IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 5, 2020, accepted August 10, 2020, date of publication August 20, 2020, date of current version September 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3018142

Optimizing the Post-Disaster Control of Islanded Microgrid: A Multi-Agent Deep Reinforcement Learning Approach

HUANHUAN NIE[1], (Graduate Student Member, IEEE), YING CHEN[1], (Member, IEEE), YUE XIA[2], (Member, IEEE), SHAOWEI HUANG[1], (Member, IEEE), AND BINGQIAN LIU[3]

[1]State Key Laboratory of Power Systems, Department of Electrical Engineering, Tsinghua University, Beijing 100084, China
[2]College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China
[3]Electric Power Research Institute of State Grid Fujian Electric Power Company Ltd., Fuzhou 350007, China

Corresponding author: Yue Xia (xiayuexiayue@163.com)

This work was supported by the Science and Technology Project of STATE GRID Corporation of China, Data-driven modeling and simulation of Energy Internet(5700-201955468A-0-0-00).

ABSTRACT Extreme disasters may cause the power supply to the distribution system (DS) to be interrupted. The DS is forced to operate in island mode and forms an islanded microgrid (MG). In order to improve the post-disaster resilience of the DS and to provide longer power supply for as many loads as possible with limited generation resources, this paper proposes a multi-agent deep reinforcement learning (DRL) method which realizes a dual control on the source and load sides of the MG. The problem of resilience improvement is converted to a sequential decision making problem, where the objective is to maximize the cumulative MG utility value over the power outage duration. A multi-agent DRL model is proposed to solve the sequential decision making problem. A dual control policy including energy storage management and load shedding strategy is put forward to maximize the utility value of the MG. A reinforcement learning (RL) environment based on OpenAI and OpenDSS for islanded MG is constructed as a simulator, which has a general interface compatible with, and also can be published to, OpenAI Gym. Numerical simulations are performed for an MG equipped with wind turbines, diesel generators, and storage devices to validate the effectiveness of the proposed method. The influences of available generation resources and power outage duration on the control policy are discussed, which validates the strong adaptability of the proposed method in different conditions.

INDEX TERMS Control optimization, load shedding, microgrid, OpenAI, OpenDSS, reinforcement learning, resilience.

I. INTRODUCTION

A. BACKGROUND

Recent years, the frequent occurrence of extreme disasters, such as earthquake, hurricane and flood, has exerted a significant impact on the normal operation of infrastructure and resulted in significant inconvenience and economic losses to residents due to the loss of electricity, water and communication. A Congressional Research Service study in 2012 estimates the inflation-adjusted cost of weather-related outages at 25 to 70 billion dollars annually in the U.S. [1]. The severe power outages caused by these extreme disasters have highlighted the importance and urgency of improving the resilience of distribution system (DS). Resilience is used to measure the ability of a DS to withstand and recover from extreme disasters [2].

B. RELATED WORKS

In the last decades, various methods are proposed to enhance the resilience of power grid, and these methods can be mainly divided into two categories based on the timeline: pre-disaster preparation and post-disaster decision making.

From the viewpoint of pre-disaster preparation, some studies focus on the natural disaster impacts on electric power systems, trying to understand the causes of the blackouts and explore ways to prepare and harden the grid [3]–[5]. A resilient defender-attacker-defender game framework is proposed in [6] to coordinate the hardening and distributed

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

generation resource allocation with the objective of minimizing the system damage against disasters. In [7], the weather information is integrated into the distribution damage assessment which helps to understand how different weather metrics impact the distribution grid. Some others, from the perspective of post-disaster decision making, focus on faster restoration of the system. Research of using microgird (MG) to restore the DS is reviewed in [5], [8]. Research [9] proposes a novel distribution system operational approach by forming multiple MGs energized by DG from the radial distribution system in real-time operations to restore critical loads from the power outage. A hierarchical energy management framework based on multi-MGs is proposed in [10] for resilience enhancement. A cost-effective system-level restoration scheme is presented in [11] to improve power grid resilience. An MG dispatch solution is proposed in [12] for emergency electric service restoration after a disaster. A methodology for MG management and control to maximize the duration of electricity supply in emergency situations is proposed in [13]. The feasibility of control strategies to be adopted for the operation of an MG when it becomes isolated is described in [14].

Microgrids (MGs) can enhance post-disaster resilience by improving generation availability (e.g., fuel cells, microturbines, wind turbines, photovoltaic panels) when the utility power of the DS is unavailable [5]. During extreme natural disasters and aftermath, the generation resources within the DS are limited and hard to supplement due to the direct or indirect damage to power grid and transportation [12]. Therefore, it is necessary to manage the generation resources within the system appropriately to prevent a complete outage. In addition, load shedding can be adopted where the non-critical load is shed gradually for continuous power supply to critical load [2], [13].

However, the uncertainties in renewable energy, load, system energy storage and the power outage duration, as well as the complex hybrid control at both load side and source side bring many technical challenges. These uncertainties make prediction to the future more troublesome, and how to make decisions based on known information becomes more difficult. The complex hybrid control at both load side and source side results in a large search space and high optimization cost, and the strategy updating becomes more difficult.

To address these challenges, effective methods are required. Classical optimization/convex optimization is one of the conventional methods. It has the following special characteristics: need a specific mathematical model but require the prediction of the future. Robust optimization (RO) and stochastic programming (SP) can deal with the uncertainty. They often formulate a multi-stage or multi-layer optimization problem and transform the problem into a deterministic problem to solve. But the computation time and model complexity of this kind of method will be enormous in complicated and high-dimension scenarios, and the feasibility cannot be guaranteed. Meanwhile, the solution obtained by RO or SP is a pre-determined solution. This means that the actual operation plan is implemented according to the pre-determined solution, and real-time control cannot be realized.

In research [15], the state-based strategy is proposed. The strategy is made based on observed states during the unfolding events. Both RO and SP are not suitable for mapping sequentially real-time varying states to optimal strategies. To overcome the problem, Markov decision process (MDP) is employed to make state-based decisions in a stochastic environment caused by weather events. It chooses the action according to the MDP state (or the available information at each decision point). Although dynamic programming can be used to solve the MDP problem, it cannot deal with the uncertainty. It also has the weakness of the curse of dimensionality . The state space is huge in the high-dimension problem. Approximate dynamic programming (ADP) or RL can be used to deal with the curse of dimensionality. The policy learned by RL can realize real-time control.

Since AlphaGo was proposed in 2016 and defeated a world champion in the game of Go [16], deep reinforcement learning (DRL) has set off a new research boom again. With model-free algorithm and empirical learning, DRL solves many of the tough problems of the past, such as robot control [17], autonomous driving [18] and many kinds of games playing [16]. DRL is such a powerful tool in the scenarios with large uncertainties that can handle the aforementioned challenges effectively. After sufficient learning, well-learned RL agent can obtain a decision policy to realize real-time control. Some researchers also use RL to control MG [19] and DS [20]. But the application of DRL in resilience control is under-researched. Inspired by the successful application of DRL in the game field, this paper uses DRL to enhance the post-disaster resilience of DS, and this method relies only on current information without prediction of the future.

C. CONTRIBUTIONS

In this paper, we consider in the aftermath of a natural disaster, the power supply of the DS is interrupted. The outage duration of DS depends on the repair process and the severity of the damage caused by a disaster. Before the repair process of DS is completed, the DS has to supply its loads with its internal resources. In order to improve the post-disaster resilience of the DS, a longer power supply to as many loads as possible with limited generation resources over the power outage duration of the DS is necessary. In this paper, the resilience enhancing problem is converted to a decision making problem, a hybrid control including the energy storage management and load shedding policy is proposed to make full use of limited generation resources within the system, thus improving the resilience. The major contributions of this paper include:

- A multi-agent DRL model based on MDP is developed for the sequential decision making problem. A dual optimal control policy on the source and load sides is achieved to improve the resilience. Test results validate the strong adaptability of the proposed method under

various conditions such as different available generation resources and MG power outage duration.
- A RL environment for islanded MG operation based on OpenAI Gym is constructed and is used as the task simulator, which provides an easy-to-use interface of RL tasks. Limitations of the generation resources and power flow, as well as the uncertainties within the MG, are all considered in the environment.

The remainder of this paper is organized as follows. Section II formulates the decision making problem for an MG with limited generation resources. Section III develops the MDP and RL models for the sequential decision making problem and proposes a multi-agent DRL control algorithm. Section IV describes the islanded MG model and constructs an RL environment based on OpenAI Gym. In Section V, the numerical results are presented to validate the proposed algorithm. Conclusions are drawn in Section VI.

II. PROBLEM FORMULATION

A. GENERIC MG MODEL

An MG is a small-scale low voltage DS that comprises controllable loads, several small modular generation and storage systems, and provides electrical and heat [combined heat and power (CHP)] supply to local loads [21]. The MG can be generalized into four types of devices from the standpoint of energy generation and consumption: intermittent distributed generators (DGs), such as wind turbines and photovoltaic modules; dispatchable DGs, such as fuel and natural gas generators; local loads with different priorities; and electric energy storage devices. It is worth noting that some devices in the MG have varying operating characteristics. Due to the influence of weather, human behaviours, and other factors, the output power of intermittent DGs and load demand have great uncertainties and vary significantly under different conditions. Continuous operation of dispatchable DGs and storage devices depends on the available generation resources, i.e., fuel reserve (FR) of DGs and state of charge (SOC) of battery storage devices [2].

When a disaster strikes DS and interrupts the generation availability from the main power grid, the islanded MG forms and has to use internal generation resources to power its loads. After a period of time T_D, which is often the power outage duration of DS, the DS service is restored thanks to the repair of power grid staffs. DS returns to normal conditions.

