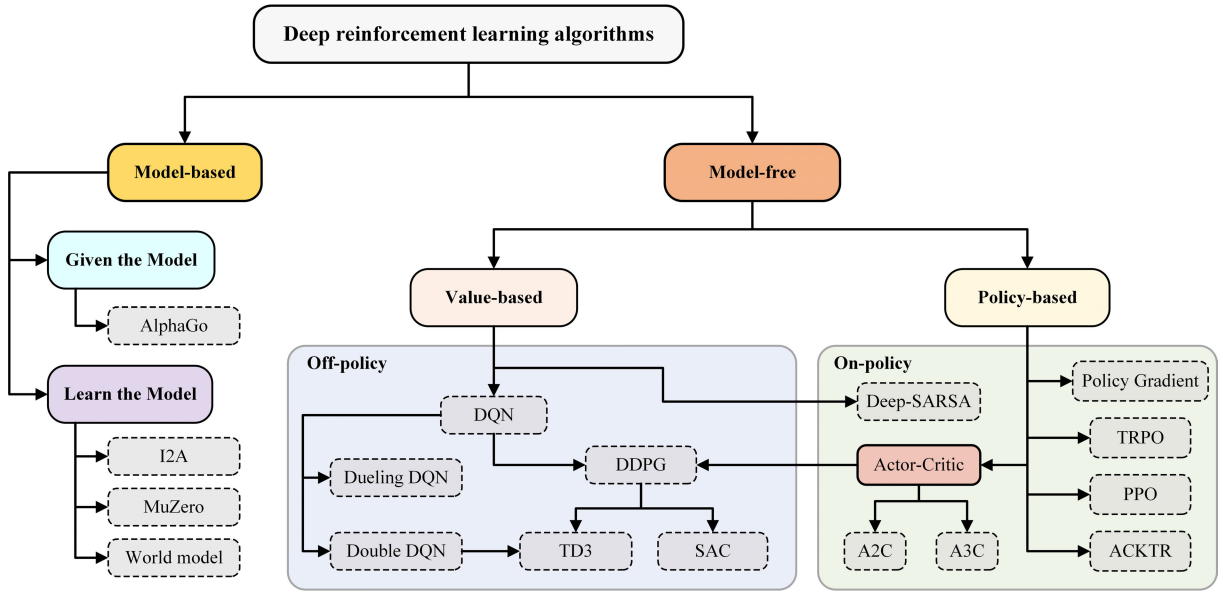


Fig. 7. *Taxonomy of DRL algorithms (boxes with thick borders denote different categories, while others indicate specific algorithms).*

return. In this way, the policy-based algorithm offers a simple policy parameterization and fast convergence speed, which is suitable for problems with continuous or high-dimensional action spaces. Nevertheless, the policy-based methods also suffer from sampling inefficiency and overestimation. However, a combination of these two categories has conveniently given rise to the AC framework. In the rest of this section, we discuss several typical value- and policy-based DRL algorithms.

E. Value-Based DRL Algorithm

The RL goal is to improve its policy to acquire better rewards. As for the value-based algorithm, it tends to optimize the action-value function $Q(s,a)$ for obtaining preferences for the action choice. Usually, value-based algorithms, such as Q-learning and SARSA, need to alternate between the value function estimation under the current policy and the policy improvement with the estimated value function, as shown in (8). However, it is not trivial to predict the accurate value of a complicated action-value function, especially when state and action spaces are continuous. The conventional tabular methods, such as Q-learning, cannot cope with these complex cases because of the limitation of computational resources. Also, state representations in practice would need to be manually designed with aligned data structures, which are also difficult to specify. To this end, the DL technique is introduced to assist RL methods to estimate the action-value function, which is the core concept of value-based DRL algorithms. Next, typical value-based DRL algorithms, including DQN and its variants, are depicted with detailed theories and explanations.

1) Deep Q-Network: As one of the breakthroughs in DRL, the DQN structure shown in Fig. 8 implements DNN as the function approximator for estimating $Q^*(s,a)$, instead of the Q-table. However, the value iteration is proved to be unstable and might even diverge when a nonlinear function approximator, e.g., neural network, is used to represent the action-value function [37]. This instability is attributed to the fact that small updates of $Q(s,a)$ might significantly change the agent policy. Therefore, the data distribution and the correlations between $Q(s,a)$ and the target value $r_{t+1}+\gamma \max_{a'} Q_{t+1}(s',a')$ are quite diverse.

Two key ideas, which include experience replay and fixed target Q-network, are adopted to address the instability issue as described in the following.

1) *Experience Replay:* In each time epoch t, DQN stores the experience of agent (s_t, a_t, r_t, s_{t+1}) into the replay buffer and then draws a mini-batch of samples from this buffer randomly to train the DNN. Then, the Q values estimated by the trained DNN will be applied to generate new experiences, which will be appended into the replay buffer in an iterative way. The experience replay mechanism has several advantages over the fit Q-learning. First, both old and new experiences are used in the experience replay mechanism to learn the Q-function, which provides higher data efficiency. Second, the experience replay avoids the situation where samples used for DNN training are determined by previous parameters, which smooths out changes in the data distribution and removes correlations in the observation sequence.
2) *Fixed Target Q-Network:* To further improve the neural network stability, a separate target network is developed to generate Q-learning targets, instead of the desired Q-network. At times, the target network will be synchronized with the primary Q-network by copying directly (hard update) or exponentially decaying

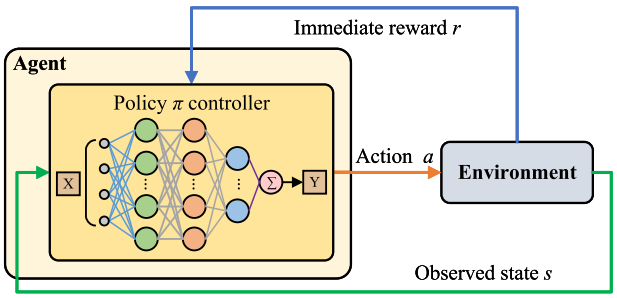


Fig. 8. ***Structure diagram of DRL.***

average (soft update). In this way, the target network is updated regularly but at a rate that is slower than the primary Q-network. This could significantly reduce the divergence and the correlation between the target and estimated Q values.

The DQN algorithm with experience replay and fixed target Q-network is presented in Algorithm 3. Before learning starts, replay buffer D, primary network Q, and target network $\hat{Q}$ are initialized with random parameters. Then, at each episode t, the agent selects an action a_t with ϵ-greedy policy and observe reward r_t, to enter a new state s_{t+1}. After that, the transition (s_t, a_t, r_t, s_{t+1}) is stored in the replay buffer for further sampling. Stochastic gradient descent with respect to the network parameter θ is performed to optimize the DNN loss function, which is defined in (11) as the deviation between the target and primary networks. Finally, target network parameters are updated by the primary network for every certain step until the epoch is terminated

$$\mathcal{L} = \left[r_j + \gamma \max_{a_{j+1}} \hat{Q}(s_{j+1}, a_{j+1}; \theta') - Q(s_j, a_j; \theta) \right]^2. \quad (11)$$

In conclusion, DQN absorbs the advantages of both DL and RL techniques, which are critical for SG application [38], [39], [40].

2) Double DQN: DQN, which has been implemented successfully, has struggled with large overestimations of action values, especially in noisy environments [41]. These over-estimations stem from positive deviation since Q-learning always selects the maximum action value as the approximation for maximal expected reward, which is denoted by the Bellman equation in (8). Therefore, the next Q values are usually overestimated since samples are used to select the optimal action, i.e., with the largest expected reward, and the same samples are also utilized for evaluating the action-value. To this end, a variant algorithm called DDQN is proposed to address the over-estimation problem of DQN [42]. The central idea of DDQN is to decouple correlations in the action selection and value evaluation by using two different networks at these two stages. In particular, the target Q-network in the DQN architecture provides a natural candidate for the extra network. More specifically, the action selection is still executed by the primary network with parameters θ. In other words, DDQN still selects the action with the maximal estimated action value according to the current state, as denoted by θ. However, the value evaluation of current policy is fairly performed by the extra network, i.e., the target network in DDQN with parameter θ'. Therefore, the DDQN loss function could be expressed as

$$\left[r_j + \gamma \hat{Q}(s', \arg\max_{a'} Q(s', a'; \theta); \theta') - Q(s, a; \theta) \right]^2 \quad (12)$$

where $\hat{Q}(s', a'; \theta')$ and $Q(s, a; \theta)$ represent the target network with parameter θ' for value evaluation and the primary network with parameter θ for action selection, respectively. In this way, the estimated value of future expected reward is evaluated using a different policy, which could manage the overestimation issue and outperform the original DQN algorithm [43].

3) Dueling DQN: For certain states, different actions are not relevant to the expected reward and there is no need to learn the effect of each action for such states. For instance, the values of the different actions are very similar in various states, and thus, the action taken would be less important. However, the conventional DQN could accurately estimate the Q value of this state only when all data are collected for each discrete action. This could result in a slower learning speed as the algorithm is not concerned with the actions that are not taken. To address this issue, a network architecture called the dueling DQN is proposed, which explicitly separates the representation

Algorithm 3 DQN Algorithm

Input: Initialize replay buffer $\mathcal{D}$, the primary Q-network $\mathcal{Q}$ with stochastic weights θ and the target Q-network $\hat{\mathcal{Q}}$ with stochastic weights θ'.

for *each episode* t **do**
- With probability ϵ select a random action a_t, otherwise select $a_t = \text{argmax}_a Q^*(s, a; \theta)$.
- Execute action a_t and observe the immediate reward r_t and next state s_{t+1}.
- Store transition (s_t, a_t, r_t, s_{t+1}) in the experience replay buffer $\mathcal{D}$.
- Sample random minibatch of transitions (s_j, a_j, r_j, s_{j+1}) from $\mathcal{D}$.
- Perform a gradient descent step with respect to the network parameter θ to minimize the loss: $\left[r_j + \gamma \max_{a_{j+1}} \hat{Q}(s_{j+1}, a_{j+1}; \theta') - Q(s_j, a_j; \theta) \right]^2$.
- Synchronize $\hat{\mathcal{Q}} = \mathcal{Q}$ every certain interval steps.

end

of action-value function $Q(s,a)$ into the state function $\mathscr{V}(s)$ and state-dependent action advantages $\mathscr{A}(a)$ [44]. Accordingly, the Q value function of dueling DQN would be decoupled into state value and action advantage parts, where

$$Q(s,a) = \mathscr{V}(s) + \mathscr{A}(a). \tag{13}$$

On the one hand, the value part, i.e., state-value function $\mathscr{V}(s)$, concentrates on estimating the importance of current state s. On the other hand, the action advantage is denoted by the state-dependent advantage function $\mathscr{A}(a)$, which estimates the importance of choosing the action a compared with other actions. Intuitively, the dueling architecture could draw lessons from valuable states, without learning the effect of each state action.

However, it might not be suitable if we directly separate the Q value function as shown in (13) since it might be unidentifiable in mathematics, that is, there might exist different combinations of $\mathscr{V}(s)$ and $\mathscr{A}(a)$, where all satisfy (13) for a given $Q(s,a)$. To deal with the identifiability issue, the advantage function estimator is refined to have a zero advantage at the selected action by force

$$Q(s,a;\boldsymbol{\alpha},\boldsymbol{\beta}) = \mathscr{V}(s;\boldsymbol{\beta}) + \left(\mathscr{A}(s,a;\boldsymbol{\alpha}) - \frac{1}{|\mathcal{A}|}\sum_{a'}\mathscr{A}(s,a';\boldsymbol{\alpha})\right) \tag{14}$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are parameters of the two estimators $\mathscr{V}(s;\boldsymbol{\beta})$ and $\mathscr{A}(s,a';\boldsymbol{\alpha})$, respectively. It should be noted that the subtraction in (14) helps with identifiability, which does not change the relative rank of the $\mathscr{A}$ values and preserves the original policy based on Q values from (13). In addition, the stability of policy optimization is enhanced since the advantages in (14) would only need to adapt to the average value, instead of pursuing an optimal solution. The training of dueling DQN requires more network layers compared with the standard DQN, which achieves a better policy evaluation in the presence of large action spaces.

F. Policy-Based DRL Algorithm

Different from the value-based algorithm, policy-based algorithms depend on the use of gradient descent for optimizing the parameterized policies, regarding the expected reward, instead of optimizing the action-value function. The abstract policies in DRL are called parameterized policies as they are represented by parametric neural networks. In particular, policy-based approaches would directly perform the learning of the parameterized policy of agent in DRL without learning or estimating the action-value function. Accordingly, policy-based DRL algorithms do not suffer from specific concerns, which have been encountered with traditional RL methods. These concerns mainly consist of higher complexities that arise from continuous states and actions, the uncertainty stemming from the stochastic environment, and the inaccuracy of estimated action value.

Another benefit of the policy-based algorithm is that the policy gradient methods could naturally model stochastic policies, while the value-based algorithms need to explicitly represent its exploration like greedy to model the stochastic policies. Furthermore, gradient information is utilized to guide the optimization in policy-based algorithms, which contributes to the network training convergence. In general, the policy-based algorithms could be divided into stochastic and deterministic polices, according to their representation. Therefore, several popular policy-based algorithms are introduced here for both policies.

1) Stochastic Policy: As mentioned before, the basic idea of policy-based algorithm is to represent the policy by a parametric neural network $\pi_{\boldsymbol{\theta}}(a|s)$, where the agent randomly chooses an action a at state s according to parameter $\boldsymbol{\theta}$. Then, policy-based algorithms typically optimize the policy $\pi_{\boldsymbol{\theta}}$ with respect to the goal $J(\pi_{\boldsymbol{\theta}})$, through sampling the policies and adjusting the policy parameters $\boldsymbol{\theta}$ in the direction of more cumulative reward, which could be expressed as follows:

$$J(\pi_{\boldsymbol{\theta}}) = \mathbb{E}_{\tau\sim\pi_{\boldsymbol{\theta}}}[R(\tau)] = \mathbb{E}_{\tau\sim\pi_{\boldsymbol{\theta}}}\left[\sum_{k=0}^{\infty}\gamma^k r_{t+k+1}\right] \tag{15}$$

where policy gradient-based optimization uses an estimator for the gradients on the expected return collected from samples to improve the policy with gradient ascent. Here, trajectory τ is a sequence of state–action pairs sampled by current policy π_θ, which records how the agent interacts with the environment. Thus, the gradient regarding the policy parameter is defined as the policy gradient, which could be calculated as follows: $\Delta\boldsymbol{\theta} = \alpha\nabla_{\boldsymbol{\theta}}J(\pi_{\boldsymbol{\theta}})$. On this basis, the policy gradient theorem is proposed to denote the optimal ascent direction of expected reward [45], as illustrated in the following equation:

$$\begin{aligned}\nabla_{\boldsymbol{\theta}}J(\pi_{\boldsymbol{\theta}}) &= \mathbb{E}_{\tau\sim\pi_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{T}\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(\tau)R(\tau)\right] \\ &= \mathbb{E}_{\tau\sim\pi_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{T}\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a_t|s_t)R(\tau)\right].\end{aligned} \tag{16}$$

In this way, policy-based algorithms are updated along the direction of ascent gradients, which is denoted as follows:

$$\boldsymbol{\theta} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta} = \boldsymbol{\theta} + \alpha\nabla_{\boldsymbol{\theta}}J(\pi_{\boldsymbol{\theta}}). \tag{17}$$

Based on policy gradient, several typical policy-based DRL algorithms, including the TRPO and the PPO, are

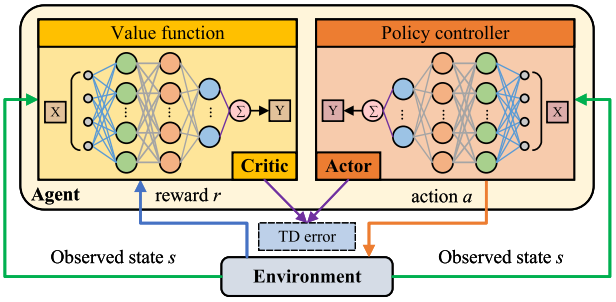


Fig. 9. ***Structure diagram of AC algorithm.***

proposed. In recent years, it has witnessed the successful applications of these algorithms in both academy and industry. In the rest part of this section, we will introduce these policy-based algorithms as well as their variants in detail.

a) Actor–critic: It could be observed from (16) that a straightforward gradient ascent is performed on the policy parameters $\boldsymbol{\theta}$, in order to gradually improve the performance of policy $\pi_{\boldsymbol{\theta}}$. Despite this concision, the conventional policy gradient method is considered to suffer from a large variance while predicting the gradient [46]. Indeed, the complexity and randomness of reward R_t may grow exponentially with the trajectory length, which is difficult to handle. To this end, the AC architecture is proposed to alleviate the large variance problem, which aims to take advantage of all the merits from both value- and policy-based methods while overcoming their drawbacks [47]. More specifically, the principal concept of architecture is to split the model into two components, i.e., the actor decides which action should be taken while the critic feeds the quality of its action back to the actor as well as the suggestion of corresponding adjustments.

In Fig. 9, the actor takes the state as input and outputs the optimal action, which essentially controls the behavior of agent through learning the optimal policy. In comparison, the critic evaluates the selected action by computing the value function $\mathcal{V}^{\pi}(s)$. Generally, the training of these two components is performed separately and gradient ascent is adopted to update their parameters. Here, the critic $\mathcal{V}_{\boldsymbol{\psi}}^{\pi_{\boldsymbol{\theta}}}$ is optimized to minimize the square of TD error δ_t, which is similar to the loss function of DQN as

$$\begin{aligned} \delta_t &= R_t + \gamma \mathcal{V}_{\boldsymbol{\psi}}^{\pi_\theta}(s_{t+1}) - \mathcal{V}_{\boldsymbol{\psi}}^{\pi_\theta}(s_t) \\ J_{\mathcal{V}_{\boldsymbol{\psi}}^{\pi_\theta}}(\boldsymbol{\psi}) &= \mathbb{E}_{\tau\sim\pi_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{T}\delta_t^2\right] \\ \boldsymbol{\psi} &= \boldsymbol{\psi} + \alpha_{\boldsymbol{\psi}}\nabla_{\boldsymbol{\psi}} J_{\mathcal{V}_{\boldsymbol{\psi}}^{\pi_{\boldsymbol{\theta}}}}(\boldsymbol{\psi}) \end{aligned} \tag{18}$$

where $\boldsymbol{\psi}$ represents the parameters of the critic and $\alpha_{\boldsymbol{\psi}}$ denotes the learning rate. It should be mentioned that the accumulative return in (18) is substituted by the TD error, which further reduces the variance of gradient. As for the actor, it follows the principle of policy gradient method to update its policy, which is stated as:

$$\begin{aligned} J(\pi_{\boldsymbol{\theta}}) &= \mathbb{E}_{\tau\sim\pi_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{T}\log\pi_{\boldsymbol{\theta}}(a_t|s_t)\delta_t\right] \\ \boldsymbol{\theta} &= \boldsymbol{\theta} + \alpha_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}} J(\pi_{\boldsymbol{\theta}}). \end{aligned} \tag{19}$$

The pseudocode of the complete network is summarized in Algorithm 4. Before the learning starts, actor network parameters $\boldsymbol{\theta}$, critic network parameters $\boldsymbol{\psi}$, and hyperparameters, including learning rates and discount factor, are initialized as random parameters. After that, the agent selects an action a_t according to the current policy $\pi_{\boldsymbol{\theta}}$, i.e., $a_t \sim \pi_{\boldsymbol{\theta}}(\cdot|s)$, shifts to the next state s_{t+1}, and receives the immediate reward R_t. For each step, the TD error δ_t is first calculated for further actor network selections and critic network evaluations. Finally, the policy gradient theorem is applied to update the parameters of both actor and critic networks, as shown in (18) and (19), respectively.

In conclusion, the AC algorithm is situated in the intersection of policy- and value-based methods, which is regarded as the breakthrough in DRL and derives a series of state-of-the-art DRL algorithms, such as A2C, A3C [48], TRPO, PPO, and SAC. Furthermore, the architecture inspires the deterministic policy methods, such as the DDPG algorithm, which will be discussed in this section later.

b) Trust region policy optimization: Although the AC method achieves a combination of policy- and value-based methods, it still suffers from the learning step-size pitfall

Algorithm 4 AC Algorithm

Input: Initialize actor network parameters $\boldsymbol{\theta}_0$ and critic network parameters $\boldsymbol{\psi}_0$; Initialize learning rates of actor and critic networks, respectively.

for *each episode* t **do**
- Actor network selection: $J(\pi_{\boldsymbol{\theta}}) = \mathbb{E}_{\tau\sim\pi_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{T}\log\pi_{\boldsymbol{\theta}}(a_t|s_t)\delta_t\right]$
- Critic network evaluation: $J_{\mathcal{V}_{\boldsymbol{\psi}}^{\pi_\theta}}(\boldsymbol{\psi}) = \mathbb{E}_{\tau\sim\pi_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{T}\delta_t^2\right]$
- Take action a_t and observe next state s_{t+1} and reward R_t according to current policy $\pi_{\boldsymbol{\theta}}(\cdot|s)$.
- Collect sample (a_t, s_t, R_t, s_{t+1}) into the trajectory.
- Calculate the TD error as follows: $\delta_t = R_t + \gamma\mathcal{V}_{\boldsymbol{\psi}}^{\pi_\theta}(s_{t+1}) - \mathcal{V}_{\boldsymbol{\psi}}^{\pi_\theta}(s_t)$.
- Replace $\boldsymbol{\psi} = \boldsymbol{\psi} + \alpha_{\boldsymbol{\psi}}\nabla_{\boldsymbol{\psi}} J_{\mathcal{V}_{\boldsymbol{\psi}}^{\pi_{\boldsymbol{\theta}}}}(\boldsymbol{\psi})$.
- Replace $\boldsymbol{\theta} = \boldsymbol{\theta} + \alpha_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}} J(\pi_{\boldsymbol{\theta}})$.

end

Output: Parameters pair of the actor and critic $(\boldsymbol{\theta}, \boldsymbol{\psi})$

just like the standard gradient descent algorithm. Indeed, the gradient $\nabla_{\boldsymbol{\theta}} J(\pi_{\boldsymbol{\theta}})$ only provides the local first-order information at current parameters $\boldsymbol{\theta}$, which completely ignores the curvature of the reward landscape. However, the suitable adjustment of learning step is very important for policy gradient methods. On the one hand, the algorithm might suffer a performance collapse if the learning step $\alpha_{\boldsymbol{\theta}}$ is large. On the other hand, if the step size is set small, the learning would be conservative to converge. What is more, the gradient $\nabla_{\boldsymbol{\theta}} J(\pi_{\boldsymbol{\theta}})$ in policy gradient methods requires an estimation from samples collected by the current policy $\pi_{\boldsymbol{\theta}}$, which in turn affects the quality of the collected samples and makes the learning performance more sensitive to the step-size selection.

