applied sciences

MDPI

Article

Multi-Microgrid Energy Management Strategy Based on Multi-Agent Deep Reinforcement Learning with Prioritized Experience Replay

Guodong Guo * and Yanfeng Gong

State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources, North China Electric Power University, Changping District, Beijing 102206, China
* Correspondence: gbjdsf@163.com; Tel.: +86-183-22595221

Abstract: The multi-microgrid (MMG) system has attracted more and more attention due to its low carbon emissions and flexibility. This paper proposes a multi-agent reinforcement learning algorithm for real-time energy management of an MMG. In this problem, the MMG is connected to a distribution network (DN). The distribution network operator (DSO) and each microgrid (MG) are modeled as autonomous agents. Each agent makes decisions to suit its interests based on local information. The decision-making problem of multiple agents is modeled as a Markov game and solved by the prioritized multi-agent deep deterministic policy gradient (PMADDPG), where only local observation is required for each agent to make decisions, the centralized training mechanism is applied to learn coordination strategy, and a prioritized experience replay (PER) strategy is adopted to improve learning efficiency. The proposed method can deal with the non-stationary problems in the process of a multi-agent game with partial observable information. In the execution stage, all trained agents are deployed in a distributed manner and make decisions in real time. Simulation results show that according to the proposed method, the training process of a multi-agent game is accelerated, and multiple agents can make optimal decisions only by local information.

Keywords: multi-microgrid; multi-agent; energy management; reinforcement learning

Citation: Guo, G.; Gong, Y. Multi-Microgrid Energy Management Strategy Based on Multi-Agent Deep Reinforcement Learning with Prioritized Experience Replay. *Appl. Sci.* **2023**, *13*, 2865. https://doi.org/10.3390/app13052865

Academic Editors: Muhammad Waseem, Shah Fahad, Arman Goudarzi and Hafiz Abdul Muqeet

Received: 2 January 2023
Revised: 20 February 2023
Accepted: 21 February 2023
Published: 23 February 2023

1. Introduction

To address severe climate challenges, many countries have proposed the goal of "net zero emissions" strategies [1]. As an important means of local integration and local consumption of distributed renewable energy, the microgrid (MG) is of great significance for reducing carbon emissions.

There have been many studies on the optimal scheduling of low-carbon MGs. Ref. [2] proposed a risk-based multi-objective energy exchange optimization for the networked MGs considering renewable power generation uncertainties. Refs. [3,4] proposed management methods and operation strategies of integrated energy storage systems and demand–response, respectively, for the uncertainty of renewable energy. However, with the introduction of social funds in the MG construction, the traditional centralized optimal scheduling method cannot reflect the interests of different agents, and the multi-agent method provides a solution to the multi-agent management problem. In [5], power suppliers and consumers in the MG were modeled as autonomous agents, maximizing their own benefits through local decisions of each agent. Refs. [6,7] proposed a scheduling model based on a multi-agent game and reinforcement learning for an integrated energy microgrid.

With the development of distributed generation, multiple MGs can operate under the same distribution network, forming a multi-microgrid (MMG) system. Ref. [8] proposed a cloud edge computing method for economic dispatch of active distribution network agents with MMG agents. Ref. [9] presents a stochastic programming model for the planning of MMGs in distribution networks.

The decision-making process for a MMG is a partially observable Markov game. Although the multi-agent method can reflect the respective interests of different agents, there are still the following problems that need to be solved in the actual process: (1) The traditional game theory model is limited to a simple environment with complete information [10], making it difficult to solve the observable multi-agent decision-making problem this way. (2) The training process of multiple agents with partially observable information is non-stationary, and it is difficult to reach a stable equilibrium point. The authors of [11] used the alternating direction method of multipliers (ADMM) to solve the optimization problem of a multi-agent microgrid. The algorithm requires all parameters in advance, and it needs to obtain accurate renewable energy output, real-time prices. and other information. The authors of [3,5–7] simulated the game process of MMG agents based on a Q-learning algorithm and found the best strategy, but Q-learning cannot handle high-dimensional continuous information, and its application scope is limited. The authors of [9,12] adopted a heuristic algorithm but still needs an accurate model. Ref. [13] adopted a data-driven Monte Carlo method to make day-ahead decisions on retail electricity prices at the point of common coupling (PCC) of MMG, which does not require an explicit mathematical model but cannot be applied to real-time decision-making problems. With the development of artificial intelligence technology, advanced deep reinforcement learning provides a new way for the artificial intelligence driving of MMG agents. The authors of [14] applied an asynchronous advantage actor–critic (A3C) algorithm to MMG energy management. Unlike the above algorithm, it can be applied to continuous multi-dimensional action and state space. A novel real-time energy management strategy for an autonomous multi-energy management system based on the deep deterministic policy gradient (DDPG) algorithm was proposed in [15], which can learn from past experiences. Although the DDPG algorithm has excellent performance in solving uncertain problems [16], it cannot guarantee stability in the multi-agent scenario [17]. The multi-agent deep deterministic policy gradient (MADDPG) algorithm [18] is used to solve Markov games in problems such as swarm intelligence perception [19] and active/reactive power coordinated scheduling [20].