B. SEQUENTIAL DECISION MAKING PROBLEM FOR AN ISLANDED MG

1) PROBLEM DESCRIPTION

In order to improve the resilience of the DS, a longer power supply to as many loads as possible with limited generation resources within time T_D is necessary. Utility value of the system power supply can be used as a measure to the post-disaster resilience. A more adequate and reasonable utilization of the energy after a disaster can result in a more resilient power system. The problem of resilience improvement can be converted to increase the cumulative utility value of the DS in time period T_D. The purpose of the islanded MG control is to achieve a policy maximizing the cumulative MG utility value over the time period T_D with the limited generation resources.

2) PROBLEM MODELING

The islanded MG control problem over the time period T_D is modeled as a sequential decision making problem. Sequential decisions are characterized by a decision-maker choosing among various actions after taking an observation of the system at different points in time, in order to control and optimize the performance of a dynamic stochastic system [22].

In this paper, the decision at each point in time is the dispatchable DGs output control on the source side and load shedding action on the load side. The objective is to maximize the cumulative MG utility value over the time period T_D and can be written as

$$\max_{\pi} \int_0^{T_D} R^{\pi}(t)\mathrm{d}t \tag{1}$$

where π represents the decision policy. T_D is the time period for MG control, usually is the power outage time of DS, it can be also selected by the MG operator. After T_D the DS is restored or the supplemental generation resources are available. $R^{\pi}(t)$ is the utility value function of MG under policy π. $R(t)$ can be measured by the load supply income, planned and unplanned outage loss. At time instant t it can be calculated as

$$R(t) = \sum_{k=1}^{N_L} (o_{L_k})^T b_{L_k} p_{L_k}(t) \tag{2}$$

with

$$b_{L_k} = [i_{L_k}, c^p_{L_k}, c^u_{L_k}]^{\mathrm{T}} \tag{3}$$

$$o_{L_k} = [\mathbb{I}(s_{L_k} = n), \mathbb{I}(s_{L_k} = p), \mathbb{I}(s_{L_k} = u)]^{\mathrm{T}} \tag{4}$$

where $p_{L_k}(t)$ denotes the active power of load L_k at t. N_L is the number of total loads. b_{L_k} reflects the supply income and outage cost of L_k in \$/kWh; $i_{L_k}, c^p_{L_k}, c^u_{L_k}$ are the supply income, planned and unplanned outage cost of L_k, respectively. o_{L_k} reflects the operation state vector of load L_k, s_{L_k} represents the power supply status of L_k which includes normal operation n, planned outage p, and unplanned outage u. $\mathbb{I}(x)$ is an indicator function. If x is true, the value is 1, otherwise it is 0.

Time period T_D can be also discretized into N decision stages, and the objective in (1) is then expressed as

$$\max_{\pi} \sum_{n=1}^{N} R^{\pi}(n) \tag{5}$$

where $R^{\pi}(n)$ is the utility value of each decision point in time.

3) CONSTRAINTS

During the normal operation of microgid, the constraints including power flow and resources should be satisfied,

$$P_i(t) - jQ_i(t) = V_i^*(t) \sum_{j \in i} Y_{ij} V_j(t) \tag{6}$$

$$V_i^{\min} \leq V_i(t) \leq V_i^{\max} \tag{7}$$

$$|I_l(t)| \leq I_l^{\max} \quad \text{or} \quad |S_l(t)| \leq S_l^{\max} \tag{8}$$

$$\begin{cases} P_g^{\min} \leq P_g(t) \leq P_g^{\max} \\ Q_g^{\min} \leq Q_g(t) \leq Q_g^{\max} \end{cases} \tag{9}$$

$$E_g(t) \leq E_g^M \tag{10}$$

$$i, j \in \mathbf{B} \quad l \in \mathbf{L}\; g \in \mathbf{G} \tag{11}$$

where **B**, **L**, **G** are sets of buses, lines, and DGs in the MG, respectively; $P_i(t)$ and $Q_i(t)$ are the injected active and reactive power of bus i at time t, respectively; $V_i(t)$, $V_i^{\min}$ and $V_i^{\max}$ are the voltage of bus i, and its lower and upper limit; $V_i^*(t)$ is the conjugate of $V_i(t)$; Y_{ij} is the admittance; $I_l(t)$ and $S_l(t)$ are the current and apparent power of line l at time t, $I_l^{\max}$ and $S_l^{\max}$ are their upper limits; $P_g(t)$ and $Q_g(t)$ are the active and reactive power of DG g at time t. $E_g(t)$ is the available generation resource of DG g at time t, and E_g^M is the maximum possible value of E_g. In this paper, E_g is used to represent the fuel reserve of DGs or the SOC of the battery devices, measured by the equivalent electric energy in kWh. The energy conversion efficiency is taken into consideration during the computation of E_g. In the case of this paper, the scale of MG is relatively small. Micro gas turbine and energy storage battery are commonly used for power supply. The output of power supply changes sharply to ensure that the MG can maintain stability under the rapid change of load demand. Therefore, the ramp constraints of DGs are ignored in this paper.

It is challenging to solve the problem of formula (1)-(11) using traditional optimization methods. Inspired by the successful application of RL in the game field, learning based methods can be adopted to handle this problem. By establishing decision making model, designing learning algorithm, building learning environment and verifying the effectiveness, the solution can be effectively derived. These steps will be described in detailed in the following sections.

III. MULTI-AGENT DRL MODEL OF POST-DISASTER CONTROL OF MG

In this section, RL model based on MDP is developed for the sequential decision making problem. The basic information about MDP and RL will be described in Section III-A. Corresponding MDP and RL models are established in Section III-B. A multi-agent DRL model developed for this decision making problem is proposed in Section III-C.

A. MDP MODEL FOR THE SEQUENTIAL DECISION MAKING PROBLEM

A MDP is a discrete time stochastic control process which provides a mathematical framework for modeling decision making in situations where outcomes are partly random and partly under the control of a decision-maker. A MDP usually comprises: a state space $\mathcal{S}$, a action space $\mathcal{A}$, an initial state distribution with density $p_0(s_0)$, a stationary transition dynamics distribution with conditional density $p(s_{t+1}|s_t, a_t)$ satisfying the Markov property $p(s_{t+1}|s_0, a_0, s_1, a_1, \cdots, s_t, a_t) = p(s_{t+1}|s_t, a_t)$, for any trajectory $s_0, a_0, \cdots, s_T, a_T$ in the state-action space, where $s_t \in \mathcal{S}$, $a_t \in \mathcal{A}$, and a reward function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$.

In this MG sequential decision making problem, the MDP elements can be designed as follows: the state should be designed to include the information required to make appropriate decision, including power flow, system remaining available generation resources, and the remaining power outage time.

$$s = [P_L, Q_L, P_G, Q_G, V_M, E_G, t_r] \tag{12}$$

where P_L, Q_L, P_G, Q_G denote the active and reactive power of load demand and DGs, respectively. V_M denotes the voltage magnitude of system, E_G is the remaining available generation resources of DGs, $t_r \in [0, T_D]$ is the remaining power outage time. The action is dispatchable DGs output control and load shedding action.

$$a = [P_G, Q_G, L_S] \tag{13}$$

where L_S denotes the load shedding action. The reward should be consistent with the system objective, so the utility value of the MG $R^\pi(t)$ can be designed as a reward.

$$r = \sum_{k=1}^{N_L} (o_{L_k})^{\mathrm{T}} b_{L_k} p_{L_k} \tag{14}$$

In the MDP, if the probabilities or rewards are unknown, the problem is one of RL [23], the transition probabilities can be accessed through a simulator that is typically restarted many times from a uniformly random initial state. In this MDP, it is impossible to get the transition probabilities because of the uncertainties, so RL can be used to solve this problem.

B. RL MODEL

We study RL and control problems in which an agent interact with an environment by sequentially choosing actions over a sequence of time steps, in order to maximize a cumulative reward. A policy in RL is used to select action in the MDP. The agent interacts with the environment using its policy and gives a trajectory of states, actions, and rewards $s_0, a_0, r_0, s_1, a_1, r_1, \cdots, s_T, a_T, r_T$ over $\mathcal{S} \times \mathcal{A} \times \mathbb{R}$. The return G_t is total discounted reward from time-step t onwards,

$$G_t = r_t + \gamma r_{t+1} + \cdots = \sum\nolimits_{k=0}^{\infty} \gamma^k r_{t+k} \tag{15}$$

where γ is the discount factor indicating how much the next step affects the current step, $0 < \gamma < 1$. Value function $V^\pi(s)$ or action-value function $Q^\pi(s, a)$ are defined to be the expected return,

$$\begin{cases} V^{\pi}(s) = \mathbb{E}[G_0|s_0 = s; \pi] \\ Q^{\pi}(s, a) = \mathbb{E}[G_0|s_0 = s, a_0 = a; \pi] \end{cases} \tag{16}$$

where the initial state s_0 is s, initial action a_0 is a. The agent's goal is to obtain a policy which maximizes the cumulative discounted reward from the initial state. The optimal policy is defined as: $\pi > \pi' \Leftrightarrow V^{\pi}(s) > V^{\pi'}(s), \forall s \in \mathcal{S}$, we have the optimal $Q^{*}(s, a) = \max_{\pi} Q^{\pi}(s, a)$.

