*electronics*

MDPI

*Article*

# Research on the Control of Multi-Agent Microgrid with Dual Neural Network Based on Priority Experience Storage Policy

**Fengxia Xu [1,*], Shulin Tong [1], Chengye Li [2] and Xinyang Du [2]**

[1] School of Mechanical and Electrical Engineering, Qiqihar University, Qiqihar 161006, China
[2] State Grid Heilongjiang Provincial Electric Power Co., Ltd. Qitaihe Power Supply Company, Qitaihe 154699, China
* Correspondence: xufengxia_hit@163.com

**Abstract:** In this paper, an improved dual neural network control method based on multi-agent system is proposed to solve the problem of rating the frequency deviation and voltage deviation of the microgrid system due to the uneven impedance distribution of the circuit. The microgrid multi-agent system control model is constructed; the microgrid operation problem is transformed into Markov decision-making process, and the frequency error model of distributed secondary control adjusting system is established. In the course of training, the priority experience replay mechanism is introduced to accelerate the training reward return by using the experience of high feedback reward, and the frequency and voltage bias of the microgrid system are reduced. The model of isolated island microgrid of distributed power supply communication topology is established, and the control strategy of double neural network is simulated. Compared with the traditional sagging control method, the double neural network algorithm proposed in this paper stabilizes the frequency of the grid at rated frequency and improves the convergence speed. Simulation results show that the proposed method is helpful to provide stable and high-quality power resources for enterprises.

**Keywords:** microgrids; multi-agent systems; deep reinforcement learning; neural network



**Citation:** Xu, F.; Tong, S.; Li, C.; Du, X. Research on the Control of Multi-Agent Microgrid with Dual Neural Network Based on Priority Experience Storage Policy. *Electronics* **2023**, *12*, 565. https://doi.org/10.3390/electronics12030565

Academic Editor: Ali Mehrizi-Sani

Received: 7 December 2022
Revised: 12 January 2023
Accepted: 20 January 2023
Published: 22 January 2023



## 1. Introduction

Microgrid are mini-grids composed of distributed power sources, loads, and energy storage devices [1]. As an important model for future smart grids, they have received extensive attention from scholars at home and abroad [2–4]. At present, the control architecture of a microgrid is mainly divided into centralized and distributed; centralized controllers are usually affected by network communication, etc., which leads to the failure of generation fluctuation, while distributed control strategy is an ideal structure; each distributed power controller is independent and equal to ensure the stability of the system and ensure the information sharing of each distributed power [5–8], so it is widely used in the microgrid. However, the flexibility of microgrid and the droop control method used by distributed power supplies cause the frequency and voltage of the system to deviate from the rated value [9,10], In Ref. [11] for the AC-DC microgrid, a distributed compensation dynamic feedback method is proposed to improve the accuracy by droop gain and total load current used in the distributed compensation strategy. However, the traditional control methods are usually limited by the fuzzy microgrid itself parameters, and the instability of renewable energy sources, loads, and other factors in the microgrid has an enormous impact, so regulating the output power of the distributed power supply by designing a secondary frequency controller to control the frequency of the system becomes a vital issue.

With the development of artificial intelligence technology, multi-agent systems with autonomous and spontaneous characteristics are suitable for distributed power control of microgrids; the deep reinforcement learning algorithm has been observed by power system personnel for solving sequential decision problems of microgrids [12]; the agents

interact with the environment, and there are also complex relationships between agents such as collaboration and competition so that the agents' communication of multi-agent systems can achieve cooperative transmission of information to achieve global control [13]. Over the years, deep reinforcement learning methods have become promising practical issues in microgrids, such as energy management and load frequency control [14,15]. In Ref. [16], the authors use multi-agent reinforcement learning algorithms to accomplish the economic optimal control objectives of each microgrid under the premise of satisfying the supply–demand balance. In Ref. [17], the authors combines artificial intelligence techniques to ensure power quality in microgrids, maximizing the integrated performance of each agent, so the algorithms using deep reinforcement learning (DRL) have received attention from power system researchers for their ability to solve sequential decision problems [18]. The current reinforcement learning algorithms can be divided into two categories based on value functions and policy gradients. In studying deep reinforcement learning algorithms based on value functions, Ref. [19] establishes algorithms using Deep Q-Network to solve the cooperative control problem of energy storage devices in the established microgrid model. In Ref. [20], the authors used algorithms of DQN and DDQN to solve the power usage optimization problem in the microgrid, and the overestimation problem solved by the dual neural network reduces the error estimation suitable for minimizing the cost problem. In another large class of research on deep reinforcement learning algorithms based on policy gradients, in [21,22] the authors develop a microgrid model for solving the cost minimization of optical energy storage and air conditioning systems to demonstrate the feasibility of policy neural network dealing with uncertain models. The Actor-Critic framework is proposed to be able to solve the continuous state space problem well according to the deep reinforcement learning algorithm combining value function and policy gradient [23–25]. In Ref. [26], the authors establish a hybrid AC-DC microgrid for droop control problem using the multi-agent Actor-Critic method to solve the microgrid problem in segments to meet the state scale of different systems. In Ref. [27], the authors solve the microgrid DC bus voltage regulation problem using online reinforcement learning method with deterministic policy gradients. The integrated performance, frequency response, and voltage regulation of distributed power sources in smart grids all have important effects. In [28], a collaborative RL algorithm for economic scheduling is proposed for the challenges of trading between microgrids and upper primary networks and operational risks, but frequency control and power scheduling are not considered.

Combining distributed control with the deep reinforcement learning algorithm in this paper, we propose a Priority Experience Storage Actor-Critic Neural Network for the control problem of frequency and voltage deviation due to primary control of the microgrid, which treats the microgrid as a multi-agent system to achieve control frequency and voltage stabilization of the system, which ensures the high precision optimal solution of the control objective and makes the microgrid system have better dynamic performance.

The main structure of this paper is as follows. Section 2 describes the problem of modeling microgrid systems at the primary control. Section 3 describes the modeling framework and the application of the deep reinforcement learning algorithm PES-Actor-Critic neural network in power grids. Section 4 is an experimental simulation with an example of a distributed grid, and Section 5 is a summary of the whole paper giving conclusions.

Based on the above analysis, the main contributions of this paper are summarized as follows:

(1) In order to solve the nonlinear coupling problem of microgrid systems, this paper proposes an Actor-Critical Neural Network, which combines deep reinforcement learning and the Priority Experience Storage Policy to improve the dual neural network structure algorithm. The control mode adopts distributed method to realize the information transmission between each neighboring agent so that the frequency control of each agent reaches the optimal expectation.
(2) The improved dual neural network adopts the method of strategic gradient updating, iterative optimal adjustment of data deviation caused by primary control layer of the

microgrid using target value neural network and predictive value neural network, solves the traditional neural network hoist problem, and ensures load sharing of each agent.

(3) Compared with the traditional control algorithm, this paper combines the method of preemptive experience playback to make the control algorithm have faster convergence speed, and the analysis of plug-and-play characteristics of microgrid ensures better robustness of grid system.

## 2. Microgrid Control System

This paper's islanded microgrid control system mainly includes the battery, gas turbine, wind turbine, photovoltaic, load, and interface with sizeable external grid, as shown in Figure 1. In order to control the islanded microgrid system more conveniently and accurately and improve the microgrid system's agent decision-making level, the microgrid is divided into primary control and secondary control. The primary control uses a droop control method using a multi-agent system to realize distributed control strategies for each distributed power source as different agents and to meet the global control objectives through coordination between neighboring agents. The total consumption in the secondary control multi-agent system is highly correlated with the agents that do not provide energy. The agents must be reasonably allocated to each intelligence after calculating the power deviation, considering the power balance, output upper and lower limits, and other constraints.

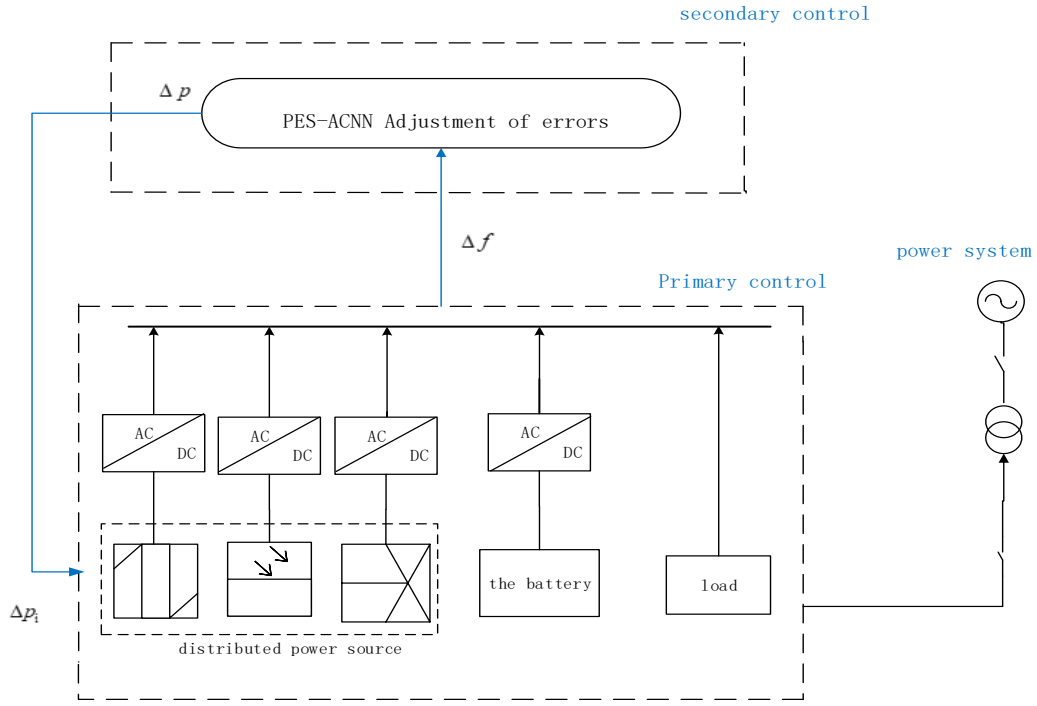



**Figure 1.** The Microgrid control model. Indicates the deviation between the rated frequency and power of the system.

### *2.1. Microgrid Distributed Power Device Model*

1. Battery model

The battery model is an essential part of the microgrid and plays a role in mitigating the uncertainty of renewable energy and load in the microgrid operation to some extent. In this paper, the storage model is established by simulating the battery with a linear model and considering the charging and discharging power and the storage state SOC; the charge

state of the battery is related to the charge power in the previous period, and the charging and discharging expressions are as follows:

$$Soc_{t+1} = Soc_t - \Delta t(\frac{s_t^{dis} p_t^{dis}}{\eta_{dis} S_{es}} + \frac{s_t^{ch} \eta_{ch} P_t^{ch}}{S_{es}}) \tag{1}$$

where, $s_t^{ch}, s_t^{sh}$ is the energy storage charging and discharging state at time $t$; $P_t^{ch}, P_t^{dis}$ corresponds to the charging and discharging power, respectively; $\eta_{ch}, \eta_{sh}$ indicates the charging and discharging efficiency of the storage system; $S_{es}$ is the rated capacity of the energy storage device; $\Delta t$ is the interval time. The energy storage equipment charging and discharging satisfies the constraints.