Another shortcoming of policy gradient method in the standard AC model is that the update occurs in the parameter space rather than the policy space. This makes it more difficult to tune in the step size $\alpha_{\boldsymbol{\theta}}$ since the same step size may correspond to totally different updated magnitudes in the policy space, which is dependent on the current policy $\pi_{\boldsymbol{\theta}}$. To this end, an algorithm, called the TRPO, is developed, which is based on the concept of trust region for adjusting the step size more precisely in the policy gradient [49]. It should be noted that the goal of policy-based method is to find an updated policy $\pi'_{\boldsymbol{\theta}}$ that improves the current policy $\pi_{\boldsymbol{\theta}}$. Fortunately, the improvement from the current policy to the updated one could be measured by the advantage function $\mathscr{A}^{\pi_{\boldsymbol{\theta}}}(s,a)$ [50], which was introduced in the dueling DQN. It is illustrated that (20) provides an insightful connection between the performances of $\pi'_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}}$

$$\begin{aligned} \mathscr{A}^{\pi_{\boldsymbol{\theta}}}(s,a) &= Q^{\pi_{\boldsymbol{\theta}}}(s,a) - \mathscr{V}_{\boldsymbol{\psi}}^{\pi_{\boldsymbol{\theta}}}(s) \\ J(\pi'_{\boldsymbol{\theta}}) &= J(\pi_{\boldsymbol{\theta}}) + \mathbb{E}_{\tau\sim\pi'_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{\infty}\gamma^t \mathscr{A}^{\pi_{\boldsymbol{\theta}}}(a_t,s_t)\right] \end{aligned} \tag{20}$$

where τ denotes the state–action trajectory sampled by updated policy $\pi'_{\boldsymbol{\theta}}$. Obviously, learning the optimal policy is equivalent to optimizing the bonus term $\mathbb{E}_{\tau\sim\pi'_{\boldsymbol{\theta}}}[\sum_{t=0}^{\infty}\gamma^t \mathscr{A}^{\pi_{\boldsymbol{\theta}}}(a_t,s_t)]$. The above expectation is based on the updated policy $\pi'_{\boldsymbol{\theta}}$ that is difficult to optimize directly. Thus, TRPO optimizes an approximation of this expectation, denoted by $\mathcal{L}_{\pi_\theta}(\pi'_\theta)$, which is stated as

$$\mathcal{L}_{\pi_\theta}\left(\pi'_\theta\right) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{\infty}\gamma^t \frac{\pi'_\theta\left(a_t \mid s_t\right)}{\pi_\theta\left(a_t \mid s_t\right)}\mathscr{A}^{\pi_\theta}\left(s_t,a_t\right)\right] \tag{21}$$

where $\pi'_{\boldsymbol{\theta}}$ is directly approximated by $\pi_{\boldsymbol{\theta}}$, which seems to be coarse, but its approximation error (22) is proved to be theoretically bounded and thus ensures its effectiveness [51]. The bounded approximation error is presented as

$$\left|J(\pi'_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}}) - \mathcal{L}_{\pi_\theta}\left(\pi'_\theta\right)\right| \le C\cdot D_{KL}^{\max}\left(\pi_\theta \| \pi'_\theta\right) \tag{22}$$

where C is a constant independent to $\pi'_{\boldsymbol{\theta}}$ and $C\cdot D_{KL}^{\max}(\pi_\theta\|\pi'_\theta)$ represents the maximum Kullback-Leibler (KL) divergence, which is a statistical distance measuring the difference between π'_θ and π_θ. Therefore, it is reasonable to optimize $\mathcal{L}_{\pi_\theta}(\pi'_\theta)$ if $D_{KL}^{\max}(\pi_\theta\|\pi'_\theta)$ is small, which is actually the principle of TRPO. On this basis, the original problem is converted into an optimization problem, which is stated as

$$\begin{aligned} &\max_{\pi'_{\boldsymbol{\theta}}} \mathcal{L}_{\pi_\theta}\left(\pi'_\theta\right) \\ &\text{s.t. } \mathbb{E}\left[D_{KL}^{\max}\left(\pi_\theta\|\pi'_\theta\right)\right] \le \xi \end{aligned} \tag{23}$$

where ξ is a predefined constant denoting the maximum allowable difference between $\pi'_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}}$. Afterward, the first-order approximation for the objective function and the second-order approximation for constraints are adopted to solve this optimization problem. In fact, the gradient of $\mathcal{L}_{\pi_\theta}(\pi'_\theta)$ at the current policy could be expressed by (24), which is similar to the AC

$$\begin{aligned} g &= \nabla_\theta \mathcal{L}_{\pi_\theta}\left(\pi'_\theta\right) \\ &= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{\infty}\nabla_\theta \log \pi_\theta\left(a_t \mid s_t\right)\gamma^t \mathscr{A}^{\pi_\theta}\left(s_t,a_t\right)\right]. \end{aligned} \tag{24}$$

Accordingly, the TRPO algorithm solves the approximated optimization problem at the current policy as

$$\begin{aligned} \theta' &= \arg\max_{\theta'} g^\top\left(\theta'-\theta\right) \\ &\text{s.t. } \left(\theta'-\theta\right)^\top H\left(\theta'-\theta\right) \le \xi \end{aligned} \tag{25}$$

where H represents the Hessian matrix of $\mathbb{E}[D_{KL}^{\max}(\pi_\theta\|\pi'_\theta)]$. It is illustrated by (25) that the gradients are calculated in the first order and the constraint is depicted in the second order. This approximation problem can be analytically solved by the methods of Lagrangian duality [52], resulting in the following analytic form solution:

$$\boldsymbol{\theta}' = \boldsymbol{\theta} + \sqrt{\frac{2\xi}{g^\top H^{-1} g}} H^{-1} g. \tag{26}$$

In summary, TRPO trains the stochastic policy in an on-policy way, where it explores by sampling according to the newest version of its stochastic policy. During the training procedure, the policy usually becomes less uncertain, progressively, since the update rule encourages it to exploit rewards that it has already obtained. Empirically, the TRPO method performs well on previous problems that require precise problem-specific hyperparameter tuning, which are solvable with a set of reasonable parameters. However, one challenge with the implementation of TRPO lies in calculating the estimation of KL divergence between parameters, as it increases the complexity and the

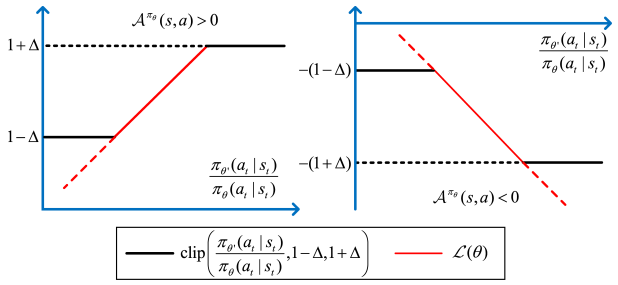


Fig. 10. ***Diagram of clipping mechanism in the PPO algorithm.***

computation time of TRPO and thus limits it applicability in practical terms. To this end, some improvements and simplifications are developed to tackle this specific trouble, which will be discussed next.

c) Proximal policy optimization: TRPO is relatively complex and suffers from a computational burden when calculating the conjugate gradients for constrained optimization. The complexity of computing the second-order Hessian matrix H^{-1} reaches $\Omega(N^3)$, which is quite expensive to undertake in the real world. Here, N denotes the number of parameters. Therefore, another policy gradient approach is developed in [53], using the PPO, to enforce a simpler and more efficient solution for calculating the similarity between updated and current policies. Unlike TRPO that tends to optimize (23) with a hard constraint, PPO improves the objective function of TRPO by converting the constraint into a penalty term. Indeed, the Lagrangian duality theorem is applied to adjoin a constraint to the objective function through a multiplier. The dual problem after adjoining the constraint is mathematically equivalent to the primal formulation under a constraint qualification condition [54]. Therefore, the objective function in PPO is rewritten in (27) after adjoining the constraint to the primal objective

$$\max_{\pi'_\theta} \mathcal{L}_{\pi_\theta}\left(\pi'_\theta\right) - \lambda \cdot \mathbb{E}\left[D_{\mathrm{KL}}\left(\pi_\theta \| \pi'_\theta\right)\right] \tag{27}$$

where λ is the Lagrange multiplier associated with the inequality constraint. For each ξ in (23), there exists a corresponding constant λ, which provides (23) and (27) with the same optimal solutions. Thus, it is significant to adjust the value of Lagrange multiplier adaptively. Here, the KL divergence is particularly checked for adjusting λ. This method, which is referred to as the PPO-penalty algorithm [55], is illustrated in Algorithm 5. The PPO-penalty approximately solves a KL-constrained update, like TRPO, by penalizing the KL divergence in the objective function and then adjusting the multiplier λ over the course of training so that it is scaled appropriately.

Another variant method of PPO, i.e., PPO-clip, would clip the objective value intuitively for the policy gradient, which brings about a more conservative update [56]. Here, the PPO-clip does not contain the KL-divergence term in the objective or constraint. Instead, it depends on specialized clipping in the objective function to remove incentives for the new policy to diverge from the old one. To achieve this goal, the ratio of the new to the old policy is represented as $\ell_t(\theta') = (\pi'_\theta(a_t \mid s_t))/(\pi_\theta(a_t \mid s_t))$. Then, the clipping mechanism is introduced as a regulator to prevent the dramatic policy update from affecting the learning performance of agents. In particular, PPO tends to clip $\ell_t(\theta')$ within $[1-\Delta, 1+\Delta]$ to ensure that the updated policy π'_θ is adjacent to $\pi_{\boldsymbol{\theta}}$. In other words, if $\ell_t(\theta')$ falls outside the interval, the advantage function will be clipped, as shown in Fig. 10. Finally, the minimum of clipped and unclipped objectives is selected as the learning objective. Therefore, PPO would maximize the lower bound of the target objective while maintaining a controllable update from $\pi_{\boldsymbol{\theta}}$ to $\pi'_{\boldsymbol{\theta}}$

$$\begin{aligned}\mathcal{L}^{\mathrm{PPO}}\left(\pi'_\theta\right) = \mathbb{E}_{\pi_\theta}\big[\min\big(&\ell_t\left(\theta'\right)\mathscr{A}^{\pi_\theta}(s_t,a_t),\\ &\mathrm{clip}(\ell_t\left(\theta'\right), 1-\Delta, 1+\Delta)\mathscr{A}^{\pi_\theta}(s_t,a_t)\big)\big].\end{aligned} \tag{28}$$

Algorithm 5 PPO Algorithm With Penalty

Input: Initialize policy parameters $\boldsymbol{\theta}$ and value function parameters ψ; Initialize reward discount factor γ and KL penalty coefficient λ; Initialize adaptive parameters $a = 1.5$ and $b = 2$, respectively;

for *each episode* t **do**
  Take action a_t and observe next state s_{t+1} and reward R_t according to current policy $\pi_{\boldsymbol{\theta}}(\cdot|s)$.
  Collect sample (a_t, s_t, R_t, s_{t+1}) into the trajectory.
  Estimate the advantage function as follows: $\mathscr{A}^{\pi_{\boldsymbol{\theta}}}(s,a) = Q^{\pi_{\boldsymbol{\theta}}}(s,a) - \mathscr{V}_{\boldsymbol{\psi}}^{\pi_{\boldsymbol{\theta}}}(s)$.
  for $m \in \{1, 2, \ldots, M\}$ **do**
    $J_{\mathrm{PPO}}(\pi_{\boldsymbol{\theta}}) = \mathbb{E}\left[\frac{\pi'_\theta(a_t|s_t)}{\pi_\theta(a_t|s_t)}\mathscr{A}_t - \lambda \cdot D_{KL}\left(\pi_\theta \| \pi'_\theta\right)\right]$
    Update θ by a gradient method w.r.t $J_{\mathrm{PPO}}(\pi_{\boldsymbol{\theta}})$.
  end
  for $b \in \{1, 2, \ldots, B\}$ **do**
    $L_{BL}(\psi) = -\sum_{t=1}^{\infty}\left(\sum_{\tau-t}\gamma^{\tau-t}R_\tau - \mathscr{V}_\psi(s_t)\right)^2$
  end
  Compute $d = \mathbb{E}\left[D_{KL}\left(\pi_\theta \| \pi'_\theta\right)\right]$.
  if $d < d_{target}/a$ **then**
    Update λ with $\lambda \leftarrow \lambda/b$;
  else if $d > d_{target} \times a$ **then**
    Update λ with $\lambda \leftarrow \lambda \times b$;
  end
end

Output: Parameters of the $(\boldsymbol{\theta}, \psi)$ pair.

Hence, PPO is motivated to take the largest possible advantage of current data, without stepping out so far that could accidentally cause any performance collapse. Unlike TRPO which tends to solve this problem with a complicated second-order method, PPO is a member of the first-order approaches, which adopts clipping tricks to maintain the proximity between old and new policies. The PPO algorithm performs comparably or even better than the other state-of-the-art methods, is significantly simpler to implement, and has thus become the default DRL in many popular platforms due to its ease of use and good performance [57].

2) Deterministic Policy: The content described above belongs to the stochastic policy gradient, which aims to optimize the stochastic policy $\pi(a|s)$ and represent the action as a probabilistic distribution according to the current state, where $a \sim \pi(\cdot|s)$. On the contrary, the deterministic policy considers the action as a deterministic output of policy, i.e., $a = \mu(s)$, instead of sampling the probability from the given distribution. In addition, it is derived that the deterministic policy (29) follows the policy gradient theorem despite the fact that they have different explicit expressions [58]:

$$\nabla_\theta J(\mu_\theta) = \mathbb{E}\left[\nabla_\theta \mu_\theta(s) \nabla_a Q^\mu(s,a)|_{a=\mu_\theta(s)}\right] \tag{29}$$

where $\mu(s)$ denotes the deterministic policy rather than $\pi(s)$, in order to eliminate the ambiguity in the distinction with stochastic policy $\pi(a|s)$. It is illustrated in (29) that the derivation of reward with respect to the action is integrated, while the integration of policy is absent. This makes the deterministic policy easier to train in high-dimensional action spaces when compared with the stochastic one. Nowadays, some methods that combine DQN with the deterministic policy are quite popular, which take advantage of both methods and perform well in most environments, especially with continuous action spaces. Next, we will discuss several typical deterministic policy gradient algorithms, including DDPG and its extensions, with detailed theories and explanations.

a) Deep deterministic policy gradient: The DDPG approach is viewed as the combination of DNN and deterministic policy gradient algorithm. DDPG is devoted to addressing the problem with continuous action spaces that DQN cannot tackle easily. Hence, DDPG is regarded as an extension of DQN in the continuous action spaces, with the help of deterministic policy gradient. More specifically, DDPG adopts the AC architecture from the policy gradient framework, which maintains a deterministic policy function $\mu(s)$ (actor) as well as a value function $Q(s,a)$ (critic). The policy gradient algorithm is used to optimize the policy function assisted by the value function. The AC used in DDPG is different from the previous one since this actor is a deterministic policy function. Nevertheless, the value function in DDPG is the same as that in DQN, which utilizes the TD error to update itself.

The overall pseudocode of DDPG presented in Algorithm 6 initializes the replay buffer $\mathcal{R}$ and parameters of four networks. Then, it selects the action according to the current policy and exploration noise as $a_t = \mu(s_t|\theta^\mu) + \mathcal{N}_t$, in order to enhance the DDPG exploration capacity. After execution, the action a_t and receive reward r_t are transferred to the next state s_{t+1} to store the transition (s_t, a_t, r_t, s_{t+1}) in buffer $\mathcal{R}$. On this basis, DDPG simultaneously maintains two models, i.e., actor and critic, in order to manage the problems with continuous action spaces. As for the critic network, it aims to approximate the output of value function, which uses the same structure as DQN, i.e., a primary network and a target network. Then, the critic network updates its state by minimizing the loss as

$$\begin{aligned} y_i &= r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'}) \\ \mathcal{L} &= \frac{1}{N}\sum_i^N \left(y_i - Q(s_i, a_i|\theta^Q)\right)^2 \end{aligned} \tag{30}$$

where y_i represents the estimated Q value of target network with parameter $\theta^{Q'}$ and $\mathcal{L}$ denotes the loss error between primary and target networks. It is found that the critic network shares the same double network architecture with DQN, which improves the accuracy of value estimation and maintains the stability during training.

As for the actor network, DDPG tends to learn a deterministic policy $\mu(s)$ and select the action that maximizes the value function $Q(s,a)$. Therefore, the gradient ascent method is performed to optimize the policy as

$$\begin{aligned} \nabla_{\theta^\mu} J &= \mathbb{E}\left[\nabla_{\theta^\mu} Q(s, a|\theta^Q)\right] \\ &= \mathbb{E}\left[\nabla_a Q(s, a|\theta^Q) \nabla_{\theta_\mu} \mu(s|\theta^\mu)\right] \\ &\approx \frac{1}{N}\sum_i^N \nabla_a Q(s, a|\theta^Q) \nabla_{\theta^\mu} \mu(s|\theta^\mu) \end{aligned} \tag{31}$$

where the chain rule of mathematics is applied to calculate the expected return from the start distribution J. Unlike the target network which is regularly updated by the primary network in DQN, DDPG develops a novel updating mechanism called the soft update, which changes network parameters by exponential smoothing rather than directly copying the parameters as follows:

$$\begin{aligned} \theta^{Q'} &\leftarrow \rho\theta^Q + (1-\rho)\theta^{\theta^{Q'}} \\ \theta^{\mu'} &\leftarrow \rho\theta^\mu + (1-\rho)\theta^{\theta^{\mu'}} \end{aligned} \tag{32}$$

where ρ represents the update coefficient, which is far less than 1 so that the learning work is updated very slowly and smoothly, thus promoting the learning stability.

In summary, DDPG combines ideas from both DQN and AC techniques, which extends the Q-learning into the continuous action spaces and produces a lasting influence for subsequent DRL algorithms. On the one hand, the

Algorithm 6 DDPG Algorithm

Input: Initialize replay buffer $\mathcal{R}$. Randomly initialize actor network parameters θ^{μ} and critic network parameters θ^{Q}. Initialize target network Q' and μ' with parameters $\theta^{Q'} \leftarrow \theta^{Q}$, $\theta^{\mu'} \leftarrow \theta^{\mu}$.

for *each episode* **do**
  Initialize a random process $\mathcal{N}$ for action exploration.
  Receive initial observation state s_1
  for $t = 1, \ldots, T$ **do**
    Selection action as $a_t = \mu(s_t|\theta^{\mu}) + \mathcal{N}_t$ according to the current policy and exploration noise.
    Execute action a_t and observe reward r_t as well as the new state s_{t+1}.
    Store transition (s_t, a_t, r_t, s_{t+1}) in the replay buffer $\mathcal{R}$.
    Set $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'}))$.
    Update critic by minimizing the loss: $\mathcal{L} = \frac{1}{N}\sum_i \left(y_i - Q(s_i, a_i|\theta^{Q})\right)^2$.
    Update the actor policy using the sampled policy gradient:
    $\nabla_{\theta^{\mu}} J \approx \frac{1}{N}\sum_i^N \nabla_a Q(s, a)|\theta^{Q}) \nabla_{\theta^{\mu}} \mu(s|\theta^{\mu})$.
    Update the target networks:
    $\theta^{Q'} \leftarrow \rho\theta^{Q} + (1-\rho)\theta^{\theta^{Q'}}$
    $\theta^{\mu'} \leftarrow \rho\theta^{\mu} + (1-\rho)\theta^{\theta^{\mu'}}$
  end
end

Output: Parameters pair of the actor and critic

AC architecture is transformed into off-policy methods because of DQN, and thus, networks are trained with samples from replay buffer that further enhances the sample efficiency. What is more, the introduction of replay buffer can also alleviate the correlations between samples, which brings about more robust and stable learning performance. On the other hand, DDPG can cope with high-dimensional problems with continuous action spaces owing to the existence of AC model. In addition, the development of soft target update trick not only smooths the network training but also impels other DRL algorithms such as TD3 and SAC.

b) Twin delay DDPG: Even though DDPG achieves a great performance, it is still frequently beset with respect to hyperparameters tuning. A common shortcoming of traditional DDPG is the overestimation of Q values, which also appears in DQN as DDPG shares the same function approximator in the action section with DQN. To this end, an improved version of DDPG named TD3 is proposed to address this issue, via introducing three critical tricks [59].

First, TD3 draws lessons from double-DQN that learn two Q-functions instead of one, from which the name of twin is originated. Then, it selects the smaller one in these two Q values, in order to form the targets in the Bellman error loss function

$$\begin{aligned} Q_{\theta_1'}(s', a') &= Q_{\theta_1'}(s', \mu_{\psi_1}(s')) \\ Q_{\theta_2'}(s', a') &= Q_{\theta_2'}(s', \mu_{\psi_2}(s')) \\ y_1 &= R_i + \gamma \min_{i=1,2} Q_{\theta_i'}(s', \mu_{\psi_i}(s')) \end{aligned} \tag{33}$$

where using the smaller Q value for the target helps mitigate the overestimation in the Q-function.

Second, it should be mentioned that the target network is an effective tool to improve the DRL stability. This is because the deep function approximator needs multiple gradient updates to converge, and target networks can supply a stable objective during the learning process. In the absence of a fixed target network, residual errors would accumulate with each update. Therefore, target networks could be used to reduce the error of multiple gradient updates, and the policy updates based on high-error states that would lead to a good divergence. Then, the policy network should have a lower updating frequency than the value network to minimize the estimation error before policy updates. Accordingly, the updating frequency of the policy network is reduced by applying the TD3 algorithm, which is also the name *delay* stems from. Generally speaking, the less frequent policy updates, the smaller variance of Q value function is, which results in a higher quality policy.

Third, one of the issues with the deterministic policy is that such a method might be overfitted to narrow peaks in value estimation. In other words, if the Q-function approximator produces an incorrect sharp peak for some actions, the policy will quickly exploit this spike and then bring about fragile or incorrect behavior. Hence, the target policy smoothing regularization is developed to address this issue by adding noise to the target action to smooth out the Q value function and avoid any overfitting

$$\begin{aligned} \Delta &\sim \text{clip}(N(0, \sigma), -c, c) \\ y &= R + \gamma Q_{\theta'}(s', \mu_{\psi'}(s') + \Delta) \end{aligned} \tag{34}$$

where Δ represents the truncated normal distribution noise to each action as a regularization, which is clipped into the valid range $[-c, c]$.

To conclude, TD3 is the successor of DDPG that aims to address the overestimation problem of critic network in the conventional DDPG. In particular, TD3 gains an improved performance over the baseline DDPG via introducing three key tricks, including the clipped DDQN for AC, delayed policy and target updates, and target policy smoothing regularization. However, as an extension of DDPG, TD3 is still sensitive to hyperparameter tuning, and different hyperparameter settings will lead to different performances, which might be a potential direction of future research. In summary, the fundamental knowledge of RL, DL, and DRL is presented in this section with a

detailed explanation. On this basis, various advanced DRL techniques as well as their extensions are discussed to analyze their advantages and drawbacks. It could be concluded that different DRL algorithms are applicable to deal with different problems in different scenarios. Therefore, in Section III, applications of DRL in SG operations for various problems are reviewed with discussion.

III. APPLICATIONS OF DRL IN SG OPERATION

The SG applications are devoted to achieving a sustainable, secure, reliable, and flexible energy delivery through bidirectional power and information flows. SG applications possess the following features.

1) SG offers a more efficient way to maintain the optimal dispatch with a lower generation cost and higher power quality via the integration of distributed sources and flexible loads, such as RE and EVs [60], [61], [62], [63], [64].
2) SG achieves the secure and stable operation of power system via the deployment of effective operational control technologies, including the AGC, AVC, and LFC [65], [66], [67], [68].
3) SG provides a transaction platform for customers and suppliers affiliated to different entities, thus enhancing the interactions between suppliers and customers, which facilitates the development of electricity market [69], [70], [71].
4) SG equipment encompasses numerous advanced infrastructures, including sensors, meters, and controllers, which are also applied to emerging conditions, such as network security and privacy preservation [72], [73], [74], [75].

On this basis, a typical SG architecture is shown in Fig. 11, which illustrates that the SG operation involves four fundamental segments, i.e., power generation, transmission, distribution, and customers. As for the generation, traditional thermal energy is converted to electrical power, while large-scale RE integration is a promising trend in SG applications. After that, the electrical energy is delivered from the power plant to the power substations via the high-voltage transmission lines. Then, substations lower the transmission voltage and distribute the energy to individual customers such as residential, commercial, and industrial loads. During the transmission and distribution stages, numerous smart meters are deployed in SGs to ensure their secure and stable operations. In addition, such advanced infrastructures bring about emerging concerns, e.g., network security and privacy concern, that traditional power systems would seldom encounter.