Q-learning is a classical RL algorithm. In Q-learning, a value called Q-value $Q(s, a)$ is stored for each state s and action a. Q-value function is updated by Bellman equation.

$$Q(s, a) = Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)] \tag{17}$$

where α is the learning rate, s' is the resulting state after taking action a in state s, a' is an action that can be selected in state s'. In state s, the action-choosing policy of Q-learning is to select the action a that maximizes $Q(s, a)$ according to ϵ-greedy algorithm.

When there is a large state space, Q-learning is not quite practical where a very big Q-table is needed to storage Q-value for each state and action. Deep Q-network (DQN) [24], [25] refers to a neural network function approximator with weights θ as a Q-network to fit Q-value function. The Q-network can be trained by minimizing the loss function $L_k(\theta_k)$ that changes at each iteration k,

$$L_k(\theta_k) = \mathbb{E}[(y_k - Q(s, a; \theta_k))^2] \tag{18}$$

where $y_k = \mathbb{E}_{s' \sim \mathcal{S}}[r + \gamma \max_{a'} Q_k(s', a'; \theta_{k-1})|s, a]$ is the target for iteration k. Differentiating the loss function with respect to the weights we arrive at the following gradient:

$$\begin{aligned} \nabla_{\theta_k} L_k(\theta_k) = \mathbb{E}_{s,s' \in \mathcal{S}, a \in \mathcal{A}}[(r + \gamma \max_{a'} Q_k(s', a'; \theta_{k-1}) \\ - Q(s, a; \theta_k)) \nabla_{\theta_k} Q(s, a; \theta_k)] \end{aligned} \tag{19}$$

DQN solves problems with high-dimensional observation spaces successfully. Nevertheless, the curse of dimensionality is serious when the action space is high-dimensional or continuous. Deep Deterministic Policy Gradient (DDPG) [26] is proposed to solve the continuous control problem which combines actor-critic approach with DQN and Deterministic Policy Gradient (DPG) algorithm [27]. Compared to DQN, DDPG adds an actor network to generate a specific action according to the current state $a_t = \mu_{\theta}(s_t)$. The actor is updated by following applying the chain rule to the expected return from initial state with respect to the actor parameters:

$$\begin{aligned} \nabla_{\theta^{\mu}} \mathcal{J} &\approx \mathbb{E}_{s_t \sim \rho^{\beta}}[\nabla_{\theta^{\mu}} Q(s, a; \theta^{Q})|_{s=s_t, a=\mu(s_t; \theta^{\mu})}] \\ &= \mathbb{E}_{s_t \sim \rho^{\beta}}[\nabla_{a} Q(s, a; \theta^{Q})|_{s=s_t, a=\mu(s_t)} \\ &\qquad \nabla_{\theta^{\mu}} \mu(s; \theta^{\mu})|_{s=s_t}] \end{aligned} \tag{20}$$

where $\mathcal{J}$ is the expected return G_0 from initial state. ρ^{β} is the initial state distribution. θ^{Q} and θ^{μ} are the parameters of critic-network and actor-network, respectively. Details about DDPG is available in [26].

C. MULTI-AGENT DRL MODEL

The islanded MG control is a discrete-continuous hybrid action space control problem. On the source side, the DGs output control is continuous control. On the load side, the load shedding control is discrete control. If only one agent is used to deal with such a discrete-continuous hybrid action space control problem, the policy update will be extremely difficult.

P-DQN is proposed in [28] to cope with the discrete-continuous hybrid action control problem in game King of Glory. But its continuous action is a low-level parameter which is associated with the high-level discrete action. Both the discrete and continuous action must be executed simultaneously. In our decision making problem, the actions on load and source sides have the same level and can be executed at different time. That makes P-DQN not straightly usable.

We propose a double agent DRL model, which has a load agent and a DGs agent on the load and source sides, respectively. The two agents control the MG together for maximum utility value. The design mechanism of double agent DRL model is shown in FIGURE 1. The load agent and DGs agent are achieved using DQN and DDPG, respectively. Two agents interact with the environment independently, without communication with each other. They get the same state from the environment, execute their own actions to alter the state of the environment and get their own rewards according to reward shaping. However, in this mechanism, to one agent, the other agent becomes part of the environment, so the two agents interact with and influence each other essentially.

Two agents receive the same state s_t specified in Section IV-A. They can also conduct feature selection considering the difference of their tasks. The action of load agent a_t^L, a discrete value such as $a_t^L = 0, 1, 2, \ldots$, is the load shedding action L_S specified in Section IV-A, and each of the discrete value represents shedding specific priorities of loads in the MG. The action of DGs agent a_t^G is the output of controllable DGs. The merged action $a_t = (a_t^G, a_t^L)$ will be executed on the islanded MG.

Another important task is how to design the rewards of two agents. Because of the difference in the tasks of two agents, reward shaping should be used to redesign the reward for better learning performance. Note that the load agent decides how much power the system needs, and the DGs agent decides how to provide it. Since load agent has direct influence on the utility value of MG, it is designed as a far-sighted agent with a large γ of 0.99 whose goal is to maximize the utility value of MG over T_D. The reward of load agent is designed as

$$r_t^L = R(t) = \sum_{L_i} (o_{L_i})^{\mathrm{T}} b_{L_i} p_{L_i}(t) \tag{21}$$

it is exactly the MG utility value at t.

Nevertheless, on the source side, an explicit expression of control objective is hard to obtain, although we know the ground truth that the regulation of generation resources can truly influence the power supply duration of MG, thus influencing the resilience. A simple thought is to set the rewards

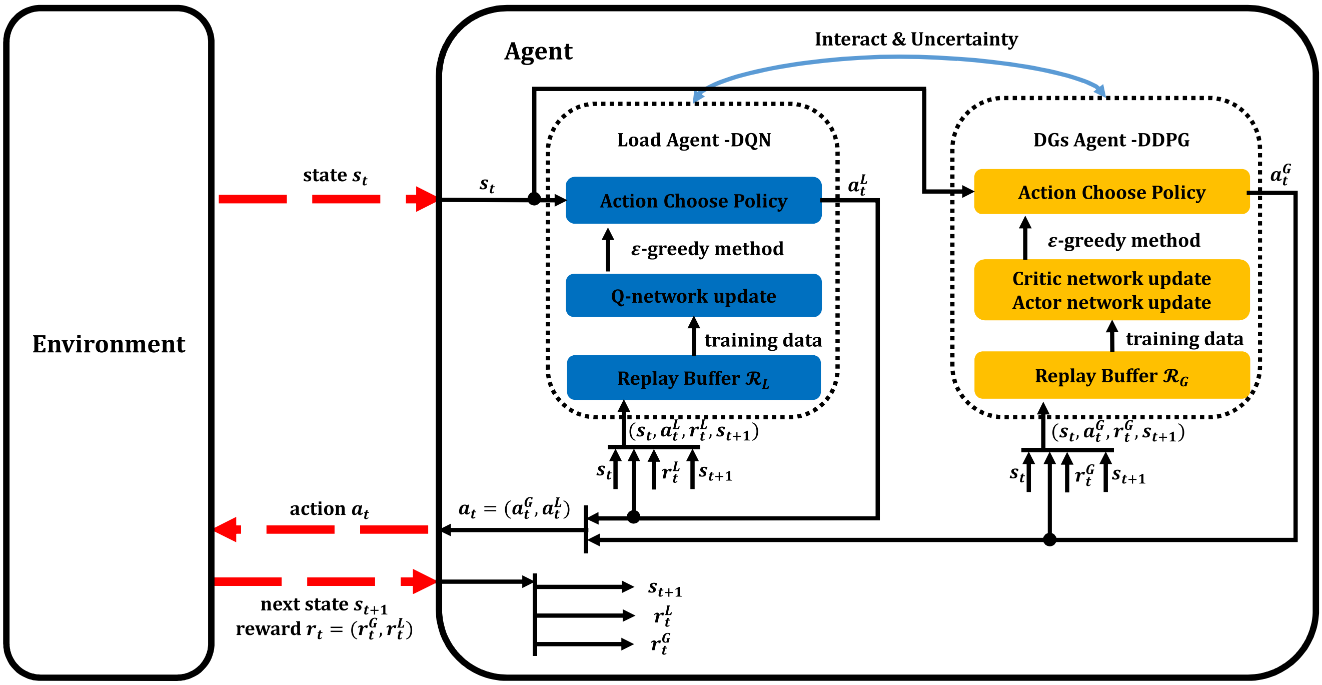

FIGURE 1. Design mechanism of double agent DRL model. The double agent DRL model has a load agent and a DGs agent on the load and source sides, respectively. The load agent and DGs agent are achieved using DQN and DDPG, respectively. Two agents interact with the environment independently, without communication with each other.

of two agents the same, but the convergence of this design is so poor in our multiple experiments that a new reward design is needed. Inspired by the idea of optimal power flow (OPF), we introduce a factor f_G to measure the resilience potential of each generator. Considering that the key to improving resilience is to use limited resources to provide more durable power for more loads, the generator with greater and more durable power supply potential is regarded as the one with higher resilience potential. On the source side, improving the resilience means minimizing "resilience potential cost", just like minimizing the economic cost of OPF. The reward of DGs agent and resilience potential are designed as

$$r_t^G = -\sum_{G_i} f_{G_i}(t) p_{G_i}(t) - \sigma H(p_G(t)) \qquad (22)$$

$$f_{G_i}(t) = S_{G_i} E_{G_i}(t) \qquad (23)$$

where r_t^G is the reward for DGs agent. f_{G_i} is the resilience potential factor of DG G_i, it depends on the DG capacity S_{G_i} and remaining sources $E_{G_i}(t)$. σ is the penalty factor, $H(p_G(t))$ represents the violation of constraints described in (7)-(10). The violation is added to reward r_t^G as penalty. DGs agent is designed as a short-sighted agent with a small γ of 0.01 whose goal is to optimize the resilience potential cost at time t. In this way, in the long run, the load shedding strategy ensures continuous power supply for as many loads as possible. In the short term, the generator output strategy guarantee the optimal resilience potential cost.