$$Soc_t, P_t^{ch}, P_t^{sh} \geq 0 \tag{2}$$

$$Soc_{\min} \leq Soc_t \leq Soc_{\max} \tag{3}$$

$$P_t^{ch} \leq P_{\max}^{ch} \tag{4}$$

$$P_t^{dis} \leq P_{\max}^{dis} \tag{5}$$

where $P_{\max}^{ch}, P_{\max}^{dis}$ is the maximum charging and discharging power of the energy storage device for the period, and $Soc_{\max}, Soc_{\min}$ is the maximum and minimum value of the capacity of the energy storage device.

2. Gas turbine model

The gas turbine is one of the distributed generation components and belongs to the controllable generation unit of the microgrid. The role is to be able to provide an adjustable power supply to the microgrid by using a generator of traditional fossil fuel natural gas, effectively reducing the dependence of the microgrid on external sources. This paper uses a gas turbine unit model as an example of a power generation model whose fuel cost function is expressed as follows:

$$C_t^{MT} = a(P_t^{MT})^2 + bP_t^{MT} + c \tag{6}$$

where $a$, $b$, $c$ are cost factors; $C_t^{MT}$ denotes the cost of fuel; $P_t^{MT}$ denotes the output power of the gas turbine.

The constraint on the output power of the gas turbine is:

$$P_{\min}^{MT} \leq P_t^{MT} \leq P_{\max}^{MT} \tag{7}$$

$$P_{down}^{MT} \leq P_t^{MT} - P_{t-1}^{MT} \leq P_{up}^{MT} \tag{8}$$

where $P_{\min}^{MT}, P_{\max}^{MT}$ is the minimum and maximum power of the gas turbine, and $P_{down}^{MT}, P_{up}^{MT}$ is the climbing constraint.

3. Wind and solar power units

Wind and solar power units belong both are among the components of distributed generation units, which are renewable generation energy sources in the microgrid converted to electrical power generation through natural resources, and due to their uncontrollability the generation level is defined as $P_t^{res}$; assuming that renewable energy generation at time step $t$ moments is sampled from the probability distribution, $P_t^{P^{res}}$ defined as follows:

$$P_t^{res} \sim P_t^{P^{res}}(P_{t-1}^{res}, \ldots, P_{t-n}^{res}) \tag{9}$$

In this paper, the data of renewable generations are represented by real data of isolated microgrids, and the distribution $P_t^{P^{res}}$ is indexed by time $t$ to represent the variation of generation power.

4. Power balance

The bus of the microgrid needs to maintain power balance in each distributed power source access, including the full grid connection of renewable energy sources, controllable generations, and storage, and the system's constraints have been given above, from which a model can be established as follows:

$$P_t^{res} + P_t^{MT} + P_r^{ES} + P_t^{grid} = P_t^K \tag{10}$$

where $P_t^{grid}$ denotes the value of the power input to the microgrid from the larger grid; $P_t^{MT} P_t^{ES} P_t^{res}$ represent the power output of the gas turbine, the battery charging and discharging power, and the control of the renewable generations at moment $t$, respectively.

In the microgrid multi-agent system, the agents are designed for distributed control based on the neighbor information of the communication network, corresponding to the distributed power supply in the microgrid system; represented by the graph, G is defined as the ensemble {N, $\alpha$, A}, where N denotes a single agent, N = {1, 2 ... $n$}; $\alpha \subseteq N \times N$ denotes the set of edges; A is the adjacency matrix reacting to the degree of interaction of each node; the elements of A are defined as $A = (a_{ij})_{n \times n}$ the communication line from node $j$ to $i$.The interaction of information is generally expressed using the Laplace matrix $L = D - A$, where $D$ is the incoming degree matrix representing the received information about the neighboring agents; $L$ is defined as follows:.

$$L = \begin{cases} l_{ij} = \sum\limits_{i \neq j} a_{ij}, i = j \\ l_{ij} = -a_{ij}, i \neq j \end{cases} \tag{11}$$

*2.2. Droop Control of the Microgrid*

The primary microgrid control uses the droop control method to regulate the active and reactive power of the microgrid, using the output active and reactive power of the inverter in the simulated traditional power system to handle the frequency and voltage in the microgrid or to control the active and reactive power in response to the change of frequency and voltage in the system. Then, the power is reasonably distributed in each distributed power source. Generally, P-f and Q-V droop control are used in medium- and high-voltage power systems, and P-V and Q-f inverted droop control are used in low voltage power systems, and the nodal currents of the lines are schematically shown in Figure 2.

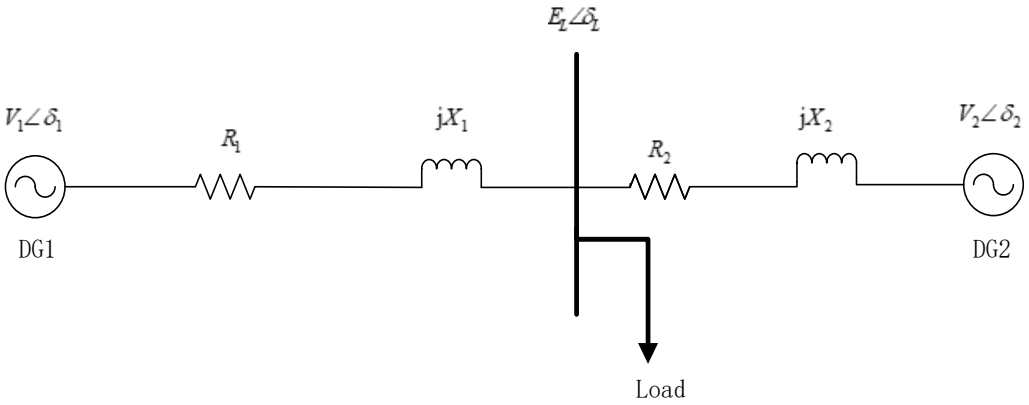



**Figure 2.** Schematic diagram of line node currents.

Where $V_i\angle\delta_i$ is the voltage ($i$ = 1, 2 ... ) of the inverter output of the distributed power supply, ($DG_i$), $E_L\angle\delta_L$ is the common side AC bus voltage; $R_i$ and $jX_i$ are the resistance

and inductance; the line impedance line is denoted as $Z_i\angle\theta = R_i + jX_i$. The resulting expressions for the active and reactive power of the inverter are as follows:

$$\begin{cases} P = \frac{V}{Z}[(V - E\cos\delta)\cos\theta + E\sin\delta\sin\theta] \\ Q = \frac{V}{Z}[(V - E\cos\delta)\cos\theta - E\sin\delta\sin\theta] \end{cases} \tag{12}$$

Normally the angle of power $\delta$ is minimal and defaults to $\cos\delta \approx 1, \sin\delta \approx \delta$ Bringing in the power equation gives:

$$\begin{cases} P = \frac{V}{Z}[(V - E)\cos\theta + E\delta\sin\theta] \\ Q = \frac{V}{Z}[(V - E)\sin\theta - E\delta\cos\theta] \end{cases} \tag{13}$$

It can be seen that the power of the inverter is related to the impedance angle $\theta$. When the line impedance is inductive, the active power $P$ is mainly related to the power angle $\delta$, and the reactive power $Q$ is primarily associated with the voltage drop $V - E$, and usually, $P$-$f$ and $Q$-$V$ droop control is used at this time; when the line impedance is resistive, the active power $P$ is mainly related to the voltage drop $V - E$, and the reactive power is primarily related to the power angle $\delta$, and usually $P$-$V$, and $Q$-$f$ droop control is used at this time.

Microgrids are usually low-voltage grids, and the output frequency and voltage of the distributed power inverter can be directly controlled by using $P$-$f$ and $Q$-$V$ droop control, which corresponds to the expression of the relationship.

$$\begin{cases} f = f_0 - \mathrm{m_i}(P - P^*) \\ V = V_0 - \mathrm{n_i}(Q - Q^*) \end{cases} \tag{14}$$

where $f_0$ is the rated frequency; $P^*$ is the rated active power; $\mathrm{m_i}$ is the active droop factor; $V_0$ is the voltage amplitude; $Q^*$ is the rated reactive power; $\mathrm{n_i}$ is the reactive droop factor. The resulting characteristic curve for droop control is shown in Figure 3.