In order to support SG operations, DRL applications are also divided into four categories, including optimal dispatch, operational control, electricity market, and other emerging issues, such as network security and privacy concern. These problems usually have similar economic and technical objectives for reducing operational costs,

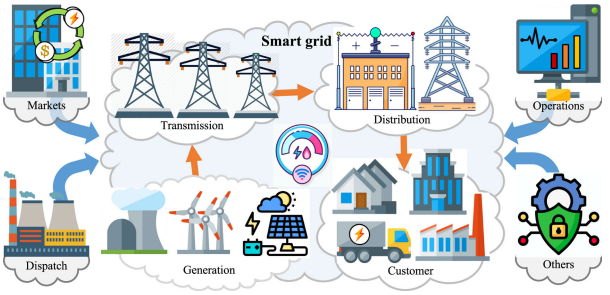


Fig. 11. *Typical architecture of SG operation. This figure is cited from [76].*

suppressing voltage fluctuations, and strengthening the SG security and stability. Despite the applicability of traditional power system methods, it is envisaged that they may be inadequate for SG applications characterized by high levels of renewable and variable energy penetrations and increased human participation in load management. Traditional optimization methods could struggle with identifying the best solutions for these problems due to the high uncertainty in prevailing SG operations and the high dimensionality of distributed systems with coupled variables that are metastasizing in SGs.

In essence, it would be difficult to establish the corresponding accurate models. Fortunately, DRL agents could automatically learn the pertinent knowledge in such cases while interacting with the environment, which is independent of the accurate environment model. However, the purpose of applying DRL is not to completely replace conventional optimization methods. Instead, DRL can serve as a complement to existing approaches and enhance them in a data-driven manner. In this way, DRL has the advantage of addressing such SG problems more effectively due to its data-driven and model-free nature. In the rest part of this section, the DRL applications to optimal dispatch, operational control, electricity market, and other emerging areas are analyzed and investigated in detail.

A. Optimal Dispatch

Compared with traditional power systems, SG integrates more distributed RE to promote sustainability [76]. Under this circumstance, the conventional centralized high-voltage power transmission might not be considered an economic operation since the RE sources are usually distributed and closely located to load centers. As a result, DN, self-sufficient microgrid, and IES are gradually becoming more independent of transmission network operations, which are also highlighted as a major developing trend in SG applications [77], [78], [79], [80]. In addition, it has witnessed the rapid development of EVs in recent years, which has already become a critical SG component [81], [82], [83], as shown in Fig. 12. To this end, applications of DRL regarding optimal dispatch on DN, microgrid, IES, and EV are summarized as follows.

1) Distribution Network: In recent years, DN operations have faced significant challenges mainly due to the

Table 2 DN Optimal Dispatch Based on DRL

Ref.	Management method	Solving Algorithm	Algorithm performance		
			Convergence	Privacy protection	Scalability
[84]	Voltage regulation	DDPG	✓		✓
[85]	Voltage regulation	DDPG	✓		✓
[86]	Model-free volt-var control	SAC	✓		✓
[87]	Model-free volt-var control	SAC	✓		✓
[88]	Two-stage volt-var control	MADDPG	✓		
[89]	Two-stage volt-var control	DQN	✓		
[90]	Many-objective DNR	DQN-EA	✓		✓
[91]	Online DNR	DQN	✓		✓
[92]	Model-free dynamic DNR	SAC	✓		✓
[93]	Demand response	Federated AC	✓	✓	✓
[94]	Resilience enhancement	A2C	✓		✓

increasing deployment of DERs and EVs. Specifically, the uncertain RE output could impact the distribution and the direction of DN power flow, which may further lead to the increase of power loss and voltage fluctuations. Hence, traditional methods based on mathematical optimization methods might not deal effectively with this highly uncertain environment. More importantly, these traditional methods significantly depend on the accurate DN parameters, which are difficult to acquire in practice. To address these limitations, DRL methods are applied in DNs, which could provide more flexible control decisions and promote the operation of DN. Generally, the reward can be designed to achieve certain goals, such as minimizing power losses, improving voltage profile, or maximizing RE utilization. The literature about the applications of DRL on the DN is listed in Table 2, which is summarized from two aspects, i.e., management method and solving algorithm. In addition, the performance of reviewed methods is analyzed from the perspectives of convergence, privacy protection, and scalability, where the tick mark means an outstanding performance and blank means the corresponding article that does not refer to the performance.

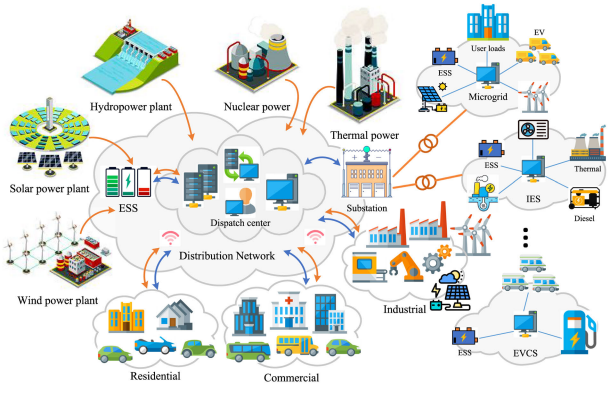


Fig. 12. ***Optimal dispatch issues of SG operation. This figure is cited from [76].***

On the one hand, DRL could provide better flexible control decisions to promote DN operations, including voltage regulation. For instance, Cao et al. [84] and Kou et al. [85] proposed a multiagent DDPG (MADDPG)-based approach for the DN voltage regulation with a high penetration of photovoltaics (PVs), which shows a better utilization of PV resources and control performance. A novel DRL algorithm named constrained SAC is proposed in [86] and [87] to solve Volt–Var control problems in a model-free manner. Comprehensive numerical studies demonstrate the efficiency and scalability of the proposed DRL algorithm, compared with state-of-the-art DRL and convectional optimization algorithms. Sun and Qiu [88] and Yang et al. [89] proposed a two-stage real-time Volt–Var control method, in which the model-based centralized optimization and the DQN algorithm are combined to mitigate the voltage violation of DN.

On the other hand, DRL algorithms are also applied to determine the optimal network configuration of DN. For example, Li et al. [90] developed a many-objective DNR model to assess the tradeoff relationship for better operations of DN, in which a DQN-assisted evolutionary algorithm (DQN-EA) is proposed to improve searching efficiency. Similarly, an online DNR scheme based on deep Q-learning is introduced in [91] to determine the optimal network topology. Simulation results indicate that the computation time of the proposed algorithm is low enough for practical applications. In addition, Gao et al. [92] developed a data-driven batch-constrained SAC algorithm for the dynamic DNR, which could learn the network reconfiguration control policy from historical datasets without interacting with the DN. In [93], the federated learning and AC algorithm are combined to solve the demand response problem in DN, which considers the privacy protection, uncertainties, as well as power flow constrains of DN simultaneously. In addition, a DRL framework based on A2C algorithm is proposed in [94], which aims at enhancing the long-term resilience of DN using hardening strategies. Simulation results show its effectiveness and

scalability in promoting the resilience of DN compared with traditional mathematical methods.

2) Microgrid Network: Microgrid is a local electric power system with DERs, ESS, and flexible loads [95]. Various objectives are proposed for the microgrid optimization dispatch, such as maximizing the operator's revenues, minimizing operational costs, promoting the users' satisfaction, reducing power delivery losses, increasing the RE utilization, and promoting the system stability. Indeed, decision variables of microgrid dispatch (35) mainly include the electricity price, generation allocation strategies, device availability, and operation state. Hence, for the mth microgrid

$$\min \sum_{t=1}^{T} \left(\sum_{k\in m} \pi_{\mathrm{dg}}\left(P_k^{\mathrm{dg}}(t)\right) + \eta_m \pi(t) P_m^{\mathrm{grid}}(t) + \eta_{\mathrm{ess}} \left|\mathrm{SOC}(t) - \mathrm{SOC}(t-1)\right| + \sum_{z=1}^{Z} \pi_m^z q_m^z(t) u_m^z(t) \right). \tag{35}$$

The first part of (35) represents the DER generation cost, in which $\pi_{\mathrm{dg}}(P_k^{\mathrm{dg}}(t))$ is the quadratic polynomial correlated to electricity generation P_k^{dg}. The second part is the energy cost of microgrid for purchasing electricity from the main grid where η_m denotes the loss coefficient of power delivery. $\pi(t)$ and $P_m^{\mathrm{grid}}(t)$ represent the electricity sale price and energy purchased from the main grid, respectively. The third part denotes the ESS degradation cost where η_{ess} is the degradation cost coefficient and SOC represents the ESS energy state. The last part of (35) is the internal cost of microgrid where π_m^z denotes the cost of the zth load block. In particular, q_m^z represents the zth load block and u_m^z is the binary variable denoting the load participation status in demand response.

To ensure the microgrid secure operation, the following constraints should be considered. For example, the energy stored at ESS during Δt should be equal to the difference of charged and discharged energy:

$$\mathrm{SOC}(t) = \mathrm{SOC}(t-1) + \eta_{\mathrm{ch}} P_{\mathrm{ch}}^{\mathrm{ESS}} \Delta t - \frac{P_{\mathrm{dc}}^{\mathrm{ESS}}}{\eta_{\mathrm{dc}}} \Delta t \tag{36}$$

where (36) describes the energy balance of ESS, η_{ch} and η_{dc} denote the charging and discharging coefficients of ESS, respectively, and $P_{\mathrm{ch}}^{\mathrm{ESS}}$ and $P_{\mathrm{dc}}^{\mathrm{ESS}}$ represent the energy amount of ESS charging and discharging, respectively. In addition, Δt is the duration of the tth time interval.

In microgrid operations, active power supply and demand should be balanced

$$P_m^{\mathrm{grid}}(t) + \sum_{k\in m} P_k^{\mathrm{dg}}(t) + P_{\mathrm{dc}}^{\mathrm{ESS}}(t) + \sum_{z=1}^{Z} q_m^z(t) u_m^z(t) = P_m^{\mathrm{load}}(t) + P_{\mathrm{ch}}^{\mathrm{ESS}}(t) \tag{37}$$

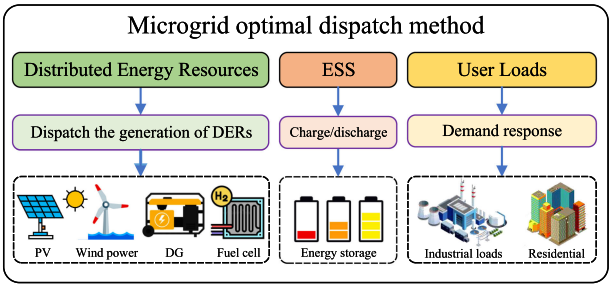


Fig. 13. ***Microgrid optimal dispatch method. This figure is cited from [76].***

where $P_m^{\mathrm{load}}(t)$ represents the power load of the mth microgrid.

The managed objects of microgrid optimal dispatch could be divided into three categories, i.e., DERs, ESS, and user loads, as shown in Fig. 13. The DER management, which consists of PV, wind power, DG, fuel cell, and alike, is primarily concerned with the DER generation dispatch [96]. As for ESS, the optimal management is achieved by charge/discharge controls, in order to coordinate the microgrid supply and demand [97]. In addition, demand response is widely applied to dispatch the user loads, which aims to reduce the operational cost and enhance the service reliability [98]. However, it is difficult to optimize the microgrid dispatch, considering the variability of RE and uncertain loads. What is more, the existence of high-dimensional variables and nonlinear constraints makes the solving trouble. To this end, DRL is introduced to solve the microgrid optimal dispatch problem in some literature, which achieves a higher computational efficiency as well as a better scalability, simultaneously. In the rest part of this topic, the detailed applications of DRL on microgrid are shown in Table 3.

a) Distributed energy resources: Typically, the reward function can be designed to maximize the RE utilization or minimize the energy cost. Dridi et al. [99] proposed a novel DRL approach based on deep LSTM for microgrid energy management through the generation dispatch of DERs and ESS, which shows better results compared to Q-learning. The A3C algorithm is introduced in [100] to dispatch energy while considering prevailing risks. Experiment results denote a higher accuracy for energy scheduling in the proposed risk-aware model than those of traditional methods. In addition, a finite-horizon DDPG (FH-DDPG)-based DRL algorithm is proposed in [101] for energy dispatch with DGs, PV panels, and ESS. The case study using isolated microgrid data shows that the proposed approach can offer efficient decisions even with partially available state information.

b) Energy storage system: In the context of ESS, the reward function can be based on costs associated with energy usage, such as electricity prices or battery degradation costs. Sanchez Gorostiza and Gonzalez-Longatt [102] introduced the DDPG algorithm to derive ESS energy

Table 3 Microgrid Network Optimal Dispatch Based on DRL

Ref.	Management objects	Solving Algorithm	Algorithm performance		
			Convergence	Privacy protection	Scalability
[99]	DERs, ESS	DRL-LSTM	✓		
[100]	DERs	A3C	✓		
[101]	DERs, ESS	FH-DDPG	✓		
[103]	ESS	MATD3	✓		✓
[102]	ESS	DDPG	✓		
[104]	DERs, ESS	A2C	✓	✓	
[106]	User loads	MCDRL	✓	✓	
[105]	DERs, user loads	PER-DDPG	✓		
[107]	User loads	A2C	✓	✓	✓

dispatch policies without fully observable state information. The proposed algorithm has derived an energy dispatch policy for ESS. A multiagent TD3 (MATD3) is developed in [103] for ESS energy management. Simulation results demonstrate its efficiency and scalability while handling high-dimensional problems with continuous action space. In addition, the curriculum learning is integrated into A2C to improve sample efficiency and accelerate the training process in [104], which speeds up the convergence during the DRL training and increases the overall profits.

c) User loads: The reward can be formulated as the reduction in peak load or the cost savings achieved through demand response. In [105], a prioritized experience replay DDPG (PER-DDPG) is applied to the microgrid dispatch model considering demand response. Simulation studies indicate its advantage in reducing operational costs compared with traditional dispatch methods. Du and Li [106] proposed an MCDRL approach for demand-side management, which tends to have a strong exploration capability and protect consumer privacy. In addition, the A2C algorithm is developed in [107] to address the demand response problem, which not only shows the superiority and flexibility of the proposed approach but also preserves customer privacy.

3) Electric Vehicles: The use of EVs has been growing rapidly across the globe, in particular within the past decade, which is mostly due to its low environmental impacts [108], [109], [110], [111]. Specifically, reducing the charging cost through dispatching the behaviors of charging and discharging is the hot spot of research. Due to the flexibility of EVs charging/discharging, some literature focuses on the coordinated dispatch of EVs and RE, which is devoted to promoting the utilization of RE by EVs. However, the uncertainty of RE and user loads results in the difficulty of model construction. At the same time, the proliferation of EVs makes it more difficult to optimize the solution of the operation problem, which is mostly due to the large number of variables.

On the one hand, traditional methods tend to estimate before optimization and decision-making while addressing the randomness of EV charging behaviors. On the other hand, multistage optimization is introduced to handle the problem caused by high-dimensional variables. Nevertheless, the optimization results of these methods are dependent on the predictive accuracy. Accordingly, DRL is applied to deal with the EV optimal dispatch problem, which is a data-driven method and to some extent insensitive to prediction accuracy. The reward can incorporate factors related to user comfort and convenience, such as the queuing time waiting for charging, the available range of EV travel, or the ability to meet specific user preferences. In the rest part of this section, we present the detailed DRL applications to the EV optimal dispatch, as shown in Table 4.

For instance, Zhang et al. [112] proposed a novel approach based on DQN to dispatch the EVs charging and recommend the appropriate traveling route for EVs. Simulation studies demonstrate its effectiveness in significantly reducing the charging time and origin–destination distance. In [113], a DRL approach with embedding and attention mechanism is developed to handle the EV routing problem with time windows. Numerical studies show that it is able to efficiently solve large-size problems, which are not solvable with other existing methods. In addition, a charging control DDPG algorithm is introduced to learn the optimal strategy for satisfying the requirements of users while minimizing the charging expense in [114]. The SAC algorithm is applied to deal with the congestion control problem in [115], which proves to outperform other decentralized feedback control algorithms in terms of fairness and utilization.

Taking the security into account, Li et al. [116] proposed a CPO approach based on the safe DRL to minimize the charging cost, which does not require any domain knowledge about the randomness. Numerical experiments demonstrate that this method could adequately satisfy the charging constraints and reduce the charging cost. A novel MADDPG algorithm for traffic light control is proposed to reduce the traffic congestion in [117]. Experimental results show that this method can significantly reduce congestion in various scenarios. Qian et al. [118] developed a multiagent DQN (MA-DQN) method to model the

Table 4 EV Optimal Dispatch Based on DRL

Ref.	Optimization targets	Solving Algorithm	Algorithm performance		
			Convergence	Privacy protection	Scalability
[112]	Reduce charging time	DQN	✓		✓
[113]	EV route selection	REINFORCE	✓		✓
[114]	Minimize charging expense	DDPG	✓		
[115]	Adaptive congestion control	SAC	✓	✓	
[116]	Minimize charging cost	CPO	✓		✓
[117]	Reduce traffic congestion	MADDPG	✓		✓
[118]	Determine the EVCS optimal price	MA-DQN	✓		✓
[119]	Minimize travel time and cost	DQN	✓		
[120]	Dispatch EV charging behavior	SAC	✓		✓

pricing game in the transportation network and determine the optimal charging price for electric vehicle charging station (EVCS). Case studies are conducted to verify the effectiveness and scalability of the proposed approach. In [119], a DQN-based EV charging navigation framework is proposed to minimize the total travel time and charging cost in the EVCS. Experimental results demonstrate the necessity of the coordination of SG with an intelligent transportation system. In addition, the continuous SAC algorithm is applied to crack the EV charging dispatch problem considering the dynamic user behaviors and electricity price in [120]. Simulation studies show that the proposed SAC-based approach could learn the dynamics of electricity price and driver's behavior in different locations.

4) Integrated Energy System: In order to solve the problem of sustainable supply of energy and environmental pollution, IES has attracted extensive attention all over the world. It regards the electric power system as the core platform and integrates RE at the source side and achieves the combined operation of cooling, heating, as well as electric power at the load side [121]. However, the high penetration of RE and flexible loads make the IES become a complicated dynamic system with strong uncertainty, which poses huge challenges to the secure and economic operation of IES. Moreover, conventional optimization methods often rely on accurate mathematical model and parameters, which are not suitable for IES optimal dispatch problem while considering strong randomness. Fortunately, DRL is introduced to address the IES optimal dispatch problem, which is a model-free method and achieves a series of successful applications. When applying DRL to IES, the design of the reward function can vary depending on the specific objectives and constraints of the system. In the rest part of this section, comprehensive reviews about DRL-based IES optimal dispatch are discussed as follows.

On the one hand, DRL methods are applied to cope with optimal dispatch problem of IES at the source side. For example, Yang et al. [122] proposed a DDPG-based dynamic energy dispatch method for IES while considering the uncertainty of renewable generation, electric load, as well as heat load. Numerical simulations on a typical day scenario demonstrate that the developed method avoids dependence on uncertainty knowledge and has a strong adaptability for inexperienced scenarios. In [123], a dynamic energy conversion and dispatch model for IES is developed based on DDPG, which takes the uncertainty of demand as well as the flexibility of wholesale prices into account. Case studies illustrate that the proposed algorithm can effectively improve the profit of system operator and smooth the fluctuations of user loads. Similarly, the optimal dispatch problem of IES with RE integrated is first formulated as a discrete MDP in [123], which is solved by the proposed DRL method based on PPO subsequently. Finally, simulation results show that this method can distinctly minimize the operation cost of IES. In addition, the IES economical optimization problem with wind power and power-to-gas technology is discussed in [124], which develops a cycling decay learning rate DDPG to obtain the optimal operation strategy. Zhang et al. [125] investigated the optimal energy management of IES considering solar power, diesel generation, and ESS, which introduces the PPO algorithm to solve the optimization problem and realizes 14.17% of cost reduction in comparison with other methods.

On the other hand, DRL approaches are also introduced to deal with the IES optimal dispatch problem at the load side [126]. For instance, Zhou et al. [127] established the constrained CHP dispatch problem as an MDP. Afterward, an improved policy gradient DRL algorithm named distributed PPO is developed to handle the CHP economic dispatch problem. Simulation results demonstrate that the proposed algorithm could cope with different operation scenarios while obtaining a better optimization performance than other methods. In [128], a DRL algorithm based on DQN is used to realize the dynamic selection of optimal subsidy price for IES with regenerative electric heating, which aims to maximize the load aggregator profits while promoting demand response. Numerical studies show that the power grid can save 56.6% of its investment and users save up to 8.7% of costs. In addition, a model-free and data-driven DRL method based on DDPG with prioritized experience replay strategy is proposed to

address the IES energy management problem in [129], which also illustrates its superior performance in reducing the energy cost. In addition, Li et al. [130] constructed a coordinated power dispatch framework, which is based on the MADDPG for combining imitation learning and curriculum learning simultaneously. Case studies verify the effectiveness of the proposed algorithm in the dispatch performance against renewable power fluctuations and stochastic loads.

In conclusion, this section reviews the applications of DRL for the optimal dispatch issues in SG operations, and the reviewed methods are summarized along with the references in Tables 2–4. It could be illustrated that the optimal dispatch problems in SG operations are usually established as an MDP. On this basis, RL could be utilized to deal with the model-free optimization problem with high uncertainty, while the curse of dimensionality is handled by the deep depth of NN. Therefore, DRL is capable of coping with such high-dimensional optimal dispatch problems with uncertainties, which achieves better performances than conventional methods. What is more, policy-based algorithms including both deterministic and stochastic methods receive more attention, compared with the value-based DRL approaches. In Section III-B, the adoption of DRL for the operational control of SG is discussed, which is confronted with more difficult challenges due to strict operating rules.

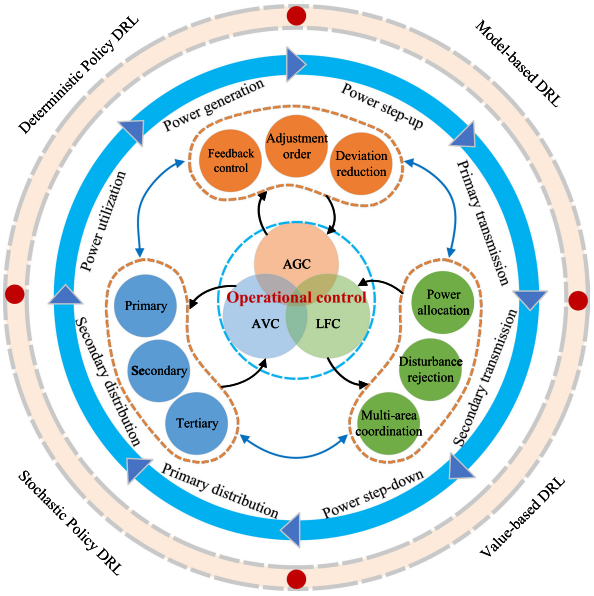


Fig. 14. ***Operational control issues in SG.***

B. Operational Control

The operational control of SG aims at maintaining its secure and stable operation, as shown in Fig. 14. However, it has become more complicated and challenging with the prevalence of RE. Indeed, the problem of voltage and frequency fluctuations induced by the ever-increasing penetration of stochastic and intermittent RE becomes more serious, which threatens the secure and stable operation of SG.

To this end, conventional operational control methods are proposed to maintain the balance on active power, voltage, and frequency stability, such as the AGC, AVC, and LFC. Unfortunately, traditional operation methods are not suitable for managing the variability of large-scale RE and loads in SG [131], [132], [133]. At present, DRL-based methods are widely applied in the field of operational control, which is due to their self-learning, self-optimization, and decision-making merits [134]. In the rest part of this section, detailed DRL applications to the operational control of SG are described as follows.