Under this kind of mechanism the two agents may interact with each other. If one agent performs not so well, the other will be influenced. For example, the load agent is so stupid to shed all loads all the time, and the DGs agent can not work at all under this case. Our solution is training the two agents alternately until the two agents can work together well. When one agent is learning, the other stays fixed, and exchanges training with the other. Considering that the effect of the load agent on the utility value of the system is more direct, we will train the load agent for first. The full algorithm of this double-agent DRL is shown as Algorithm 1. Initialize DGs agent and Load agent with DDPG and DQN, respectively. The training scenarios are generated by sampling the prior probability distributions described in Appendix. At each decision point of an episode, choose action use ϵ-greedy method. After conducting the action, storage the experience in replay buffer. Train the two agents iteratively, and update the neural network based on the gradient information. The policy obtained after training is evaluated in the test scenario. In our design mechanism, the two agents have no communication with each other. How to introduce an appropriate communication mechanism to improve control performance is left as future work.

IV. RL ENVIRONMENT OF ISLANDED MG

There are mainly two objects in RL: environment and agent. The environment is an object the agent interacts with and tries to learn about. Sometimes there are so many uncertainties in the environment that what the agent can do is interacting with the environment constantly to acquire experiences to improve itself. Furthermore, the environment also

Algorithm 1 Double-Agent Deep Reinforcement Learning Using DQN and DDPG

```
For DGs Agent:
  Randomly initialize critic network $Q_G(s, a^G; \theta^{Q_G})$ and actor $\mu(s; \theta^{\mu})$ with weights $\theta^{Q_G}$ and $\theta^{\mu}$
  Initialize target network $Q_G{}'$ and $\mu'$ with weights $\theta^{Q_G{}'} \leftarrow \theta^{Q_G}$ and $\theta^{\mu'} \leftarrow \theta^{\mu}$
  Initialize replay buffer $\mathcal{R}_G$
For Load Agent:
  Randomly initialize critic network $Q_L(s, a^L; \theta^{Q_L})$ with weights $\theta^{Q_L}$
  Initialize target network $Q_L{}'$ with weights $\theta^{Q_L{}'} \leftarrow \theta^{Q_L}$
  Initialize replay buffer $\mathcal{R}_L$
Initialize $1 \ll N_{TI} \ll M$ for training iteration
for $episode = 1, M$ do
  $flag = int(episode/N_{TI})\%2$
  Initialize a random process $\mathcal{N}$ for action exploration
  Receive initial state $s$
  for $t = 1, T_D$ do
    With probability $\epsilon$ select a random action $a_t$
    With probability $1 - \epsilon$ select action $a_t = [a_t^G, a_t^L]$
      Load action $a_t^L = \max_{a^L} Q_L(s_t, a^L; \theta^{Q_L})$
      DGs action $a_t^G = \mu(s_t; \theta^{\mu}) + \mathcal{N}_t$ according to current policy and exploration noise
    Execute action $a_t$ and observe $r_t$,$s_{t+1}$
    Storage transition $(s_t, a_t^L, r_t^L, s_{t+1})$ in $\mathcal{R}_L$ and $(s_t, a_t^G, r_t^G, s_{t+1})$ in $\mathcal{R}_G$
    if $flag == 0$ then
      Sample a random minibatch of N transitions $(s_t, a_t^L, r_t^L, s_{t+1})$ from $\mathcal{R}_L$
      Set $y_i^L = r_i^L + \gamma^L \max_{a^{L'}} Q_L{}'(s_{i+1}, a^{L'}; \theta^{Q_L{}'})$
      Perform a gradient descent step on $L = \sum_i (y_i^L - Q_L(s_i, a_i^L; \theta^{Q_L}))^2/N$
      Update the target networks : $\theta^{Q_L{}'} \leftarrow \tau\theta^{Q_L} + (1 - \tau)\theta^{Q_L{}'}$
    end if
    if $flag == 1$ then
      Sample a random minibatch of N transitions $(s_t, a_t^G, r_t^G, s_{t+1})$ from $\mathcal{R}_G$
      Set $y_i^G = r_i^G + \gamma^G Q_G{}'(s_{i+1}, \mu'(s_{i+1}; \theta^{\mu'}); \theta^{Q_G{}'})$
      Update critic by minimizing the loss: $L = \sum_i (y_i^G - Q_G(s_i, a_i^G; \theta^{Q_G}))^2/N$
      Update the actor policy using the sampled policy gradient:
      $\nabla_{\theta^{\mu}} \mathcal{J} \approx \sum_i \nabla_{a^G} Q_G(s, a^G; \theta^{Q_G})|_{s=s_i, a^G=\mu(s_i)} \nabla_{\theta^{\mu}} \mu(s; \theta^{\mu})|_{s=s_i}/N$
      Update the target networks: $\theta^{Q_G{}'} \leftarrow \tau\theta^{Q_G} + (1 - \tau)\theta^{Q_G{}'} \quad \theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1 - \tau)\theta^{\mu'}$.
    end if
  end for
end for
```

provides agents with a platform to train and test. In this paper, the environment and agent are exactly MG and MG operator, respectively.

In this section, we will introduce how to establish an RL environment. This environment is exactly a game of islanded MG control, in which the agent need to learn to control the grid to keep it stable for a period of time, and the quality of the operation is reflected in the game score. Firstly, the uncertainties from the components, such as DGs, loads, wind turbines and batteries, within the MG will be described. Then an RL environment for islanded MG operation based on OpenAI Gym [29] and OpenDSS [30] is constructed, which has a general interface compatible with OpenAI Gym. The mechanism for how uncertainty is reflected in the environment will be introduced.

A. UNCERTAINTIES WITHIN THE MG

In the real world, for an agent, the environment is unknown and full of uncertainty. The task of agent is to learn to understand the environment. The uncertainties within the MG are mainly from renewable energy, load demand [31], system energy storage [2] and the power outage duration [3] of DS.

Renewable energy in MG mainly includes wind turbines and solar panels. Their power outputs are intermittent and uncertain because of the weather situation and other factors. We often use beta distribution to describe their outputs uncertainties [31]–[33]. Meanwhile, the uncertainties of load demands can be described using normal distribution [31]. The system energy storage and the power outage duration T_D are usually different under different circumstances. For example, when the DS is out of power, different resource

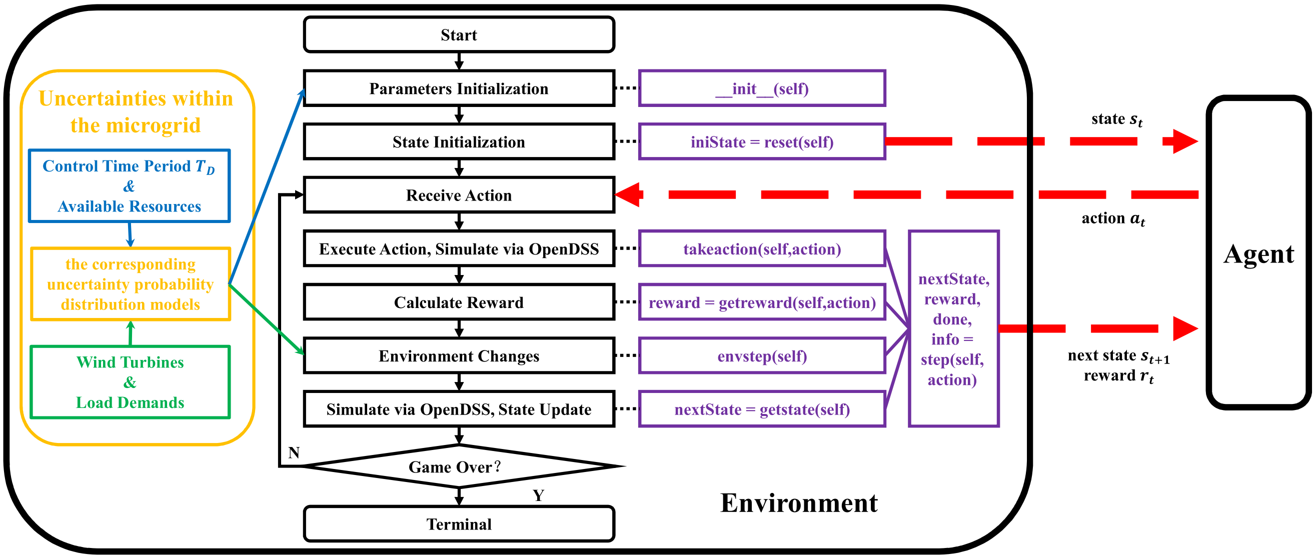
Uncertainties within the microgrid
Control Time Period T_D & Available Resources
the corresponding uncertainty probability distribution models
Wind Turbines & Load Demands
Start
Parameters Initialization
State Initialization
Receive Action
Execute Action, Simulate via OpenDSS
Calculate Reward
Environment Changes
Simulate via OpenDSS, State Update
Game Over?
N
Y
Terminal
__init__(self)
iniState = reset(self)
takeaction(self,action)
reward = getreward(self,action)
envstep(self)
nextState = getstate(self)
nextState, reward, done, info = step(self, action)
state s_t
action a_t
next state s_{t+1}
reward r_t
Agent
Environment

FIGURE 2. Design mechanism of environment. The environment is an object the agent interacts with and tries to learn about. An RL environment for islanded MG operation based on OpenAI Gym and OpenDSS is constructed. The uncertainties from the components, such as DGs, loads, wind turbines and batteries, within the MG are considered.

allocation schemes may cause the system energy storage to vary greatly. Moreover, many factors involving power system characteristics, geographic characteristics, climatic variables and repair process will influence the power outage duration T_D. Both system energy storage and power outage duration T_D are important for the decision-maker to control the MG. In this paper, in order to simulate the senses with different system energy storage and power outage duration, we use uniform distribution to describe the uncertainties of them. The detailed uncertainty model of these elements are shown in Appendices.

