IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 15 July 2024, accepted 10 August 2024, date of publication 16 August 2024, date of current version 26 August 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3445143*

RESEARCH ARTICLE

# A Distributed Deep Reinforcement Learning Approach for Reactive Power Optimization of Distribution Networks

**JINLIN LIAO**[1,2,3] **AND JIA LIN**[4]

[1]Economic Technology Research Institute, State Grid Fujian Electric Power Company Ltd., Fuzhou 350013, China
[2]Distribution Network Planning and Operation Control Technology in Multiple Disaster Superimposed Areas State Grid Corporation Laboratory, Fuzhou 350013, China
[3]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[4]State Grid Fujian Electric Power Company Ltd., Fuzhou 350001, China

Corresponding author: Jinlin Liao (liaojinlin@alumni.sjtu.edu.cn)

This work was supported by the Science and Technology Project of State Grid Fujian Electric Power Company Ltd., under Grant SGFJJY00GHJS2250008.

**ABSTRACT** An actor-critic based distributed deep reinforcement learning approach is proposed to optimize the reactive power of the distribution network under the access of distributed photovoltaics, wind turbines and other power sources. This approach can optimize and dispatch the resources of the distribution network in real time under the change of power output such as distributed photovoltaics and wind turbines, so as to optimize the reactive power of the distribution network. First, this paper builds an optimization model with the objective function of minimizing the reactive power of the distribution network, and considers the operating constraints. Then, the agents of the proposed approach are trained, and the well-trained agents can schedule and optimize the resources of the distribution network in real time. Finally, based on the actual source-load output data in a certain place, reactive power optimization simulation experiments are carried out on the IEEE 33-bus, IEEE 123-bus simulation systems and the actual power distribution system in a region of China. Simulation results show that the proposed distributed deep reinforcement learning approach (DDRLA) can optimize distribution network reactive power online in real time.

**INDEX TERMS** Actor-critic, distributed power generation, distribution network, reactive power optimization, distributed deep reinforcement learning.

## I. INTRODUCTION

With the large-scale wind power, photovoltaic and other distributed power sources accessing the distribution network in large quantities, the proportion of power sources with uncertain output in the system is gradually increasing, and the distribution network is facing challenges in the consumption of distributed power sources, flexible resource regulation and control, etc. [1]. Due to the obvious fluctuation of wind power and photovoltaic with weather changes, its output has randomness, in addition to the random fluctuation of the load, which brings power quality reduction, network loss increases and other problems to the distribution network, and the security of the distribution system is difficult to ensure [2].

Previously, most reactive power optimization studies focused on a single objective, such as minimizing operating costs [3], active power loss [4], or emissions [5]. With the construction of new power distribution systems, electric vehicles [6], photovoltaic (PV) generations [7], etc. are connected to distribution networks. Researchers' goal has changed from a single-objective reactive power optimization problem to a multi-objective optimization problem [8] that considers operating indicators [9], economic indicators [10], power quality indicators [11], etc. These single-objective optimization problems can be solved by dynamic programming (DP) methods [12], fuzzy decision methods [13], etc.

The associate editor coordinating the review of this manuscript and approving it for publication was Fabio Mottola.

The goal of reactive power optimization in distribution network is to effectively ensure the voltage stability of each node and reduce voltage fluctuation and network loss under the constraints of safe operation of the power grid [14]. Many studies have been carried out on reactive power optimization in traditional power distribution networks, such as genetic algorithm (GA) [15], simulated annealing (SA) algorithm [16], and particle swarm optimization (PSO) algorithm [17], etc. optimized solution. Ai et al. [18] proposed a reactive power optimization algorithm for distribution network with PV generation to address the problems of power quality degradation. A multi-objective reactive power optimization model is established using the Non-dominated Sorting Genetic Algorithm III (NSGA-III) to effectively solve the problem. The experimental results show that except for the proposed NSGA-III where active power reduction can reach about 25%, only less than 10% of PVs active power is reduced in all other cases. Traditional reactive power optimization methods face challenges in achieving global optimality and tend to have slow computational speeds. Li et al. [19] utilized simulated annealing algorithm to solve the mathematical models, which contains PVs, wind turbines, and electric vehicles in active distribution networks. The experimental results based on the on the modified IEEE 14-bus distribution network show that the voltage deviation and system losses are significantly decreased after optimization. Liu et al. [20] proposed an improved PSO algorithm combined with the $\varepsilon$-greedy strategy to solve the multi-objective reactive power optimization model. The simulation results show that in terms of the active power loss and the static voltage stability, the proposed improved PSO algorithm has better reactive power optimization capability compare with the standard PSO algorithm and NSGA-II. Henceforth, the strategy significantly explores the possibility of finding optimal solutions in the local space during the early stages of the iteration. Additionally, it mitigates the tendency to fall into local optima in the later stages of the iteration, enhancing the overall effectiveness and robustness of the optimization process. Linlin et al. [21] proposed a multi-objective reactive power and voltage optimization model and introduced the grey wolf optimization algorithm to effectively improve the system node voltage quality and improve the stable operation level of the system. Niu et al. [22] proposed a new adaptive range composite differential evolution (ARCoDE) algorithm designed to efficiently and accurately solve the optimal reactive power dispatch (ORPD) problem. Thanks to a novel adaptive range strategy for control parameters, the proposed ARCoDE algorithm excels in both exploration and exploitation. It can effectively manage the ORPD problem, which includes complex constraints and a mix of discrete and continuous variables. Saddique et al. [23] proposed a novel algorithm called the Sine-Cosine Algorithm (SCA) is employed to solve the ORPD problem. To demonstrate the superiority of the proposed algorithm, its results are compared with recently published outcomes obtained using PSO, modified Gaussian Barebones Teaching-Learning Based Optimization (BBTLBO), Ant Bee Colony Optimization (ABCO), Whale Optimization Algorithm (WOA), and Backtracking Search Algorithm (BSA). The results achieved with SCA indicate a significant improvement in power loss minimization. The analysis clearly shows that the proposed algorithm is robust, effective, and computationally efficient in solving the ORPD problem compared to existing meta-heuristic algorithms.

However, these methods have disadvantages such as slow calculation speed, easy to fall into local optimum, and dependence on models and prediction data [24]. With the increase of distribution network scale and the number of reactive power controllable devices, the complexity of these methods for solving reactive power optimization problems is greatly increased [25], and they are no longer suitable for real-time online reactive power optimization problems.

In recent years, artificial intelligence technology [26] in the field of reinforcement learning has been rapidly developed. The principle of deep reinforcement learning (DRL) [27] is that the agent constantly gets feedback from the environment during interaction with the environment, and constantly tries to make mistakes and learns according to the feedback from the environment in order to optimize the decision. A well-trained agent can give optimization strategies in real time according to the changes in the environment. In distribution network reactive power optimization, a well-trained agent can adapt itself to the uncertainty of source load [28] and optimize the distribution network reactive power and network loss.

The process of the agent interacting with the distribution network environment in the distribution network is called Markov Decision Process (MDP) [29], in which the agent makes different optimization strategies according to different environments in the distribution network, i.e., adjusts reactive power regulation equipment according to different source-load-output situations in the distribution network. The agent gets different network losses under different actions, and through continuous interaction and learning with the environment, the final trained agent is able to adjust the reactive power regulation equipment in real time according to the changes of the source-load-output of the distribution network to optimize the return value [30]. Therefore, this paper proposes an Actor-Critic based Distributed Deep Reinforcement Learning Approach (DDRLA) for the distribution network reactive power optimization and voltage fluctuation problem considering distributed power sources and load uncertainty. The proposed method ensures that the voltage fluctuation of the distribution network optimize the grid network loss under the condition that the voltage fluctuation of the distribution network is within the constraint range. Finally, the feasibility of the proposed Actor-Critic based distributed deep reinforcement learning method is verified through example proofs and reactive power optimization simulation experiments on the improved the IEEE 33-bus simulation system.