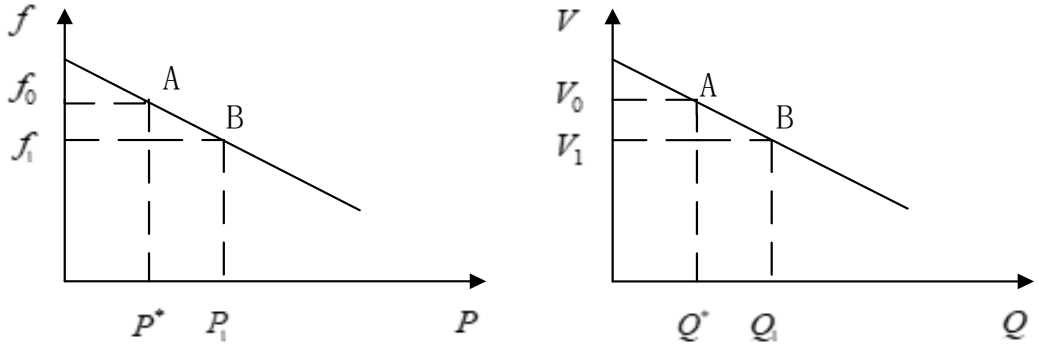



**Figure 3.** Droop control characteristic curve. A represents rated power and moves to point B when the system load increases.

The rated frequency $f_0$ of the system at point A during stable operation corresponds to the active power of the system as $p^*$ and the output voltage $V_0$ in the $V$-$Q$ curve reach the reactive power of $Q^*$. When the load in the microgrid system increases, the operating point of the system moves from A to B. The design remains stable when the frequency and voltage decrease and the active power and reactive power increase; thus, droop control is used to achieve the power balance of the microgrid system.

However, the traditional droop control will affect the frequency and voltage stability of the system to produce deviation; the allowable range of frequency fluctuation of our power quality standard is $\pm$2%, and the content of voltage fluctuation is $\pm$5%. We thus propose a secondary control algorithm to reduce the system frequency and voltage error so that the micro-grid system can operate more stably.

## 3. Secondary Control Based on PES-ACNN Model Framework

In order to cope with the frequency deviation problem of conventional microgrid systems, an improved deep reinforcement learning algorithm based on the PES_ACNN model for the multi-agent system is proposed to train the control parameter data of the microgrid sub-frequency system optimally based on the frequency deviation of the microgrid, which is received by the agent at certain intervals $\Delta f$ as the optimization target of secondary control.

In the multi-agent systems of microgrids, due to each agent model system having an independent Actor-Critic neural network structure, all data are processed by feature extraction of the neural network as input, and the error between the predicted value and the target value in the model is updated iteratively by the strategy gradient descent and finally as output after passing the activation function. To make efficient use of experience during the network model training, a preferred experience pooling strategy is introduced to improve the speed of convergence of the algorithm to ensure a timely solution to energy optimization problems in microgrids. The model training flow chart is shown in Figure 4.

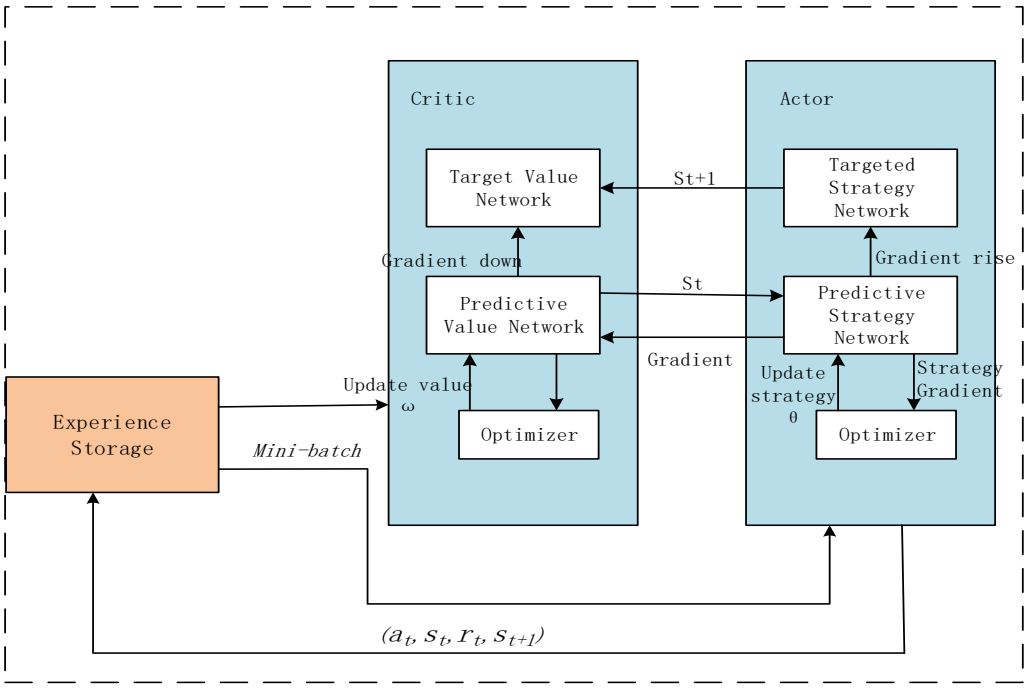



**Figure 4.** PES-ACNN neural network model training flowchart.

### *3.1. Deep Reinforcement Learning Algorithms*

The reinforcement learning algorithm is based on the Q-Learning algorithm, whose main framework consists of the agent and the environment. The core of its algorithm in the microgrid system is the interaction between the distributed power source and the grid, which is continuously updated with strategies to achieve optimal control.

The distributed agent of the microgrid takes different actions with the environment through its state for the feedback, thus choosing to adjust the following action to change its state process. This process can be seen as the Markov decision process, using $Q$ table to store the system state and action corresponding to the value function $Q(s, a)$. According to the Bellman equation as follows:

$$G_t = Q(s,a) = E[R_t | s_t = s, a_t = a] = E[r_t + \gamma Q(s_{t+1}, a_{t+1}) + \gamma^2(s_{t+2}, a_{t+2}) + \ldots \gamma^n(s_{t+n}, a_{t+n})] \quad (15)$$

In this training process, a tuple of $Q$-valued training model devices ($s_t$, $a_t$, $r_t$, $s_{t+1}$) is built as samples for training using the Markov decision process; $s_t$ is the current state; $a_t$ is

the present action; $r_t$ is the immediate reward after performing the action; $s_{t+1}$ is the next state; $t$ is the moment; the $Q$ function recursive update strategy is:

$$Q_{t+1}(s_t, a_t) \leftarrow Q_t(s_t, a_t) + \alpha(r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)) \quad (16)$$

where $\alpha$ is the learning rate, and $\gamma$ is the discount factor.

Since the state and action of the $Q$ function in the reinforcement learning algorithm have a high-dimensional complexity, to solve this problem, a neural network $Q(s, a; \omega)$ can be introduced as a function approximator to estimate the $Q(s, a)$ function, denoted as a deep $Q$ network DQN and compared to Q-Learning, which approximates the input function of the state and action of the $Q$ function as the $Q$ of the action obtained after the analysis of the neural network values and selects the maximum $Q$ value as the following action, as shown in Figure 5.

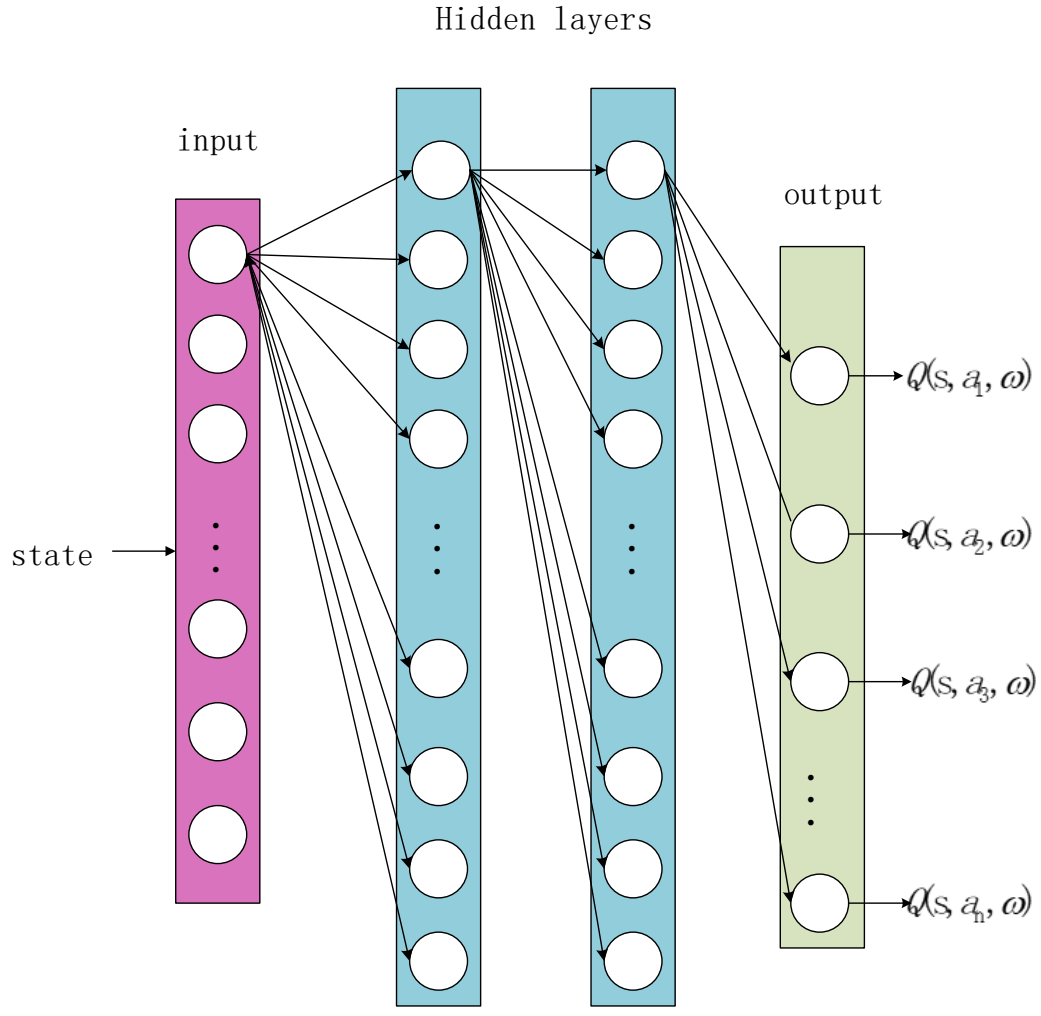



**Figure 5.** Neural network value function approximation.

The weight $\omega$ in a deep neural network represents the mapping of the system state to the $Q$ value, which solves the problem of the state space being continuous, so a loss function $L_i(\omega)$ needs to be defined to update the neural network weights $\omega$ with the corresponding $Q$ values. The loss function is the difference between the objective function and the prediction function.