1) Automatic Generation Control: GC is used for adjusting the power output of multiple generators at different power plants, in response to changes in the load side. As the conventional AGC methods are not adequate to handle the strong uncertainty induced by the ever-increasing penetration of RE, DRL is applied to deal with the above problems. Accordingly, the reward is generally defined as the minus of total generation cost. For instance, Vijayshankar et al. [135] proposed a model-free DRL framework based on SARSA for real-time yaw control of wind farms to accurately track the power reference signal. Simulation studies indicate that a wind farm could achieve a better tracking performance with this control paradigm. In [136], an intelligent automatic control framework is proposed to address the coordination problems between AGC controllers in multiarea power systems, which adopts an imitation guided-exploration multiagent twin-delayed DDPG (IGE-MATD3) algorithm. As demonstrated by the simulation results, the intelligent AGC framework can improve dynamic control performance and reduce the regulation mileage payment in each area. In addition, Xi et al. [137] proposed a novel DRL algorithm (namely DPDPN) to allocate power order among the various generators, which combines RL and DNN to obtain the optimal coordinated control of source–grid–load. Experimental results demonstrate that the power system control performance and adaptability are improved using the proposed algorithm compared with conventional methods.

In [138], a multiple experience pool replay twin delayed deep deterministic policy gradient (MEPR-TD3) algorithm is proposed to handle AGC dispatch problem in the IES, which achieves the comprehensive optimum in control performance and economy profit. The control performance of AGC for wind power ramping based on DRL is investigated in [139], in which the DQN is used to AGC parameter fitting. Simulation results verify the feasibility and effectiveness of analyzing the relationship between AGC performance and wind power ramping, on the basis of the proposed AGC parameter fitting model. In addition, Li et al. [140] proposed a hierarchical multiagent deep deterministic policy gradient (HMA-DDPG) algorithm for AGC dispatch. Numerical analysis verifies that the

Table 5 SG AGC Based on DRL

Ref.	Fields	System	Optimization targets	Learning Algorithm	Agent
[135]	AGC	Wind farms	Real-time yaw control of wind farm	SARSA	Single-agent
[136]	AGC	Interconnected power grids	Reduce area regulation payment	IGE-MATD3	Multi-agent
[137]	AGC	Interconnected power grids	Power order optimal allocation	DPDPN	Multi-agent
[138]	AGC	Integrated energy system	Comprehensive optimum of system	MEPR-TD3	Single-agent
[139]	AGC	Integrated power grid	Reduce area control deviation	DQN	Single-agent
[140]	AGC	Interconnected power grid	Comprehensive optimum of system	HMA-DDPG	Multi-agent
[141]	AGC	Integrated smart grid	Improve control performance	SI-DDPG	Single-agent
[142]	AGC	Interconnected smart grid	Maintain system stability	TD3	Single-agent
[143]	AGC	Smart grid	Improve control performance	DFRL	Single-agent
[144]	AGC	Wind farm	Maximize revenue	Rainbow	Singe-agent
[145]	AGC	Integrated smart grid	Compensate power balance	Q-learning	Single-agent
[146]	AGC	Various generation units	Minimizing regulating error	Q-learning	Single-agent
[147]	AGC	Emission-free ship	Generation Management	DQN	Single-agent

sectional AGC dispatch based on HMA-DDPG can adjust the AGC unit outputs with the changes in system state, thus guaranteeing an optimal economic, secure, and stable SG operation.

In [141], a swarm intelligence-based DDPG (SIDDPG) algorithm is designed to acquire the control knowledge and implement a high-quality decision for AGC. Simulation results on a two-area SG validate the SI-DDPG effectiveness for improving the area control performance. In addition, a threshold solver based on TD3 is presented in [142] to dynamically update the thresholds of AGC, which is verified to be effective in maintaining the SG stability with a lower operation cost. A preventive strategy for the AGC application in SG operation is proposed in [143]. The strategy is on the basis of DFRL, which achieves the highest control performance compared with ten other conventional AGC methods. Yang et al. [144] presented a DRL model about wind farm AGC to maximize the revenue of wind power producers, which utilizes the rainbow algorithm to train the wind farm controller against uncertainties.

In [145], an intelligent controller based on Q-learning for the AGC application in the SG operation is proposed to compensate the power balance between generation against the load demand. Numerical simulations validate the feasibility of the SG controller with network-induced effects. In addition, a multiarea AGC scheme based on Q-learning is designed in [146] to dynamically allocate the AGC regulating commands among various AGC units. Comprehensive tests on practical data demonstrate the validation of the proposed method in minimizing the generation cost and regulating error. In addition, Hasanvand et al. [147] presented a reliable and optimal AGC method based on DQN to manage the generators in electric ship. Real-time simulation is conducted to verify the performance and efficacy of suggested AGC scheme for the electric ship.

2) Autonomous Voltage Control: Voltage control is another critical aspect of SG operational control, which can maintain bus voltage magnitudes within a desirable range. Although most of the existing model-based AVC methods could mitigate voltage violations, they are significantly dependent on the accurate SG knowledge data, which is often difficult to acquire in real time. Thus, the use of DRL allows controllers to learn the control strategy through interactions with a system-like simulation model, where the reward is defined as a penalty for the voltage deviation from its nominal value. Wang et al. [148] proposed a multiagent AVC algorithm based on MADDPG to mitigate voltage fluctuations, which could learn gradually and master the system operation rules by input and output data.

More specifically, MADDPG utilizes a centralized training approach with decentralized execution, as presented in Fig. 15. During the training phase, MADDPG agents employ a centralized critic network that observes all agents' actions to estimate the value function. This enables them to learn coordination and collaboration among agents. However, during the execution phase, each agent acts independently based on its own observations and makes decisions based solely on its own observations. This decentralized execution allows agents to interact with the environment and make decisions autonomously, without a

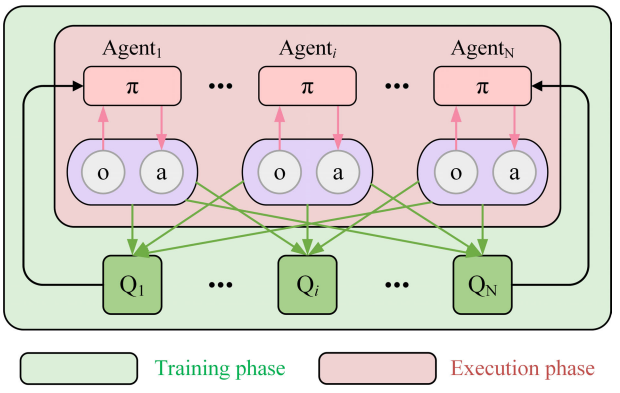


Fig. 15. ***Illustration diagram of MADDPG.***

Table 6 SG AVC Based on DRL

Ref.	Fields	System	Optimization targets	Learning Algorithm	Agent
[148]	AVC	Power grid	Mitigate voltage fluctuations	MADDPG	Multi-agent
[149]	AVC	Power grid	Autonomous voltage control	DQN and DDPG	Single-agent
[150]	AVC	Smart grid	Fast voltage regulation	DDPG	Single-agent
[151]	AVC	Buck converters	Enhance converter's performance	DQN	Single-agent
[152]	AVC	Unbalanced LV networks	Minimize voltage deviations	SAC	Single-agent
[153]	AVC	Decentralized smart grid	Fast voltage regulation	MATD3	Multi-agent
[154]	AVC	Multi-area smart grid	Ensure voltage stability	ARS	Multi-agent
[155]	AVC	Smart grid	Maintain voltage stability	PARS	Single-agent
[156]	AVC	Interconnected Smart grid	Mitigate voltage fluctuations	GC-DDQN	Single-agent
[157]	AVC	Distributed smart grid	Decentralized Voltage control	MASAC	Multi-agent
[158]	AVC	Active distribution network	Reduce voltage violation	SAC	Single-agent
[159]	AVC	Dynamic smart grid	Fast voltage stabilization	DMRL	Single-agent

need for explicit coordination with other agents. In [149], a DRL-based AVC scheme is developed for autonomous grid operation, which takes control actions to ensure secure SG operations for various randomly generated operating conditions. Numerical studies on a realistic 200-bus test system demonstrate the effectiveness and promising performance of the proposed method. In addition, a physical-model-free AVC approach based on DDPG is presented in [150], which can cope with fast voltage fluctuations. A model-free DRL control strategy based on DQN is proposed in [151], which aims to enhance the bus voltage regulation performance of converters.

The comparison of simulation results indicates the efficiency of the proposed control strategy for managing large signal perturbations. Wang et al. [152] proposed a novel DRL-based voltage regulation scheme for unbalanced low-voltage DNs, which is devoted to minimizing the expected total daily voltage regulation cost while satisfying operational constraints. An attention-enabled MATD3 algorithm is designed in [153] for decentralized AVCs, which is demonstrated to be effective in dealing with uncertainties, reducing communication requirements, and achieving fast decision-making processes. In addition, a novel hierarchical DRL, referred to as the ARS algorithm, is proposed in [154] where the lower level DRL agents are trained in an areawise decentralized manner, and the higher level agent is trained to coordinate the actions executed by lower level agents. Numerical experiments verify the advantages and various intricacies of the hierarchical method applied to the IEEE 39-bus power system. Huang et al. [155] formulated a derivative-free PARS algorithm for AVC via load shedding, which can overcome the control problems of existing DRL algorithms, including computational inefficiency and poor scalability. Simulation results illustrate that the proposed method offers better computational efficiency, more robustness in learning, excellent scalability, and better generalization capacity, compared with other approaches.

In [156], a DDQN framework, which applies the graph convolutional network (GCN), referred to as GC-DDQN, is proposed to tackle topology changes in the AVC problem, where the GCN model assists the DRL algorithm to better capture topology changes and spatial correlations in nodal features. A model-free centralized training and decentralized execution multiagent SAC (MASAC) framework is designed in [157] for AVC with high penetration of PVs. Comparative simulation studies demonstrate the superiority of the proposed approach in reducing the communication requirements. Nguyen and Choi [158] presented a three-stage AVC framework in SG using the online safe SAC method to reduce voltage violations, mitigate peak loads, and manage active power losses by coordinating the three stages with different control timescales. Numerical simulations for the IEEE 123-bus system demonstrate the high efficiency and safety of the presented method for regulating voltages. In addition, a novel deep meta-reinforcement learning (DMRL) algorithm is developed in [159], which combines the meta-strategy optimization with PARS to maintain voltage stability. Experimental results show that the performance of the proposed method surpasses those of state-of-the-art DRL and model predictive control approaches.

3) Load Frequency Control: LFC is also a complicated decision-making problem in SG applications. To this end, DRL is introduced for restoring the frequency and tie-line power flows to their nominal values after disturbances. Therefore, the reward could be defined as negative frequency and tie-line flow deviations. A novel control strategy for distributed LFCs is developed in [160], which is based on the multiagent DDQN with action discovery (DDQN-AD) algorithm. The approach shows a faster convergence speed and stronger learning ability compared with other traditional methods. In [161], a TDAC control strategy is proposed for LFC to deal with strong random disturbances caused by RE. Simulation studies show that TDAC has an excellent exploratory stability and learning capability, which improves the power system dynamic performance and achieves the regional optimal coordinated control. In addition, a multistep unified RL method is proposed in [162] for managing the LFC in multiarea interconnected power grid, which proves to outperform

Table 7 SG LFC Based on DRL

Ref.	Fields	System	Optimization targets	Learning Algorithm	Agent
[160]	LFC	Integrated Energy system	Optimal coordinated control	DDQN-AD	Multi-agent
[161]	LFC	Smart grid	Reduce control bias	TDAC	Single-agent
[162]	LFC	Interconnected power grid	Load frequency control	Double Q-learning	Multi-agent
[163]	LFC	Smart grid	Minimize frequency deviation	DRL	Multi-agent
[164]	LFC	Multi-area smart grid	Optimal coordinated control	MADDPG	Multi-agent
[165]	LFC	Fuzzy microgrids	Frequency stabilization	DDPG	Single-agent
[166]	LFC	Multi-unit smart grid	Enhance control performance	DDPG	Single-agent
[163]	LFC	Island microgrids	Distributed frequency control	MA-QDRL	Multi-agent
[167]	LFC	Smart grid with ESS	Support frequency control	DDPG	Single-agent
[168]	LFC	Multiple smart grids	Damp oscillation	DDPG	Single-agent

other traditional algorithms in terms of convergence and dynamic performance. Yan and Xu [163] developed a data-driven LFC method based on DRL in the continuous action domain for minimizing the frequency deviations under uncertainties. Numerical simulations verify the effectiveness and advantages of the proposed method over other existing approaches.

A data-driven cooperative approach for LFC, which is based on MADDPG in a multiarea SG, is presented in [164]. The approach offers optimal coordinated control strategies for LFC controller via centralized learning and decentralized implementation. Experimental results for a three-area SG demonstrate that the proposed algorithm can effectively minimize control errors against stochastic frequency variations. Khooban and Gheisarnejad [165] considered the DDPG to generate the supplementary control action for LFC, which is appraised for its systematic feasibility and applicability. In addition, a novel model-free LFC scheme is presented in [166], which adopts DDPG to learn the near-optimal strategies under various scenarios. Numerical simulations on benchmark systems verify the effectiveness of the proposed scheme in achieving satisfactory control performances. Yan et al. [169] developed a data-driven algorithm for distributed frequency control of island microgrids based on multiagent quantum DRL (MAQDRL). Numerical tests illustrate that the designed method can effectively regulate the frequency with better time delay tolerance. In [167], a DDPG-based data-driven approach for optimal control of ESS is proposed to support LFC. Simulation results in a three-area SG demonstrate the effectiveness of the proposed approach in supporting frequency regulation. In addition, the DDPG algorithm is combined with sensitivity analysis theory in [168], in order to learn the sparse coordinated LFC policy of multiple power grids. Numerical experiments verify that the proposed approach can obtain better performance of damping oscillation and robustness against wind power uncertainty.

To conclude, this section reviews the applications of DRL for the operational control of SGs, which are inherently coupled with generation adjustment, voltage regulation, frequency stabilization, and so on. The reviewed methods are summarized along with adequate references in Tables 5–7. It could be observed that the most widely used DRL framework for the operational control of SG is centralized, while the decentralized manner is an irresistible trend with the prevalence of distributed generation. What is more, CNN is the most popular network architecture for aforementioned DRL algorithms to extract features, while the novel GCN is gradually applied to capture the topology information of SG that typical CNN cannot complete. Despite the successful applications of DRL in operational control, they are still deemed to be computationally inefficient and offer poor scalability to a certain extent, according to the statements of some related literature. To this end, SG calls for more advanced DRL frameworks to support its secure and stable operation via offering robust strategy for its operational control. In Section III-C, the DRL adoption in SG markets is discussed, which involves multiple entities and complex relationships.

C. Electricity Market

The reforming of electricity power market has drawn much attention during the progressively undergoing restructuring of modern power systems. The emerging electricity market is regarded as the potential solution for improving the power system efficiency and optimizing SG operations [170]. In this situation, electricity retailers have appeared in various liberalized electricity markets, as the intermediary between electricity power producers and consumers. However, the electricity market with retailers contains increasing uncertainties and complexities in both supply and retail sectors, which is a challenge that affects the decision of participants. Indeed, the decision-making progress of electricity market is extremely complicated, as shown in Fig. 16, which mainly consists of energy bidding and retail pricing strategies [171]. On the one hand, the energy bidding process is a vital decision-making step for suppliers, which requires generality in different situations. On the other hand, the retail pricing strategy is the core challenge for retailers to promote profitability, which should have the adaptability to cope with a dynamic and complex environment.

Accordingly, conventional methods are proposed to promote the implement of electricity market, such as the

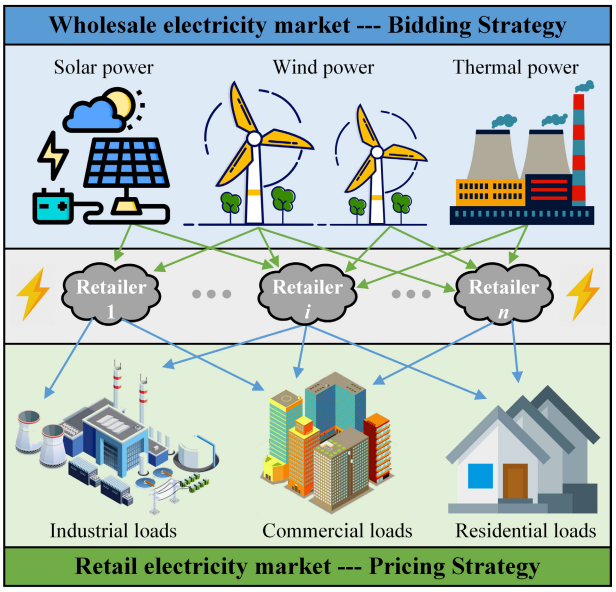


Fig. 16. *Hierarchical electricity market model.*

equilibrium model, bilevel optimization, and game theory. Nevertheless, the traditional methods are not suitable to handle the decision-making problem under strong uncertainties [172]. Fortunately, the development of digitization in SG offers an application of data-driven algorithms, especially the DRL approach. At present, DRL is gradually becoming an effective tool for electricity participants to make decisions during the execution of energy transactions. In the rest part of this section, detailed applications of DRL on electricity power market are discussed as follows.

1) Optimal Bidding Strategy: Energy bidding is the critical step during the decision-making progress in a wholesale electricity market. Although some methods have been developed to address the optimal bidding strategy, the strategy still encounters significant data acquisition and decision-making challenges. The DRL approach can provide an opportunity to handle this data-driven concern. Accordingly, the reward is defined based on payoffs obtained from successful bids. The energy bidding is formulated in [173] as an MDP with continuous state and action spaces, which is solved by DDPG for calculating the optimal bidding policy.

Du et al. [174] developed a model-free and data-driven approach, referred to as MADDPG, to approximate the Nash equilibrium in an electricity market with incomplete information. Simulation results demonstrate that the proposed algorithm could find a superior bidding strategy for all market participants with increased profit gains, compared with conventional bidding methods. A coordinated bidding and operation model, which is based on the DRL algorithm, is proposed in [175] and [176] to improve the real-time wholesale electricity market benefits for wind farms, which considers the cooperation of wind power bidding and ESS operation with uncertainty. The case study illustrates that the bidding policy learned by the DRL method can effectively improve the wind farm benefit while ensuring robustness. In addition, the DDPG algorithm is proposed in [177] to model generation bidding strategies, which is verified to be more accurate than those of the traditional RL algorithms and can converge to the Nash equilibrium even in an incomplete information environment.

In addition, Guo et al. [178] developed a data-driven bidding objective function identification framework with three procedures. First, the bidding decision process is formulated for participants as a standard MDP. Then, a DIRL method, which is based on maximum entropy, is introduced to identify individual reward functions. Finally, the DQN method is customized to simulate the individual bidding behaviors based on the obtained objective functions. The effectiveness and feasibility of the proposed framework and methods are verified by the real data from the Australian power market. A model-based DRL called MB-A3C is presented for the strategic energy bidding issues of wind farms in [179], by which generated policies are verified to be less cost than those provided by both previous model-based and model-free approaches. Also, Tao et al. [180] proposed a bidding strategy for EV aggregators, which is based on the data analytic and DDPG algorithm. Compared with the stochastic strategy, the profit of the proposed approach is 63.3% higher than that of the random strategy.

In addition, the SAC method is utilized in [181] to learn the optimal bidding strategy in a complex SG with incomplete information. Comprehensive simulations are conducted to verify the effectiveness of the proposed method in balancing supply and demand across the SG in a distributed manner. An asynchronous version of the fit Q iteration algorithm is proposed for continuous intraday market bidding in [182], which is compared against a number of benchmark policies and outperforms them on average 5%. Zhang et al. [183] considered a multi-DQN-based bidding strategy for EV owners to formulate the optimal bidding strategy and maximize the economic benefits, in which a target network and a value evaluation network are proposed for each agent to learn the optimal bidding strategy. Extensive experimental results demonstrate that the proposed strategy can achieve better economic benefits and assist EV owners spend less time on charging in comparison to those of Q-learning-based methods. In addition, an electricity–gas joint bilateral bidding energy market model is constructed in [184], which employs the improved PPO algorithm to learn the optimal bidding policy. Comparative simulations illustrate that the innovative market model as well as the PPO method is able to obtain the multienergy collaborative optimization and improve the energy utilization efficiency.

2) Optimal Pricing Strategy: Since the conventional pricing mechanism suffers from prevailing shortcomings to

deal with dynamics and uncertainties [185], DRL has been adopted to offer a pricing strategy for electricity retailers. Accordingly, the reward could be defined as the market agent's total revenue for selling electricity. Liu et al. [186] established a quarter-hourly dynamic time-sharing pricing model, which is based on the DDPG algorithm. It considers various market factors such as peak-valley time-of-use tariff, demand response, and balanced market deviations. Simulation results show that the proposed scheme with a higher daily pricing frequency could guide the users' charging behavior more effectively, tapping to a greater extent into the retail electricity market's economic potentials and clamping the load fluctuations in power grids. In [187], a novel DRL method is proposed to solve the EV pricing problem, which combines DDPG principles with a prioritized experience replay strategy. Numerical studies demonstrate that the proposed approach outperforms state-of-the-art RL methods in terms of both the solution optimality and computational requirements. The optimal retail pricing problem is formulated as an MDP in [188], which is solved by the proposed DDPG method with a shared LSTM-based representation network. As indicated by simulation results, the proposed framework enhances the perception capacity, further improves the optimization performance, and provides a more profitable pricing strategy.

In addition, Lee and Choi [189] developed a pricing strategy for EVCSs based on the privacy-preserving distributed SAC algorithm to maximize the benefits of EVCSs integrated with PV and ESS. Numerical experiments show that the proposed method outperforms in terms of convergence, sensitivity, and adaptability. In [190], a differentiated pricing mechanism based on TD3 algorithm is proposed to motivate EV users to avoid the over-utilization. Simulation studies demonstrate that the designed pricing scheme can maximize the utilization of charging facility while ensuring the satisfaction of service quality. A DRL-based pricing strategy of energy aggregator for profit maximization is presented in [191], which considers the behavior of opponents, uncertainty of RE, and varying bounds of the charging and discharging events in an unstable environment.

In addition, Lu et al. [192] proposed a Q-learning-based decision system for assisting the selection of electricity pricing plans, which extracts the hidden features behind the time-varying pricing plans from the continuous high-dimensional state space. Experimental studies demonstrate that the developed decision model can establish a precise predictive policy for individual users, effectively decreasing their energy consumption dissatisfaction and cost. In [193], a novel online pricing strategy is formulated to prevent power outages, which adopts the multiagent DRL (MADRL) to control EV charging demands and support the grid stability and the EVCS profitability. Extensive simulation studies illustrate the significant improvement in the robustness and effectiveness of the developed solution in terms of revenue and energy saving.

Fig. 17. *Typical architecture diagram of P2P energy trading market.*

A real-time pricing strategy for multienergy generation system is discussed in [194], which integrates a distributed online MARL algorithm to solve the MDP model without acquisition of the transition probabilities. As demonstrated by simulation, the proposed pricing approach shows a good performance in ensuring the revenue of both supply and demand sides.

3) P2P Energy Trading: The abovementioned works are mainly based on the centralized collaborative method, which suffers from certain shortcomings, including privacy leakage and low efficiency [195], [196], [197]. With respect to these challenges, P2P energy trading is considered as the potential solution for distributed and flexible control of energy flow among peers, which allows direct energy transactions between producers and consumers [198]. The P2P energy trading market participants are described as prosumers, who can buy and sell electricity to achieve win–win market transactions [199], as shown in Fig. 17. Therefore, P2P participants could gain more revenues for trading energy in an electricity market.

However, the decentralized electricity market involves complex decision-making processes with high-dimensional data and uncertainties, which is difficult to solve by traditional methods. With the development of AI, DRL is gradually applied in the P2P energy trading market due to its scalability and privacy protection features. One possible design is to reward agents for maximizing their individual profits or minimizing their costs. The P2P transactions in an electricity market are established as an MDP in [200] and solved by the SARSA algorithm with average discrete processing. The case study of a community with multiple users is conducted to verify the DRL's effectiveness, economy, and security for solving the P2P energy trading problem. Liu et al. [201] introduced the DQN for autonomous agents in the consumer-centric electricity market, which considers both local energy priority transactions and public shared energy facilities. Numerical studies verify that the proposed data-driven method can handle the P2P decision-making problem and promote the benefits of the whole community in electricity markets.