The shortcomings of the proposed DDRLA are as follows. Actor-Critic involves two neural networks. Both neural networks update parameters in a continuous state each time. Moreover, there is a correlation before and after each parameter update of the neural network. This results in the neural network potentially having difficulty converging. The contributions are summarized as follows.

1) Better than traditional heuristic optimization method, GA, the DDRLA can adaptively adjust the actions of SC, OLTC, and DG in real time to optimize reactive power.

2) Compared with DQN and DDPG, DDRLA performs better reactive power optimization effects during testing.

3) DDRLA has the capability to be applied to large-scale power distribution systems.

The remainder of this paper is organized as follows. Section II introduces the distributed deep reinforcement learning approach based on Actor-Critic. In Section III, the model for reactive power optimization problem in distribution network is proposed. In Section IV, three experiments are designed and conducted to validate smaller network loss, voltage deviation effect and a real-time performance of the proposed DDRLA. Section V concludes this paper.

## II. DISTRIBUTED DEEP REINFORCEMENT LEARNING APPROACH BASED ON ACTOR-CRITIC

The essence of reinforcement learning lies in interactive learning, that is, the agent can interact with the environment. The agent selects the corresponding action according to the current environment state to respond to the environment. The environment gives the corresponding reward value to the agent's action, and the agent adjusts the action strategy according to the reward value. The agent's goal is to optimize the expected value of the cumulative reward sum in constant interaction with the environment. By continuously interacting with the environment and learning through trial and error, the agent can choose an action strategy to optimize the environment to optimize the reward.

### A. ABBREVIATIONS

DP Dynamic programming
GA Genetic algorithm
SA Simulated annealing
PSO Particle swarm optimization
NSGA-III Non-dominated sorting genetic algorithm
DRL Deep reinforcement learning
MDP Markov Decision Process
KL Kullback–Leibler
DDRLA Distributed Deep Reinforcement Learning Approach
SC Switching Capacitor
OLTC On-Line Tap Changer
DG Distributed Generation
DQN Deep Q-Network
DDPG Deep Deterministic Policy Gradient

### B. MARKOV DECISION PROCESS

Reinforcement learning is a type of feedback-based learning [31]. It is modelled using the Markov decision process: a Markov decision process is represented by a quintuple $\langle S, A, R, T, \gamma \rangle$, where $S$ represents the state space, $A$ represents the action space, $R$ represents the reward function, $T$ represents the state transfer function, and $\gamma$ is the reward discount factor. The process of the interaction between the agent and the environment can be described as: at time t, the agent observes that the state of the environment is $s_t$, executes an action $a_t$, the agent obtains a reward $r_t$, and the environment moves to the next state $s_{t+1}$. As the agent interacts with the environment, an interaction trajectory $(s_t, a_t, r_t, s_{t+1}, a_{t+1}, r_{t+1}, \cdots)$ is formed.

### C. POLICY GRADIENT

Value-based methods, such as Q-learning [32], train a deep learning network through TD errors, judge the Q value of the action state through the deep learning network, and select the action with the largest Q value through a greedy strategy. The strategy method also uses a deep neural network to evaluate the state, and the ultimate goal of the strategy is to maximize the cumulative return expectation of formula (1):

$$\max \sum_{t=1}^{T} \left(R\left(s_t\right) | \pi_\theta \right) = \max \sum_{t=1}^{T} \left(r_t | s_t, \pi_\theta \right) \tag{1}$$

The core idea of the strategy method [33] is to judge whether an action is good or bad. If the action is good, then increase the probability of this action being selected; otherwise, reduce the probability of this action being selected. The agent interacts with the environment to obtain a periodic interaction sequence $\tau = (s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_T, a_T, r_T)$, and the cumulative return of the trajectory $\tau$ can be expressed as follows:

$$\overset{\wedge}{R}(\tau) = \sum\nolimits_{t=1}^{T} R\left(s_t, a_t\right) \tag{2}$$

The cumulative advantage of the control strategy is as follow:

$$\overset{\wedge}{A}(\tau) = \overset{\wedge}{R}(\tau) - V_\phi(\tau) = \sum\nolimits_{t=1}^{T} R\left(s_t, a_t\right) - E\left[\sum_{\tau} r(\phi)\right] \tag{3}$$

where $V_\phi(\tau)$ is the expected reward value of the Critic network with the parameter $\phi$ at the trajectory $\tau$.

Use $P(\tau, \theta)$ to represent the probability of the trajectory $\tau$ under the parameter $\theta$, then the goal of reinforcement learning can be updated as:

$$\max \Re(\theta) = \sum_{\tau} p(\tau, \theta) R(\tau) \tag{4}$$

Gradient descent and Monte Carlo [34] methods are used to approximate the derivative of the objective function:

$$\theta \leftarrow \theta + \alpha \nabla_\theta \sum_{\tau} \log P(\tau, \theta) R(\tau) \tag{5}$$

Using the Monte Carlo method, the policy gradient is approximated with the experience of $m$ trajectories. Each trajectory contains the complete sequence of actions taken by the agent in the environment from the start state until the termination state to approximate the policy gradient, which can be expressed as:

$$\hat{g} = \frac{1}{m} \sum_{i=1}^{m} \nabla_\theta \sum_{\tau} \log P\left(\tau_i, \theta\right) R\left(\tau_i\right) \tag{6}$$

If is $R(\tau)$ greater than 0, then the probability of trajectory $\tau$ becomes larger; otherwise, the probability becomes smaller.

### D. ACTOR-CRITIC

DDRLA's Actor-Critic network consists of two main components: the actor network and the critic network. The actor network takes the state or observation as input and processes it through several hidden layers to produce the parameters of the action distribution, representing the policy. The critic network also takes the state as input, processes it through hidden layers, and outputs a single value estimating the expected return from that state. These networks can share initial layers or be entirely separate. The actor network's updates are constrained by a clipped objective to ensure stable policy changes, while the critic network minimizes the mean squared error between predicted values and actual returns. Both networks are optimized simultaneously using gradient-based methods to achieve effective and stable learning.

The role of $R(\tau)$ is to evaluate the entire trajectory and control the probability of the trajectory, the formula (5) can be updated to the action strategy gradient, namely:

$$\hat{g} = E\left[\sum_{t=1}^{T} R_t \nabla_\theta \sum_{\tau} \log \pi_\theta\left(a_t \left| s_t\right.\right)\right] \tag{7}$$

where $E[\cdot]$ is the expected value at different strategies corresponding to different probabilities.

This article uses the TD error under the action $a_t$ to evaluate the quality of the action, namely:

$$R_t = \left(V^\pi\left(s_{t+1}\right) + r_t - V^\pi\left(s_t\right) \left| a_t\right.\right) \pi \tag{8}$$

where $V^\pi\left(s_{t+1}\right)$ is the expected reward value of the critic network with strategy $\pi$ in state $s_{t+1}$. And $V^\pi\left(s_t\right) \left| a_t\right.$ is the expected reward value of the critic network with strategy $\pi$ in state $s_t$ after taking action $a_t$.