$$y_t = E_s[r_{t+1} + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \omega_t)] \tag{17}$$

$$L_i(\omega_t) = E_s[(y_t - Q(s, a; \omega_t))^2] \tag{18}$$

Update the weights of the agents by finding the gradient of the loss function and performing stochastic gradient descent:

$$\nabla_{\omega_t} L_i(\omega_t) = E_s[(r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \omega_t{}^-) - Q(s_t, a_t; \omega_t)) \nabla_{\omega_t} Q(s_t, a_t; \omega_t)] \tag{19}$$

The secondary frequency control of the microgrid is used to reduce the frequency deviation by optimizing the generation of distributed power sources. The frequency deviation of the microgrid is discretized through the proposed model framework defined as $(\Delta f_1, \Delta f_2, \Delta f_3, \ldots \Delta f_n)$, which corresponds to an ambient state of $S(s_1, s_2, s_3 \ldots s_n)$. The value of the environment state interval affects the convergence speed and accuracy of the controller; the space defined too densely affects the convergence speed leading to reduction and too sparsely affects the accuracy difficulty of achieving the required goal. The frequency regulation range of the power system is $50 \pm 0.2$ hz; in order to make the system more accurate, set the control target of the controller as $50 \pm 0.05$ hz. Therefore, the state space S of this model is defined as $\{(-\infty, -0.05), [-0.05, -0.01), [-0.01, -0.005), [-0.005, 0.005), [0.005, 0.01), [0.01, 0.05), [0.05, +\infty)\}$. In the microgrid, the S defined control actions based on the state distribution shall be discrete controller regulation commands, including actual power load, output power, and charge; discharge power, defining A as $\{-0.1, -0.5, -0.01, -0.05, 0, 0.05, 0.01, 0.5, 0.1\}$ Define the reward function as follows:

$$r_i(s, a) \begin{cases} 0 & |\Delta f| \le 0.05 \\ -\mu_1 \cdot |\Delta f| & 0.005 < |\Delta f| \le 0.01 \\ -\mu_2 \cdot |\Delta f| & 0.01 < |\Delta f| \le 0.05 \\ -\mu_3 \cdot |\Delta f| & 0.05 < |\Delta f| < 0.1 \\ -\mu_4 \cdot |\Delta f| & |\Delta f| \ge 0.1 \end{cases} \tag{20}$$

where the regulation dead zone is considered as $|\Delta f| \le 0.005$ when the intelligence receives zero rewards and the remaining intervals receive the corresponding negative reward value; $-\mu_1 \sim -\mu_4$ is the reward factor set to 5, 10, 15, and 20. The setting of the reward function is the key to the control of the controller, and the control target can be clarified by the reward function.

The traditional DQN model can enormously impact the agent systems due to accuracy issues. For the deviation of predicted and actual values, we propose the PES-ACNN model to reduce this error.

### *3.2. Actor-Critic Neural Network*

The Actor-Critic Neural Network is a neural network approach based on the combination of value network function and policy network function with complex mapping ability to handle unstructured and inaccurate data adaptively. The Actor network makes a probability-based selection of behavior; the Critic network scores based on the Actor's behavioral algorithm, and the Actor network then modifies its own behavior selection based on the score of the Critic network. The state value function is expressed as follows:

$$V(s; \theta, \omega) = \sum_a \pi(a|s; \theta) \cdot q(s, a; \omega) \tag{21}$$

where $\pi(a|s; \theta)$ denotes the strategy network; $\theta$ is the strategy network parameter, and the purpose is to learn and optimize the strategy to make the strategy perform better; $q(s, a; \omega)$ is the value network; $\omega$ is the value network parameter, and the purpose is to learn and evaluate the value function to make the value function evaluation more accurate.

Critic networks can be considered reinforcement learning algorithms based on value network functions, the core of which is to evaluate improvement strategies using value functions. The core of the dynamic programming-based reinforcement learning algorithm is the bootstrap algorithm, which inevitably generates overestimation problems, and the model has uncertainty; based on the Monte Carlo approximation method, due to probability, the impact problem leads to tension, so the value function uses time difference algorithm to make the intelligence constantly interact with the environment to update and use the current value function to update the strategy, according to the Bellman equation state value function formula expressed as follows:

$$\begin{aligned} V_\pi(s_t) &= E_{At}[Q_\pi(s_t, A_t)] \\ &= E_{At}[E_{st+1} + [R_t + \gamma V_\pi(s_{t+1})]] \\ &= E_{A_t, s_{t+1}}[R_t + \gamma V_\pi(s_{t+1})] \end{aligned} \tag{22}$$

The iterative update of the neural network using the Monte Carlo approximation is represented as follows:

$$\begin{aligned} V(s_{t+1}) &= v(s_t) + \alpha[R_t + \gamma v(s_{t+1}) - v(s_t)] \\ &= v(s_t) + \alpha\delta_t \end{aligned} \tag{23}$$

where, $\alpha$ is the hyperparameter learning rate, and $\delta_t$ is the error.

To make the error of the value network function less, updating the neural network parameters using gradient descent can be expressed as follows:

$$\omega \leftarrow \omega - \alpha \cdot \delta_t \cdot \frac{\partial v(s_t; \omega)}{\partial \omega} \tag{24}$$

The Actor network can be considered a reinforcement learning method based on policy network functions, which utilizes a scoring of the Critic network value function learning evaluation to select actions. The policy update network is stochastic; the action space can be continuous, and the reinforcement learning update method based on the policy gradient function utilizes parameterizing the function. Thus, a gradient ascent is used to update the parameters until convergence corresponds to the optimal policy. The policy gradient-based reinforcement learning objective function is expressed as follows:

$$\max J(\theta) = E_\pi[G_t | S_t = s_0] = v_\pi(s_0) \tag{25}$$

The approximate policy gradient is used in the formula to update the policy network, and the policy gradient WITH BASELINE is expressed as follows:

$$\frac{\partial V_\pi(s_t)}{\partial \theta} = E_{A_t \sim \pi}[\frac{\partial \ln \pi(A_t | s_t; \theta)}{\partial \theta} \cdot (Q_\pi(s_t, A_t) - V_\pi(s_t))] \tag{26}$$

where $\theta$ is the neural network parameters; $V_\pi(s_t)$ is the baseline, which does not affect the expectation but can reduce the variance to accelerate convergence. Calculating the expectation in the strategy gradient for random sampling of the strategy gradient using the Monte Carlo approximation for unbiased estimation of the strategy gradient is as follows:

$$\begin{aligned} g(a_t) &= \frac{\partial \ln \pi(a_t | s_t; \theta)}{\partial \theta}(Q_\pi(s_t, a_t) - V_\pi(s_t))] \\ &= \frac{\partial \ln \pi(a_t | s_t; \theta)}{\partial \theta}(R_t + \gamma \cdot V_\pi(s_{t+1}; \omega) - V_\pi(s_t; \omega)) \\ &= \frac{\partial \ln \pi(a_t | s_t; \theta)}{\partial \theta}\delta_t \end{aligned} \tag{27}$$

The strategy gradient up update network parameter $\theta$ is as follows:

$$\theta \leftarrow \theta + \beta \cdot \delta_t \frac{\partial \ln \pi(a_t | s_t; \theta)}{\partial \theta}. \tag{28}$$

where $\beta$ is the hyperparameter learning rate, and $\delta_t$ is the error.

The Actor-Critic based reinforcement learning method is combined with the integration idea of the value function and policy function, using the policy network to learn the optimization strategy to choose a better policy method and using the value network to output the value function and learning the evaluation to make the evaluation of the value function more accurate. It is an excellent solution to the reinforcement learning framework to deal with continuous state space problems and also to deal with continuous action space problems with good stability and to realize an incremental iterative online reinforcement learning strategy.

However, this method also brings relative drawbacks; the training experience data extracted by the agents in the microgrid system based on the state features are generated once per iteration in the Actor-Critic network, which leads to low utilization of experience and slow convergence rate to meet the requirements. To make the samples more efficient in solving the instability problem, the offline strategy of DQN is invoked, using the strategy of priority experience storage to reuse the training experience and disrupting the order of experience to reduce the correlation work of the samples.

### *3.3. Priority Experience Storage Policy*

In the framework of the Actor-Critic algorithm, the performance of the agents can be substantially improved by using experience replay. The experience data ($s_t$, $a_t$, $r_t$, $s_{t+1}$) obtained by the distributed power agents in the microgrid interacting with the environment are stored in a portfolio. These experiences replay arrays are repeatedly used afterwards to train the neural network for updating, and since the size of the experience pool is fixed, noted as $b$, only $b$ data can be retained in the array, and as the training proceeds, the newly acquired experience overwrites the old experience until the new experience gradually reaches a consistent representation of the algorithm convergence of the algorithm. Usually, the size of the array $b$ is a hyperparameter that needs to be tuned, which affects the training results; $b$ is usually $10^5$~$10^6$.

In practice, it is necessary to wait for enough experience with playback data before the neural network starts to update. The probability of the intelligence receiving a high reward experience at the early stage of exploration is much lower than that of a low reward experience. Due to the exploratory influence of the algorithm, it is difficult for intelligence to reuse the high reward experience even if it explores it and explores other experiences, leading to unsatisfactory training results and fluctuations and resulting in reduced efficiency.