In addition, the P2P energy trading problem in a community market with many participating households is investigated in [202], which accounts for heterogeneity with respect to their DER portfolios. In order to address this problem, a novel DRL algorithm named MADDPG with parameter sharing (MADDPG-PS) is proposed in this article, which achieves a significant operating cost and peak demand benefits. Samende et al. [203] presented an MADDPG-based algorithm for P2P electricity trading considering SG constraints. It minimizes the energy costs of prosumers who are participating in the P2P market. Numerical experiments on real-world datasets indicate that the proposed algorithm can reduce the energy cost while satisfying network constraints.

In [204], a distributed DQN-based method is developed to manage the energy trading between multiple virtual power plants through P2P and utility. Simulation results show that the designed method can adjust its action according to the available energy demand and uncertain environment adaptively. An improved MADDPG method-based double-side auction market is formulated in [205], in order to address the automated P2P energy trading problem among multiple consumers and prosumers. Case studies demonstrate that the proposed algorithm can promote the economic benefits of prosumers in P2P energy trading. In addition, Zhang et al. [206] developed an MADDPG-based P2P energy trading model among microgrids to improve the resource utilization and operational economy. Simulation results illustrate that the designed algorithm could reduce the operation cost of each microgrid by 0.09%–8.02%, compared to baselines.

Taking the privacy concern of P2P trading into account, Ye et al. [207] proposed a scalable and privacy-preserving P2P energy trading scheme based on the MAAC algorithm. Simulation studies, including a real-world, large-scale scenario with 300 residential participants demonstrate that the proposed approach significantly outperforms the state-of-the-art MADRL algorithms in reducing the operation cost and peak demand. In addition, Wang et al. [208] provided a novel hybrid community P2P market framework for multienergy systems, where a data-driven market surrogate model-enabled DRL method is proposed to facilitate P2P transaction within constraints. Specifically, a market surrogate model based on deep belief network is developed to characterize P2P transaction behaviors of peers in the market without disclosing their private data. In addition, an MADDPG-based energy trading algorithm is developed in [209] to formulate the optimal policy for each microgrid in the electricity market. Moreover, blockchain is adopted to guarantee the privacy of energy transaction data.

In summary, this section reviews the DRL applications in and SG electricity market, which mainly involves three actions, i.e., bidding, pricing, and P2P trading. DRL offers an effective tool for market participants to make optimal decisions, even without using the complete information about electricity market. These approaches are summarized along with the references in Table 8. It is illustrated that most DRL framework for electricity market only contains single agent, while the MADRL indicates a promising prospect with the development of decentralized electricity market, e.g., P2P trading market. First, the prevalence of distributed electricity market calls for DRL algorithms with multiple agents, in which each agent is responsible for a local market. Second, the increasing concern about privacy leakage starves for MADRL approaches where multiple agents cooperatively train the model without the need of sharing datasets. Moreover, the policy-based DRL methods are adopted extensively in SG electricity market operations that are compared with the value-based one, due to the complexity in both supply and retail sectors. In Section III-D, we conduct a discussion on DRL applications in SG operations that will highlight future research trends.

D. Emerging Areas

In recent years, industry has witnessed the SG digitization and modernization via the numerous deployments of advanced metering infrastructures. On this basis, SG will maintain secure, economic, and sustainable operations, compared with those in traditional power systems. Meanwhile, the widespread popularity of smart meters and RE also brings about some emerging issues that conventional power systems have seldom encountered, including network security and privacy concerns. Since these problems are rather new in SG operations, typical methods may not cope with them in an effective manner. To this end, the data-driven DRL approaches are introduced in these emerging areas to assist SGs in tackling the aforementioned issues. In the rest part of this section, detailed applications of DRL on network security and privacy preservation are depicted as follows.

1) Network Security: With rapid SG developments in active DNs, various sensing, communication, and control devices are deployed to maintain a secure SG operation. However, these cyber-physical components have also expanded the landscape of cyber threat, which have further resulted in SG vulnerabilities to malicious cyberattacks [210], [211], [212]. Even though regular defense strategies, such as intrusion prevention systems and firewalls, are provided in SGs, such methods might not be very effective while facing the many unknown vulnerabilities [213]. To this end, DRL is applied in SGs to offer additional defense strategies for mitigating the blackout risks during cyberattacks. Accordingly, the reward should be designed to incentive actions that enhance network security and discourage actions that compromise it. For example, DRL is applied to assist SG operators in counteracting malicious cyberattacks in [214], which investigates the possibility of defending SG using a DQN agent. Simulation results not only demonstrate the effectiveness of the proposed DQN algorithm but also pave the way for defending the SG under a sophisticated cyberattack.

Liu et al. [215] proposed a cybersecurity assessment approach based on DQN to determine the optimal

Table 8 Applications of DRL on the Wholesale, Retail, and P2P Electricity Markets

Ref.	Fields	System	Optimization targets	Learning Algorithm	Agent
[173]	Bidding	Wholesale electricity market	Optimize bid response function	DDPG	Single-agent
[174]	Bidding	Generation company	Enhance power selling profit	MADDPG	Multi-agent
[175]	Bidding	Wind farms	Promote benefits of wind farms	SAC	Single-agent
[176]	Bidding	Wind farms	Maximize benefits of wind farms	SAC	Single-agent
[177]	Bidding	Generation company	Solve Nash equilibrium	DDPG	Single-agent
[178]	Bidding	Australian electricity market	Improve individual utility	DIRL	Single-agent
[179]	Bidding	Wind farms	Profits optimization	MB-A3C	Multi-agent
[180]	Bidding	EV aggregators	Profits optimization	DDPG	Single-agent
[181]	Bidding	Complex smart grid	Balance supply and demand	SAC	Single-agent
[182]	Bidding	European Intraday market	Profits maximization	Fitted Q iteration	Single-agent
[183]	Bidding	smart grid with EVs	Benefits maximization	Multi-DQN	Multi-agent
[184]	Bidding	Integrated energy system	Improve energy utilization	PPO	Single-agent
[186]	Pricing	Retail electricity market	Enhance economic potential	DDPG	Single-agent
[187]	Pricing	Aggregators and EVs	Reduce electricity cost of EV	DDPG with PER	Single-agent
[188]	Pricing	Electricity market	Promote long-term profits	DDPG-LSTM-S	Single-agent
[189]	Pricing	EVCSs	Benefits maximization	SAC	Single-agent
[190]	Pricing	EVCSs	Promote energy utilization	TD3	Single-agent
[191]	Pricing	Energy aggregator	Minimize operation cost	DDPG	Single-agent
[192]	Pricing	Electricity market	Decrease user dissatisfaction	Q-learning	Single-agent
[193]	Pricing	EVCSs	Adjust EV charging demand	MADRL	Multi-agent
[194]	Pricing	Multi-energy system	Ensure system revenue	MARL	Multi-agent
[200]	P2P	Multi-energy microgrids	Average operation cost	MAAC	Multi-agent
[201]	P2P	User-side energy system	Whole community's benefit	SARSA	Single-agent
[202]	P2P	Community electricity market	Operating cost	MADDPG-PS	Multi-agent
[203]	P2P	P2P electricity market	Energy cost	MADDPG	Multi-agent
[204]	P2P	Multiple Virtual power plant	Reduce operation cost	DQN	Single-agent
[205]	P2P	DA electricity market	Promote economic benefits	MADDPG	Multi-agent
[206]	P2P	Energy trading market	Improve resource utilization	MADDPG	Multi-agent
[207]	P2P	Privacy-preserving P2P market	Peak demand	MAAC	Multi-agent
[208]	P2P	Multi-energy systems	Energy cost	DDPG	Single-agent
[209]	P2P	Multi-microgrids	Utility of microgrid	MADDPG	Multi-agent

attack transition policy. Numerical and real-time simulation experiments verify the performance of developed algorithm without the need for full observation of system. A DQN-based DRL algorithm is developed in [216] for the low-latency detection of cyberattacks in SGs, which aims at minimizing the detection delay while maintaining a high accuracy. Case studies verify that the DQN-based algorithm could achieve very low detection delays while ensuring a good performance. In addition, a DRL-based approach is proposed in [217] to detect data integrity attacks, which checks whether the system is currently under attack by introducing LSTM to extract state features of previous time steps. Simulation studies illustrate that the proposed detection approach outperforms the benchmarked metrics, including the delay error rate and false rate.

Moreover, Chen et al. [218] proposed the model-free defense strategy for SG secondary frequency control with the help of DRL, which proves to be effective through validation based on the IEEE benchmark systems. In [219], an MADDPG algorithm is proposed for SSA, which integrates the DRL and edge computing to conduct efficient SSA deployment in SGs. In addition, a comprehensive risk assessment model of excessive traffic concentration in an SG is established in [220], which considers the link delay and load balancing simultaneously. Then, a DQN-based route planning algorithm is designed to find the optimal route, which not only meets the delay requirements but also enhances the resistivity of SG. To address the ever-increasing FDI attack in SG, Zhang et al. [221] proposed a resilient optimal defensive strategy with distributed DRL method, which devotes itself to correcting false price information and making the optimal recovery strategy for fighting against the FDI attack. Numerical studies reveal that the distributed DRL algorithm provides a promising way for the optimal SG defense against cyberattacks. In [222], a DQN detection scheme is presented to defend against data integrity attacks in SG. Experimental results demonstrate that the developed method surpasses the existing DRL-based detection scheme in terms of accuracy and rapidity. In addition, an MADRL with prioritized experience replay algorithm is proposed to identify the critical lines under coordinated multistage cyberattacks, which contributes to deploying the limited defense resources optimally and mitigating the impact of cyberattacks.

2) Privacy Preservation: To an extent, the widespread deployment of advanced meters in SG has also raised serious concerns from the privacy perspective, which is

regarded as one of the main oppositions for SG modernization [223]. In fact, the fine-grained smart meter data carries sensitive information about consumers, posing a potential threat for preserving privacy. Traditional methods have been proposed for privacy preservation in SGs, such as data aggregation and encryption [224], data downsampling [225], and random noise addition [226]. However, these approaches may restrict the potential applications of SG data in an uncontrolled manner, e.g., time delay of fault detection and degradation of detection precision. In this regard, DRL is introduced to provide the optimal operational strategy while ensuring the privacy security of consumers.

When applying DRL to privacy protection in power systems, the design of the reward function can vary depending on the specific goals and requirements. For instance, Lee and Choi [227] proposed a privacy-preserving method based on federated RL for the energy management of smart homes with PV and ESS. It develops a novel distributed A2C model that contains a global server and a local home energy management system. First, A2C agents for local energy management systems construct and upload their models to the global server. After that, the global server aggregates the local models to update a global model and broadcasts it to the A2C agents. Finally, the A2C agents replace the previous local models with the global one and reconstruct their local models, iteratively. In this way, data sharing between local systems is prevented, thus preserving SG privacy. In [228], a distributed DRL algorithm is employed for devising the intelligent management of household power consumption. More specifically, the interactions of SGs and household appliances are established as a noncooperative game problem, which is addressed by the DPG algorithm considering privacy protection. In addition, a privacy-aware smart meter framework is investigated in [229] that utilizes the battery to hide the actual power consumption of a household. In detail, the problem of searching the optimal charging/discharging policy for reducing information leakage with minimal additional energy cost is formulated as an MDP, which is handled by the DDQN with mutual information. As demonstrated by simulation studies, the performance of developed algorithm achieves significant improvements over the state-of-the-art privacy-aware demand shaping approaches.

In [230], a novel federated learning framework is presented for privacy-preserving and communication-efficient energy data analysis in SG. On this basis, a DQN-based incentive algorithm with two layers is devised to offer optimal operational strategies. Extensive simulations validate that the designed scheme can significantly stimulate high-quality data sharing while ensuring preserving privacy. Wang et al. [231] proposed a data privacy-aware routing algorithm based on DDPG for communication issues in SGs, to realize the latency reduction and load balancing. Experimental results show that the formulated privacy-aware routing protocol can effectively reduce the latency while maintaining excellent load balancing. A privacy-preserving Q-learning framework for the SG energy management is formulated in [232], which is verified to be effective in energy management without privacy leakage. In addition, Zhang et al. [233] developed an intelligent demand response resource trading framework, in which the dueling DQN is constructed to simulate the bilevel Stackelberg game in a privacy-protecting way. Numerical experiments demonstrate that the designed approach has an outstanding performance in reducing energy cost as well as preserving privacy.

Liu et al. [234] presented a battery-based intermittently differential privacy scheme to realize privacy protection. Afterward, it develops a DDPG-based algorithm to offer the optimal battery control policy, in order to maintain the battery power level and realize cost saving. Case studies illustrate that the proposed method has a better performance in both cost saving and privacy preservation. A DQN-based technique is applied in [235] to keep the balance between privacy protection and knowledge discovery during SG data analysis. In [236], a hierarchical SAC-based energy trading scheme is presented in electricity markets, by which the prosumers' privacy concerns are tackled because the training process would only require the local observations. Extensive simulations validate that the proposed algorithm can effectively reduce the daily cost of prosumers without privacy leakage. In addition, a DDPG-based energy management approach is developed in [237] for integration in SG systems, which addresses the privacy issues via local data executions. Experimental results demonstrate that the proposed scheme can achieve good performances while preserving the data privacy.

To conclude, this section reviews DRL applications to SG network security and privacy preservation. These methods are summarized in Table 9 along with the corresponding references. It is observed that the value-based DQN is the most popular DRL algorithm for managing network security, while policy-based DRL methods proposed for privacy preservation include both deterministic and stochastic policies. Furthermore, decentralized DRL frameworks for handling emerging SG issues are paid more attention than other architectures, which is due to additional requirements for maintaining security and privacy. However, DRL applications are relatively inadequate for managing the SG emerging issues, which call for more investigation and exploration in the future. Although there have been numerous literature studies on DRL applications in SGs, many critical problems would still need to be addressed before their practical implementations. On the one hand, DRL applications to SG systems are still relatively new and require further research before maturity. On the other hand, it is necessary to reassess the DRL advantages and limitations in SG applications, which are among the most complex and critically engineered systems in the world. Although real-world DRL applications in SG operations are relatively limited, this technology holds great potential for encountering SG applications, particularly in tackling complex decision-making and control problems. Therefore,

Table 9 Applications of DRL on the Emerging Issues in SG

Ref.	Fields	System	Optimization targets	Algorithm	Agent
[214]	Security	Cyber attack counteraction	Defend cyber attacks	DQN	Single-agent
[215]	Security	Cyber security assessment	Optimize attack transition policy	DQN	Single-agent
[216]	Security	Low latency detection	Minimize detection delay	DQN	Single-agent
[217]	Security	Integrity attack detection	Reduce latency detection	DQN	Single-agent
[218]	Security	Frequency attack defense	Secondary frequency control	DQN	Single-agent
[219]	Security	Security situational awareness	Direct SSA deployment	MADDPG	Multi-agent
[220]	Security	smart grid risk assessment	Search optimal route	DQN	Single-agent
[221]	Security	False data injection prevention	Correct false price data	DDPG	Single-agent
[222]	Security	Data integrity attack defense	Offer detection scheme	DQN	Single-agent
[227]	Privacy	Smart homes with PV and ESS	Optimize energy management	A2C	Multi-agent
[228]	Privacy	Household energy management system	Manage power consumption	DPG	Single-agent
[229]	Privacy	Household energy system with battery	Reduce information leakage	DDQN	Single-agent
[230]	Privacy	Smart grid energy data owners	Optimize operational strategy	DQN	Single-agent
[231]	Privacy	Smart grid communication system	Latency reduction & load balance	DDPG	Single-agent
[232]	Privacy	Smart grid energy management	Reduce operational cost	Q-learning	Single-agent
[233]	Privacy	Demand response resource market	Reduce energy cost	DQN	Single-agent
[234]	Privacy	Smart grid with battery	Optimize control policy	DDPG	Single-agent
[235]	Privacy	Smart grid data analysis	Hide sensitive information	DQN	Single-agent
[236]	Privacy	Community electricity market	Reduce daily cost	SAC	Single-agent
[237]	Privacy	Integrated smart grid system	Minimize energy cost	DDPG	Single-agent

a comprehensive review of DRL applications in SG operations can help comprehend unsolved problems in this domain and provide guidance to promote its development, which is one of the intentions for drafting this survey article.

IV. CHALLENGES AND OPEN RESEARCH ISSUES

We have mentioned the difficulty of SG operations mainly stems out of strong uncertainty, curse of dimensionality, and lack of accurate models. As one of the model-free approaches, RL can deal with variable RE and uncertain load demand issues by interacting with the environment in the absence of sufficient knowledge data. In addition, the curse of dimensionality can be handled with DNN. Therefore, DRL shows great potential in addressing the pertinent SG operation issues. However, current DRL methods still have a certain extent of limitation, which is mainly due to their dependence on handcrafted reward functions. It is not easy to design a reward function that encourages the desirable behaviors. Furthermore, the most reasonable reward function cannot avoid the local optimality, which belongs to the typical exploration–exploitation dilemma and has puzzled DRL applications for a long time. Hence, a relatively comprehensive survey of DRL approaches, potential solutions, and future directions is discussed in this section.

A. Security Concerns

SGs are critical infrastructures in modern power systems, which can handle sustainable, secure, economic, and reliable power system operations. To this end, it is crucial for DRL algorithms to ensure secure decisions in the learned policy that would not lead to potentially catastrophic consequences in SGs. For instance, the control commands issued by DRL should not violate physical SG constraints that could possibly result in device failures, grid instability, or even system breakdown. At present, DRL studies can be divided into three categories, including modifications in optimization criteria, modifications in exploration processes [238], and offline DRL methods [239].

1) Modifying Optimization Criterion: In general, the purpose of DRL is primarily focused on maximizing long-term rewards without explicitly considering the potential harm caused by dangerous states to the agent. In other words, the objective function of traditional DRL does not incorporate a description about decision risks. Moreover, if the objective function is designed inadequately, the DRL agent may encounter safety issues. To this end, the transformation of optimization criterion has been proposed to take the risk into account. This can be achieved through various approaches, such as directly penalizing infeasible solutions [240], penalizing the worst case scenario [241], or incorporating constrained optimality within the reward function [242]. For example, Qian et al. [110] incorporated constrained optimality within the reward function through using a Lagrangian function of power flow constraints.

2) Modifying Exploration Process: The unrestricted random exploration can potentially expose the agents to highly dangerous states. To prevent unforeseen and irreversible consequences, it is essential to evaluate the DRL agent security during training and deployment and restrict their exploration within permissible regions. Such methods can be categorized as the modification of exploration processes with a focus on ensuring security [243]. The

modification can be achieved through various approaches, such as embedding external knowledge [244] and constraining exploration within a certified safe region [245]. Cui et al. [246] formulated the online preventive control problem for mitigating transmission overloads as a constrained MDP. The constrained MDP is then solved using the interior-point policy optimization, which promotes learnings that can satisfy the pertinent constraints and improves the policy simultaneously.

3) Offline DRL: The two categories for modifying the optimization criteria and exploration process are regarded as online DRL, where the agent learns how to perform tasks by continuously interacting with the environment. In contrast, offline DRL requires an agent that can learn solely from statically offline datasets without exploration, thus ensuring the training safety from the perspective of data [247]. However, such approaches do not consider risk-related factors during policy deployment phases and, therefore, might not guarantee the security at the time of deployment [248].

In response to safety concerns, four related DRL variants are briefly introduced here, which include constrained DRL, adversarial DRL, robust DRL, and federated DRL as presented in the following.

1) *Constrained DRL:* It refers to the application of RL techniques to solve SG problems with explicit constraints. Generally, there are two types of soft and hard constraints, which are considered in the literature. Soft constraints allow for some degree of violation, whereas hard constraints must be strictly adhered to. On the one hand, there are common approaches to addressing soft constraints, including adjoining constraints to the reward through barrier or penalty functions and formulating constraints as chance constraints (i.e., setting a predefined threshold for the probability of constraint violation), or a budget constraint as follows [249]:

$$\max_{\pi} J(\pi), \quad \text{s.t. } J^{c}(\pi) \leq \bar{J}$$

where the agent goal is to find a control policy π that maximizes the expected return with respect to reward function J subject to a budget constraint for the return with respect to the cost function J^{c}. However, constrained DRL methods that focus on soft constraints alone may not guarantee safe exploration during the training phase. In addition, even after training convergence, the control actions generated by the trained policy may not always be entirely safe [250]. On the other hand, the enforcement approach is to take the conservative actions while dealing with hard constraints in constrained DRL [251]. Nevertheless, the enforcement approach usually results in significant conservation and might have large errors for complex power networks.

2) *Adversarial DRL:* It involves training a DRL agent in the presence of adversarial agents or environments that actively try to disturb the learning process or achieve their own objectives [252]. Adversarial training has been applied to enhance DRL algorithms against adversarial attacks in managing the SG cybersecurity. For instance, the attack and defense problems are formulated as MDP and adversarial MDP in [253], while the robust defense strategy is generated by adversarial training between attack and defense agents. In [254], a repeated game is formulated to mimic the real-world interactions between attackers and defenders in SGs. Furthermore, according to [255], it has been observed that a high-performing DRL agent, initially vulnerable to action perturbations, can be made more resilient against similar perturbations through the application of adversarial training. It is indeed worth mentioning that naively applying adversarial training may not be effective for all DRL tasks [256]. Adversarial training is a complex and challenging process that requires careful consideration and customization for each specific task.
3) *Robust DRL:* It incorporates robust optimization techniques to ensure that the learned policies remain effective even in the presence of uncertainties and perturbations, thereby improving the overall performance and stability of the DRL agent [257]. To be specific, robust DRL considers the worst case scenario or min–max framework to learn a control policy that maximizes the reward with respect to the worst case scenario or outcome encountered during the learning process. By training against these worst case scenarios, the agent becomes more resilient and capable of making effective decisions even in the face of uncertainties or adversarial conditions. The utilization of min–max structure in DRL algorithms has been a vibrant area of research. Previous studies primarily focus on addressing two types of uncertainties, i.e., inherent uncertainty stemming from the stochastic nature of the system and parameter uncertainty arising from incomplete knowledge about certain parameters of the MDP [258], [259]. While robust DRL has not yet received extensive attention in the context of SG, it holds significant potential as a future direction to tackle the diverse uncertainties present in the environment, such as model uncertainty, noise, and disturbances.
4) *Federated DRL:* The concern regarding SG security and privacy is one of the main obstacles in SG operations. However, extensive previous research about DRL applications in SG mainly belongs to the centralized method, which is vulnerable to cyberattack and privacy leakage. To this end, federated learning is combined with DRL to meet the requirements of privacy preservation and network security [260]. By combining federated learning and DRL, federated DRL enables collaborative learning while preserving

data privacy and reducing the communication overhead between the central server and distributed devices. For instance, Li et al. [261] proposed a federated MADRL algorithm via the physics-informed reward to solve the complex multiple microgrids energy management with privacy concern. Federated learning enables multiple agents to coordinately learn a shared decision model while keeping all the training data on device, thus preventing the risk of privacy leakage. What is more, the decentralized structure of federated learning offers a promising technique to reduce the pressure of centralized data storage. Therefore, it is meaningful to investigate a combination of federated learning and DRL in SG operations.

B. Sample Efficiency

Despite the success of DRL, it usually needs at least thousands of samples to gradually learn some useful policies even for a simple task. However, the real-world or real-time interactions between agent and environment are usually costly, and they still require time and energy consumption even in the simulation platform. This brings about a critical problem for DRL, i.e., how to design a more efficient algorithm to learn faster with fewer samples. At present, most DRL algorithms are of such a low learning efficiency that requires unbearable training time under current computational power. It is even worse for real-world interactions that potential problems of security concern, risks of failure cases, and time consumption all put forward higher requirements on the learning efficiency of DRL algorithms in practice.