In Actor-Critic, the optimization method of Actor uses the evaluation of Critic combined with the strategy gradient, and the optimization of Critic uses the value function to approximate $R_t$, then the network update target of the value function is as follows:

$$\min_\theta \left[r_t + R_{t+1}\left(S, \theta\right) - R_t\left(S_{old}, \theta_{old}\right)\right]^2 \tag{9}$$

where $S_{old}$ and $S$ are the states before and after the update, respectively. And $\theta_{old}$ and $\theta$ are the parameters of the actor network before and after the update, respectively.

Deriving formula (8), and add update amplitude control item $\beta KL\left[\pi_{old} \left| \pi_\theta\right.\right]$, the optimization formula is obtained:

$$\hat{g} = \sum_{t=1}^{T-1} r_t + R_{t+1}\left(S, \theta\right) - R_t\left(S_{old}, \theta_{old}\right) - \beta KL\left[\pi_{old} \left| \pi_\theta\right.\right] \tag{10}$$

where $\pi_{old}$ and $\pi_\theta$ denote the strategies before and after the update, respectively. $\beta$ denotes the penalty term coefficient.

This paper utilizes the $KL$ divergence (Kullback-Leibler divergence) [35] to control the update amplitude of the action strategy. The $KL$ divergence is a measure of how different one probability distribution is from another. In Bayesian theory, there is a real distribution $\pi_{old}$, which is estimated by an approximate distribution $\pi_\theta$. The $KL$ divergence measures the distance between the approximate distribution $\pi_\theta$ and the true distribution $\pi_{old}$ on the action space $S$, that is $KL\left[\pi_{old} \left| \pi_\theta\right.\right]$, which can be expressed by formula (11):

$$KL\left[\pi_{old} \left| \pi_\theta\right.\right] = E_{s \sim P}\left[\log \frac{\pi_{old}}{\pi_\theta}\right] \tag{11}$$

If the update amplitude is too large, a larger penalty term $\beta KL\left[\pi_{old} \left| \pi_\theta\right.\right]$ is given, that is, the value of $\beta$ is increased. If the update amplitude is too small, a smaller penalty term $\beta KL\left[\pi_{old} \left| \pi_\theta\right.\right]$ is given, that is, the value of $\beta$ is reduced. As shown in formula (12):

$$\begin{cases} \beta \leftarrow \tilde{\alpha}\,\beta & if\ KL\left[\pi_{old} \left| \pi_\theta\right.\right] > \beta_{high} KL_{t\arg et} \\ \beta \leftarrow \beta / \tilde{\alpha} & if\ KL\left[\pi_{old} \left| \pi_\theta\right.\right] < \beta_{low} KL_{t\arg et} \end{cases} \tag{12}$$

where $\beta_{high} KL_{t\arg et}$ represents the upper control limit of the $KL$ divergence. $\beta_{low} KL_{t\arg et}$ represents the lower control limit of the $KL$ divergence. If the $KL$ divergence is greater than the control upper limit value $\beta_{high} KL_{t\arg et}$, it means that the update speed is too fast. Otherwise, it means that the update speed is too slow. $\tilde{\alpha}$ is a constant greater than 1.

In this paper, the parameters of the Actor network are updated by the update rule of the policy gradient method:

$$\theta = \theta_{old} + \eta_1 \hat{g} \tag{13}$$

where $\theta_{old}$ and $\theta$ are the parameters before and after the update, respectively. $\eta_1$ denotes the update step of the action network.

To update the parameters of the Critic network, the squared error Loss $L_{SEL}(\phi)$ is calculated:

$$L_{SEL}(\phi) = -\sum\nolimits_{t=1}^{T} \left(\overset{\wedge}{R_t} - V_\phi(s_t)\right)^2 \tag{14}$$

Then calculate its gradient $\nabla_\phi L_{SEL}$, and finally also update the parameters of the neural network according to the update rule of the policy gradient method:

$$\phi = \phi_{old} + \eta_2 \nabla_\phi L_{SEL} \tag{15}$$

where $\phi_{old}$ and $\phi$ are the parameters before and after the update, respectively. $\eta_2$ denotes the update step of the critic network.

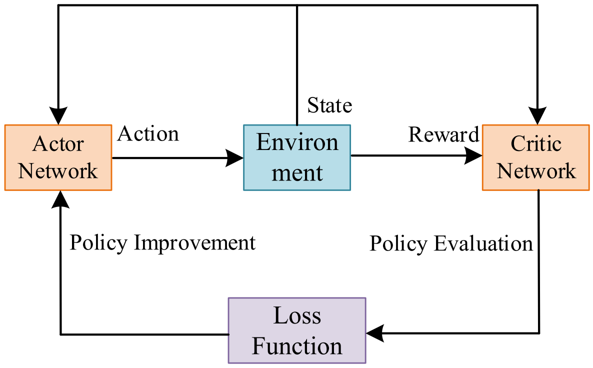

**FIGURE 1.** Structure of actor-critic network.

As shown in Fig. 1, the structure of Actor-Critic network is described as follows.

*Step 1:* Actor interacts with the environment according to the current policy. Actor chooses actions from the current state and interact with the environment, receive rewards and move to the next state. This process will generate a trajectory sequence that contains the agent's behavior in the environment and the results of its interaction with the environment.

*Step 2:* Use the state, action and reward information in the trajectory sequence to calculate the TD error. Based on the TD error, the Critic network is updated. It is an approximator of the value function and improves the accuracy of the value function estimation by minimizing the TD error.

*Step 3:* Use the updated Critic network to evaluate the trajectory sequence. By using the Critic network to estimate the cumulative return of each state, the trajectory sequence is evaluated and the expected return of each state under the current strategy is obtained.

*Step 4:* Use the policy gradient method to update the parameters of the Actor network. The policy gradient method is a gradient ascent method that uses the evaluation results of the Critic network to calculate the gradient, indicating how the probability of selecting actions in different states should be adjusted in order to maximize the expected return. Use this gradient to update the parameters of the actor network to improve the policy.

*Repeat the Above Steps:* The entire training process is an iterative process. And these steps are repeated multiple times until the Actor and Critic networks can fit the environment well and the strategy reaches a satisfactory level. In each iteration, the Actor continuously improves the policy, while the Critic gradually improves the estimation accuracy of the value function. By constantly interacting with the environment and updating network parameters, the entire system is gradually fitted and optimized in the reinforcement learning task to achieve better performance.

### E. DDRLA ALGORITHM

In the nutshell, the Actor network is responsible for giving actions based on the current state of the environment, while the Critic network is responsible for evaluating the actions. Then, the Actor network selects the action based on the evaluation of the Critic network. By using an adaptive the KL)term and multiple workers, the Actor network and Critic network can be trained efficiently.

Pseudo-code for the DDRLA algorithm is provided in Algorithm Boxes 1 and 2. $W$ is the number of workers, $D$ is the threshold value of the parameter that can provide workers with gradient updates, $N$ is the total episodes, $T$ is the number of data points collected by each worker before computing the parameter updates, and $K$ is the number of time steps for backpropagation after computing $K$ steps.

## III. MODELING

Switching Capacitor (SC) and On-Line Tap Changer (OLTC) are discrete regulation devices. They regulate reactive power in predetermined steps or gears, causing the system to absorb or release a specific amount of reactive power through the operation of a switch or transformer. These regulation devices can only regulate at predefined discrete levels. The reactive power of Distributed Generation (DG) is continuously regulated. In this paper, the DG inverter is operated on a bus with an apparent power capacity of $S_{DG}$. The DG inverter can supply or absorb reactive power continuously over a range, instead of being able to regulate only at a specific level, as is the case with discrete regulation devices. This continuous regulation allows the DG system to respond more flexibly to the changing demands of the power system, such as adjusting reactive power in real time to stabilize the voltage. The constraint on $Q_{DG}$ can be expressed as:

$$-Q_{DG,MAX} \leq Q_{DG} \leq Q_{DG,MAX} \tag{16}$$

where $Q_{DG,MAX}$ is the maximum reactive power value running on the bus, $Q_{DG} = \sqrt{S_{DG}^2 - P_{DG}^2}$; $P_{DG}$ is the active power. $\alpha_{DG} \in [-1, 1]$, $Q_{DG} = \alpha_{DG} Q_{DG,MAX}$.