This paper proposes a real-time energy management strategy for MMGs based on multi-agent deep reinforcement learning (MADRL), in which the objectives of the distribution system operator (DSO) are to improve the profit from selling power and improve the smoothness of the power exchange at the PCC, and the objective of each MG is to reduce its own operating cost. During operation, each agent can only obtain local observation information. The DSO agent adjusts the retail electricity price according to the operation state of the MMG, and each MG agent schedules according to the observation information. The decision-making problem of the MMG is modeled as a Markov game and solved by the MADDPG. The cooperation and competition between agents can be fully considered. Multiple agents use centralized training to reach equilibrium as soon as possible, and then they are deployed in a distributed manner to make decisions in real time. In addition, in order to improve training efficiency, this paper proposes the use of the prioritized-experience-replay method [21], in which the experience and priority are stored in a binary tree structure. The major contributions of this paper are summarized as follows:

1. A real-time energy management strategy is developed for the MMG to optimize the operation of the DSO and individual MGs. The decision-making problem of the MMG is modeled as a Markov game. Unlike a centralized optimization algorithm, the proposed model can take into consideration the interests of each agent and simulate the game process of agents.
2. A prioritized multi-agent deep deterministic policy gradient (PMADDPG) is proposed to solve the Markov game, where the centralized training mechanism is applied to learn a coordination strategy. The trained agents can be deployed in a distributed manner and make decisions in real time. In addition, a prioritized experience replay (PER) strategy is adopted to improve learning efficiency considering that the decision-making process of multi-agent with partial observable information is non-stationary.

3. According to the proposed method, all agents can make decisions based on local observation, which reduces the requirements for the communication network. In addition, the inferences of the neural network can allow making decisions quickly and greatly alleviate the communication burden. It avoids solving complex optimization problems through the time-consuming heuristic algorithms and realizes real-time decision making.
4. Privacy protection. In the execution stage, all agents make decisions based on local information. There is no information exchange between MMGs, and the DSO will not obtain the detailed information inside the MGs but only observe them as a whole. Thus, the privacy of MGs is protected.

The remainder of the paper is organized as follows. Section 2 formulates the multi-microgrid energy management problem, including the DSO model and MMGs model. In Section 3, the multi-microgrid energy management problem is converted into a Markov game model with multiple agents. Section 4 proposes the PMADDPG to solve the Markov game model with multiple agents. In Section 5, the numerical simulation results are shown and analyzed. Finally, Section 6 concludes the paper.

2. Modeling of Multi-Microgrid Energy Management

As an important subject of integrated distributed renewable energy [22], the power exchange of a MMG at the PCC has great uncertainty, which will impact the distribution network. To this end, this paper constructs a MMG energy management model. The interaction between the MMG and the distribution network is shown in Figure 1. The information is transmitted between the distribution network and the MMG through a two-way communication network. The DSO releases retail electricity prices to the MMG in real time, and each MG feeds back its electricity purchase or sales based on the retail price.