1) Model-Based DRL: Different from the aforementioned model-free methods, model-based DRL generally indicates an agent not only learns a policy to estimate its action but also learns a model of environment to assist its action planning, thus accelerating the speed of policy learning. Learning an accurate model of environment provides additional pertinent information that could be helpful in evaluating the agent's current policy, which can make the entire learning process more efficient. In principle, a good model could handle a bunch of problems, as AlphaGo has done. Therefore, it is meaningful to integrate model-based and model-free DRLs and promote the sample efficiency in SGs. On the one hand, model-based methods can be utilized as warm-starts or the nominal model, providing initial information or serving as a foundation for model-free DRL methods. On the other hand, model-free DRL algorithms can coordinate and fine-tune the parameters of existing model-based controllers to improve their adaptability while maintaining baseline performance guarantees. Although the amount of research in this area is currently limited, the integration of model-free DRL with existing model-based approaches is considered to be a promising direction for future research.

2) Imitation Learning Combined DRL: Imitation learning attempts to not only mimic the actions and choices of experts but also learn a generalized policy that can handle unseen situations. The combination of imitation learning and RL is a very promising research field that has been extensively studied in recent years [262], [263]. It has been applied in various domains such as autonomous driving [264], quantitative trading [265], and the optimal SG dispatch [266], to tackle the challenge of low learning efficiency in DRL. For example, Guo et al. [267] combined DRL with imitation learning for cloud resource scheduling, where DRL is devoted to tackling the challenging multiresource scheduling problem and imitation learning enables an agent to learn an optimal policy more efficiently. In conclusion, the integration of imitation learning and DRL can provide a powerful learning framework that enables fast learning, generalization, and effectiveness. This combination is significant for addressing complex tasks and improving the learning capabilities of intelligent agents.

C. Learning Stability

Unlike the stable supervised learning, present DRL algorithms are volatile to a certain extent, which means that there exist huge differences of the learning performances over time in horizontal comparisons across multiple runs. In specific, this learning instability over time generally reflects as the large local variances or the nonmonotonicity on a single learning curve. As for unstable learning, it manifests a significant performance difference between different runs during training, which leads to large variances for horizontal comparisons. What is more, the endogenous instability and unpredictability of DNN aggravate the deviation of value function approximation, which further brings about noise in the gradient estimators and unstable learning performance. Significant efforts have been dedicated to addressing the stability problem in DRL for a considerable period of time. As mentioned in this article, the utilization of a target network with delayed updates and the incorporation of a replay buffer have been shown to mitigate the issue of unstable learning. In addition, TRPO employs second-order optimization techniques to provide more stable updates and comprehensive information. It applies constraints to the updated policy to ensure conservative yet stable improvements. However, DRL remains sensitive to hyperparameters and initialization even with the above works. This sensitivity poses a significant challenge and highlights the need for further research in this area to address these issues and improve the robustness and stability of DRL algorithms.

1) *Multiagent DRL:* With the development of DRL, MADRL is proposed and has attracted much attention. In fact, MADRL is regarded as a promising and worth exploring direction, which provides a novel way to investigate the unconventional DRL situations, including swarm intelligence, unstable environments for each agent, and innovation of agent itself. MADRL not only makes it possible to explore distributed

intelligence in multiagent environments but also contributes to learning a near-optimal agent policy in large-scale SG applications. Overall, multiple agents and their interactions in MADRL can enhance the learning stability of DRL by promoting exploration, facilitating experience sharing, enabling policy coordination, and improving robustness to environmental changes. These characteristics make MADRL a promising approach for addressing the learning stability challenges in DRL.

D. Exploration

Different from classical exploration–exploitation tradeoffs in RL, exploration is another main challenge of DRL. The difficulty of exploration in RL mainly stems from sparse reward functions, large action spaces, and unstable environment, as well as the security issue of exploration in the real world. First, the sparse rewards might result in the value function and policy networks optimized on hypersurfaces that are not convex and not smooth or even discontinuous scenarios. In such circumstances, the policy after one-step optimization may not effectively facilitate the exploration of higher reward regions. Consequently, the agent encounters challenges in exploring trajectories that yield high rewards during its exploration phase. Second, the large action space also poses a challenge for exploration in DRL agents. For example, in electricity market bidding issues, the presence of a large action space makes it extremely difficult to explore an optimal policy. Third, the presence of an unstable environment also makes it difficult for agents to explore effectively. For instance, multiplayer settings cause the opponent part of the electricity market environment to some extent, which weakens the exploration capacity of agent. Finally, real-world security concerns in SG applications also raise exploration concerns in DRL. For instance, in the context of an SG controlled by an agent, it is crucial to learn from failure cases such as power outages, voltage fluctuations, and transmission congestion. However, the collection of these failure cases directly from SG is not feasible due to safety and operational concerns. Random actions for exploration in SGs are also not viable as they can potentially lead to catastrophic consequences. Therefore, alternative methods and strategies need to be employed to enable safe and effective exploration in these complex and high-stakes environments.

E. Simulation to Reality

DRL has been successful in solving a wide range of optimization tasks in simulated environments and may even surpass human performance in some specific domains such as the game of GO [268]. However, the challenges of applying DRL methods to real-world cases have not been addressed. As mentioned above, tasks involving hardware in the real world often have high demands for security and accuracy. Taking SG as an example, a single error in operation can result in catastrophic consequences. Moreover, most of the existing literature trails DRL policies solely based on high-fidelity power system simulators, which do not emphasize the gap between the simulators and real-world SG operations, i.e., reality gap. Therefore, policies trained in simulators may not always exhibit reliable performance in real-world scenarios due to the existence of reality gap. In general, methods for addressing the simulation to reality can be categorized into at least two following approaches.

1) Meta-Learning-Based DRL: Meta-learning is devoted to improving the learning performance by utilizing the previous SG experience. More specifically, meta-learning leverages the acquired SG knowledge by extracting universal learning policies and knowledge to guide the learning process in a new task. In this way, it is viable to extend the meta-learning to a new SG operation scenario, including the transfer from simulated environments to real-world systems. In addition, the combination of meta-learning and RL gives the meta-RL methods, which can reduce the sensitivity to network parameters and enhance the robustness of the SG algorithm. On this basis, it would be quite desirable to promote the meta-RL into the deep one and apply it to SG operational problems.

2) Transfer Learning Combined DRL: Similar to the meta-learning approach, transfer learning also emphasizes the storage knowledge acquired while solving one problem and applying to a different but relevant one, whose core is to find similarities between the existing and new knowledge. Since it is too expensive to learn the target domain directly from scratch, transfer learning is adopted to use existing knowledge to learn new knowledge as quickly as possible. Transfer learning can leverage knowledge from simulated tasks to speed up the learning process in real-world cases. Consequently, we can potentially combine DRL with transfer learning to handle the reality gap in SG operations.

V. CONCLUSION

In this article, we have presented a comprehensive review of DRL applications in SG operations. First, an overview of RL, DL, and DRL is provided. Then, various DRL techniques are divided into two categories according to the optimization policy, i.e., value- and policy-based algorithms. On the one hand, typical value-based DRL algorithms, including DQN and its variants, are depicted with detailed theories. On the other hand, several popular policy-based algorithms involving both stochastic and deterministic policies are introduced with specific explanations. Then, we displayed exhaustive surveys, comparisons, and analyses for DRL approaches to address a variety of SG applications, covering optimal dispatch, operation control, electricity markets, and other emerging areas in power systems. Finally, essential challenges, potential solutions, and future research directions are discussed from the perspective of

safety concerns, sample efficiency, learning stability, exploration, simulation to reality, and so on. Furthermore, this article does not propose a dichotomy between DRL and conventional methods. Instead, DRL can serve as a complement to existing approaches and enhance them in a data-driven manner. In summary, careful consideration should be devoted to identifying appropriate DRL application scenarios and utilizing them effectively in SG applications. ■

REFERENCES

[1] M. Liserre, M. A. Perez, M. Langwasser, C. A. Rojas, and Z. Zhou, "Unlocking the hidden capacity of the electrical grid through smart transformer and smart transmission," *Proc. IEEE*, vol. 111, no. 4, pp. 421–437, Apr. 2023.

[2] M. Chertkov and G. Andersson, "Multienergy systems," *Proc. IEEE*, vol. 108, no. 9, pp. 1387–1391, Sep. 2020.

[3] S. Geng, M. Vrakopoulou, and I. A. Hiskens, "Optimal capacity design and operation of energy hub systems," *Proc. IEEE*, vol. 108, no. 9, pp. 1475–1495, Sep. 2020.

[4] M. Shahidehpour, M. Yan, P. Shikhar, S. Bahramirad, and A. Paaso, "Blockchain for peer-to-peer transactive energy trading in networked microgrids: Providing an effective and decentralized strategy," *IEEE Electrific. Mag.*, vol. 8, no. 4, pp. 80–90, Dec. 2020.

[5] M. Shahidehpour, Z. Li, S. Bahramirad, Z. Li, and W. Tian, "Networked microgrids: Exploring the possibilities of the IIT-Bronzeville grid," *IEEE Power Energy Mag.*, vol. 15, no. 4, pp. 63–71, Jul. 2017.

[6] S. Z. Tajalli, A. Kavousi-Fard, M. Mardaneh, A. Khosravi, and R. Razavi-Far, "Uncertainty-aware management of smart grids using cloud-based LSTM-prediction interval," *IEEE Trans. Cybern.*, vol. 52, no. 10, pp. 9964–9977, Oct. 2022.

[7] X. Xia, Y. Xiao, W. Liang, and J. Cui, "Detection methods in smart meters for electricity thefts: A survey," *Proc. IEEE*, vol. 110, no. 2, pp. 273–319, Feb. 2022.

[8] L. Duchesne, E. Karangelos, and L. Wehenkel, "Recent developments in machine learning for energy systems reliability management," *Proc. IEEE*, vol. 108, no. 9, pp. 1656–1676, Sep. 2020.

[9] Y. Yuan et al., "Data driven discovery of cyber physical systems," *Nature Commun.*, vol. 10, no. 1, p. 4894, Oct. 2019.

[10] H. L. Liao, Q. H. Wu, Y. Z. Li, and L. Jiang, "Economic emission dispatching with variations of wind power and loads using multi-objective optimization by learning automata," *Energy Convers. Manage.*, vol. 87, pp. 990–999, Nov. 2014.

[11] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K.-R. Müller, "Explaining deep neural networks and beyond: A review of methods and applications," *Proc. IEEE*, vol. 109, no. 3, pp. 247–278, Mar. 2021.

[12] Y. Li et al., "Dense skip attention based deep learning for day-ahead electricity price forecasting," *IEEE Trans. Power Syst.*, vol. 38, no. 5, pp. 4308–4327, Sep. 2023.

[13] M. Lapan, *Deep Reinforcement Learning Hands-On: Apply Modern RL Methods to Practical Problems of Chatbots, Robotics, Discrete Optimization, Web Automation, and More*. Birmingham, U.K.: Packt Publishing Ltd, 2020.

[14] N. C. Luong et al., "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3133–3174, 4th Quart., 2019.

[15] W. Chen, X. Qiu, T. Cai, H.-N. Dai, Z. Zheng, and Y. Zhang, "Deep reinforcement learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1659–1692, 3rd Quart., 2021.

[16] Y. Keneshloo, T. Shi, N. Ramakrishnan, and C. K. Reddy, "Deep reinforcement learning for sequence-to-sequence models," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 7, pp. 2469–2489, Jul. 2020.

[17] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 26–38, Nov. 2017.

[18] N. Le, V. S. Rathour, K. Yamazaki, K. Luu, and M. Savvides, *Deep Reinforcement Learning in Computer Vision: A Comprehensive Survey*. Cham, Switzerland: Springer, 2021.

[19] X. Chen, G. Qu, Y. Tang, S. Low, and N. Li, "Reinforcement learning for selective key applications in power systems: Recent advances and future challenges," *IEEE Trans. Smart Grid*, vol. 13, no. 4, pp. 2935–2958, Jul. 2022.

[20] Z. Zhang, D. Zhang, and R. C. Qiu, "Deep reinforcement learning for power system applications: An overview," *CSEE J. Power Energy Syst.*, vol. 6, no. 1, pp. 213–225, Mar. 2020.

[21] M. Glavic, "(Deep) reinforcement learning for electric power system control and related problems: A short review and perspectives," *Annu. Rev. Control*, vol. 48, pp. 22–35, Jan. 2019.

[22] D. Cao et al., "Reinforcement learning and its applications in modern power and energy systems: A review," *J. Modern Power Syst. Clean Energy*, vol. 8, no. 6, pp. 1029–1042, Nov. 2020.

[23] T. Yang, L. Zhao, W. Li, and A. Y. Zomaya, "Reinforcement learning in sustainable energy and electric systems: A survey," *Annu. Rev. Control*, vol. 49, pp. 145–163, Jan. 2020.

[24] A. T. D. Perera and P. Kamalaruban, "Applications of reinforcement learning in energy systems," *Renew. Sustain. Energy Rev.*, vol. 137, Mar. 2021, Art. no. 110618.

[25] L. Yu, S. Qin, M. Zhang, C. Shen, T. Jiang, and X. Guan, "A review of deep reinforcement learning for smart building energy management," *IEEE Internet Things J.*, vol. 8, no. 15, pp. 12046–12063, Aug. 2021.

[26] D. Zhang, X. Han, and C. Deng, "Review on the research and practice of deep learning and reinforcement learning in smart grids," *CSEE J. Power Energy Syst.*, vol. 4, no. 3, pp. 362–370, Sep. 2018.

[27] L. Zeng, M. Sun, X. Wan, Z. Zhang, R. Deng, and Y. Xu, "Physics-constrained vulnerability assessment of deep reinforcement learning-based SCOPF," *IEEE Trans. Power Syst.*, vol. 38, no. 3, pp. 2690–2704, May 2023.

[28] T. T. Nguyen and V. J. Reddi, "Deep reinforcement learning for cyber security," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 8, pp. 3779–3795, Aug. 2023.

[29] T. Ding, Z. Zeng, J. Bai, B. Qin, Y. Yang, and M. Shahidehpour, "Optimal electric vehicle charging strategy with Markov decision process and reinforcement learning technique," *IEEE Trans. Ind. Appl.*, vol. 56, no. 5, pp. 5811–5823, Sep. 2020.

[30] H. Dong, Z. Ding, and S. Zhang, *Deep Reinforcement Learning*. Cham, Switzerland: Springer, 2020.

[31] S. Dreyfus, "Richard Bellman on the birth of dynamic programming," *Oper. Res.*, vol. 50, no. 1, pp. 48–51, Feb. 2002.

[32] J. Hu, H. Niu, J. Carrasco, B. Lennox, and F. Arvin, "Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 14413–14423, Dec. 2020.

[33] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, 1992.

[34] 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.

[35] M. Ravanelli, P. Brakel, M. Omologo, and Y. Bengio, "Light gated recurrent units for speech recognition," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 2, pp. 92–102, Apr. 2018.

[36] Y. LeCun et al., "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

[37] J. N. Tsitsiklis and B. Van Roy, "An analysis of temporal-difference learning with function approximation," *IEEE Trans. Autom. Control*, vol. 42, no. 5, pp. 674–690, May 1997.

[38] J. Li, H. Wang, H. He, Z. Wei, Q. Yang, and P. Igic, "Battery optimal sizing under a synergistic framework with DQN-based power managements for the fuel cell hybrid powertrain," *IEEE Trans. Transport. Electrific.*, vol. 8, no. 1, pp. 36–47, Mar. 2022.

[39] V. Mnih et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[40] A. Camacho, J. Varley, A. Zeng, D. Jain, A. Iscen, and D. Kalashnikov, "Reward machines for vision-based robotic manipulation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 14284–14290.

[41] H. Hasselt, "Double Q-learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 23, 2010, pp. 2613–2621.

[42] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 30, no. 1, 2016, pp. 1–13.

[43] X. Chen, H. Zhang, C. Wu, S. Mao, Y. Ji, and M. Bennis, "Optimized computation offloading performance in virtual edge computing systems via deep reinforcement learning," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4005–4018, Jun. 2019.

[44] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, "Dueling network architectures for deep reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1995–2003.

[45] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 12, 1999, pp. 1057–1063.

[46] V. Konda and J. Tsitsiklis, "Actor-critic algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 12, 1999, pp. 1008–1014.

[47] A. G. Barto, R. S. Sutton, and C. W. Anderson, "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE Trans. Syst. Man, Cybern.*, vol. SMC-13, no. 5, pp. 834–846, Sep. 1983.

[48] V. Mnih et al., "Asynchronous methods for deep reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1928–1937.

[49] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1889–1897.

[50] S. Kakade and J. Langford, "Approximately optimal approximate reinforcement learning," in *Proc. 19th Int. Conf. Mach. Learn.*, 2002, pp. 267–274.

[51] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 22–31.

[52] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[53] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.

[54] R. I. Bot, S. M. Grad, and G. Wanka, *Duality in Vector Optimization*. Cham, Switzerland: Springer, 2009.

[55] N. Heess et al., "Emergence of locomotion behaviours in rich environments," 2017, *arXiv:1707.02286*.

[56] J. Booth, "PPO dash: Improving generalization in deep reinforcement learning," 2019, *arXiv:1907.06704*.

[57] C.-Y. Tang, C.-H. Liu, W.-K. Chen, and S. D. You, "Implementing action mask in proximal policy optimization (PPO) algorithm," *ICT Exp.*, vol. 6, no. 3, pp. 200–203, Sep. 2020.

[58] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 387–395.

[59] 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.

[60] A. Navas, J. S. Gómez, J. Llanos, E. Rute, D. Sáez, and M. Sumner, "Distributed predictive control strategy for frequency restoration of microgrids considering optimal dispatch," *IEEE Trans. Smart Grid*, vol. 12, no. 4, pp. 2748–2759, Jul. 2021.

[61] Z. Chen, J. Zhu, H. Dong, W. Wu, and H. Zhu, "Optimal dispatch of WT/PV/ES combined generation system based on cyber-physical-social integration," *IEEE Trans. Smart Grid*, vol. 13, no. 1, pp. 342–354, Jan. 2022.

[62] T. Wu, C. Zhao, and Y. A. Zhang, "Distributed AC–DC optimal power dispatch of VSC-based energy routers in smart microgrids," *IEEE Trans. Power Syst.*, vol. 36, no. 5, pp. 4457–4470, Sep. 2021.

[63] Z. Zhang, C. Wang, H. Lv, F. Liu, H. Sheng, and M. Yang, "Day-ahead optimal dispatch for integrated energy system considering power-to-gas and dynamic pipeline networks," *IEEE Trans. Ind. Appl.*, vol. 57, no. 4, pp. 3317–3328, Jul. 2021.

[64] Md. R. Islam, H. Lu, Md. R. Islam, M. J. Hossain, and L. Li, "An IoT-based decision support tool for improving the performance of smart grids connected with distributed energy sources and electric vehicles," *IEEE Trans. Ind. Appl.*, vol. 56, no. 4, pp. 4552–4562, Jul. 2020.

[65] X. Sun and J. Qiu, "Hierarchical voltage control strategy in distribution networks considering customized charging navigation of electric vehicles," *IEEE Trans. Smart Grid*, vol. 12, no. 6, pp. 4752–4764, Nov. 2021.

[66] L. Xi, L. Zhang, Y. Xu, S. Wang, and C. Yang, "Automatic generation control based on multiple-step greedy attribute and multiple-level allocation strategy," *CSEE J. Power Energy Syst.*, vol. 8, no. 1, pp. 281–292, Jan. 2022.

[67] K. S. Xiahou, Y. Liu, and Q. H. Wu, "Robust load frequency control of power systems against random time-delay attacks," *IEEE Trans. Smart Grid*, vol. 12, no. 1, pp. 909–911, Jan. 2021.

[68] K.-D. Lu, G.-Q. Zeng, X. Luo, J. Weng, Y. Zhang, and M. Li, "An adaptive resilient load frequency controller for smart grids with DoS attacks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 4689–4699, May 2020.

[69] B. Hu, Y. Gong, C. Y. Chung, B. F. Noble, and G. Poelzer, "Price-maker bidding and offering strategies for networked microgrids in day-ahead electricity markets," *IEEE Trans. Smart Grid*, vol. 12, no. 6, pp. 5201–5211, Nov. 2021.

[70] H. Haghighat, H. Karimianfard, and B. Zeng, "Integrating energy management of autonomous smart grids in electricity market operation," *IEEE Trans. Smart Grid*, vol. 11, no. 5, pp. 4044–4055, Sep. 2020.

[71] A. Paudel, L. P. M. I. Sampath, J. Yang, and H. B. Gooi, "Peer-to-peer energy trading in smart grid considering power losses and network fees," *IEEE Trans. Smart Grid*, vol. 11, no. 6, pp. 4727–4737, Nov. 2020.

[72] P. Zhuang, T. Zamir, and H. Liang, "Blockchain for cybersecurity in smart grid: A comprehensive survey," *IEEE Trans. Ind. Informat.*, vol. 17, no. 1, pp. 3–19, Jan. 2021.

[73] Y. Ding, B. Wang, Y. Wang, K. Zhang, and H. Wang, "Secure metering data aggregation with batch verification in industrial smart grid," *IEEE Trans. Ind. Informat.*, vol. 16, no. 10, pp. 6607–6616, Oct. 2020.

[74] K. Kaur, G. Kaddoum, and S. Zeadally, "Blockchain-based cyber-physical security for electrical vehicle aided smart grid ecosystem," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 8, pp. 5178–5189, Aug. 2021.

[75] M. B. Gough, S. F. Santos, T. AlSkaif, M. S. Javadi, R. Castro, and J. P. S. Catalão, "Preserving privacy of smart meter data in a smart grid environment," *IEEE Trans. Ind. Informat.*, vol. 18, no. 1, pp. 707–718, Jan. 2022.

[76] Y. Li, Y. Zhao, L. Wu, and Z. Zeng, *Artificial Intelligence Enabled Computational Methods for Smart Grid Forecast and Dispatch*. Cham, Switzerland: Springer, 2023.

[77] A. M. Fathabad, J. Cheng, K. Pan, and F. Qiu, "Data-driven planning for renewable distributed generation integration," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4357–4368, Nov. 2020.

[78] K. Utkarsh, D. Srinivasan, A. Trivedi, W. Zhang, and T. Reindl, "Distributed model-predictive real-time optimal operation of a network of smart microgrids," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2833–2845, May 2019.

[79] Y. Liu, L. Guo, and C. Wang, "A robust operation-based scheduling optimization for smart distribution networks with multi-microgrids," *Appl. Energy*, vol. 228, pp. 130–140, Oct. 2018.

[80] C. Guo, F. Luo, Z. Cai, and Z. Y. Dong, "Integrated energy systems of data centers and smart grids: State-of-the-art and future opportunities," *Appl. Energy*, vol. 301, Nov. 2021, Art. no. 117474.

[81] Z. J. Lee et al., "Adaptive charging networks: A framework for smart electric vehicle charging," *IEEE Trans. Smart Grid*, vol. 12, no. 5, pp. 4339–4350, Sep. 2021.

[82] C. Li, Z. Dong, G. Chen, B. Zhou, J. Zhang, and X. Yu, "Data-driven planning of electric vehicle charging infrastructure: A case study of Sydney, Australia," *IEEE Trans. Smart Grid*, vol. 12, no. 4, pp. 3289–3304, Jul. 2021.

[83] B. Zhou et al., "Optimal coordination of electric vehicles for virtual power plants with dynamic communication spectrum allocation," *IEEE Trans. Ind. Informat.*, vol. 17, no. 1, pp. 450–462, Jan. 2021.

[84] D. Cao, W. Hu, J. Zhao, Q. Huang, Z. Chen, and F. Blaabjerg, "A multi-agent deep reinforcement learning based voltage regulation using coordinated PV inverters," *IEEE Trans. Power Syst.*, vol. 35, no. 5, pp. 4120–4123, Sep. 2020.