### A. REACTIVE POWER OPTIMIZATION MODEL DESIGN

The goal of reactive power optimization in distribution network is to ensure that the voltage can operate within the normal range and minimize the active network loss. The objective function for reactive power optimization is defined as:

$$\min \sum_{i=1}^{N} P_{loss,i} \tag{17}$$

where $N$ is the number of command cycles in a day; $P_{loss,i}$ is the active network loss.

Constraints include constraints on node voltage $U_d$, reactive power $Q_d$, and change in action value $SG_d$, as shown below:

$$\begin{cases} U_{\min} \leq U_d \leq U_{\max} \\ Q_{\min} \leq Q_d \leq Q_{\max} \\ SG_{\min} \leq SG_d \leq SG_{\max} \end{cases} \tag{18}$$

**Algorithm 1** DDRLA (chief)

```
for $i \in \{1, \cdots, N\}$ do
    for $j \in \{1, \cdots, M\}$ do
        Wait until the gradient on $\theta$ is available for at least $D$ workers out of $W$ workers.
        Average the gradient values and update the global $\theta$.
    for $j \in \{1, \cdots, M\}$ do
        Wait until the gradient on $\phi$ is available for at least $D$ workers out of $W$ workers.
        Average the gradient values and update the global $\phi$.
    end for
end for
```

**Algorithm 2** DDRLA (worker)

```
Randomly initialize Actor network $\mu(s|\theta)$, Critic network $Q(s, a|\phi)$ and the Kullback–Leibler (KL) penalty.
for the number of iterations of$i \in \{1, \cdots, N\}$ do
    Receive initial observation state $s_1$.
    for $\omega \in \{1, \cdots, T/K\}$ do
        Select action $a_t = \mu(s_t|\theta)$ according to the current policy $\pi_\theta$.
        Perform Action $a_t$.
        Get reward according to reward function $r_t$.
        Observe new state $s_{t+1}$.
        Collect $\{s_t, a_t, r_t\}$ for trajectory $\tau = (s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_T, a_T, r_T)$.
        Estimate the reward $\overset{\wedge}{R}(\tau) = \sum_{t=1}^{T} R(s_t, a_t)$.
        Estimate advantages $\overset{\wedge}{A}(\tau) = \overset{\wedge}{R}(\tau) - V_\phi(\tau)$.
        Storing partial trajectory information.
    end for
    $\pi_{old} \leftarrow \pi_\theta$
    for $m \in \{1, \cdots, M\}$ do
        $R_t = (V^\pi(s_{t+1}) + r_t - V^\pi(s_t)|a_t), \hat{g} = E\left[\sum_{t=1}^{T} R_t \nabla_\theta \sum_{\tau} \log \pi_\theta(a_t|s_t)\right]$
        if $KL[\pi_{old}|\pi_\theta] > 4KL_{t\arg et}$ then
            Interrupt and continue to the next external iteration $i + 1$.
        end if
        Calculate $\hat{g} = \sum_{t=1}^{T-1} r_t + R_{t+1}(S, \theta) - R_t(S_{old}, \theta_{old})$.
        Send gradient $\theta$ to chief.
        Wait until gradient is accepted or discarded; then update parameters.
    end for
    for $b \in \{1, \cdots, B\}$ do
        $L_{SEL}(\phi) = -\sum_{t=1}^{T} \underset{t}{\overset{\wedge}{R}} - V_\phi(s_t)$
        Calculate $\nabla_\theta L_{SEL}$.
        Send gradient $\phi$ to chief.
        Wait until gradient is accepted or discarded; then update parameters.
    end for
    if $KL[\pi_{old}|\pi_\theta] > \beta_{high} KL_{t\arg et}$ then
        $\lambda \leftarrow \overset{\sim}{\alpha}\lambda$
    else if $KL[\pi_{old}|\pi_\theta] < \beta_{low} KL_{t\arg et}$ then
        $\lambda \leftarrow \lambda/\overset{\sim}{\alpha}$
    end if
end for
```

### B. DEEP REINFORCEMENT LEARNING

During the learning process, the agent interacts with the environment of the distribution grid system and achieves reactive power optimization of the power system by executing actions. The agent optimizes the reactive power allocation by constantly observing the state of the distribution network system and adjusting the strategy function according to the current state to select an appropriate action strategy. During the training process, the agent continuously learns and adjusts according to the feedback information of the environment to obtain better optimization performance. Through such interaction and learning, the agent gradually masters the methods and techniques for reactive power optimization of the distribution network system, so that the reactive power allocation can be more flexible and efficiently adapted to different power system operating states in order to achieve the optimal balance of voltage stability and energy utilization.

### C. STATE AND ACTION

The action space can be expressed as $[a_1, a_2, \cdots, a_N]^T$. $a_i$ is the action set corresponding to SC, OLTC, and DG, $a_i \in A_i$, $A_i$ is the search space for the $i$-th action, $i \in \{1, 2, \cdots, N\}$. That is, the optimal combination of these actions needs to be chosen to achieve the optimization goal.

In this paper, three important variables and matrices are introduced to describe the decision-making process of reactive power optimization in distribution networks. These variables provide a comprehensive description of the state of the distribution network and the operation of the regulation devices during the different decision phases. The voltages on all buses in the distribution network system as a state space can be represented as:

$$s_i = \{U_i, SG_i, E_i\} \tag{19}$$

where $U_i$ is the node voltage matrix of the distribution network in the $i$-th decision-making stage, and the dimension is $n \times m$, $n$ is the number of measurable nodes, $m$ is the measurement times of the action cycle; $SG_i$ is the switching position of each regulating device in the $i$-th action cycle; $E_i$ is the action completed by each regulating device in an action period.

In this paper, the action decision-making cycle time of the distribution network system is 15 minutes, the sampling period of the reactive equipment is 15 minutes.

### D. REWARD FUNCTION

The SC node voltage needs to meet the constraints, and the reward is set to the opposite number of the sum of the network loss and the action cost. The reward is defined as:

$$r_{i,SC} = -P_{loss,i} - \lambda_{SC} \sum_{j=1}^{i} \left| G_{SC,j} - G_{SC,j-1} \right| \tag{20}$$

where $\lambda_S$ is the action adjustment coefficient, and $G_{SC,j}$ is the switching gear of SC at the $j$-th decision-making time.

The reward obtained by OLTC at the current action moment is defined as:

$$r_{i,OLTC} = -P_{loss,i} - \lambda_O \sum_{j=1}^{i} \left| G_{OLTC,j} - G_{OLTC,j-1} \right| \tag{21}$$

where $\lambda_O$ is the action adjustment coefficient, and $G_{OLTC,j}$ is the switching gear of the OLTC at the $j$-th decision-making time.