B. ISLANDED MG RL ENVIRONMENT BASED ON OpenAI AND OpenDSS

In order to construct a platform to train and test the DRL algorithms, an islanded MG control platforms based on OpenAI and OpenDSS is constructed. It is available to researchers in related fields. OpenAI Gym is a toolkit for developing and comparing the performances of RL algorithms, and defines the interface standard between agent and environment. Nonlinear power flow equations are used for power flow analysis and OpenDSS are used as power flow simulation tool. OpenDSS is a comprehensive electrical power system simulation tool primarily for electric utility power DS. OpenDSS serves as the DS power flow simulation tool in this environment design. A large amount of simulation data, which is used for training of RL, is readily obtained by performing power flow simulation based on OpenDSS.

In RL, the agent obtains observation or state information from the environment, then takes action to execute on the environment and gets reward to improve its policy. When we build an RL environment, the design of state, action, and reward should be considered. In this MG control problem, the state, action and reward are designed as (12), (13) and (14), respectively.

The design mechanism of this MG control RL environment is shown in FIGURE 2. The interaction between agent and environment and the flow of environment design in one episode is described. In each episode, the agent completes a full game playing until the simulation time arrives the power outage duration T_D, which means the game of MG control terminates. Then, a new game restarts and the agent learns during a large number of games playing.

In one episode, the flow of environment design is as follows:

(1) Start. A new episode starts.
(2) Parameters initialization. Initialize the MG case, and the related parameters including the power outage duration T_D and available generation resources which can be obtained by sampling their probability distribution models described in Section III-A. In different episodes, T_D and available generation resources at $t = 0$ may be different. In the begining of each episode, the probability model of power outage duration T_D and available generation resources are sampled to generate the specific T_D and available generation resources, just like the blue boxes and arrows in the left part of FIGURE 2.
(3) State resets. Environment gives the reset state to the agent. This is the initial state where the agent begins to control the MG.
(4) Receive and execute action. Environment receives action from the agent. The action is executed on the environment, and OpenDSS is used as a simulation tool to calculate the power flow.

TABLE 1. Device Information of the MG.

Device Name	Device Type	Parameters
BT1	Battery Storage System	$V = 1.0$p.u., $f = 60$Hz, $S^N = 500$kVA, $E^M_{BT1} = 800$kWh
BT2	Battery Storage System	$P = 10$kW, $Q = 10$kVar, $P^N = 50$kW, $E^M_{BT2} = 400$kWh
DG1	Diesel Generator	$P = 10$kW, $Q = 10$kVar, $P^N = 50$kW, $E^M_{BT2} = 1800$kWh
WT1,2	Wind Turbine	$P = 40/50$kW, $S^N = 100$kVA, $p_f = 0.95$
L1	Primary Load	$P = 120$kW, $Q = 20$kVar
L2,3	Secondary Load	$P = 35/36$kW, $Q = 5/11$kVar
L4	Tertiary Load	$P = 50$kW, $Q = 15$kVar
Tr	Transformer	$V = 4.16/0.48$kV, $S^N = 500$kVA, $r = 0.1\%$

(5) Calculate reward. The MG utility value in this time step is calculated. In this environment, we divide one day into 96 time steps, i.e. 15 minutes per time step.

(6) Environment changes. As Section III-A describes, the output of wind turbines and load demands have strong uncertainties, their real values in each time step can be obtained by sampling their probability distribution models, just like the green boxes and arrows in the left part of FIGURE 2. The outputs of wind turbines and load demands will be set according the sampling results.

(7) State update. After executing the action, state of the environment transfers to the next state, and the next state can be obtained after simulation.

(8) Determine if the game is over. The information involving next state, reward, if the game is over, and attached information are given to the agent. If the game is over, this episode terminates. Otherwise, go to next time step and repeat procedure (4)-(8) until the game ends.

The purple boxes attached to the flow in FIGURE 2 are corresponding PYTHON functions. They describe how to implement the corresponding functions. The interface between agent and environment is designed according to the standard of OpenAI Gym.

V. SIMULATION RESULTS

In this section, we validate our methods through several case studies. OpenDSS is used as power flow simulation tools. Python 3.6.5 is used to realize the RL agent. The calculation is realized on a desktop PC with a 2.60GHz CPU(Intel(R) Xeon(R) E5-2670 0) and 64GB RAM.

A. CASE INFORMATION

The MG used to validate the proposed method is shown in FIGURE 3, which includes 7 buses (B1, B2, ..., B7, excepts the point of common coupling PCC), 1 transformer Tr, 2 battery storage systems BT1 and BT2, 1 diesel generator DG1, 2 wind turbines WT1 and WT2, and 4 loads L1, L2, L3 and L4 with various priorities. The specific parameters of the MG are shown in Table 1.

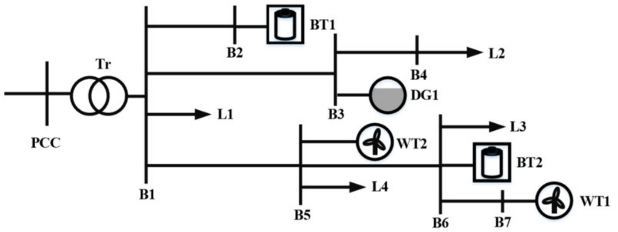

FIGURE 3. Topology of the microgid.

TABLE 2. Supply Income and Outage Loss for Load of Different Load Subsets.

Priority	i_{L_i}($/kWh)	$c^p_{L_i}$($/kWh)	$c^u_{L_i}$($/kWh)
Primary	2.0	0.8	1.0
Secondary	1.5	0.6	0.75
Tertiary	1.2	0.48	0.6

TABLE 3. Load Shedding Action.

L_S	Meaning	Loads Shed
0	Shed no loads	None
1	Shed tertiary load	L4
2	Shed secondary and tertiary loads	L2,L3,L4
3	Shed all loads	L1,L2,L3,L4

The energy source in this case includes 2 battery storage systems, 1 diesel generator and 2 wind turbines. BT1, with a capacity of 500kVA and rated power of 400kW, is served as the master source in the MG and used to balance system power flow. Thus, bus B2 is the slack bus . Both BT2 and DG1 have a rated generation power of 50kW. They are considered as adjustable DGs. The wind turbines are considered to be non-adjustable and recognized as negative loads. The loads are classified into 3 subsets according to their priorities: primary load L1, secondary loads L2 and L3, and tertiary load L4. Their supply income, planned and unplanned outage loss are defined in Table 2. Correspondingly, load shedding action L_S can be defined as shedding loads with different priorities and is shown in Table 3. Load shedding strategy is be used to shed less critical loads gradually to ensure the continuous power supply to the more critical loads. As the load shedding action L_S changes from 0 to 3, more critical loads will be shed.

B. BASE CASE STUDY

The outage duration of DS is up to the repair process and the severity of the damage caused by the disaster. The power outage duration of blackout in the United States and Canada on August 14th of 2003 [34] reached 29 hours. The India's Blackout [35] in 2012 lasted for nearly 2 days. In order to simulate the operation of DS in extreme situations, the outage

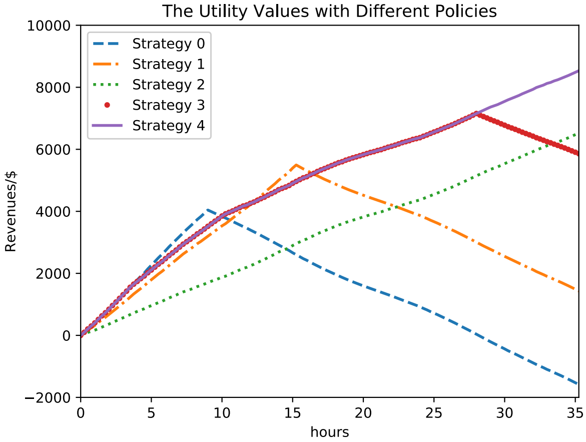

FIGURE 4. The utility values with different policies.

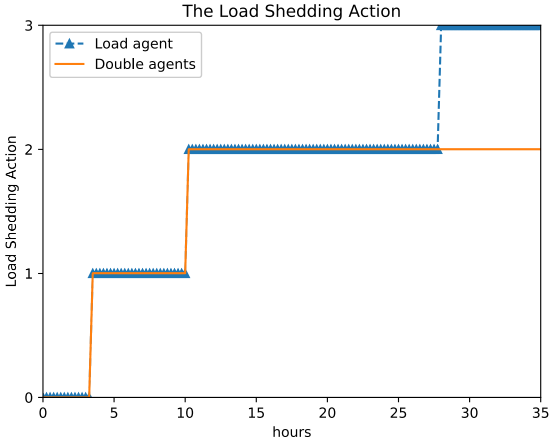

FIGURE 5. The load shedding action.

duration of DS is chosen up to 30-40 hours in this paper. Assume that the MG is disconnected from a DS with full generation resources available, and the time period T_D is set equal to 35 hours, the initial available generation resources are assumed to be 100%. Each train and test scenario is generated according to the probability distribution models described in Section IV-A. Therefore, the agent encounters a brand new scenario during the test.