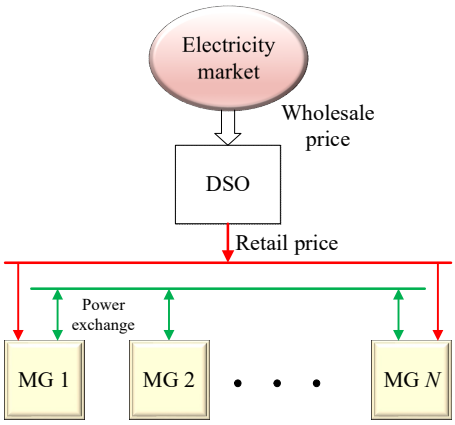

Figure 1. Multi-microgrid energy management system framework.

The objective of the DSO is to improve the profit of distribution network through an appropriate retail electricity price strategy, while smoothing the power exchange curve at the PCC. The DSO sets the retail price by solving the following problems:

$$\text{Max } E\left((1-\alpha)J_t^d/J_{base}^d - \alpha\varphi_t^d/\varphi_{base}^d\right) \tag{1}$$

$$J_t^d = \left(\lambda_t^d - \lambda_t^c\right)\tau\sum_{m=1}^{N_m}\varepsilon_m P_m^b(t) \tag{2}$$

$$\varphi_t^d = \left|P^{ex}(t) - P^{st}\right|\mathbb{I}\left(\left|P^{ex}(t) - P^{st}\right| \geq P_{thr}\right) \tag{3}$$

$$P^{ex}(t) = \sum_{m=1}^{N_m}\left(\varepsilon_m P_m^b(t) - P_m^s(t)/\varepsilon_m\right) \tag{4}$$

$$\mathbb{I}(x) = \begin{cases} 1 \text{ if} x \text{is} true \\ 0 \text{ if} x \text{is} false \end{cases} \tag{5}$$

$$P_m^b(t) * P_m^s(t) = 0 \tag{6}$$

where Equation (1) is the objective of the DSO, in which J_t^d is the profit of the DSO at time t, φ_t^d is the penalty term of the DSO for the unsmoothness of power exchange at the PCC, α is the weight factor, J_{base}^d is the normalization factor of profit, and φ_{base}^d is the normalization factor of the penalty term. The expression of J_t^d is shown in Equation (2), where λ_t^c is the wholesale electricity price at time t, λ_t^d is the retail electricity price released by DSO, and τ is the time interval. ε_m refers to the network loss factor from the MG to the PCC. The expression of φ_t^d is shown in Equation (3), where P_m^b and P_m^s represent the local purchasing and selling power of the MG. P^{ex} represents the total power exchange between the MG and the distribution network, and its expression is shown in Equation (4). $P^{ex} > 0$ represents the total net power purchased by the MG through the PCC. P^{st} is the expected power exchange, and P_{thr} is the threshold power. Equation (5) is the expression of logical function $\mathbb{I}$. When the condition is true, the function value is one; otherwise, it is zero. When the deviation of the power exchange is less than the threshold power, no penalty will be imposed on DSO. When the deviation is large, the penalty will be imposed to reduce the peak to average ratio of the power exchange at the PCC. Equation (6) represents that the MG can only perform one operation of power purchasing or sale at the same time.

For each MG, it solves the following mixed integer quadratic programming based on the received retail electricity price to reduce the operation cost:

$$\text{Min} \sum_{j=t}^{T} \gamma^{j-t} \cos t(t) \tag{7}$$

$$\cos t(t) = \left(\sum_{j \in \mathbf{M}} C_t^{g,j} + C_t^B + C_t^D + C_t^{ex} \right) \tau \tag{8}$$

where γ represents the attenuation coefficient; $\gamma \in [0,1]$. T is the total time steps, τ is the adjacent time interval. **M** is a collection of controllable generator sets. Cost (t) is the cost of the tth time step, specifically including the cost of controllable generator units, the cost of energy storage system, the cost of load demand response, and the cost of power exchange with external power grids. The mathematical expressions of all costs are as follows:

$$C_t^{g,j}\left(P_j^g(t)\right) = a_j\left(P_j^g(t)\right)^2 + b_j P_j^g(t) + c_j \tag{9}$$

$$C_t^B = k_B abs(\text{SOC}(t+1) - \text{SOC}(t)) cap \tag{10}$$

$$C_t^D = \sum_{z \in Z} \alpha_z P_z^D(t) \tag{11}$$

$$C_t^{ex} = \lambda_t^d \varepsilon_m P_m^b - \lambda_t^c P_m^s / \varepsilon_m \tag{12}$$