Priority experience replay is a special method of experience replay where weight is added to the quaternion of experience replay. Then non-uniform random sampling is conducted based on the weights. The weight of the data is noted as the absolute value of the error $|\delta_t|$, and the sampling probability is as follows:

$$p_t \propto |\delta_t| + \varepsilon \tag{29}$$

where $\varepsilon$ is a minimal value to prevent the sampling probability from approaching zero and is used to ensure that all samples are sampled with a non-zero probability. Samples with significant errors have a high possibility of being sampled, and the sampling probability is used in the non-uniform sampling method to adjust the learning rate $\alpha$, setting the learning rate to:

$$\alpha_t = \frac{\alpha}{(b \cdot p_t)^{\tau}} \tag{30}$$

where $b$ is the total number of samples played back empirically; $\tau \in (0, 1)$ is a hyperparameter.

The Priority Experience Storage Actor-Critic Neural Network Algorithm 1 is described below.

**Algorithm 1:** Priority Experience Storage Actor-Critic Algorithm

Initialize network parameters $\theta$ randomly
Initialize training memory with capacity $Tb_{\max}$
Initialize agent memory
For episode = 1,Max_episodes do:
    For t = 0, D-1 do:
Get initial state $S_t$
Take action a with $\varepsilon - greedy$ policy-based
On $\pi(s_t, \theta)$
Receive new state $S_{t+1}$ and reward $R_t$
Store transition $(s_t, a_t, r_t, s_{t+1})$ in agent memory
    **End for**
    Calculate $P_t$ for each transition in the agent memory
    Store $(s_t, a_t, P_t)$ in the training memory.
    Reset agent memory
    Reset environment
    **If** training memory is full:
Calculate $Q_\pi(S_t, A_t) \ln \pi(a_t|s_t;\theta)\, L(\theta)$ for the whole batch
Perform a gradient descent step on $\theta$
Reset training memory
**End for**

## 4. Experimental Validation and Analysis

### *4.1. Model Simulation*

In this section, an islanded microgrid model consisting of four inverter-based distributed power sources with communication topology is constructed by numerical simulation using the Simulink toolbox of MATLAB2020b software. This microgrid model is 220 V/50 Hz to verify the power distribution and frequency and voltage regulation of each distributed power source based on the deep reinforcement learning secondary control strategy proposed in this paper validity is shown in Figure 6. The circuit parameters of the microgrid are shown in Table 1.

**Table 1.** Circuit parameters of the microgrid.

| | **DG1** | **DG2** | **DG3** | **DG4** |
|---|---|---|---|---|
| DGs | C1/F $15 \times 10^{-5}$<br>L1/mH 0.2<br>R1/Ω 0.01 | C2/F $15 \times 10^{-5}$<br>L2/mH 0.2<br>R2/Ω 0.01 | C3/F $15 \times 10^{-5}$<br>L3/mH 0.2<br>R3/Ω 0.01 | C4/F $15 \times 10^{-5}$<br>L4/mH 0.2<br>R4/Ω 0.01 |
| load | Load1<br>P1/kW 90<br>Q1/kVar 30 | Load2<br>P2/kW 90<br>Q2/kVar 20 | Load3<br>P3 /kW 60<br>Q3 /kVar 30 | Load4<br>P4/kW 50<br>Q4/kVar 20 |
| routes | Line1<br>R1/Ω 0.18<br>L1/mH 0.302 | Line2<br>R2/Ω 0.21<br>L2/mH 0.302 | | Line3<br>R3/Ω 0.20<br>L3/mH 0.82 |

In the four distributed power supplies DG1~DG4, each distributed power supply acts as an agent that can transfer information with neighboring agents to complete distributed coordination control. In contrast, the communication topology between each agent is connected and contains a minimal directed spanning tree so that the agent system can achieve consistent convergence.

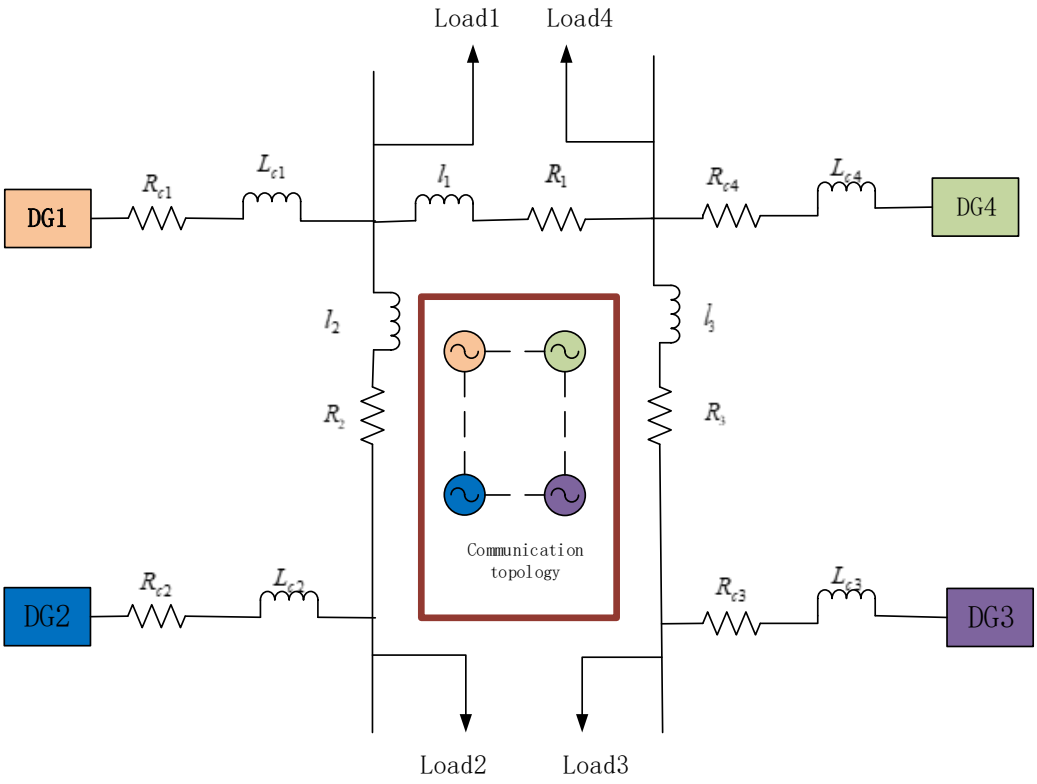



**Figure 6.** Islanded microgrid with communication topology.

### *4.2. Neural Network Model Validations*

Before the input data of the proposed secondary frequency control model for microgrid, offline numerical simulation is required to determine the appropriate number of hidden neurons for the structure of the Actor policy model and Critic value network model. The neural network contains an input, hidden layer, and output, and the layers are connected by weights, thresholds, and activation functions.

The number of neurons affects the accuracy and speed of the neural network. More neurons mean that the neural network can fit the input–output relationship more accurately, but it also takes more time. Taking the Actor action neural network model as an example to verify the influence of the number of neurons on the algorithm and choosing the input as the state feature and the output as the action distribution, we set the number of neurons in the hidden layer as 1–10; the influence of the number of neurons on the convergence is shown in Figure 7; when the number of neurons is less than four the output of the neural network is difficult to achieve convergence; when the number of neurons is greater than four can converge, in order to prevent the excessive number of neural network neurons leads to overfitting affecting the convergence speed, the change of neural network structure can be determined. Take the neural network of one of the distributed power DG1 as an example. The number of neurons n on the neural network is represented in Figure 8. Under the same data set built in the neural network, the greater the number of neurons, the better the neural network recognition effect is; when the number of neurons $n = 4$, the control input–output relationship of the algorithm can be accurately expressed convergence when the number of neurons continues to increase the neural network will reach the overfitting. When the number of neurons continues to increase, the neural network will reach the overfitting state and expand the network computation time.