In order to demonstrate the effectiveness of the proposed method, we use several strategies for comparison. On the load side, constant load shedding strategies that use actions $L_S = 0$, $L_S = 1$, and $L_S = 2$, respectively are proposed. They are denoted as Strategy 0, Strategy 1 and Strategy 2, respectively. The control policy with only load agent and double agents are denoted as Strategy 3 and Strategy 4. On the source side, manual adjustment (MA) method is proposed for comparison, the controllable DGs are used in the order of resilience potential factor described in (23) from low to high in each regulation. That means we will use the DGs with low resilience potential factor for first. All the DGs output are controlled except Strategy 4 using manual adjustment method. The test results are shown in FIGURE 4, FIGURE 5, FIGURE 6, and FIGURE 7.

FIGURE 4 shows the MG utility values with different control policies. The MG utility values under all strategies first increase with time T_D in about the first 10 hours. The MG remains normal operation during this period. As T_D further increases, however, the MG utility values under Strategy 0 and Strategy 1 begin to decrease, while MG utility values under others keep increasing. This is because under Strategy 0 and Strategy 1 the heavy load causes the system master battery to run out, as shown in FIGURE 6. While under Strategy 2 the system only remains primary load supply ($L_S = 2$), and under Strategies 3 and 4 the system sheds load of lower priority gradually as shown in FIGURE 5. Also, FIGURE 5 shows the status of loads. In the initial period of time, $L_S = 0$ and no load is shed. Then, load L4 is shed when $L_S = 1$. Gradually, more loads are shed with the consumption of system resources; all the loads are shed when $L_S = 3$. Strategy 2 is the most conservative strategy as it only supplies the primary load, its utility value may exceed that of Strategy 4 if time T_D continues to increase. It can also be seen from FIGURE 7 that the remaining resources of the system under Strategy 2 is the maximum. However, at the time instant of $T_D = 35$ h, Strategy 4 maximizes the MG utility value.

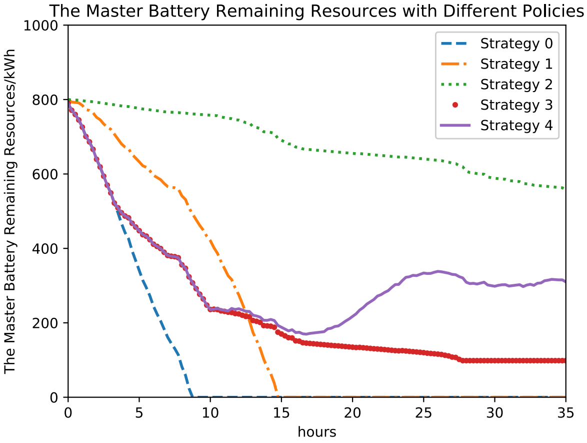

FIGURE 6. The master battery remaining resources.

The difference between Strategy 3 and Strategy 4 is the DGs control policy, which causes the difference in the load shedding action. As shown in FIGURE 8, the difference of two policies is the output of DG1. Under manual adjustment the output of DG1 is more conservative. The DGs control policy under RL is bolder, which not only meets the load demand but even charges the master battery at some time instant as shown in FIGURE 6. This manner ensures that the master battery has sufficient resources. For this reason, Strategy 4 will not shed all loads in later stages as shown in FIGURE 5.

FIGURE 9 shows the outputs of 3 adjustable devices and the loads demand (include the negative loads: wind turbines) in one test scenario under the double RL agent strategy. With the consumption of system resources, more loads are

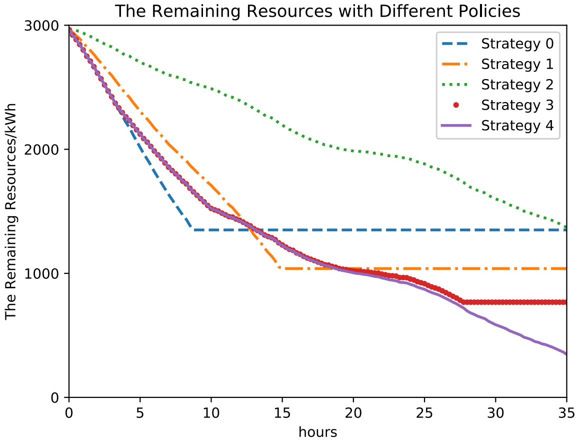

FIGURE 7. The remaining resources with different policies.

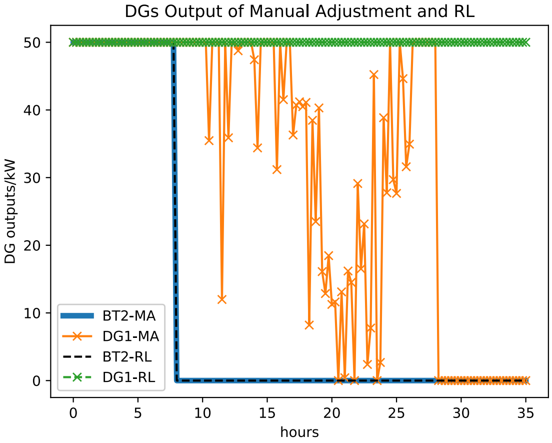

FIGURE 8. DGs output of RL and Manual Adjustment.

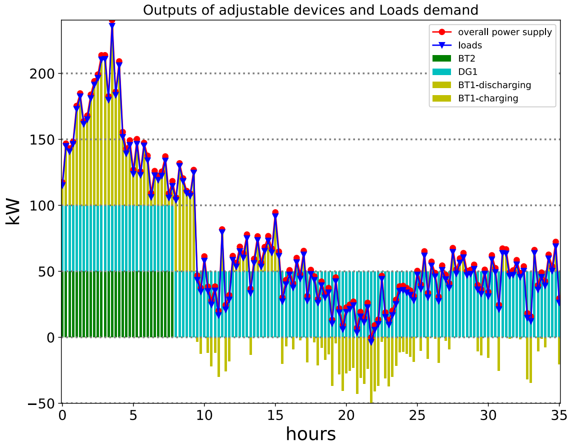

FIGURE 9. Outputs of adjustable devices and Loads demand under strategy 4.

gradually shed. As seen from FIGURE 9, the loads demand becomes relatively smaller as time increases. The overall power supply is slightly larger than loads demand because of the power losses in the transmission lines. BT1 serves as the master source and is used to balance system power flow.

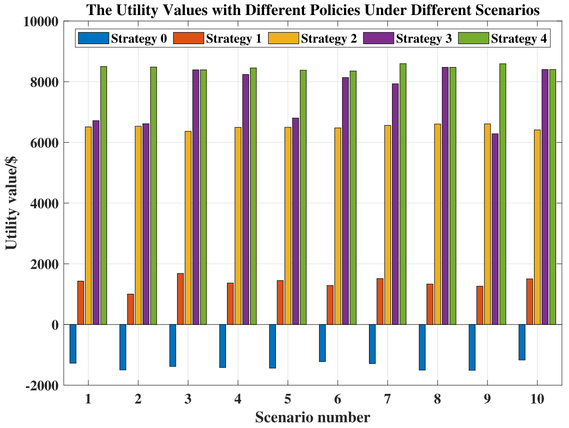

FIGURE 10. The utility values with 5 strategies under different scenarios.

It discharges when there is heavy load and charges when most loads are shed to adsorb the outputs of wind turbines.

Sample the probability distributions described in the appendix and randomly generate 10 test scenarios. Assume that the MG is disconnected from a DS with full generation resources available, and the power outage duration is also set equal to 35 hours. The utility values with 5 strategies under different scenarios are shown in FIGURE 10. Among these 10 test scenarios, the utility value of Strategy 4 (double agents) remains highest . The utility values of Strategy 3 (only load agent) are closed to those of Strategy 4 in certain scenarios but remains second highest in most scenarios. The utility values of all strategies fluctuate in different scenarios because of the uncertainty of loads and wind turbine. The results in different test scenarios demonstrate that the proposed method can cope with the uncertainty well in the system.

C. CASE STUDY UNDER VARIOUS CONDITIONS

For a given MG, the initial available generation resources and the power outage duration T_D for MG control will influence the control policy. In this section, the proposed method is tested under variations of these factors. The results demonstrate that the proposed method is able to adapt to various conditions to make full use of the limited resources and maximize the utility value of the MG.

1) INITIAL AVAILABLE GENERATION RESOURCES

When the MG is disconnected from the DS, the initial available generation resources are assumed to be 80%, 85%, 90%, 95%, and 100%, respectively. The duration T_D for MG control remains constant.

The MG utility values with double agents under various initial available generation resources are shown in FIGURE. 11, and the load shedding actions are shown in FIGURE. 12. In FIGURE. 12, we can see that with the decrease of initial available generation resources, the time for the agent to shed load of the same level is advanced, the "bigger" load shedding action occupies more proportion.

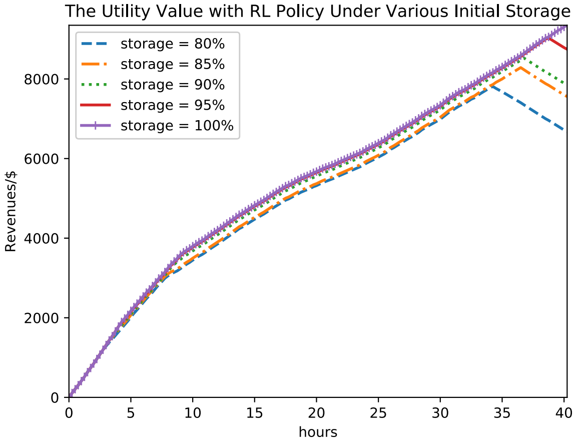

FIGURE 11. The utility values under various initial storage.