The reward obtained by DG at the current scheduling moment is defined as:

$$r_{i,DG} = -P_{loss,i} - \lambda_D \sum_{k=1}^{N_D} \left| \frac{U_{k,j} - U_{k,baseline}}{U_{\max} - U_{\min}} \right| \tag{22}$$

where $\lambda_D$ is the DG gear adjustment coefficient, $U_{k,baseline}$ is the voltage reference value, $U_{k,j}$ is the voltage of the bus connected to the DG, $U_{\max}$ and $U_{\min}$ is the voltage upper and lower limits, and $N_D$ is the total number of nodes.

### E. TIME COMPLEXITY OF ALGORITHM

From Algorithm Boxes 1 and 2, the time complexity of the algorithm is as follows:

$$O(N \cdot T) = O\left(n^2\right) \tag{23}$$

Therefore, $N$ and $T$ should not be too large when training for reinforcement learning and designing the simulation model. By providing the initial probability distribution and hyperparameter tuning, both can effectively reduce the values of $N$ and $T$, thus reducing the time complexity.

## IV. EXPERIMENTAL VERIFICATION

Select the actual data of distributed photovoltaics, wind turbines and load output in a certain area of China for half a year to analyze the computing power based on deep reinforcement learning. Randomly select two typical days (a typical day in winter and a typical day in summer) for a total of 60 days as the test set, and the rest of the time as the training set, and the decision-making interval is set to 15 minutes.

In order to verify the effectiveness of DDRLA, this paper uses the following three algorithms for comparison: (1) the traditional optimization algorithm, GA [36]; (2) Value-based reinforcement learning algorithm, Deep Q-Network (DQN) [37]; (3) Policy-based reinforcement learning algorithm, Deep Deterministic Policy Gradient (DDPG) [38].

The hardware and software platforms for the simulation tests conducted in this paper are shown in Table 1.

**TABLE 1.** Hardware and software platforms.

| Platform | Configuration information |
|---|---|
| Hardware platform | Intel(R) Core(TM) i7-8700K CPU @ 4.30 GHz;<br>32GB RAM;<br>GPU: NVIDIA GTX 2080 Ti |
| Software platform | ubuntu19. 04 (Linux);<br>Python 3.8.6;<br>Pytorch 1.8.1 |

**TABLE 2.** Neural network parameter settings.

| Parameter | Value |
|---|---|
| Learning Rate | 0.005 |
| Discount factor | 0.99 |
| Number of hidden layers of Actor network | 2 |
| Number of hidden layers of Critic network | 2 |
| Number of neurons in hidden layer of Actor network | 400,100 |
| Number of neurons in hidden layer of Critic network | 400,200 |
| Batch Size | 1600 |
| Max episodes | 5000 |

### A. DEEP LEARNING TRAINING PROCESS

In this paper, Pytorch framework is used to build a distributed deep reinforcement learning algorithm, and a specific neural network structure is designed as the model. The specific neural network structure and training parameter settings are detailed in Table 2.

Pytorch provides flexible and powerful tools to facilitate the construction of complex neural network models and supports distributed training, which can make full use of multiple computational resources for parallel computation and improve the training efficiency and performance of the algorithm. With such a build, we are able to effectively implement deep reinforcement learning algorithms and conduct accurate model training and performance evaluation in experiments.

In the training phase, the actions of the intelligences are first initialized, and then, in the simulation environment of the distribution network system, the tidal current is calculated according to the action instructions provided by the Actor-Critic network, and the corresponding action $a$ is executed. Next, the corresponding reward $r$ is calculated according to the reward function of each worker to evaluate the impact of the action on the performance of the system. Subsequently, Actor organizes the collected data $\left(s, a, r, s', a'\right)$ and environment state information and stores them in the experience playback pool. The experience playback pool is used to store the experience data of the agent at different decision-making stages for random sampling and reuse in the training process. Through such a data collection and experience playback mechanism, the agent gradually learns better strategies to make better decisions in similar states, so as to achieve the goal of distribution network reactive power optimization. When the number of cache pools reaches the set threshold, Actor updates the policy $\pi$ and the critic updates the action value parameters. critic can receive the data collected by the actor and update the action value parameters by combining the data generated by the actor. The training is repeated until convergence.

### B. ALGORITHM VALIDITY VERIFICATION IN THE IEEE 33-BUS SIMULATION SYSTEM

The calculation example is based on the IEEE 33-bus simulation system. The calculation example system includes 3 DGs, each with a capacity of 750 kW, as shown in Figure 2.

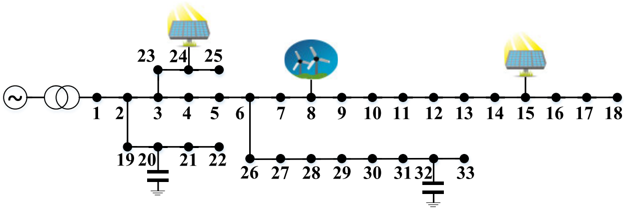

**FIGURE 2.** Distribution network example system based on IEEE 33-bus.

In Figure 2, parallel capacitors are added to nodes 21 and 32 in the system. The maximum number of actions per SC per day is 5. SC1 has 6 gears, each gear is 0.4MVar, the total capacity is 2MVar, SC2 has 5 gears, each gear is 0.3MVar, the total capacity is 1.2MVar. OLTC sets 10 adjustment gears. In the IEEE 33-bus distribution network system, it is assumed that SC1 and SC2 each have 5 or 4 gears (corresponding to 6 or 5 switching states) designated as the action of No. 1 worker, denoted as $\dim(\xi_1) = 6 \times 5 = 30$. OLTC has 11 discrete positions designated as the actions of No. 2 worker, denoted as $\dim(\xi_2) = 11$. In order to achieve a better training convergence effect, the DG of node 15 is used to participate in the adjustment of reactive power, and the DG is set to have 20 discrete gears as the actions of No. 3 worker, denoted as $\dim(\xi_3) = 20$. All actions are combined by matrix splicing $\dim(\xi_1) \times \dim(\xi_2) \times \dim(\xi_3) = 6600$.

Four different optimization methods are simulated at the typical time of 12:00 on a typical day (summer day and winter day), and the average network loss and voltage deviation are compared. The experimental results are shown in Table 3. As can be seen in Table 3, the DDRLA algorithm used in this paper has the lowest network loss in typical summer days. Compared with GA, DQN, and DDPG, the average network loss in summer days is reduced by 13.59%, 8.94%, and 3.08%. The average network loss in winter days decreased by 16.72%, 6.58%, and 3.07% respectively. In addition, the voltage deviation after DDRLA optimization is the smallest, which ensures the stability of voltage operation and minimizes voltage fluctuations. In summer days, the voltage deviation of DDRLA is 55.05%, 44.03%, 10.56% lower than that of GA, DQN, and DDPG, respectively. And in winter days, the voltage deviation of DDRLA is 55.25%, 44.99%, 13.55% lower than that of GA, DQN, and DDPG, respectively.

Therefore, DDRLA can reduce the system network loss with greater efficiency under the two typical conditions, which proves the effectiveness and superiority of DDRLA proposed in this paper.

The daily cumulative stall changes of OLTC and SC regulated equipment for the four methods under two typical days are shown in Figure 3. As can be seen from Figure 3, the cumulative gear change of the equipment after optimization using the DDRLA of this paper is smaller than that of the other two methods, which indicates that the action cost of optimizing the equipment by the method of this paper is smaller, and it has better economy.