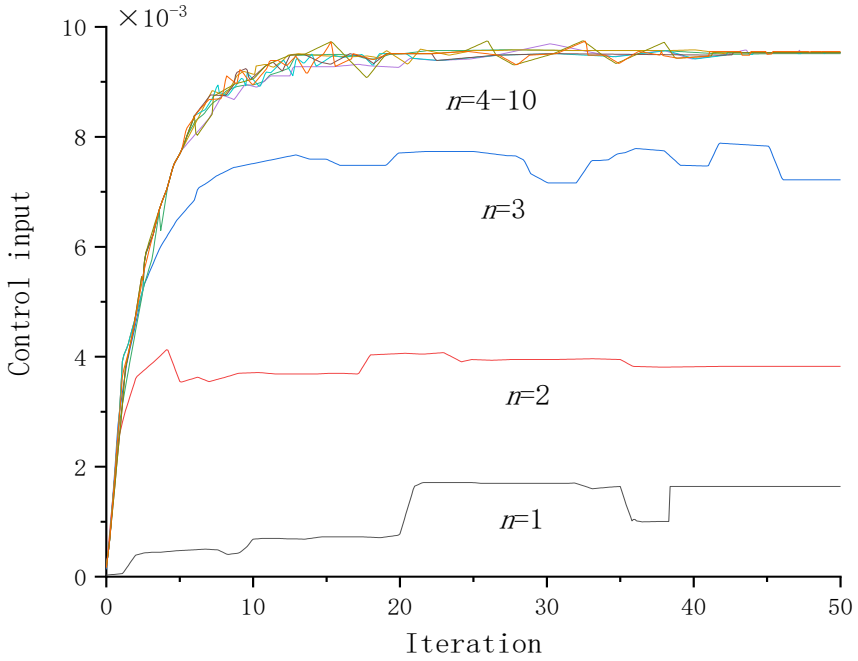



**Figure 7.** Effect of the number of neurons on convergence.

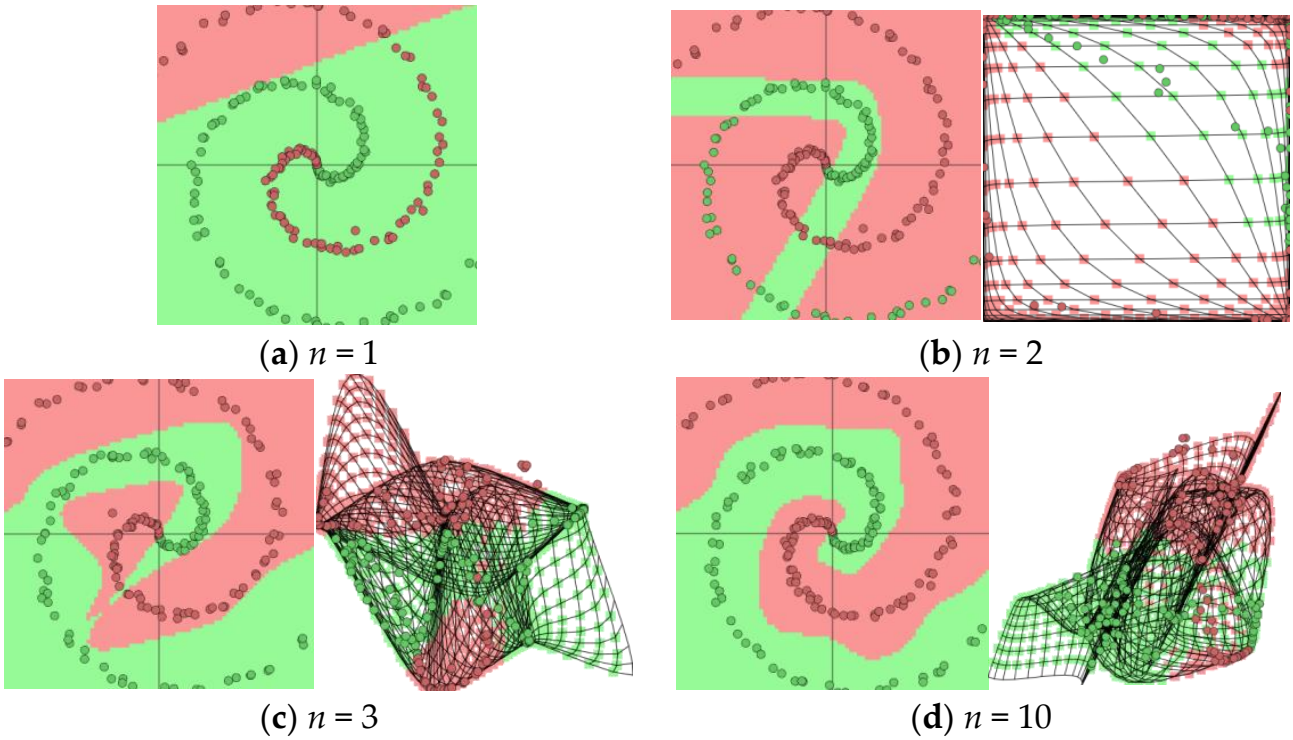
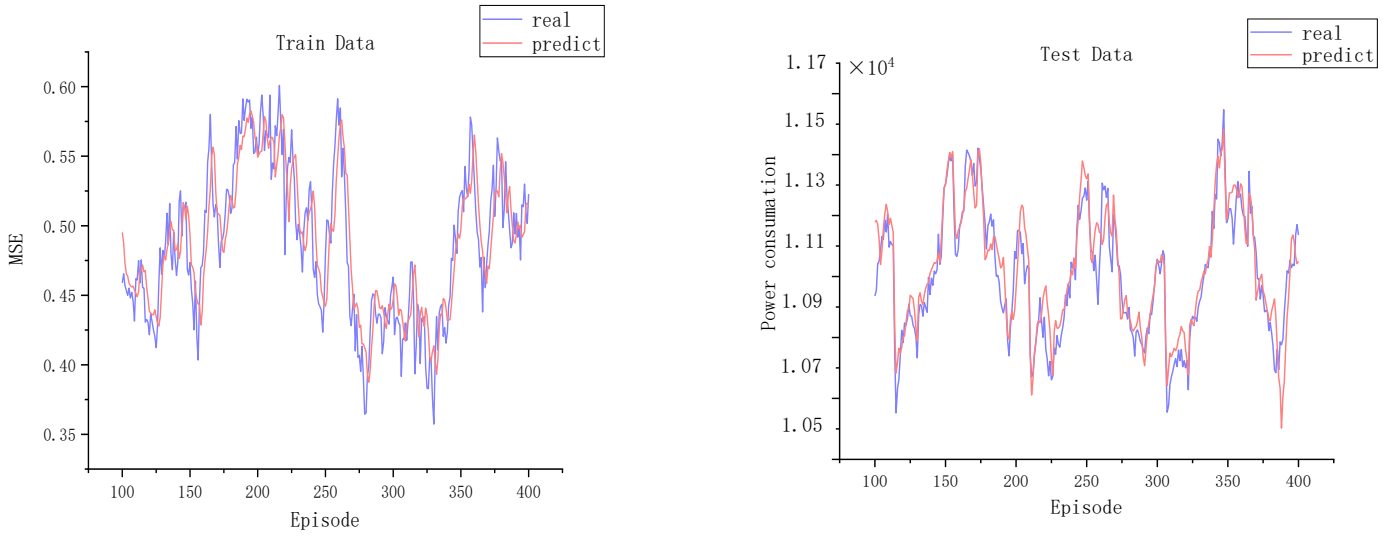



**Figure 8.** The effect of the number of neurons on the neural network.

In order to verify the influence of the neural network on the microgrid, the number of neurons $n$ = 4 was selected to predict the load of grid. Power load projections were made using city electricity data from 1 January 2018 to 31 December 2018 for a city.

The first 361 data points were used as the training model for data input, and the last four data points were used as the test model to obtain the final experimental results compared with the real data of the original power load. Changes in data projections are shown in Figure 9.




(**a**) (**b**)

**Figure 9.** Neural network load prediction (**a**) power load training data error for a city in 2018 (**b**) power load prediction for a city in 2018.

The training results of the traditional control method combined with the PES-AC neural network algorithm for the multi-agent system are shown in Figure 10. It can be seen from the figure that as the training proceeds, the reward experience obtained by the neural network rises continuously, and the reward obtained by the algorithm can be stabilized in the corresponding interval at about 1000 rounds of training; adding priority experience replay can make the neural network preferentially use the experience pool with higher reward experience during training, reduce the training fluctuation, and accelerate the convergence speed to achieve satisfactory results, as shown in Figure 11.

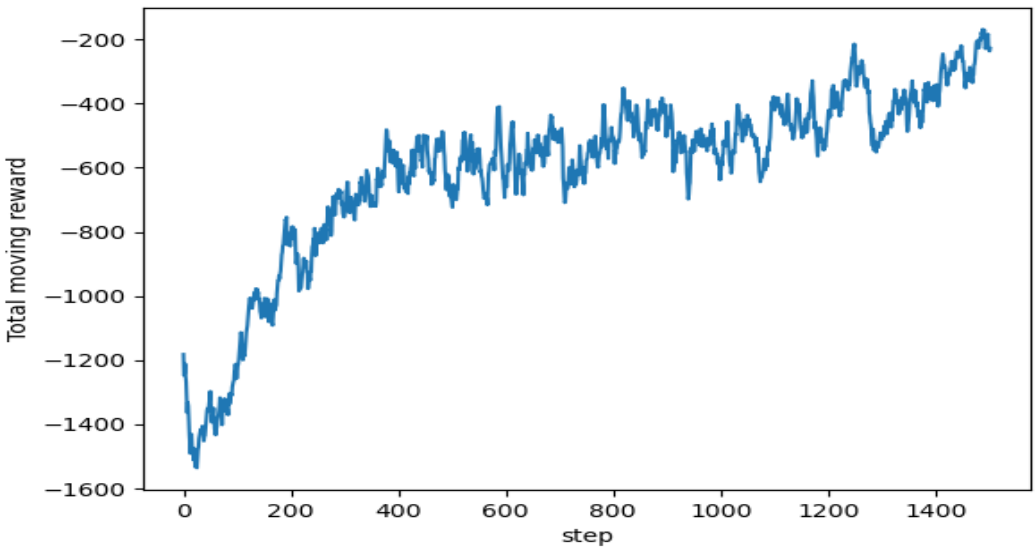



**Figure 10.** Training neural networks to achieve empirical returns. Represents the increasing reward value.

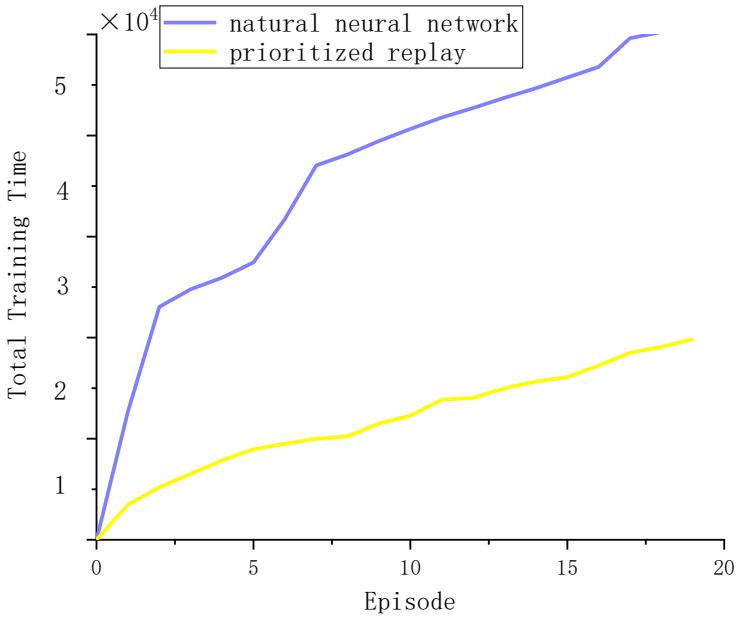



**Figure 11.** Comparison of neural network incorporation priority experience playback results.

### *4.3. Distributed Secondary Frequency Control Simulation*

The experiments in this section are conducted to verify the effectiveness of the role of the proposed secondary frequency control. The simulation of the conventional control method is performed in the constructed microgrid model, and the simulation results of the proposed secondary regulation control in this paper are compared with the same parameters throughout the simulation. At the initial moment, the microgrid test system works the load of each node is matched with the nominal frequency of the generation unit.