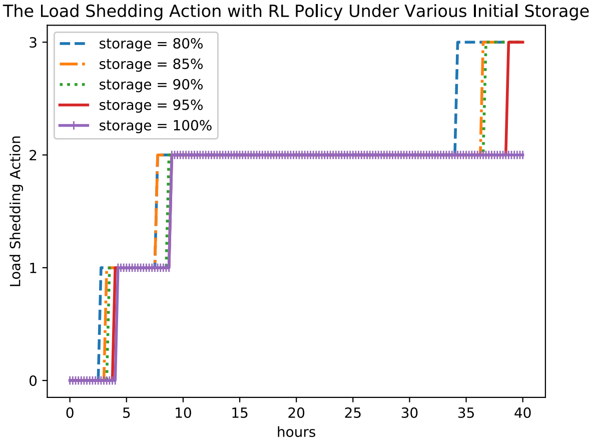

FIGURE 12. Load shedding action with RL.

TABLE 4. Utility Value of Different Strategies With Different Initial Resources ($, at T_D = 35 h).

$E_G(0)$	Strategy 0	Strategy 1	Strategy 2	Strategy 3	Strategy 4
80%	-2722.70	79.733	6383.96	5706.38	**7817.60**
85%	-2404.81	514.52	6383.96	5832.83	**8068.70**
90%	-2094.16	867.35	6383.96	5982.59	**8194.11**
95%	-1631.88	1191.90	6383.96	6157.84	**8369.36**
100%	-1322.13	1513.60	6383.96	6204.75	**8416.26**

Table 4 shows the utility value of different strategies at $T_D = 35$ h, and the utility value of double RL agents control policy remains the largest during $E_G(0)$ varies. It demonstrates the capability of the proposed method to adapt to various generation resource conditions.

2) POWER OUTAGE DURATION T_D FOR MG CONTROL

In this case, the power outage duration T_D for MG control is changed from 30 h to 40 h with an interval of 2 h, while the initial available generation resources remain constant.

TABLE 5. Utility Value of Different Strategies With Different T_D ($, $E_G(0)$ = 100%).

T_D (h)	Strategy 0	Strategy 1	Strategy 2	Strategy 3	Strategy 4
30	-219.006	2573.28	5598.14	**7636.95**	**7636.95**
32	-670.187	2134.77	6000.40	7839.77	**8039.20**
34	-1094.71	1723.25	6345.07	7500.16	**8383.87**
36	-1567.76	1263.80	6758.93	5685.33	**8820.71**
38	-2035.87	809.943	7141.73	5310.84	**9203.51**
40	-2516.23	344.251	7531.78	4926.55	**9496.66**

TABLE 6. Load Shedding Action Frequency of RL With Different T_D (%, $E_G(0)$ = 100%).

T_D (h)	$L_S = 0$	$L_S = 1$	$L_S = 2$	$L_S = 3$
30	14.05	19.83	66.12	0
32	13.18	18.60	68.22	0
34	12.41	17.52	70.07	0
36	12.41	15.86	71.72	0
38	11.76	15.03	73.20	0
40	11.18	14.29	73.91	0.62

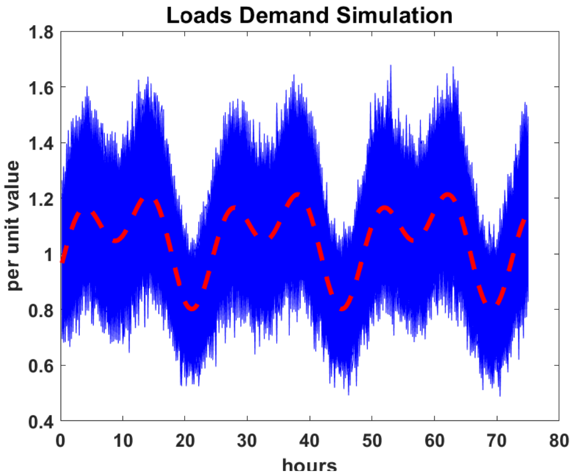

FIGURE 13. Loads demand simulation(per unit value).

The MG utility values of different strategies with different T_D are shown in Table 5. Table 5 shows the MG utility value of double agents (Strategy 4) is the largest in every T_D. Table 6 shows the proportion of each load shedding action in each T_D, more loads are shed with T_D increases. It indicates that more generation resources are used for the power supply of high priority load.

The agent can automatically adjust the load shedding strategy according to the length of control period. The results indicate that the capability of the proposed method to adapt to various control time period.

D. PERFORMANCE EVALUATION

RL is a kind of method which make decision based on the information of current state. It follows the Markov property. It does not require the future information and takes action according to the current state. However, many of the previous state-of-art methods, such as classic optimization methods, robust optimization, dynamic programming, etc., require the prediction for the future to make decision. Moreover, in the high dimension problem, the computational costs are huge.

Although there is no prediction for the future, RL can learn the trends of the future through the training process on

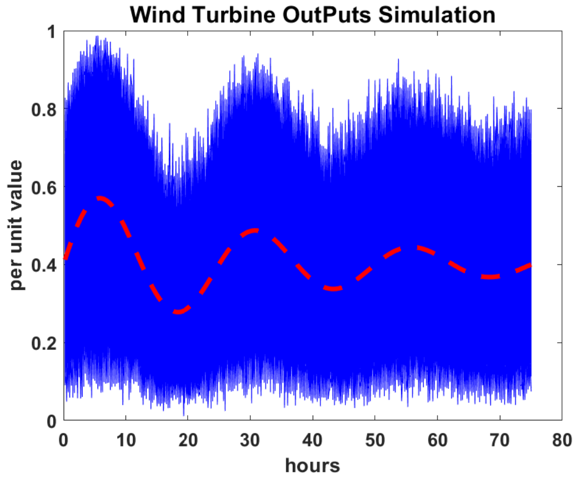

FIGURE 14. **Wind turbine outputs simulation(per unit value).**

historical data, thus making the decision. In this paper, several constant load shedding strategies are proposed as comparisons. Among these strategies, the performance of double agent RL is the best. The training process takes about 10 hours on a desktop PC with a 2.60GHz CPU(Intel(R) Xeon(R) E5-2670 0) and 64GB RAM to achieve the aforementioned training effect. After training, in one test scenario, the well-learned RL agent can derive the solution within one second. The training speed of can be further improved by using GPU and parallel simulation techniques.

Theoretically, those kinds of method which utilize the future information can achieve better performance than RL. That is because they get more information. Therefore, in this paper we do not take this kind of methods as comparison. In [36], the RL method can achieve an effect close to the optimization benchmark with perfect forecast. In [19], the proposed cooperative RL algorithm can do better than scenario-based algorithm. In our latest research, a well-learned reinforcement learning agent without the prediction for the future can achieve a performance close to the state-of-the-art dynamic programming which has perfect prediction, but with less computation time. The calculation time of the dynamic programming is about 2 minutes, while the RL can yield close results within one second. The author believes that how to utilize the information of future in RL is also a problem worthy of study.

VI. CONCLUSION

In this paper, a resilience enhancing problem is converted to a decision making problem. A multi-agent DRL model is proposed to control an islanded MG with limited generation resources. An RL environment for islanded MG operation based on OpenAI Gym is constructed, which has a general interface compatible with and can be published to OpenAI Gym. The proposed RL policy is applied to an MG with wind turbines, diesel generators and storage devices. It realizes a dual control: the energy storage management on source side and load shedding policies on the load side. The policy maximizes the utility value of the MG in a limited time period, thus improving the resilience. Test results demonstrate its effectiveness under various conditions such as different available generation resources and MG control time periods.

Our future work will use a larger scale case to validate the proposed method's ability to scale up, consider using parallel simulation to accelerate the agent training process and try to introduce an appropriate communication mechanism between agents to improve control performance.

APPENDICES

The uncertainties of renewable energy can be described as beta function,

$$f(x) = x^{\alpha-1}(1-x)^{\beta-1} \tag{24}$$

$$p = \frac{\alpha}{\alpha+\beta} \tag{25}$$

$$\sigma^2 = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} \tag{26}$$

where α, β are shape parameters of beta function. Beta function models the occurrence of real power values x when a certain prediction value p has been forecasted. α, β can be calculated as follows,

$$\alpha = \frac{p^2(1-p)}{\sigma^2} - p \tag{27}$$

$$\beta = \frac{p(1-p)^2}{\sigma^2} - (1-p) \tag{28}$$

where p is the normalized predicted power output, σ^2 is the variance of the beta distribution.

Generally speaking, there is a positive correlation between forecast error and predicted power output, a linear fit for the standard deviation as a function of the predicted power proposed in [33] is as follows:

$$\sigma_W = 0.249p + 0.035 \tag{29}$$

With the predicted DG outputs and the three formulas above, the parameters of beta functions for the current prediction data can be calculated. Meanwhile, the uncertainties of load demands can be described using normal distribution [31]:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-p)^2}{2\sigma^2}\right) \tag{30}$$

and a linear fit for the standard deviation σ_L of load as a function of the predicted power p can be

$$\sigma_L = 0.1p \tag{31}$$

The fluctuation of profiles of loads demand and wind turbine outputs are shown in FIGURE 14 and FIGURE 13 (per unit value). 1000 profiles of loads demand and the output of wind turbine are shown, and the dashed lines are the average value.