Equation (9) represents the generation cost of controllable unit j; a_j, b_j, and c_j are coefficients of the quadratic term, primary term, and constant term, respectively. Equation (10) represents the cost of the energy storage. A change in its state of charge (SOC) can characterize the use of the battery. K_B is the cost coefficient of the battery, and cap is the capacity of the battery. Equation (11) represents the cost of load demand response, α_Z is the response cost, P_z^D is the response quantity of block z, and **Z** is the collection of demand response blocks. Equation (12) is the cost of power exchange with the external power grid, which depends on the purchasing/selling power and real-time retail price. ε_m is the network loss

factor of the line. It is always greater than one, because the network gateway for settlement is at the PCC. In addition, the MG also needs to meet the following constraints:

$$P_j^{g,\min} \le P_j^g(t) \le P_j^{g,\max} \tag{13}$$

$$0 \le \sum_{z \in Z} P_z^D(t) \le k_d P^{load}(t) \tag{14}$$

$$P^{B,\min} \le P^B(t) \le P^{B,\max} \tag{15}$$

$$SOC(t+1) = SOC(t) + \varepsilon_c P^B(t)\tau \mathbb{I}\Big(P^B(t) \ge 0\Big)/cap + P^B(t)\tau \mathbb{I}\Big(P^B(t) < 0\Big)/\varepsilon_{disc} cap \tag{16}$$

$$SOC^{\min} \le SOC(t) \le SOC^{\max} \tag{17}$$

$$\sum_{j \in M} P_j^g(t) + P^{Wind}(t) + P^{PV}(t) + P_m^b = P^{load}(t) + P^B(t) + P_m^s - \sum_{z \in Z} P_z^D(t) \tag{18}$$

where Equation (13) is the constraint on the output of controllable unit; $P_j^{g,\min}$ and $P_j^{g,\max}$ are the output lower and upper limit of unit j, respectively. Equation (14) is a constraint on the power of the demand response block. K_d is the proportion of the load participating in the demand response, and the power curtailment of the demand response cannot exceed the limit. Equation (15) limits the power exchange P^B of the energy storage battery. $P^B > 0$ means that the MG charges the battery, and $P^B < 0$ means that the battery provides power support to the MG. $P^{B,min}$ and $P^{B,max}$, respectively, represent the power limits of discharge and charging of the battery. Equation (16) is the model of an energy storage system. ε_C and ε_{Disc} represent the charging and discharging efficiency of the energy storage battery, respectively. Additionally, *cap* represents the capacity of the battery. The logic function represents the current state of charge and discharge of battery, and *SOC* changes with the charging and discharging process. Equation (17) limits the range of *SOC*. SOC^{min} and SOC^{max} represent the lower and upper limits of *SOC*, respectively. Equation (18) is the balance constraint of the MG power. P^{Wind} and P^{PV} represent the output of the wind power generator and photovoltaic panel, respectively.

It is necessary for all power systems to meet the power balance constraints, and the same for multi-microgrid systems. Power-flow constraints are satisfied in power systems with a demand response [23] and distributed energy systems [24]. Actually, all the imbalance of the multi-microgrid system is balanced by the external power grid at the PCC, and P_m^b/P_m^s in Equation (2) is the power imbalance of MMGs, which represent the purchasing/selling power of the MGs. Additionally, Equation (18) is the power balance constraint. P_m^b/P_m^s provided by the external power grid maintains the power balance among the P_z^D, $P_j^g(t)$, $P^{Wind}(t)$, $P^{PV}(t)$, $P^{load}(t)$, and $P^B(t)$.

The mathematical model of each agent is established above. Multiple agents seek stable decision-making strategies by continuous games.

It is worth mentioning that with the increase in the penetration of distributed renewable energy, multiple energy storage systems have gradually become a research hotspot. Control frameworks for transactive energy storage services are proposed in [25], and optimal power-flow control of a hybrid renewable energy system with energy storage is proposed in [26]. Benefiting from the learning ability of reinforcement learning in complex environments, the method proposed in this paper can be easily transformed for multiple storage systems, specifically through the following two ways:

1. Deploying multiple energy storage systems in a microgrid. For the agent, it only needs to expand the dimensions of the action network—that is, to adjust the structure of the output layer of the neural network and add corresponding control signals to each energy storage system.
2. Multiple agents are deployed in multiple MGs. It is quite consistent with the multi-agent system proposed in this paper. It only needs to increase the number of agents

in the application process. In addition, the privacy of multiple energy storage can be protected.