**TABLE 3.** Comparison of typical daily network loss and voltage deviation in t the IEEE 33-bus distribution network system.

| Method | Average network loss /kW | | Voltage deviation /p.u. | |
|---|---|---|---|---|
| | summer day | winter day | summer day | winter day |
| GA | 127.3 | 136.4 | 7.142 | 6.713 |
| DQN | 120.8 | 121.6 | 5.735 | 5.461 |
| DDPG | 113.5 | 117.2 | 3.589 | 3.475 |
| **DDRLA** | **110.0** | **113.6** | **3.210** | **3.004** |

**TABLE 4.** Parameters of voltage regulation devices.

| Devices | Parameters | Operating Limits | Node Location |
|---|---|---|---|
| OLTC | ±10×0.01 | 5 | 1 |
| SC1~SC4 | 5×100kVar | 6 | 16, 46 58, 106 |
| DG1~DG6 | 750kW | - | 28, 48, 67, 89, 93, 113 |

**TABLE 5.** Comparison of typical daily network loss and voltage deviation in the IEEE 123-bus simulation system.

| Method | Average network loss /kW | | Voltage deviation /p.u. | |
|---|---|---|---|---|
| | summer day | winter day | summer day | winter day |
| GA | 413.5 | 449.7 | 18.129 | 16.945 |
| DQN | 398.3 | 401.2 | 14.583 | 13.975 |
| DDPG | 376.9 | 385.2 | 9.103 | 8.851 |
| **DDRLA** | **357.8** | **369.7** | **8.049** | **7.533** |

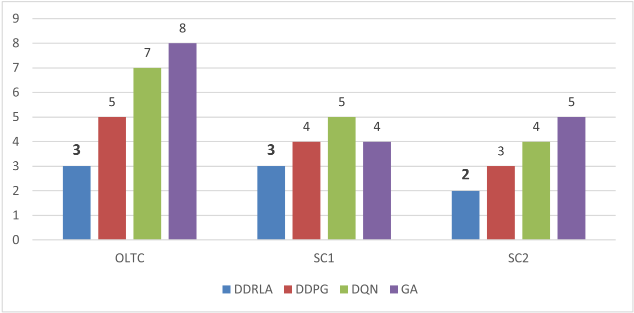

**FIGURE 3.** Daily cumulative number of stall changes for different methods of discrete regulation of equipment.

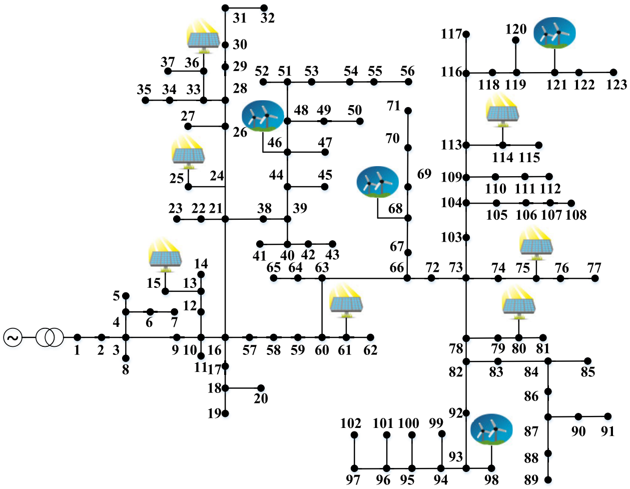

**FIGURE 4.** Distribution network example system based on IEEE 123-bus.

### C. ALGORITHM VALIDITY VERIFICATION IN THE IEEE 123-BUS SIMULATION SYSTEM

In order to validate the scalability and applicability of the proposed DDRLA, simulations of the IEEE 123-bus simulation system are carried out. The detailed parameters of the devices are shown in Table 4.

The network loss results of the IEEE 123-bus simulation system in two typical days were comparatively analyzed, as shown in Table 5. As can be seen from Table 5, the DDRLA proposed in this paper has the lowest network loss in typical summer. Compared with GA, DQN, and DDPG, the average

**TABLE 6.** Comparison of typical daily network loss and voltage deviation in the actual power distribution system in a region of China.

| Method | Average network loss /kW | | Voltage deviation /p.u. | |
|---|---|---|---|---|
| | summer day | winter day | summer day | winter day |
| GA | 168.5 | 187.2 | 9.2 | 8.6 |
| DQN | 165.9 | 167.2 | 7.4 | 7.1 |
| DDPG | 158.5 | 159.7 | 4.6 | 4.5 |
| **DDRLA** | **146.1** | **151.1** | **4.1** | **3.8** |

network loss in summer is reduced by 13.47%, 10.17% and 5.07%, respectively. The average network losses in winter decreased by 17.79%, 7.85% and 4.02% respectively. In addition, DDRLA has been optimized to minimize voltage deviation, ensuring the stability of voltage operation and minimizing voltage fluctuations. In summer days, the voltage deviation of DDRLA is 55.60%, 44.81%, 11.58% lower than that of GA, DQN, and DDPG, respectively. And in winter days, the voltage deviation of DDRLA is 55.54%, 46.10%, 14.89% lower than that of GA, DQN, and DDPG, respectively.

Therefore, the effectiveness, superiority and applicability of the DDRLA proposed in this article are proved.

### D. ALGORITHM VALIDITY VERIFICATION IN THE ACTUAL POWER DISTRIBUTION SYSTEM IN A REGION OF CHINA

In order to validate the scalability and applicability of the proposed DDRLA, simulations of the actual power distribution system in a region of China are carried out.

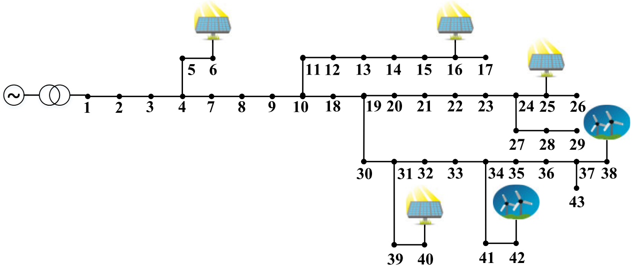

**FIGURE 5.** Actual power distribution system in a region of China.

The network loss results of the actual power distribution system in a region of China in two typical days were comparatively analyzed, as shown in Table 6. Compared to GA, DQN, and DDPG, the average network loss in summer is reduced by 13.30%, 11.89%, and 7.80%, respectively. In winter, the average network losses decreased by 19.30%, 9.63%, and 5.37%, respectively. Additionally, DDRLA has been optimized to minimize voltage deviation, ensuring stable voltage operation and reducing voltage fluctuations. During summer, DDRLA achieves voltage deviations that are 55.43%, 44.56%, and 11.26% lower than those of GA, DQN, and DDPG, respectively. In winter, DDRLA's voltage deviations are 55.45%, 45.76%, and 14.48% lower than those of GA, DQN, and DDPG, respectively.

Therefore, the superiority of the proposed algorithm is proved by the realistic examples based on the actual power distribution system in a region of China.

DDRLA achieves effective optimization due to its stability and efficiency in training. By using a clipped surrogate objective, DDRLA ensures gradual and controlled policy updates, preventing the destabilizing effects of large changes. It enhances sample efficiency through importance sampling, allowing multiple updates from the same batch of data. The simultaneous optimization of policy and value functions accelerates convergence and improves the accuracy of value estimates. DDRLA is computationally efficient and robust to hyperparameter variations, making it adaptable to a wide range of tasks. Additionally, it avoids the complexity of second-order optimization while maintaining strong performance.

### E. COMPUTATIONAL PERFORMANCE

In the online testing phase, the DDRLA algorithm was applied to the IEEE 33-bus IEEE 123-bus simulation systems and the actual power distribution system in a region of China. The average execution time of the DDRLA algorithm is 28.2ms, 46.5ms and 30.8ms, respectively, which can meet the real-time requirements of distribution systems.