[85] P. Kou, D. Liang, C. Wang, Z. Wu, and L. Gao, "Safe deep reinforcement learning-based constrained optimal control scheme for active distribution networks," *Appl. Energy*, vol. 264, Apr. 2020, Art. no. 114772.

[86] 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.

[87] H. Liu and W. Wu, "Two-stage deep reinforcement learning for inverter-based Volt-VAR control in active distribution networks," *IEEE Trans. Smart Grid*, vol. 12, no. 3, pp. 2037–2047, May 2021.

[88] X. Sun and J. Qiu, "Two-stage Volt/Var control in active distribution networks with multi-agent deep reinforcement learning method," *IEEE Trans. Smart Grid*, vol. 12, no. 4, pp. 2903–2912, Jul. 2021.

[89] Q. Yang, G. Wang, A. Sadeghi, G. B. Giannakis, and J. Sun, "Two-timescale voltage control in distribution grids using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2313–2323, May 2020.

[90] Y. Li, G. Hao, Y. Liu, Y. Yu, Z. Ni, and Y. Zhao, "Many-objective distribution network reconfiguration via deep reinforcement learning assisted optimization algorithm," *IEEE Trans. Power Del.*, vol. 37, no. 3, pp. 2230–2244, Jun. 2022.

[91] S. H. Oh, Y. T. Yoon, and S. W. Kim, "Online reconfiguration scheme of self-sufficient distribution network based on a reinforcement learning approach," *Appl. Energy*, vol. 280, Dec. 2020, Art. no. 115900.

[92] Y. Gao, W. Wang, J. Shi, and N. Yu, "Batch-constrained reinforcement learning for dynamic distribution network reconfiguration," *IEEE Trans. Smart Grid*, vol. 11, no. 6, pp. 5357–5369, Nov. 2020.

[93] S. Bahrami, Y. C. Chen, and V. W. S. Wong, "Deep reinforcement learning for demand response in distribution networks," *IEEE Trans. Smart Grid*, vol. 12, no. 2, pp. 1496–1506, Mar. 2021.

[94] N. L. Dehghani, A. B. Jeddi, and A. Shafieezadeh, "Intelligent hurricane resilience enhancement of power distribution systems via deep reinforcement learning," *Appl. Energy*, vol. 285, Mar. 2021, Art. no. 116355.

[95] Y. Li et al., "Optimal operation of multimicrogrids via cooperative energy and reserve scheduling," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3459–3468, Aug. 2018.

[96] M. Mahmoodi, P. Shamsi, and B. Fahimi, "Economic dispatch of a hybrid microgrid with distributed energy storage," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2607–2614, Nov. 2015.

[97] Y. Shi, S. Dong, C. Guo, Z. Chen, and L. Wang, "Enhancing the flexibility of storage integrated power system by multi-stage robust dispatch," *IEEE Trans. Power Syst.*, vol. 36, no. 3, pp. 2314–2322, May 2021.

[98] Y. Li et al., "Day-ahead risk averse market clearing considering demand response with data-driven load uncertainty representation: A Singapore electricity market study," *Energy*, vol. 254, Sep. 2022, Art. no. 123923.

[99] A. Dridi, H. Afifi, H. Moungla, and J. Badosa, "A novel deep reinforcement approach for IIoT microgrid energy management systems," *IEEE Trans. Green Commun. Netw.*, vol. 6, no. 1, pp. 148–159, Mar. 2022.

[100] Md. S. Munir, S. F. Abedin, N. H. Tran, Z. Han, E.-N. Huh, and C. S. Hong, "Risk-aware energy scheduling for edge computing with microgrid: A multi-agent deep reinforcement learning approach," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 3, pp. 3476–3497, Sep. 2021.

[101] L. Lei, Y. Tan, G. Dahlenburg, W. Xiang, and K. Zheng, "Dynamic energy dispatch based on deep reinforcement learning in IoT-driven smart isolated microgrids," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 7938–7953, May 2021.

[102] F. Sanchez Gorostiza and F. M. Gonzalez-Longatt, "Deep reinforcement learning-based controller for SOC management of multi-electrical energy storage system," *IEEE Trans. Smart Grid*, vol. 11, no. 6, pp. 5039–5050, Nov. 2020.

[103] T. Chen, S. Bu, X. Liu, J. Kang, F. R. Yu, and Z. Han, "Peer-to-peer energy trading and energy conversion in interconnected multi-energy microgrids using multi-agent deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 13, no. 1, pp. 715–727, Jan. 2022.

[104] H. Hua et al., "Data-driven dynamical control for bottom-up energy Internet system," *IEEE Trans. Sustain. Energy*, vol. 13, no. 1, pp. 315–327, Jan. 2022.

[105] Y. Li, R. Wang, and Z. Yang, "Optimal scheduling of isolated microgrids using automated reinforcement learning-based multi-period forecasting," *IEEE Trans. Sustain. Energy*, vol. 13, no. 1, pp. 159–169, Jan. 2022.

[106] Y. Du and F. Li, "Intelligent multi-microgrid energy management based on deep neural network and model-free reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1066–1076, Mar. 2020.

[107] Z. Qin, D. Liu, H. Hua, and J. Cao, "Privacy preserving load control of residential microgrid via deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 12, no. 5, pp. 4079–4089, Sep. 2021.

[108] Y. Li et al., "Coordinated scheduling for improving

uncertain wind power adsorption in electric vehicles—Wind integrated power systems by multiobjective optimization approach," *IEEE Trans. Ind. Appl.*, vol. 56, no. 3, pp. 2238–2250, May 2020.

[109] Y. Li, S. He, Y. Li, L. Ge, S. Lou, and Z. Zeng, "Probabilistic charging power forecast of EVCS: Reinforcement learning assisted deep learning approach," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 1, pp. 344–357, Jan. 2023.

[110] T. Qian, C. Shao, X. Wang, Q. Zhou, and M. Shahidehpour, "Shadow-price DRL: A framework for online scheduling of shared autonomous EVs fleets," *IEEE Trans. Smart Grid*, vol. 13, no. 4, pp. 3106–3117, Jul. 2022.

[111] J. Zhang, Y. Guan, L. Che, and M. Shahidehpour, "EV charging command fast allocation approach based on deep reinforcement learning with safety modules," *IEEE Trans. Smart Grid*, early access, Jun. 5, 2023, doi: 10.1109/TSG.2023.3281782.

[112] C. Zhang, Y. Liu, F. Wu, B. Tang, and W. Fan, "Effective charging planning based on deep reinforcement learning for electric vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 542–554, Jan. 2021.

[113] B. Lin, B. Ghaddar, and J. Nathwani, "Deep reinforcement learning for the electric vehicle routing problem with time windows," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 11528–11538, Aug. 2022.

[114] F. Zhang, Q. Yang, and D. An, "CDDPG: A deep-reinforcement-learning-based approach for electric vehicle charging control," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3075–3087, Mar. 2021.

[115] A. A. Zishan, M. M. Haji, and O. Ardakanian, "Adaptive congestion control for electric vehicle charging in the smart grid," *IEEE Trans. Smart Grid*, vol. 12, no. 3, pp. 2439–2449, May 2021.

[116] H. Li, Z. Wan, and H. He, "Constrained EV charging scheduling based on safe deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2427–2439, May 2020.

[117] T. Wu et al., "Multi-agent deep reinforcement learning for urban traffic light control in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8243–8256, Aug. 2020.

[118] T. Qian, C. Shao, X. Li, X. Wang, Z. Chen, and M. Shahidehpour, "Multi-agent deep reinforcement learning method for EV charging station game," *IEEE Trans. Power Syst.*, vol. 37, no. 3, pp. 1682–1694, May 2022.

[119] T. Qian, C. Shao, X. Wang, and M. Shahidehpour, "Deep reinforcement learning for EV charging navigation by coordinating smart grid and intelligent transportation system," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1714–1723, Mar. 2020.

[120] L. Yan, X. Chen, J. Zhou, Y. Chen, and J. Wen, "Deep reinforcement learning for continuous electric vehicles charging control with dynamic user behaviors," *IEEE Trans. Smart Grid*, vol. 12, no. 6, pp. 5124–5134, Nov. 2021.

[121] E. A. M. Ceseña, E. Loukarakis, N. Good, and P. Mancarella, "Integrated electricity–heat–gas systems: Techno–Economic modeling, optimization, and application to multienergy districts," *Proc. IEEE*, vol. 108, no. 9, pp. 1392–1410, Sep. 2020.

[122] T. Yang, L. Zhao, W. Li, and A. Y. Zomaya, "Dynamic energy dispatch strategy for integrated energy system based on improved deep reinforcement learning," *Energy*, vol. 235, Nov. 2021, Art. no. 121377.

[123] B. Zhang, W. Hu, D. Cao, Q. Huang, Z. Chen, and F. Blaabjerg, "Deep reinforcement learning–based approach for optimizing energy conversion in integrated electrical and heating system with renewable energy," *Energy Convers. Manage.*, vol. 202, Dec. 2019, Art. no. 112199.

[124] B. Zhang, W. Hu, D. Cao, Q. Huang, Z. Chen, and F. Blaabjerg, "Economical operation strategy of an integrated energy system with wind power and power to gas technology—A DRL-based approach," *IET Renew. Power Gener.*, vol. 14, no. 17, pp. 3292–3299, Dec. 2020.

[125] G. Zhang et al., "Data-driven optimal energy management for a wind-solar-diesel-battery-reverse osmosis hybrid energy system using a deep reinforcement learning approach," *Energy Convers. Manage.*, vol. 227, Jan. 2021, Art. no. 113608.

[126] Y. Li, F. Bu, Y. Li, and C. Long, "Optimal scheduling of island integrated energy systems considering multi-uncertainties and hydrothermal simultaneous transmission: A deep reinforcement learning approach," *Appl. Energy*, vol. 333, Mar. 2023, Art. no. 120540.

[127] S. Zhou et al., "Combined heat and power system intelligent economic dispatch: A deep reinforcement learning approach," *Int. J. Electr. Power Energy Syst.*, vol. 120, Sep. 2020, Art. no. 106016.

[128] S. Zhong et al., "Deep reinforcement learning framework for dynamic pricing demand response of regenerative electric heating," *Appl. Energy*, vol. 288, Apr. 2021, Art. no. 116623.

[129] Y. Ye, D. Qiu, X. Wu, G. Strbac, and J. Ward, "Model-free real-time autonomous control for a residential multi-energy system using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3068–3082, Jul. 2020.

[130] J. Li, T. Yu, and X. Zhang, "Coordinated load frequency control of multi-area integrated energy system using multi-agent deep reinforcement learning," *Appl. Energy*, vol. 306, Jan. 2022, Art. no. 117900.

[131] B. Yang, X. Zhang, T. Yu, H. Shu, and Z. Fang, "Grouped grey wolf optimizer for maximum power point tracking of doubly-fed induction generator based wind turbine," *Energy Convers. Manage.*, vol. 133, pp. 427–443, Feb. 2017.

[132] Q. Sun, R. Fan, Y. Li, B. Huang, and D. Ma, "A distributed double-consensus algorithm for residential We-Energy," *IEEE Trans. Ind. Informat.*, vol. 15, no. 8, pp. 4830–4842, Aug. 2019.

[133] W. Fu, K. Wang, J. Tan, and K. Zhang, "A composite framework coupling multiple feature selection, compound prediction models and novel hybrid swarm optimizer-based synchronization optimization strategy for multi-step ahead short-term wind speed forecasting," *Energy Convers. Manage.*, vol. 205, Feb. 2020, Art. no. 112461.

[134] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis, "Optimal and autonomous control using reinforcement learning: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2042–2062, Jun. 2018.

[135] S. Vijayshankar, P. Stanfel, J. King, E. Spyrou, and K. Johnson, "Deep reinforcement learning for automatic generation control of wind farms," in *Proc. Amer. Control Conf. (ACC)*, May 2021, pp. 1796–1802.

[136] J. Li, T. Yu, and X. Zhang, "Coordinated automatic generation control of interconnected power system with imitation guided exploration multi-agent deep reinforcement learning," *Int. J. Electr. Power Energy Syst.*, vol. 136, Mar. 2022, Art. no. 107471.

[137] L. Xi et al., "A deep reinforcement learning algorithm for the power order optimization allocation of AGC in interconnected power grids," *CSEE J. Power Energy Syst.*, vol. 6, no. 3, pp. 712–723, Sep. 2020.

[138] J. Li, T. Yu, X. Zhang, F. Li, D. Lin, and H. Zhu, "Efficient experience replay based deep deterministic policy gradient for AGC dispatch in integrated energy system," *Appl. Energy*, vol. 285, Mar. 2021, Art. no. 116386.

[139] D. Zhang et al., "Research on AGC performance during wind power ramping based on deep reinforcement learning," *IEEE Access*, vol. 8, pp. 107409–107418, 2020.

[140] J. Li, T. Yu, H. Zhu, F. Li, D. Lin, and Z. Li, "Multi-agent deep reinforcement learning for sectional AGC dispatch," *IEEE Access*, vol. 8, pp. 158067–158081, 2020.

[141] J. Li, J. Yao, T. Yu, and X. Zhang, "Distributed deep reinforcement learning for integrated generation-control and power-dispatch of interconnected power grid with various renewable units," *IET Renew. Power Gener.*, vol. 16, no. 7, pp. 1316–1335, May 2022.

[142] Q. Zhang, H. Tang, Z. Wang, X. Wu, and K. Lv, "Flexible selection framework for secondary frequency regulation units based on learning optimisation method," *Int. J. Electr. Power Energy Syst.*, vol. 142, Nov. 2022, Art. no. 108175.

[143] L. Yin, L. Zhao, T. Yu, and X. Zhang, "Deep forest reinforcement learning for preventive strategy considering automatic generation control in large-scale interconnected power systems," *Appl. Sci.*, vol. 8, no. 11, p. 2185, Nov. 2018.

[144] J. J. Yang, M. Yang, M. X. Wang, P. J. Du, and Y. X. Yu, "A deep reinforcement learning method for managing wind farm uncertainties through energy storage system control and external reserve purchasing," *Int. J. Electr. Power Energy Syst.*, vol. 119, Jul. 2020, Art. no. 105928.

[145] V. P. Singh, N. Kishor, and P. Samuel, "Distributed multi-agent system-based load frequency control for multi-area power system in smart grid," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5151–5160, Jun. 2017.

[146] H. Wang, Z. Lei, X. Zhang, J. Peng, and H. Jiang, "Multiobjective reinforcement learning-based intelligent approach for optimization of activation rules in automatic generation control," *IEEE Access*, vol. 7, pp. 17480–17492, 2019.

[147] S. Hasanvand, M. Rafiei, M. Gheisarnejad, and M.-H. Khooban, "Reliable power scheduling of an emission-free ship: Multiobjective deep reinforcement learning," *IEEE Trans. Transport. Electrific.*, vol. 6, no. 2, pp. 832–843, Jun. 2020.

[148] 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.

[149] J. Duan et al., "Deep-reinforcement-learning-based autonomous voltage control for power grid operations," *IEEE Trans. Power Syst.*, vol. 35, no. 1, pp. 814–817, Jan. 2020.

[150] D. Cao et al., "Model-free voltage control of active distribution system with PVs using surrogate model-based deep reinforcement learning," *Appl. Energy*, vol. 306, Jan. 2022, Art. no. 117982.

[151] C. Cui, N. Yan, B. Huangfu, T. Yang, and C. Zhang, "Voltage regulation of DC–DC buck converters feeding CPLs via deep reinforcement learning," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 69, no. 3, pp. 1777–1781, Mar. 2022.

[152] S. Wang, L. Du, X. Fan, and Q. Huang, "Deep reinforcement scheduling of energy storage systems for real-time voltage regulation in unbalanced LV networks with high PV penetration," *IEEE Trans. Sustain. Energy*, vol. 12, no. 4, pp. 2342–2352, Oct. 2021.

[153] D. Cao, J. Zhao, W. Hu, F. Ding, Q. Huang, and Z. Chen, "Attention enabled multi-agent DRL for decentralized Volt-VAR control of active distribution system using PV inverters and SVCs," *IEEE Trans. Sustain. Energy*, vol. 12, no. 3, pp. 1582–1592, Jul. 2021.

[154] S. Mukherjee, R. Huang, Q. Huang, T. L. Vu, and T. Yin, "Scalable voltage control using structure-driven hierarchical deep reinforcement learning," 2021, *arXiv:2102.00077*.

[155] R. Huang et al., "Accelerated derivative-free deep reinforcement learning for large-scale grid emergency voltage control," *IEEE Trans. Power Syst.*, vol. 37, no. 1, pp. 14–25, Jan. 2022.

[156] R. R. Hossain, Q. Huang, and R. Huang, "Graph convolutional network-based topology embedded deep reinforcement learning for voltage stability control," *IEEE Trans. Power Syst.*, vol. 36, no. 5, pp. 4848–4851, Sep. 2021.

[157] D. Cao et al., "Data-driven multi-agent deep reinforcement learning for distribution system decentralized voltage control with high penetration of PVs," *IEEE Trans. Smart Grid*, vol. 12, no. 5, pp. 4137–4150, Sep. 2021.

[158] H. T. Nguyen and D.-H. Choi, "Three-stage inverter-based peak shaving and Volt-VAR control in active distribution networks using online safe deep reinforcement learning," *IEEE Trans. Smart*

Grid, vol. 13, no. 4, pp. 3266–3277, Jul. 2022.
[159] R. Huang et al., "Learning and fast adaptation for grid emergency control via deep meta reinforcement learning," *IEEE Trans. Power Syst.*, vol. 37, no. 6, pp. 4168–4178, Nov. 2022.
[160] L. Xi, L. Yu, Y. Xu, S. Wang, and X. Chen, "A novel multi-agent DDQN-AD method-based distributed strategy for automatic generation control of integrated energy systems," *IEEE Trans. Sustain. Energy*, vol. 11, no. 4, pp. 2417–2426, Oct. 2020.
[161] L. Xi, J. Wu, Y. Xu, and H. Sun, "Automatic generation control based on multiple neural networks with actor-critic strategy," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 6, pp. 2483–2493, Jun. 2021.
[162] L. Xi, L. Zhou, Y. Xu, and X. Chen, "A multi-step unified reinforcement learning method for automatic generation control in multi-area interconnected power grid," *IEEE Trans. Sustain. Energy*, vol. 12, no. 2, pp. 1406–1415, Apr. 2021.
[163] Z. Yan and Y. Xu, "Data-driven load frequency control for stochastic power systems: A deep reinforcement learning method with continuous action search," *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 1653–1656, Mar. 2019.
[164] Z. Yan and Y. Xu, "A multi-agent deep reinforcement learning method for cooperative load frequency control of a multi-area power system," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4599–4608, Nov. 2020.
[165] M. H. Khooban and M. Gheisarnejad, "A novel deep reinforcement learning controller based type-II fuzzy system: Frequency regulation in microgrids," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 5, no. 4, pp. 689–699, Aug. 2021.
[166] C. Chen, M. Cui, F. Li, S. Yin, and X. Wang, "Model-free emergency frequency control based on reinforcement learning," *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2336–2346, Apr. 2021.
[167] Z. Yan, Y. Xu, Y. Wang, and X. Feng, "Deep reinforcement learning-based optimal data-driven control of battery energy storage for power system frequency support," *IET Gener., Transmiss. Distrib.*, vol. 14, no. 25, pp. 6071–6078, Dec. 2020.
[168] G. Zhang, W. Hu, D. Cao, Q. Huang, Z. Chen, and F. Blaabjerg, "A novel deep reinforcement learning enabled sparsity promoting adaptive control method to improve the stability of power systems with wind energy penetration," *Renew. Energy*, vol. 178, pp. 363–376, Nov. 2021.
[169] R. Yan, Y. Wang, Y. Xu, and J. Dai, "A multiagent quantum deep reinforcement learning method for distributed frequency control of islanded microgrids," *IEEE Trans. Control Netw. Syst.*, vol. 9, no. 4, pp. 1622–1632, Dec. 2022.
[170] M. Shahidehpour, H. Yamin, and Z. Li, *Market Operations in Electric Power Systems: Forecasting, Scheduling, and Risk Management*. Hoboken, NJ, USA: Wiley, 2002.
[171] Y. Liu, D. Zhang, and H. B. Gooi, "Data-driven decision-making strategies for electricity retailers: A deep reinforcement learning approach," *CSEE J. Power Energy Syst.*, vol. 7, no. 2, pp. 358–367, Mar. 2021.
[172] Y. Ye, D. Qiu, M. Sun, D. Papadaskalopoulos, and G. Strbac, "Deep reinforcement learning for strategic bidding in electricity markets," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1343–1355, Mar. 2020.
[173] H. Xu, H. Sun, D. Nikovski, S. Kitamura, K. Mori, and H. Hashimoto, "Deep reinforcement learning for joint bidding and pricing of load serving entity," *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6366–6375, Nov. 2019.
[174] Y. Du, F. Li, H. Zandi, and Y. Xue, "Approximating Nash equilibrium in day-ahead electricity market bidding with multi-agent deep reinforcement learning," *J. Modern Power Syst. Clean Energy*, vol. 9, no. 3, pp. 534–544, May 2021.
[175] X. Wei, Y. Xiang, J. Li, and J. Liu, "Wind power bidding coordinated with energy storage system operation in real-time electricity market: A maximum entropy deep reinforcement learning approach," *Energy Rep.*, vol. 8, pp. 770–775, Apr. 2022.
[176] X. Wei, Y. Xiang, J. Li, and X. Zhang, "Self-dispatch of wind-storage integrated system: A deep reinforcement learning approach," *IEEE Trans. Sustain. Energy*, vol. 13, no. 3, pp. 1861–1864, Jul. 2022.
[177] Y. Liang, C. Guo, Z. Ding, and H. Hua, "Agent-based modeling in electricity market using deep deterministic policy gradient algorithm," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4180–4192, Nov. 2020.
[178] H. Guo, Q. Chen, Q. Xia, and C. Kang, "Deep inverse reinforcement learning for objective function identification in bidding models," *IEEE Trans. Power Syst.*, vol. 36, no. 6, pp. 5684–5696, Nov. 2021.
[179] M. Sanayha and P. Vateekul, "Model-based deep reinforcement learning for wind energy bidding," *Int. J. Electr. Power Energy Syst.*, vol. 136, Mar. 2022, Art. no. 107625.
[180] Y. Tao, J. Qiu, and S. Lai, "Deep reinforcement learning based bidding strategy for EVAs in local energy market considering information asymmetry," *IEEE Trans. Ind. Informat.*, vol. 18, no. 6, pp. 3831–3842, Jun. 2022.
[181] A. Taghizadeh, M. Montazeri, and H. Kebriaei, "Deep reinforcement learning-aided bidding strategies for transactive energy market," *IEEE Syst. J.*, vol. 16, no. 3, pp. 4445–4453, Sep. 2022.
[182] I. Boukas et al., "A deep reinforcement learning framework for continuous intraday market bidding," *Mach. Learn.*, vol. 110, no. 9, pp. 2335–2387, Sep. 2021.
[183] Y. Zhang, Z. Zhang, Q. Yang, D. An, D. Li, and C. Li, "EV charging bidding by multi-DQN reinforcement learning in electricity auction market," *Neurocomputing*, vol. 397, pp. 404–414, Jul. 2020.
[184] L. Yang, Q. Sun, N. Zhang, and Y. Li, "Indirect multi-energy transactions of energy Internet with deep reinforcement learning approach," *IEEE Trans. Power Syst.*, vol. 37, no. 5, pp. 4067–4077, Sep. 2022.
[185] C. Schlereth, B. Skiera, and F. Schulz, "Why do consumers prefer static instead of dynamic pricing plans? An empirical study for a better understanding of the low preferences for time-variant pricing plans," *Eur. J. Oper. Res.*, vol. 269, no. 3, pp. 1165–1179, Sep. 2018.
[186] D. Liu, W. Wang, L. Wang, H. Jia, and M. Shi, "Dynamic pricing strategy of electric vehicle aggregators based on DDPG reinforcement learning algorithm," *IEEE Access*, vol. 9, pp. 21556–21566, 2021.
[187] D. Qiu, Y. Ye, D. Papadaskalopoulos, and G. Strbac, "A deep reinforcement learning method for pricing electric vehicles with discrete charging levels," *IEEE Trans. Ind. Appl.*, vol. 56, no. 5, pp. 5901–5912, Sep. 2020.
[188] H. Xu, J. Wen, Q. Hu, J. Shu, J. Lu, and Z. Yang, "Energy procurement and retail pricing of electricity retailers via deep reinforcement learning with long short-term memory," *CSEE J. Power Energy Syst.*, vol. 8, no. 5, pp. 1338–1351, Sep. 2022.
[189] S. Lee and D.-H. Choi, "Dynamic pricing and energy management for profit maximization in multiple smart electric vehicle charging stations: A privacy-preserving deep reinforcement learning approach," *Appl. Energy*, vol. 304, Dec. 2021, Art. no. 117754.
[190] A. Abdalrahman and W. Zhuang, "Dynamic pricing for differentiated PEV charging services using deep reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1415–1427, Feb. 2022.
[191] Y.-C. Chuang and W.-Y. Chiu, "Deep reinforcement learning based pricing strategy of aggregators considering renewable energy," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 6, no. 3, pp. 499–508, Jun. 2022.
[192] T. Lu, X. Chen, M. B. McElroy, C. P. Nielsen, Q. Wu, and Q. Ai, "A reinforcement learning-based decision system for electricity pricing plan selection by smart grid end users," *IEEE Trans. Smart Grid*, vol. 12, no. 3, pp. 2176–2187, May 2021.
[193] V. Moghaddam, A. Yazdani, H. Wang, D. Parlevliet, and F. Shahnia, "An online reinforcement learning approach for dynamic pricing of electric vehicle charging stations," *IEEE Access*, vol. 8, pp. 130305–130313, 2020.
[194] L. Zhang, Y. Gao, H. Zhu, and L. Tao, "Bi-level stochastic real-time pricing model in multi-energy generation system: A reinforcement learning approach," *Energy*, vol. 239, Jan. 2022, Art. no. 121926.
[195] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Trans. Depend. Secure Comput.*, vol. 15, no. 5, pp. 840–852, Sep. 2018.
[196] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3154–3164, Dec. 2017.
[197] R. Khalid, N. Javaid, A. Almogren, M. U. Javed, S. Javaid, and M. Zuair, "A blockchain-based load balancing in decentralized hybrid P2P energy trading market in smart grid," *IEEE Access*, vol. 8, pp. 47047–47062, 2020.
[198] A. A. Al-Obaidi and H. E. Z. Farag, "Decentralized quality of service based system for energy trading among electric vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 6586–6595, Jul. 2022.
[199] Y. Li, C. Yu, Y. Liu, Z. Ni, L. Ge, and X. Li, "Collaborative operation between power network and hydrogen fueling stations with peer-to-peer energy trading," *IEEE Trans. Transport. Electrific.*, vol. 9, no. 1, pp. 1521–1540, Mar. 2023.
[200] D. Wang, B. Liu, H. Jia, Z. Zhang, J. Chen, and D. Huang, "Peer-to-peer electricity transaction decisions of the user-side smart energy system based on the SARSA reinforcement learning," *CSEE J. Power Energy Syst.*, vol. 8, no. 3, pp. 826–837, May 2022.
[201] Y. Liu, D. Zhang, C. Deng, and X. Wang, "Deep reinforcement learning approach for autonomous agents in consumer-centric electricity market," in *Proc. 5th IEEE Int. Conf. Big Data Anal. (ICBDA)*, May 2020, pp. 37–41.
[202] D. Qiu, Y. Ye, D. Papadaskalopoulos, and G. Strbac, "Scalable coordinated management of peer-to-peer energy trading: A multi-cluster deep reinforcement learning approach," *Appl. Energy*, vol. 292, Jun. 2021, Art. no. 116940.
[203] C. Samende, J. Cao, and Z. Fan, "Multi-agent deep deterministic policy gradient algorithm for peer-to-peer energy trading considering distribution network constraints," *Appl. Energy*, vol. 317, Jul. 2022, Art. no. 119123.
[204] J. Li et al., "Energy trading of multiple virtual power plants using deep reinforcement learning," in *Proc. Int. Conf. Power Syst. Technol. (POWERCON)*, Dec. 2021, pp. 892–897.
[205] D. Qiu, J. Wang, J. Wang, and G. Strbac, "Multi-agent reinforcement learning for automated peer-to-peer energy trading in double-side auction market," in *Proc. 13th Int. Joint Conf. Artif. Intell.*, Aug. 2021, pp. 2913–2920.
[206] T. Zhang, D. Yue, L. Yu, C. Dou, and X. Xie, "Joint energy and workload scheduling for fog-assisted multimicrogrid systems: A deep reinforcement learning approach," *IEEE Syst. J.*, vol. 17, no. 1, pp. 164–175, Mar. 2023.
[207] Y. Ye, Y. Tang, H. Wang, X.-P. Zhang, and G. Strbac, "A scalable privacy-preserving multi-agent deep reinforcement learning approach for large-scale peer-to-peer transactive energy trading," *IEEE Trans. Smart Grid*, vol. 12, no. 6, pp. 5185–5200, Nov. 2021.
[208] X. Wang, Y. Liu, J. Zhao, C. Liu, J. Liu, and J. Yan, "Surrogate model enabled deep reinforcement learning for hybrid energy community operation," *Appl. Energy*, vol. 289, May 2021, Art. no. 116722.
[209] Y. Xu, L. Yu, G. Bi, M. Zhang, and C. Shen, "Deep reinforcement learning and blockchain for