The microgrid system can achieve proportional active power distribution under decentralized primary control with droop control, but the frequency of the system decreases below the rated frequency and finally stabilizes as shown in Figure 12.

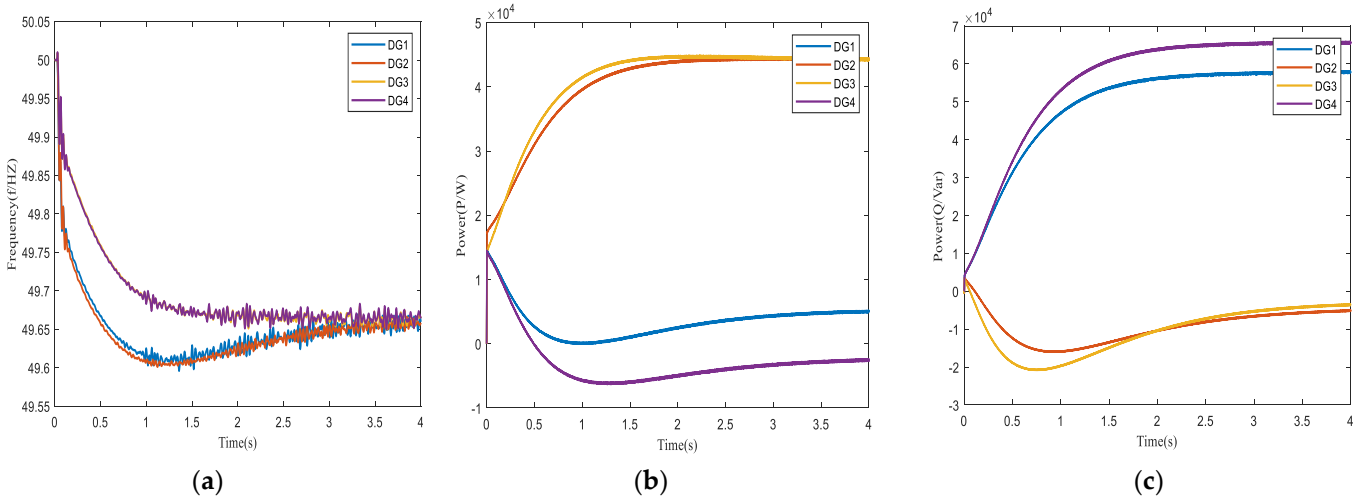



**Figure 12.** Parameter results of the microgrid system under primary droop control (**a**) Frequency variation of the system (**b**) Variation of active power of the system (**c**) Variation of reactive power of the system.

When secondary frequency regulation is used, the system ensures power distribution under the same conditions while allowing the system to also operate at a frequency that quickly recovers to the microgrid rated frequency permitted error, as shown in Figure 13.

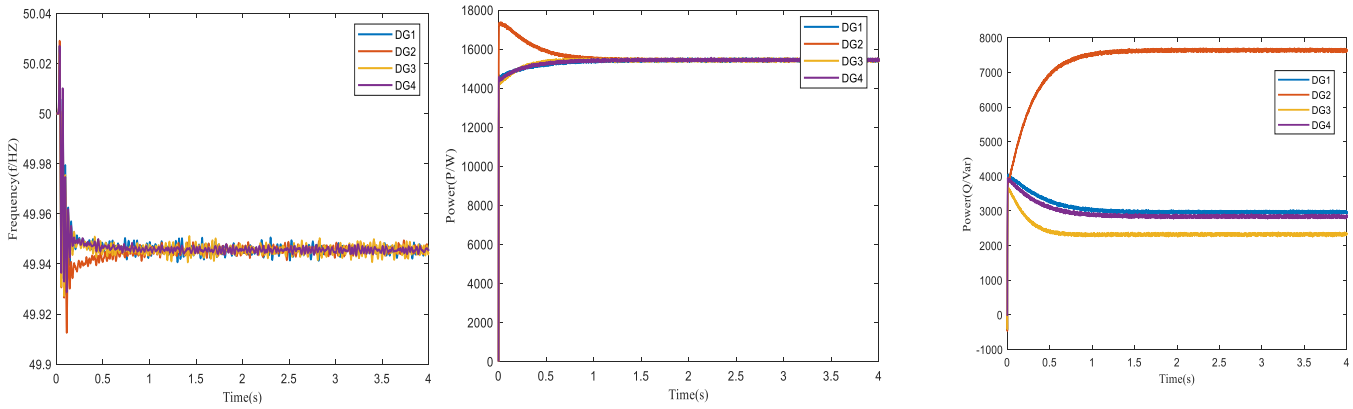



**Figure 13.** Parameter Results of Secondary Control System Used in Microgrid System (**a**) Frequency variation of the system (**b**) Variation of active power of the system (**c**) Variation of reactive power of the system.

To verify the robustness of the microgrid, both the activity and reactivity of the system are increased to match the load variation when the load increases under the same conditions, assuming that the dynamic response of the load can be accessed at the microgrid test system with stable operation at the rated frequency.

In the traditional microgrid control method, the load connected to 20 kw at t = 0.4 s shows that the frequency of the distributed power supply in the system is stable at 49.55 hz, and the load connected to 25 kw again at t = 0.6 s, and the system frequency eventually stabilizes at 49.6 hz, with a large deviation from the rated frequency shown in Figure 14. After using the improved dual neural network algorithm, the microgrid system is connected to 20 kw load at t = 0.4 s, and it can be seen from the figure that the system resumes stable operation at t = 3 s, and the frequency returns to within the allowable deviation from the rated frequency, and the active and reactive power simulation results of the system are shown in Figure 15. The results show that the microgrid system has plug-and-play characteristics and is robust using a secondary frequency control strategy.

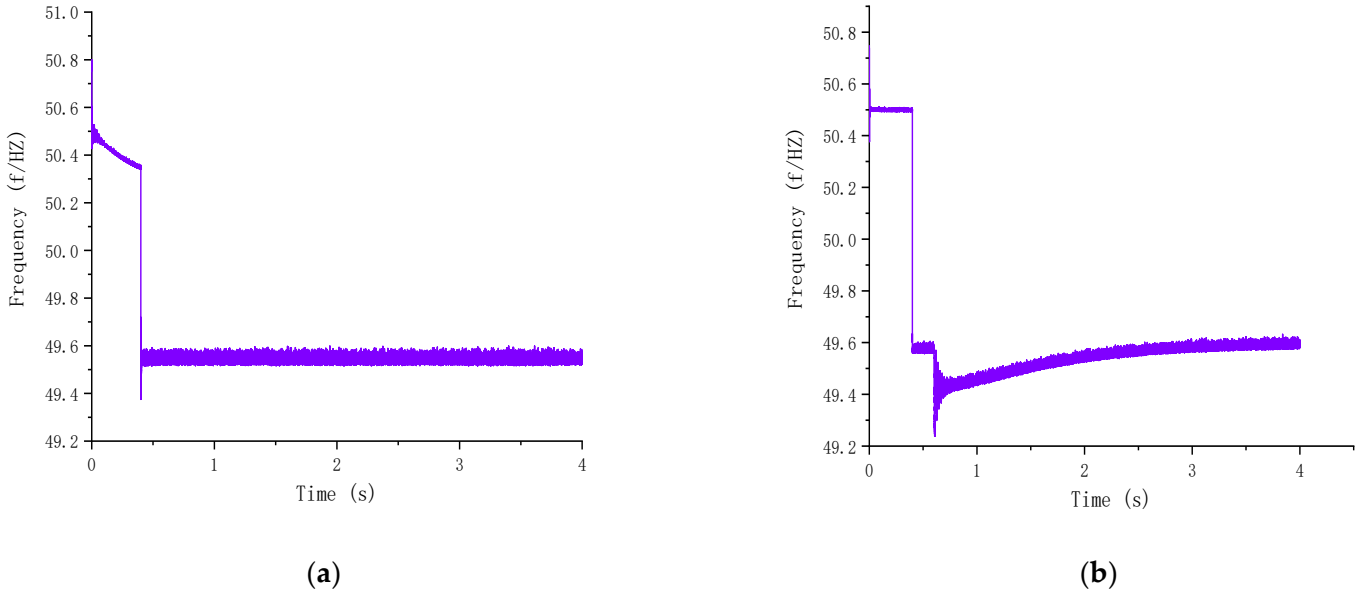



**Figure 14.** Parameter results of microgrid system under primary droop control (**a**) Frequency of access load 20 kw (**b**) Frequency of access load 25 kw.