Therefore, the DDRLA algorithm proposed in this article is reasonable when dealing with high-dimensional action spaces. The space is high-dimensional, and the DDRLA algorithm proposed in this article is reasonable.

## V. CONCLUSION

This paper proposes an Actor-Critic-based distributed deep reinforcement learning approach (DDRLA) to optimize reactive power in distribution networks. A well-trained agent can adaptively make decisions to adjust node voltage and reduce grid losses. This proposed DDRLA achieves better optimization results by designing multi-worker online training. Compared with the genetic algorithm (GA), deep Q network (DQN), deep deterministic policy gradient (DDPG) method, the DDRLA used in this paper ensures the minimum voltage fluctuation while the network loss of the optimized distribution network system is minimal. It has a remarkable effect in improving the safe, reliable, real-time performance and efficient operation of the distribution network.

In the IEEE 33-bus simulation system, compared with GA, DQN, and DDPG, the average network loss of DDRLA in summer is reduced by 13.59%, 8.94%, and 3.08%, respectively, and the average network loss in winter is reduced by 16.72%, 6.58%, and 3.07%, respectively. In addition, voltage operation, DDRLA has a better optimization effect than GA, DQN, and DDPG. In summer days, the voltage deviation of DDRLA is 55.60%, 44.81%, 11.58% lower than that of GA, DQN, and DDPG, respectively. And in winter days, the voltage deviation of DDRLA is 55.54%, 46.10%, 14.89% lower than that of GA, DQN, and DDPG, respectively. Besides, the average execution time of the DDRLA algorithm is only 28.2ms, which meets the real-time performance.

What is more, DDRLA is also applied to the IEEE 123-bus simulation system, compared with GA, DQN, and DDPG, the average network loss of DDRLA in summer is reduced by 13.47%, 10.17% and 5.07%, respectively, and the average network loss in winter is reduced by 17.79%, 7.85% and 4.02%, respectively. In terms of voltage deviation, in summer days, DDRLA is 55.60%, 44.81%, 11.58% lower than GA, DQN, and DDPG, respectively. And in winter days, DDRLA is 55.54%, 46.10%, 14.89% lower than GA, DQN, and DDPG, respectively. And the average execution time of the DDRLA algorithm is only 46.5ms, which meets the real-time performance in the more complex system.

To verify the superiority of the proposed algorithm, it is applied to the actual power distribution system in a region of China. Compared to GA, DQN, and DDPG, the average network loss in summer is reduced by 13.30%, 11.89%, and 7.80%, respectively. In winter, the average network losses decrease by 19.30%, 9.63%, and 5.37%, respectively. Additionally, DDRLA is optimized to minimize voltage deviation, thereby ensuring stable voltage operation and reducing voltage fluctuations. On summer days, the voltage deviation of DDRLA is 55.43% lower than GA, 44.56% lower than DQN, and 11.26% lower than DDPG. Similarly, on winter days, DDRLA achieves a voltage deviation reduction of 55.45% compared to GA, 45.76% compared to DQN, and 14.48% compared to DDPG.

In future work, the proposed DDRLA algorithm will be further examined by varying different parameters. The outcomes obtained with these varying parameters will be compared and analyzed in detail. The currently proposed algorithm needs to be improved in three aspects: algorithmic exploration performance, large-scale distribution system adaptability and algorithmic stability. Future work can be carried out from the following three aspects to improve the shortcomings of the DDRLA, as follows.

1) Improve the exploration scope and utilization balance to improve the performance of the algorithm.
2) Improve algorithm training speed to adapt to the needs of large-scale environments.
3) Improve algorithm stability to solve problems such as gradient disappearance and gradient explosion.

## ACKNOWLEDGMENT

The authors would like to acknowledge the support of Economic and Technology Institute of State Grid Fujian Electric Power Co., Ltd. and State Grid Fujian Electric Power Co., Ltd. for their selfless support.

## REFERENCES

[1] A. Kaneko, Y. Hayashi, T. Anegawa, H. Hokazono, and Y. Kuwashita, "Evaluation of an optimal radial-loop configuration for a distribution network with PV systems to minimize power loss," *IEEE Access*, vol. 8, pp. 220408–220421, 2020.

[2] T. Yachida, R. Okuyama, N. Morishima, Y. Ashizaki, and Y. Itaya, "Design and installation of STATCOM system for wind and photovoltaic power plant," in *Proc. Int. Power Electron. Conf. (IPEC-Himeji-ECCE Asia)*, May 2022, pp. 765–769.

[3] M. N. Mojdehi and P. Ghosh, "Minimization of energy usage and cost for EV during reactive power service," in *Proc. IEEE Int. Conf. Smart Energy Grid Eng. (SEGE)*, Oshawa, ON, Canada, Aug. 2015, pp. 1–6.

[4] M. Junxia, G. Binqing, L. Fuchao, and D. Peidong, "Study on power loss of distribution network with distributed generation and its reactive power optimization problem," in *Proc. Int. Conf. Power Syst. Technol.*, Chengdu, China, Oct. 2014, pp. 1213–1216.

[5] B. Mahdad, T. Bouktir, and K. Srairi, "OPF with environmental constraints with SVC controller using decomposed parallel GA: Application to the Algerian network," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Calgary, AB, Canada, Jul. 2009, pp. 1–8.

[6] Q. Li, Q. Huang, W. Qin, Y. Xing, H. Zhao, X. Song, E. Hu, and Z. Shen, "Resilience lifting method of active distribution network considering charging and discharging scheduling of electric vehicles," in *Proc. IEEE 5th Conf. Energy Internet Energy Syst. Integr. (EI2)*, Taiyuan, China, Oct. 2021, pp. 1973–1977.

[7] H. Bai, Y. Fu, Y. Li, W. Yang, Y. Cai, X. Li, and J. Zhi, "Distributed control of photovoltaic-energy storage system for low-voltage distribution networks considering the consistency of power and SOC," in *Proc. 5th Int. Conf. Power Energy Technol. (ICPET)*, Tianjin, China, Jul. 2023, pp. 579–583.

[8] H. Zhou, Z. Tan, and Y. Liu, "Multi-objective reactive power compensation optimization for power grid with large scale offshore wind farms," in *Proc. 7th Int. Conf. Power Renew. Energy (ICPRE)*, Shanghai, China, Sep. 2022, pp. 809–814.

[9] Y. Liu, Q. Zhang, J. Li, and Y. Peng, "Research on reactive power optimization configuration of distribution network based on small hydroelectric generating units," in *Proc. IEEE Int. Conf. Artif. Intell. Comput. Appl. (ICAICA)*, Dalian, China, Jun. 2022, pp. 534–539.

[10] J. Lu, S. Chen, B. Li, S. Zhu, Y. Tan, W. Liu, and X. Zhao, "An optimal reactive power compensation allocation method considering the economic value affected by voltage sag," in *Proc. IEEE Int. Power Electron. Appl. Conf. Expo. (PEAC)*, Shenzhen, China, Nov. 2018, pp. 1–6.

[11] X. Yi, M. Tian, C. Chen, and G. Zhang, "Reactive power optimization and reconstruction of distribution network based on improved GSA algorithm," in *Proc. IEEE/IAS Ind. Commercial Power Syst. Asia (I&CPS Asia)*, Chengdu, China, Jul. 2021, pp. 674–679.

[12] D. Maharana, R. Kommadath, and P. Kotecha, "Dynamic Yin-Yang pair optimization and its performance on single objective real parameter problems of CEC 2017," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Donostia, Spain, Jun. 2017, pp. 2390–2396.