peer-to-peer energy trading among microgrids," in *Proc. Int. Conferences Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData) IEEE Congr. Cybermatics (Cybermatics)*, Nov. 2020, pp. 360–365.

[210] Y. Li, X. Wei, Y. Li, Z. Dong, and M. Shahidehpour, "Detection of false data injection attacks in smart grid: A secure federated deep learning approach," *IEEE Trans. Smart Grid*, vol. 13, no. 6, pp. 4862–4872, Nov. 2022.

[211] Z. Li, M. Shahidehpour, and F. Aminifar, "Cybersecurity in distributed power systems," *Proc. IEEE*, vol. 105, no. 7, pp. 1367–1388, Jul. 2017.

[212] M. Shahidehpour, F. Tinney, and Y. Fu, "Impact of security on power systems operation," *Proc. IEEE*, vol. 93, no. 11, pp. 2013–2025, Nov. 2005.

[213] Z. Zhang, S. Huang, Y. Chen, B. Li, and S. Mei, "Cyber-physical coordinated risk mitigation in smart grids based on attack-defense game," *IEEE Trans. Power Syst.*, vol. 37, no. 1, pp. 530–542, Jan. 2022.

[214] T. Bailey, J. Johnson, and D. Levin, "Deep reinforcement learning for online distribution power system cybersecurity protection," in *Proc. IEEE Int. Conf. Commun., Control, Comput. Technol. Smart Grids (SmartGridComm)*, Oct. 2021, pp. 227–232.

[215] X. Liu, J. Ospina, and C. Konstantinou, "Deep reinforcement learning for cybersecurity assessment of wind integrated power systems," *IEEE Access*, vol. 8, pp. 208378–208394, 2020.

[216] Y. Li and J. Wu, "Low latency cyberattack detection in smart grids with deep reinforcement learning," *Int. J. Electr. Power Energy Syst.*, vol. 142, Nov. 2022, Art. no. 108265.

[217] D. An, F. Zhang, Q. Yang, and C. Zhang, "Data integrity attack in dynamic state estimation of smart grid: Attack model and countermeasures," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 3, pp. 1631–1644, Jul. 2022.

[218] C. Chen, M. Cui, X. Fang, B. Ren, and Y. Chen, "Load altering attack-tolerant defense strategy for load frequency control system," *Appl. Energy*, vol. 280, Dec. 2020, Art. no. 116015.

[219] W. Lei, H. Wen, J. Wu, and W. Hou, "MADDPG-based security situational awareness for smart grid with intelligent edge," *Appl. Sci.*, vol. 11, no. 7, p. 3101, Mar. 2021.

[220] Z. Jin et al., "Cyber-physical risk driven routing planning with deep reinforcement-learning in smart grid communication networks," in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, Jun. 2020, pp. 1278–1283.

[221] H. Zhang, D. Yue, C. Dou, and G. P. Hancke, "Resilient optimal defensive strategy of micro-grids system via distributed deep reinforcement learning approach against FDI attack," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, May 27, 2022, doi: 10.1109/TNNLS.2022.3175917.

[222] D. An, Q. Yang, W. Liu, and Y. Zhang, "Defending against data integrity attacks in smart grid: A deep reinforcement learning-based approach," *IEEE Access*, vol. 7, pp. 110835–110845, 2019.

[223] Y. Wang, Q. Chen, T. Hong, and C. Kang, "Review of smart meter data analytics: Applications, methodologies, and challenges," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 3125–3148, May 2019.

[224] A. Mohammadali and M. S. Haghighi, "A privacy-preserving homomorphic scheme with multiple dimensions and fault tolerance for metering data aggregation in smart grid," *IEEE Trans. Smart Grid*, vol. 12, no. 6, pp. 5212–5220, Nov. 2021.

[225] C. E. Kement, B. Tavli, H. Gultekin, and H. Yanikomeroglu, "Holistic privacy for electricity, water, and natural gas metering in next generation smart homes," *IEEE Commun. Mag.*, vol. 59, no. 3, pp. 24–29, Mar. 2021.

[226] Z. Zheng, T. Wang, A. K. Bashir, M. Alazab, S. Mumtaz, and X. Wang, "A decentralized mechanism based on differential privacy for privacy-preserving computation in smart grid," *IEEE Trans. Comput.*, vol. 71, no. 11, pp. 2915–2926, Nov. 2022.

[227] S. Lee and D.-H. Choi, "Federated reinforcement learning for energy management of multiple smart homes with distributed energy resources," *IEEE Trans. Ind. Informat.*, vol. 18, no. 1, pp. 488–497, Jan. 2022.

[228] H.-M. Chung, S. Maharjan, Y. Zhang, and F. Eliassen, "Distributed deep reinforcement learning for intelligent load scheduling in residential smart grids," *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2752–2763, Apr. 2021.

[229] M. Shateri, F. Messina, P. Piantanida, and F. Labeau, "Privacy-cost management in smart meters with mutual-information-based reinforcement learning," *IEEE Internet Things J.*, vol. 9, no. 22, pp. 22389–22398, Nov. 2022.

[230] Z. Su et al., "Secure and efficient federated learning for smart grid with edge-cloud collaboration," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1333–1344, Feb. 2022.

[231] X. Wang et al., "QoS and privacy-aware routing for 5G-enabled industrial Internet of Things: A federated reinforcement learning approach," *IEEE Trans. Ind. Informat.*, vol. 18, no. 6, pp. 4189–4197, Jun. 2022.

[232] Z. Wang, Y. Liu, Z. Ma, X. Liu, and J. Ma, "LiPSG: Lightweight privacy-preserving Q-learning-based energy management for the IoT-enabled smart grid," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3935–3947, May 2020.

[233] Y. Zhang, Q. Ai, and Z. Li, "Intelligent demand response resource trading using deep reinforcement learning," *CSEE J. Power Energy Syst.*, early access, Sep. 10, 2021, doi: 10.17775/CSEEJPES.2020.05540.

[234] X. Liu, H. Wang, G. Chen, B. Zhou, and A. U. Rehman, "Intermittently differential privacy in smart meters via rechargeable batteries," *Electr. Power Syst. Res.*, vol. 199, Oct. 2021, Art. no. 107410.

[235] U. Ahmed, J. C. Lin, and G. Srivastava, "5G-empowered drone networks in federated and deep reinforcement learning environments," *IEEE Commun. Standards Mag.*, vol. 5, no. 4, pp. 55–61, Dec. 2021.

[236] L. Yan, X. Chen, Y. Chen, and J. Wen, "A hierarchical deep reinforcement learning-based community energy trading scheme for a neighborhood of smart households," *IEEE Trans. Smart Grid*, vol. 13, no. 6, pp. 4747–4758, Nov. 2022.

[237] T. Li, Y. Xiao, and L. Song, "Integrating future smart home operation platform with demand side management via deep reinforcement learning," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 2, pp. 921–933, Jun. 2021.

[238] J. García and F. Fernández, "A comprehensive survey on safe reinforcement learning," *J. Mach. Learn. Res.*, vol. 16, no. 42, pp. 1437–1480, Aug. 2015.

[239] X. Wang, R. Wang, and Y. Cheng, "Safe reinforcement learning: A survey," *Acta Automatica Sinica*, vol. 49, no. 9, pp. 1–23, Sep. 2023.

[240] Z. Yi et al., "An improved two-stage deep reinforcement learning approach for regulation service disaggregation in a virtual power plant," *IEEE Trans. Smart Grid*, vol. 13, no. 4, pp. 2844–2858, Jul. 2022.

[241] Z. Zhu, K. W. Chan, S. Xia, and S. Bu, "Optimal bi-level bidding and dispatching strategy between active distribution network and virtual alliances using distributed robust multi-agent deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 13, no. 4, pp. 2833–2843, Jul. 2022.

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

[243] Q. Huang, R. Huang, W. Hao, J. Tan, R. Fan, and Z. Huang, "Adaptive power system emergency control using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1171–1182, Mar. 2020.

[244] E. Marchesini, D. Corsi, and A. Farinelli, "Exploring safer behaviors for deep reinforcement learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 7, 2022, pp. 7701–7709.

[245] Y. Ye, H. Wang, P. Chen, Y. Tang, and G. Strbac, "Safe deep reinforcement learning for microgrid energy management in distribution networks with leveraged spatial–temporal perception," *IEEE Trans. Smart Grid*, vol. 14, no. 5, pp. 3759–3775, Sep. 2023.

[246] H. Cui, Y. Ye, J. Hu, Y. Tang, Z. Lin, and G. Strbac, "Online preventive control for transmission overload relief using safe reinforcement learning with enhanced spatial–temporal awareness," *IEEE Trans. Power Syst.*, early access, Mar. 15, 2023, doi: 10.1109/TPWRS.2023.3257259.

[247] R. F. Prudencio, M. R. O. A. Maximo, and E. L. Colombini, "A survey on offline reinforcement learning: Taxonomy, review, and open problems," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Mar. 22, 2023, doi: 10.1109/TNNLS.2023.3250269.

[248] H. Niu, Y. Qiu, M. Li, G. Zhou, J. Hu, and X. Zhan, "When to trust your simulator: Dynamics-aware hybrid offline-and-online reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 36599–36612.

[249] Z. Yan and Y. Xu, "A hybrid data-driven method for fast solution of security-constrained optimal power flow," *IEEE Trans. Power Syst.*, vol. 37, no. 6, pp. 4365–4374, Nov. 2022.

[250] A. R. Sayed, C. Wang, H. Anis, and T. Bi, "Feasibility constrained online calculation for real-time optimal power flow: A convex constrained deep reinforcement learning approach," *IEEE Trans. Power Syst.*, early access, Nov. 9, 2022, doi: 10.1109/TPWRS.2022.3220799.

[251] T. L. Vu, S. Mukherjee, R. Huang, and Q. Huang, "Barrier function-based safe reinforcement learning for emergency control of power systems," in *Proc. 60th IEEE Conf. Decis. Control (CDC)*, Dec. 2021, pp. 3652–3657.

[252] I. Ilahi et al., "Challenges and countermeasures for adversarial attacks on deep reinforcement learning," *IEEE Trans. Artif. Intell.*, vol. 3, no. 2, pp. 90–109, Apr. 2022.

[253] Y. Wang and B. Pal, "Destabilizing attack and robust defense for inverter-based microgrids by adversarial deep reinforcement learning," *IEEE Trans. Smart Grid*, early access, Mar. 30, 2023, doi: 10.1109/TSG.2023.3263243.

[254] S. Paul, Z. Ni, and C. Mu, "A learning-based solution for an adversarial repeated game in cyber-physical power systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4512–4523, Nov. 2020.

[255] K. L. Tan, Y. Esfandiari, X. Y. Lee, and S. Sarkar, "Robustifying reinforcement learning agents via action space adversarial training," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2020, pp. 3959–3964.

[256] H. Zhang et al., "Robust deep reinforcement learning against adversarial perturbations on state observations," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 33, 2020, pp. 21024–21037.

[257] S. Li, Y. Wu, X. Cui, H. Dong, F. Fang, and S. Russell, "Robust multi-agent reinforcement learning via minimax deep deterministic policy gradient," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 4213–4220.

[258] H. Dong and X. Zhao, "Wind-farm power tracking via preview-based robust reinforcement learning," *IEEE Trans. Ind. Informat.*, vol. 18, no. 3, pp. 1706–1715, Mar. 2022.

[259] A. Roy, H. Xu, and S. Pokutta, "Reinforcement learning under model mismatch," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 3046–3055.

[260] Y. Li, R. Wang, Y. Li, M. Zhang, and C. Long, "Wind power forecasting considering data privacy protection: A federated deep reinforcement learning approach," *Appl. Energy*, vol. 329, Jan. 2023, Art. no. 120291.

[261] Y. Li, S. He, Y. Li, Y. Shi, and Z. Zeng, "Federated

multiagent deep reinforcement learning approach via physics-informed reward for multimicrogrid energy management," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jan. 3, 2023, doi: 10.1109/TNNLS.2022.3232630.

[262] P. Rashidinejad, B. Zhu, C. Ma, J. Jiao, and S. Russell, "Bridging offline reinforcement learning and imitation learning: A tale of pessimism," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 11702–11716.

[263] X. Chen, Z. Zhou, Z. Wang, C. Wang, Y. Wu, and K. Ross, "BAIL: Best-action imitation learning for batch deep reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 18353–18363.

[264] H. Liu, Z. Huang, J. Wu, and C. Lv, "Improved deep reinforcement learning with expert demonstrations for urban autonomous driving," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2022, pp. 921–928.

[265] Y. Liu, Q. Liu, H. Zhao, Z. Pan, and C. Liu, "Adaptive quantitative trading: An imitative deep reinforcement learning approach," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 2, Apr. 2020, pp. 2128–2135.

[266] G. Krishnamoorthy, A. Dubey, and A. H. Gebremedhin, "Reinforcement learning for battery energy storage dispatch augmented with model-based optimizer," in *Proc. IEEE Int. Conf. Commun., Control, Comput. Technol. Smart Grids (SmartGridComm)*, Oct. 2021, pp. 289–294.

[267] W. Guo, W. Tian, Y. Ye, L. Xu, and K. Wu, "Cloud resource scheduling with deep reinforcement learning and imitation learning," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3576–3586, Mar. 2021.

[268] D. Silver et al., "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, Oct. 2017.

ABOUT THE AUTHORS

Yuanzheng Li (Senior Member, IEEE) received the M.S. degree in electrical engineering from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2011, and the Ph.D. degree in electrical engineering from the South China University of Technology (SCUT), Guangzhou, China, in 2015.

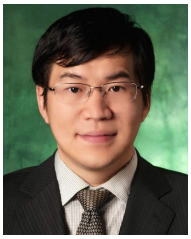
He is currently an Associate Professor with HUST. He has published several peer-reviewed articles in international journals. His current research interests include deep learning, reinforcement learning, smart grid operations, optimal power system/microgrid scheduling and decision-making, stochastic optimization considering large-scale integration of renewable energy into the power system, and multiobjective optimization.

Chaofan Yu received the B.S. degree in automation from Guangxi University (GXU), Nanning, China, in 2020. He is currently working toward the M.S. degree at the China-EU Institute for Clean and Renewable Energy, Huazhong University of Science and Technology (HUST), Wuhan, China.

His current research interests include electric vehicles, optimal scheduling of large-scale renewable energy integrated power system, and artificial intelligence and its application in the smart grid.

Mohammad Shahidehpour (Life Fellow, IEEE) is currently a University Distinguished Professor, a Bodine Chair Professor of Electrical and Computer Engineering, and the Director of the Robert W. Galvin Center for Electricity Innovation, Illinois Institute of Technology (IIT), Chicago, IL, USA. He has 45 years of experience with electric power system operation and planning. His sponsored project on perfect power systems has converted the entire IIT campus to an islandable microgrid. He has coauthored six books and more than 800 technical articles on electric power system operation and planning.

Dr. Shahidehpour is a member of the National Academy of Engineering and a Fellow of the American Association for the Advancement of Science and the National Academy of Inventors. He received the IEEE Burke Hayes Award for his research on hydrokinetics, the IEEE Power and Energy Society (PES) Outstanding Power Engineering Educator Award, the IEEE/PES Ramakumar Family Renewable Energy Excellence Award, the IEEE/PES Douglas M. Staszesky Distribution Automation Award, and the Edison Electric Institute's Power Engineering Educator Award. He was the founding Editor-in-Chief of IEEE TRANSACTIONS ON SMART GRID.



Tao Yang (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from Washington State University, Pullman, WA, USA, in 2012.

From August 2012 to August 2014, he was an ACCESS Postdoctoral Researcher with the ACCESS Linnaeus Centre, Royal Institute of Technology, Stockholm, Sweden. He then joined the Pacific Northwest National Laboratory, Richland, WA, USA, as a Postdoctor, where he was promoted to a Scientist/Engineer II in 2015. He was an Assistant Professor with the Department of Electrical Engineering, The University of North Texas, Denton, TX, USA, from 2016 to 2019. He is currently a Professor with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, China. His research interests include industrial artificial intelligence, integrated optimization and control, distributed control and optimization with applications to process industries, cyber-physical systems, networked control systems, and multiagent systems.

Dr. Yang received the Ralph E. Powe Junior Faculty Enhancement Award and the Best Student Paper Award (as an advisor) at several international conferences. He is an Associate Editor of IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY and IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.

Zhigang Zeng (Fellow, IEEE) received the Ph.D. degree in systems analysis and integration from the Huazhong University of Science and Technology, Wuhan, China, in 2003.

He is currently a Professor with the School of Automation and the Key Laboratory of Image Processing and Intelligent Control of the Education Ministry of China, Huazhong University of Science and Technology. He has published more than 100 international journal articles. His current research interests include the theory of functional differential equations and differential equations with discontinuous right-hand sides and their applications to dynamics of neural networks, memristive systems, and control systems.

Dr. Zeng has been a member of the Editorial Board of *Neural Networks* since 2012, *Cognitive Computation* since 2010, and *Applied Soft Computing* since 2013. He was an Associate Editor of IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS from 2010 to 2011. He has been an Associate Editor of IEEE TRANSACTIONS ON CYBERNETICS since 2014 and IEEE TRANSACTIONS ON FUZZY SYSTEMS since 2016.

Tianyou Chai (Life Fellow, IEEE) received the Ph.D. degree in control theory and engineering from Northeastern University, Shenyang, China, in 1985.

He became a Professor at Northeastern University in 1988. He is the Founder and the Director of the Center of Automation, Northeastern University, which became the National Engineering and Technology Research Center and the State Key Laboratory. He was the Director of the Department of Information Science, National Natural Science Foundation of China, from 2010 to 2018. He has developed control technologies with applications to various industrial processes. He has published more than 320 peer-reviewed international journal articles. His current research interests include modeling, control, optimization, and integrated automation of complex industrial processes.

Dr. Chai is a member of the Chinese Academy of Engineering and a Fellow of International Federation for Automatic Control (IFAC). His paper titled "Hybrid intelligent control for optimal operation of shaft furnace roasting process" was selected as one of the three best papers for the Control Engineering Practice Paper Prize for the term 2011–2013. For his contributions, he has won five prestigious awards of the National Natural Science, the National Science and Technology Progress, and the National Technological Innovation, the 2007 Industry Award for Excellence in Transitional Control Research from IEEE Multi-Conference on Systems and Control, and the 2017 Wook Hyun Kwon Education Award from the Asian Control Association.