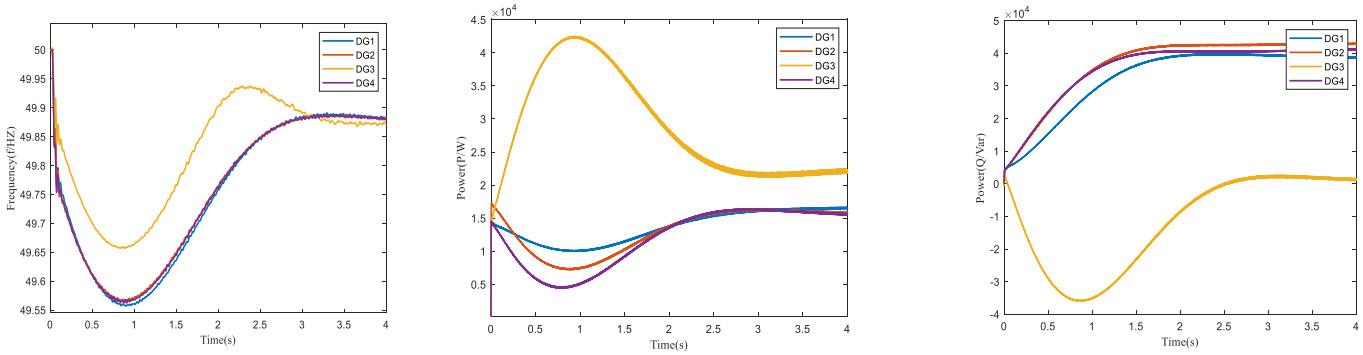



**Figure 15.** Variation of microgrid secondary control access load parameters (**a**) System access load frequency variation (**b**) Variation of active power of system access load (**c**) Variation in reactive power of system access loads.

## 5. Conclusions

This paper proposes a multi-agent deep reinforcement learning microgrid control method based on the PES-ACNN model. Compared with the traditional algorithm introducing a priority experience storage strategy, this method converts online reinforcement learning to offline learning. It improves the convergence speed and stability of the Actor-Critic neural network training process by accumulating experience through a large number of offline learning. The improvement of the algorithm reduces the frequency deviation of the distributed power supply in the isolated microgrid under droop control and reduces the frequency error of the distributed power supply in the microgrid multi-agent system, and each agent solves the power distribution problem by exchanging information with neighboring agent bodies, combines the economic environment and other benefit objectives to achieve the optimal control action of the microgrid, and accelerates the rapid regulation capability of the microgrid under external disturbances.

**Author Contributions:** Conceptualization, F.X.; Methodology, F.X.; Software, S.T.; Validation, S.T.; Investigation, C.L. and X.D.; Resources, X.D.; Data curation, C.L.; Writing—review & editing, F.X.; Visualization, S.T.; Project administration, F.X. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by The Science and Technology Project of State Grid Heilongjiang Electric Power Co., Ltd., grant number 522424220005, Heilongjiang Provincial Natural Science Foundation of China, grant number LH2021F057, and Heilongjiang Provincial institutions of higher learning basic scientific research funds scientific research project 135509803.

**Data Availability Statement:** The data that support the findings of this study are available on request from the corresponding author, [S.T.], upon reasonable request.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Farrelly, M.A.; Tawfik, S. Engaging in Disruption: A Review of Emerging Microgrids in Victoria, Australia. *Renew. Sustain. Energy Rev.* **2020**, *117*, 109491. [CrossRef]
2. Gao, F.; Bozhko, S.; Asher, G.; Wheeler, P.; Patel, C. An Improved Voltage Compensation Approach in A Droop-Controlled DC Power System for the More Electric Aircraft. *IEEE Trans. Power Electron.* **2015**, *31*, 7369–7383. [CrossRef]
3. Aklilu, Y.T.; Ding, J. Survey on Blockchain for Smart Grid Management, Control, and Operation. *Energies* **2021**, *15*, 193. [CrossRef]
4. De Caro, F.; Andreotti, A.; Araneo, R.; Panella, M.; Rosato, A.; Vaccaro, A.; Villacci, D. A Review of the Enabling Methodologies for Knowledge Discovery from Smart Grids Data. *Energies* **2020**, *13*, 6579. [CrossRef]
5. Dou, C.-X.; Liu, B. Multi-Agent Based Hierarchical Hybrid Control for Smart Microgrid. *IEEE Trans. Smart Grid* **2013**, *4*, 771–778. [CrossRef]
6. Li, F.-D.; Wu, M.; He, Y.; Chen, X. Optimal Control in Microgrid Using Multi-Agent Reinforcement Learning. *ISA Trans.* **2012**, *51*, 743–751. [CrossRef]

7. Wang, Y.; Nguyen, T.L.; Xu, Y.; Li, Z.; Tran, Q.-T.; Caire, R. Cyber-Physical Design and Implementation of Distributed Event-Triggered Secondary Control in Islanded Microgrids. *IEEE Trans. Ind. Appl.* **2019**, *55*, 5631–5642. [CrossRef]
8. Al-Tameemi, Z.H.A.; Lie, T.T.; Foo, G.; Blaabjerg, F. Optimal Coordinated Control of DC Microgrid Based on Hybrid PSO–GWO Algorithm. *Electricity* **2022**, *3*, 346–364. [CrossRef]
9. Ziouani, I.; Boukhetala, D.; Darcherif, A.-M.; Amghar, B.; El Abbassi, I. Hierarchical Control for Flexible Microgrid Based on Three-Phase Voltage Source Inverters Operated in Parallel. *Int. J. Electr. Power Energy Syst.* **2018**, *95*, 188–201. [CrossRef]
10. Bidram, A.; Davoudi, A. Hierarchical Structure of Microgrids Control System. *IEEE Trans. Smart Grid* **2012**, *3*, 1963–1976. [CrossRef]
11. Guo, F.; Xu, Q.; Wen, C.; Wang, L.; Wang, P. Distributed Secondary Control for Power Allocation and Voltage Restoration in Islanded DC Microgrids. *IEEE Trans. Sustain. Energy* **2018**, *9*, 1857–1869. [CrossRef]
12. Ning, C.; You, F. Data-Driven Stochastic Robust Optimization: General Computational Framework and Algorithm Leveraging Machine Learning for Optimization under Uncertainty in the Big Data Era. *Comput. Chem. Eng.* **2018**, *111*, 115–133. [CrossRef]
13. Dou, C.; Li, Y.; Yue, D.; Zhang, Z.; Zhang, B. Distributed Cooperative Control Method Based on Network Topology Optimisation in Microgrid Cluster. *IET Renew. Power Gener.* **2020**, *14*, 939–947. [CrossRef]
14. Lu, R.; Hong, S.H.; Yu, M. Demand Response for Home Energy Management Using Reinforcement Learning and Artificial Neural Network. *IEEE Trans. Smart Grid* **2019**, *10*, 6629–6639. [CrossRef]
15. Du, Y.; Li, F. Intelligent Multi-Microgrid Energy Management Based on Deep Neural Network and Model-Free Reinforcement Learning. *IEEE Trans. Smart Grid* **2020**, *11*, 1066–1076. [CrossRef]
16. Fang, X.; Zhao, Q.; Wang, J.; Han, Y.; Li, Y. Multi-Agent Deep Reinforcement Learning for Distributed Energy Management and Strategy Optimization of Microgrid Market. *Sustain. Cities Soc.* **2021**, *74*, 103163. [CrossRef]
17. Das, S.R.; Ray, P.K.; Sahoo, A.K.; Singh, K.K.; Dhiman, G.; Singh, A. Artificial Intelligence Based Grid Connected Inverters for Power Quality Improvement in Smart Grid Applications. *Comput. Electr. Eng.* **2021**, *93*, 107208. [CrossRef]
18. Liu, Y.; Zhang, D.; Gooi, H.B. Optimization Strategy Based on Deep Reinforcement Learning for Home Energy Management. *CSEE J. Power Energy Syst.* **2020**, *6*, 572–582. [CrossRef]
19. Lei, L.; Tan, Y.; Dahlenburg, G.; Xiang, W.; Zheng, K. Dynamic Energy Dispatch Based on Deep Reinforcement Learning in IoT-Driven Smart Isolated Microgrids. *IEEE Internet Things J.* **2021**, *8*, 7938–7953. [CrossRef]
20. Li, H.; Wan, Z.; He, H. Real-Time Residential Demand Response. *IEEE Trans. Smart Grid* **2020**, *11*, 4144–4154. [CrossRef]
21. Wang, J.; Kurth-Nelson, Z.; Tirumala, D.; Soyer, H.; Leibo, J.; Munos, R.; Blundell, C.; Kumaran, D.; Botvinick, M. Learning to Reinforcement Learn. *arXiv* **2016**, arXiv:1611.05763.
22. Mnih, V.; Badia, A.P.; Mirza, M.; Graves, A.; Harley, T.; Lillicrap, T.P.; Silver, D.; Kavukcuoglu, K. Asynchronous Methods for Deep Reinforcement Learning. *Int. Conf. Mach. Learn.* **2016**, *48*, 1928–1937.
23. Wang, Z.; Bapst, V.; Mnih, V.; Munos, R.; de Freitas, N.; Heess, N.; Kavukcuoglu, K. Sample efficient actor-critic with experience replay. *arXiv* **2017**, arXiv:1611.01224.
24. Van Hasselt, H.; Guez, A.; Silver, D. Deep Reinforcement Learning with Double Q-Learning. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-16), Phoenix, AR, USA, 12–17 February 2016; Volume 30. [CrossRef]
25. Phan, T.V.; Nguyen, T.G.; Bauschert, T. DeepMatch: Fine-Grained Traffic Flow Measurement in SDN with Deep Dueling Neural Networks. *IEEE J. Select. Areas Commun.* **2021**, *39*, 2056–2075. [CrossRef]
26. Bizzarri, F.; del Giudice, D.; Linaro, D.; Brambilla, A. Partitioning-Based Unified Power Flow Algorithm for Mixed MTDC/AC Power Systems. *IEEE Trans. Power Syst.* **2021**, *36*, 3406–3415. [CrossRef]
27. Xia, Y.; Xu, Y.; Wang, Y.; Dasgupta, S. A Distributed Control in Islanded DC Microgrid Based on Multi-Agent Deep Reinforcement Learning. In Proceedings of the IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, Singapore, 18–21 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 2359–2363.
28. Liu, W.; Zhuang, P.; Liang, H.; Peng, J.; Huang, Z. Distributed Economic Dispatch in Microgrids Based on Cooperative Reinforcement Learning. *IEEE Trans. Neural Netw. Learning Syst.* **2018**, *29*, 2192–2203. [CrossRef]