[13] M. R. Adaryani and A. Karami, "Artificial bee colony algorithm for solving multi-objective optimal power flow problem," *Int. J. Electr. Power Energy Syst.*, vol. 53, pp. 219–230, Dec. 2013.

[14] C. Yong, Y. Li, Z. Zeng, Z. Zhang, Z. Zhang, and Y. Liu, "Coordinated active and reactive power optimization considering load characteristics for active distribution network," *Chin. J. Electr. Eng.*, vol. 6, no. 4, pp. 97–105, Dec. 2020.

[15] S. Abdelhady, A. Osama, A. Shaban, and M. Elbayoumi, "A real-time optimization of reactive power for an intelligent system using genetic algorithm," *IEEE Access*, vol. 8, pp. 11991–12000, 2020.

[16] X. Zhou, M. Ling, Q. Lin, S. Tang, J. Wu, and H. Hu, "Effectiveness analysis of multiple initial states simulated annealing algorithm, a case study on the molecular docking tool AutoDock vina," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 20, no. 6, pp. 3830–3841, Nov. 2023.

[17] J. Li, H. Huang, B. Lou, Y. Peng, Q. Huang, and K. Xia, "Wind farm reactive power and voltage control strategy based on adaptive discrete binary particle swarm optimization algorithm," in *Proc. IEEE Asia Power Energy Eng. Conf. (APEEC)*, Mar. 2019, pp. 99–102.

[18] Y. Ai, M. Du, Z. Pan, and G. Li, "The optimization of reactive power for distribution network with PV generation based on NSGA-III," *CPSS Trans. Power Electron. Appl.*, vol. 6, no. 3, pp. 193–200, Sep. 2021.

[19] C. Li, Q. Lu, H. He, J. Zhao, Y. Jiang, B. Xu, Y. Yan, J. Bian, and W. Du, "Reactive power optimization of active distribution networks based on simulated annealing algorithm," in *Proc. IEEE 7th Inf. Technol. Mechatronics Eng. Conf. (ITOEC)*, Chongqing, China, Sep. 2023, pp. 1022–1026.

[20] X. Liu, P. Zhang, H. Fang, and Y. Zhou, "Multi-objective reactive power optimization based on improved particle swarm optimization with $\varepsilon$-*greedy* strategy and Pareto archive algorithm," *IEEE Access*, vol. 9, pp. 65650–65659, 2021.

[21] Y. Linlin, Z. Lihua, M. Gaojun, Z. Feng, and L. Wanxun, "Research on multi-objective reactive power optimization of power grid with high proportion of new energy," *IEEE Access*, vol. 10, pp. 116443–116452, 2022.

[22] M. Niu, N. Z. Xu, H. N. Dong, Y. Y. Ge, Y. T. Liu, and H. T. Ngin, "Adaptive range composite differential evolution for fast optimal reactive power dispatch," *IEEE Access*, vol. 9, pp. 20117–20126, 2021.

[23] M. S. Saddique, S. Habib, S. S. Haroon, A. R. Bhatti, S. Amin, and E. M. Ahmed, "Optimal solution of reactive power dispatch in transmission system to minimize power losses using sine-cosine algorithm," *IEEE Access*, vol. 10, pp. 20223–20239, 2022.

[24] M. Khairy, M. B. Fayek, and E. E. Hemayed, "PSO2: Particle swarm optimization with PSO-based local search," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2011, pp. 1826–1832.

[25] J. Y. Li, Z. J. Liu, B. Tang, R. S. Qin, Z. Da Yu, M. X. Liu, and G. L. Yang, "Reactive power compensation device based on magnetic-valve controllable reactor in the treatment of low voltage at the end of line," in *Proc. 7th Int. Conf. Power Renew. Energy (ICPRE)*, Shanghai, China, Sep. 2022, pp. 1–6.

[26] S. Stock, D. Babazadeh, and C. Becker, "Applications of artificial intelligence in distribution power system operation," *IEEE Access*, vol. 9, pp. 150098–150119, 2021.

[27] Y. Pei, J. Zhao, Y. Yao, and F. Ding, "Multi-task reinforcement learning for distribution system voltage control with topology changes," *IEEE Trans. Smart Grid*, vol. 14, no. 3, pp. 2481–2484, May 2023.

[28] Y. Wang, M. Mao, L. Chang, and N. D. Hatziargyriou, "Intelligent voltage control method in active distribution networks based on averaged weighted double deep Q-network algorithm," *J. Mod. Power Syst. Clean Energy*, vol. 11, no. 1, pp. 132–143, Jan. 2023.

[29] L. Z. Velimirovic, A. Janjic, and J. D. Velimirovic, "Fault location and isolation in power distribution network using Markov decision process," in *Proc. 14th Int. Conf. Adv. Technol., Syst. Services Telecommun. (TELSIKS)*, Nis, Serbia, Oct. 2019, pp. 408–411.

[30] H. Gao, R. Wang, S. He, L. Wang, J. Liu, and Z. Chen, "A cloud-edge collaboration solution for distribution network reconfiguration using multi-agent deep reinforcement learning," *IEEE Trans. Power Syst.*, vol. 39, no. 2, pp. 3867–3879, Mar. 2024.

[31] D. H. Abdulazeez and S. K. Askar, "Offloading mechanisms based on reinforcement learning and deep learning algorithms in the fog computing environment," *IEEE Access*, vol. 11, pp. 12555–12586, 2023.

[32] 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.

[33] Z. Xu, J. Tang, J. Meng, W. Zhang, Y. Wang, C. H. Liu, and D. Yang, "Experience-driven networking: A deep reinforcement learning based approach," in *Proc. IEEE Conf. Comput. Commun.*, Apr. 2018, pp. 1871–1879.

[34] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A survey of Monte Carlo tree search methods," *IEEE Trans. Comput. Intell. AI Games*, vol. 4, no. 1, pp. 1–43, Mar. 2012.

[35] X. Li, B. Han, G. Li, L. Luo, K. Wang, and X. Jiang, "Dynamic topology awareness in active distribution networks under DG uncertainties using GMM-PSEs and KL divergence," *IEEE Trans. Sustain. Energy*, vol. 12, no. 4, pp. 2086–2096, Oct. 2021.

[36] K. Iba, "Reactive power optimization by genetic algorithm," *IEEE Trans. Power Syst.*, vol. 9, no. 2, pp. 685–692, May 1994, doi: 10.1109/59.317674.

[37] M. Ali, A. Mujeeb, H. Ullah, and S. Zeb, "Reactive power optimization using feed forward neural deep reinforcement learning method: (Deep reinforcement learning DQN algorithm)," in *Proc. Asia Energy Electr. Eng. Symp. (AEEES)*, May 2020, pp. 497–501.

[38] Y. Tang, W. Hu, D. Cao, N. Hou, Y. Li, Z. Chen, and F. Blaabjerg, "Artificial intelligence-aided minimum reactive power control for the DAB converter based on harmonic analysis method," *IEEE Trans. Power Electron.*, vol. 36, no. 9, pp. 9704–9710, Sep. 2021.

**JINLIN LIAO** received the B.S. degree in electrical engineering and automation from Shandong University, Jinan, China, in 2018, and the M.S. degree in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 2021.

His current research interests include machine learning methods for distribution network planning and operation and resilient distribution networks.

**JIA LIN** received the B.S. degree in electrical engineering from Southeast University, Nanjing, China, in 2007, and the M.S. degree in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 2010.

She is currently a System Planning and Operating Engineer. Her research interests include intelligent system planning, energy information study, and data mining.