REFERENCES

[1] R. J. Campbell and S. Lowry, *Weather-Related Power Outages and Electric System Resiliency*. Washington, DC, USA: Congressional Research Service, Library of Congress, 2012.

[2] H. Gao, Y. Xu, C.-C. Liu, and Y. Chen, "Dynamic load shedding for an islanded microgrid with limited generation resources," *IET Gener., Transmiss. Distrib.*, vol. 10, no. 12, pp. 2953–2961, Sep. 2016.

[3] R. Nateghi, S. D. Guikema, and S. M. Quiring, "Comparison and validation of statistical methods for predicting power outage durations in the event of hurricanes," *Risk Anal.*, vol. 31, no. 12, pp. 1897–1906, Dec. 2011.

[4] S. D. Guikema, S. M. Quiring, and S.-R. Han, "Prestorm estimation of hurricane damage to electric power distribution systems," *Risk Anal.*, vol. 30, no. 12, pp. 1744–1752, Dec. 2010.

[5] Y. Wang, C. Chen, J. Wang, and R. Baldick, "Research on resilience of power systems under natural disasters—A review," *IEEE Trans. Power Syst.*, vol. 31, no. 2, pp. 1604–1613, Mar. 2016.

[6] W. Yuan, J. Wang, F. Qiu, C. Chen, C. Kang, and B. Zeng, "Robust optimization-based resilient distribution network planning against natural disasters," *IEEE Trans. Smart Grid*, vol. 7, no. 6, pp. 2817–2826, Nov. 2016.

[7] C. Chen, J. Wang, and D. Ton, "Modernizing distribution system restoration to achieve grid resiliency against extreme weather events: An integrated solution," *Proc. IEEE*, vol. 105, no. 7, pp. 1267–1288, Jul. 2017.

[8] C. Abbey, D. Cornforth, N. Hatziargyriou, K. Hirose, A. Kwasinski, E. Kyriakides, G. Platt, L. Reyes, and S. Suryanarayanan, "Powering through the storm: Microgrids operation for more efficient disaster recovery," *IEEE Power Energy Mag.*, vol. 12, no. 3, pp. 67–76, May 2014.

[9] C. Chen, J. Wang, F. Qiu, and D. Zhao, "Resilient distribution system by microgrids formation after natural disasters," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 958–966, Mar. 2016.

[10] Y. Bian and Z. Bie, "Multi-microgrids for enhancing power system resilience in response to the increasingly frequent natural hazards," *IFAC-PapersOnLine*, vol. 51, no. 28, pp. 61–66, 2018.

[11] A. Arab, A. Khodaei, S. K. Khator, and Z. Han, "Electric power grid restoration considering disaster economics," *IEEE Access*, vol. 4, pp. 639–649, 2016.

[12] M. Choobineh and S. Mohagheghi, "Emergency electric service restoration in the aftermath of a natural disaster," in *Proc. IEEE Global Humanitarian Technol. Conf. (GHTC)*, Oct. 2015, pp. 183–190.

[13] D. Q. Oliveira, A. C. Zambroni de Souza, A. B. Almeida, M. V. Santos, B. I. L. Lopes, and D. Marujo, "Microgrid management in emergency scenarios for smart electrical energy usage," in *Proc. IEEE Eindhoven PowerTech*, Jun. 2015, pp. 1–6.

[14] J. A. P. Lopes, C. L. Moreira, and A. G. Madureira, "Defining control strategies for MicroGrids islanded operation," *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 916–924, May 2006.

[15] C. Wang, P. Ju, S. Lei, Z. Wang, F. Wu, and Y. Hou, "Markov decision process-based resilience enhancement for distribution systems: An approximate dynamic programming approach," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2498–2510, May 2020.

[16] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis, "Mastering the game of go without human knowledge," *Nature*, vol. 550, no. 7676, p. 354, 2017.

[17] J. Tan, T. Zhang, E. Coumans, A. Iscen, Y. Bai, D. Hafner, S. Bohez, and V. Vanhoucke, "Sim-to-real: Learning agile locomotion for quadruped robots," 2018, *arXiv:1804.10332*. [Online]. Available: http://arxiv.org/abs/1804.10332

[18] X. Pan, Y. You, Z. Wang, and C. Lu, "Virtual to real reinforcement learning for autonomous driving," 2017, *arXiv:1704.03952*. [Online]. Available: http://arxiv.org/abs/1704.03952

[19] W. Liu, P. Zhuang, H. Liang, J. Peng, and Z. Huang, "Distributed economic dispatch in microgrids based on cooperative reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2192–2203, Jun. 2018.

[20] E. Foruzan, L.-K. Soh, and S. Asgarpoor, "Reinforcement learning approach for optimal distributed energy management in a microgrid," *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 5749–5758, Sep. 2018.

[21] IRE Series, *Microgrids and Active Distribution Networks*. London, U.K.: The institution of Engineering and Technology, 2009.

[22] J. R. Busemeyer, "Dynamic decision making," *Int. Encyclopedia Social Behav. Ences*, vol. 47, no. 2, pp. 3903–3908, 2001.

[23] Y. Shoham, R. Powers, and T. Grenager, "Multi-agent reinforcement learning: A critical survey," Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep., May 2003. [Online]. Available: http://jmvidal.cse.sc.edu/library/shoham03a.pdf

[24] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," 2013, *arXiv:1312.5602*. [Online]. Available: http://arxiv.org/abs/1312.5602

[25] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[26] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*. [Online]. Available: http://arxiv.org/abs/1509.02971

[27] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. ICML*, 2014.

[28] J. Xiong, Q. Wang, Z. Yang, P. Sun, L. Han, Y. Zheng, H. Fu, T. Zhang, J. Liu, and H. Liu, "Parametrized deep Q-Networks learning: Reinforcement learning with discrete-continuous hybrid action space," 2018, *arXiv:1810.06394*. [Online]. Available: http://arxiv.org/abs/1810.06394

[29] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "OpenAI gym," 2016, *arXiv:1606.01540*. [Online]. Available: https://arxiv.org/abs/1606.01540

[30] D. Montenegro, M. Hernandez, and G. A. Ramos, "Real time OpenDSS framework for distribution systems simulation and analysis," in *Proc. 6th IEEE/PES Transmiss. Distrib., Latin Amer. Conf. Expo. (T&D-LA)*, Sep. 2012, pp. 1–5.

[31] Z. Wang and J. Wang, "Self-healing resilient distribution systems based on sectionalization into microgrids," *IEEE Trans. Power Syst.*, vol. 30, no. 6, pp. 3139–3149, Nov. 2015.

[32] A. Fabbri, T. GomezSanRoman, J. RivierAbbad, and V. H. MendezQuezada, "Assessment of the cost associated with wind generation prediction errors in a liberalized electricity market," *IEEE Trans. Power Syst.*, vol. 20, no. 3, pp. 1440–1446, Aug. 2005.

[33] S. Bofinger, A. Luig, and H. G. Beyer, "Qualification of wind power forecasts," in *Proc. Global Windpower Conf.*, Paris, France, vol. 2, no. 5, 2002.

[34] B. Liscouski and W. Elliot, "Final report on the august 14, 2003 blackout in the united states and canada: Causes and recommendations," *Rep. US Dept. Energy*, vol. 40, no. 4, p. 86, 2004.

[35] L. Lei Lai, D. Ramasubramanian, H. Tian Zhang, F. Yuan Xu, S. Mishra, and C. S. Lai, "Lessons learned from july 2012 Indian blackout," in *Proc. 9th IET Int. Conf. Adv. Power Syst. Control, Operation Manage. (APSCOM)*, 2012, p. 174.

[36] T. Chen and W. Su, "Indirect customer-to-customer energy trading with reinforcement learning," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 4338–4348, Jul. 2019.

HUANHUAN NIE (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2019, where he is currently pursuing the master's degree with the Department of Electrical Engineering and Applied Electronic Technology.

His research interests include reinforcement learning and cyber-physical system modeling.

YING CHEN (Member, IEEE) received the B.E. and Ph.D. degrees in electrical engineering from Tsinghua University, Beijing, China, in 2001 and 2006, respectively.

He is currently an Associate Professor with the Department of Electrical Engineering and Applied Electronic Technology, Tsinghua University. His research interests include parallel and distributed computing, electromagnetic transient simulation, cyber-physical system modeling, and cyber security of smart grid.

YUE XIA (Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from China Agricultural University, Beijing, China, in 2009 and 2011, respectively, and the Ph.D. degree in electrical engineering from the Technische Universität Berlin, Germany, in 2016.

From 2017 to 2019, he held a postdoctoral position with the Department of Electrical Engineering, Tsinghua University. He is currently an Associate Professor with the College of Information and Electrical Engineering, China Agricultural University. His research interests include power electronic systems, electrical machines, wind power, power system transients modeling, and reinforcement learning.

SHAOWEI HUANG (Member, IEEE) received the B.S. and Ph.D. degrees from the Department of Electrical Engineering, Tsinghua University, Beijing, China, in July 2006 and June 2011, respectively.

From 2011 to 2013, he held a postdoctoral position with the Department of Electrical Engineering, Tsinghua University, where he is currently an Associate Professor. His research interests include power systems modeling and simulation, power system parallel and distributed computing, complex systems and its application in power systems, and artificial intelligence.

BINGQIAN LIU received the M.S. degree in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 2016.

She is currently an Engineer with the Electric Power Research Institute of State Grid Fujian Electric Power Company Ltd., Fuzhou, China. Her research interests include intelligent power distribution inspection, low-voltage distribution network information technology, and the ubiquitous power Internet of Things.