3. Multi-Agent Markov Game Model

The MMG management system can be regarded as a decision-making model for multiple agents due to the differences in interest subjects and optimization objectives. The whole system can be divided into a DSO agent A^D and various MG agents A_i^m, i = 1, 2,..., N_m. The decision of each agent will be affected by the joint decision of other agents, and the Markov decision process is expanded into a Markov game in the multi-agent field. A Markov game of N agents can be represented as (S, A_1,..., A_N,..., O_1,..., O_N, R_1,..., R_N, TR), where S represents the set of overall states of all agents, A_i represents the set of actions of agent i, O_i represents the set of environmental states that agent i can observe, R_i represents the set of rewards of agent i, and TR represents the state transition function. The Markov game process of agent i is as follows: agent i observes its current state $o_i^t \in O_i$ at time step t, and then selects actions $a_i^t \in A_i$ according to the observed state to obtain corresponding rewards $r_i^t \in R_i$ and the next observed state $o_i^{t+1} \in O_i$. The strategy adopted by agent i is the mapping from state space to action space: $O_i \times A_i \rightarrow$ [0,1]. The mapping of the state transition function TR is shown as follows: $S \times A_1 \times \ldots \times A_N \rightarrow S$. Reward is a function of the agent's state and action, $S \times A_i \rightarrow R$. Each agent selects strategies according to its observation to maximize the expected discount reward:

$$R_i^t = \sum_{n=0}^{T} \gamma^n r_i^{t+n} \tag{19}$$

where γ is the discount factor and T is the time step of the observation interval.

The Markov game of the DSO agent and MG agents are given below. The observation state space o_t^D of the DSO agent at time step t is shown in Equation (20):

$$o_t^D : \left\{ P_{t-1}^{ex}, P_{t-2}^{ex}, \ldots, P_{t-L}^{ex}, \lambda_{t-1}^{c}, \lambda_{t-2}^{c}, \ldots, \lambda_{t-L}^{c}, h_t \right\} \tag{20}$$

where P_{t-L}^{ex} is the total power exchange of MG at time step $t-L$, λ_{t-L}^{c} is the wholesale electricity price at time step $t-L$, and h_t is the time component. It can be seen that the DSO agent needs to obtain the power-exchange information and wholesale electricity price information of the last L time steps when making decisions. The action $a_t^D : \lambda_t^D$ it takes at time step t is to set the retail electricity price of time step t.

The reward obtained by DSO agent in time step t is shown in (21):

$$r_t^d = (1-\alpha) J_t^d \ / \ J_{base}^d - \alpha \varphi_t^d \ / \ \varphi_{base}^d \tag{21}$$

It is composed of electricity selling profits and penalty factors; α is the weight factor.

The state information of MG i is shown in Equation (22):

$$o_{i,t}^m : \left\{ \boldsymbol{\lambda}_{\mathbf{t}}^{\mathbf{d}}, \mathbf{L_{i,t}}, \mathbf{P_{i,t}^{re}}, SOC_{i,t}, h_t \right\} \ \forall i \in \mathbf{N} \tag{22}$$

where $\mathbf{N}$ is the collection of MGs; $\boldsymbol{\lambda}_{\mathbf{t}}^{\mathbf{d}}$ is the retail electricity price vector; $\mathbf{L_{i,t}}$ is the load vector; $\mathbf{P_{i,t}^{re}}$ is the renewable energy output vector, including wind power output and photovoltaic output; $SOC_{i,t}$ is the state of charge of the energy storage battery in MG I; and h_t is the time corresponding to the time step t. The agent makes decisions based on the observation information, and each state vector is as follows:

$$\boldsymbol{\lambda}_{\mathbf{t}}^{\mathbf{d}} = \left[\lambda_t^d, \lambda_{t-1}^d, \ldots, \lambda_{t-L+1}^d \right] \tag{23}$$

$$\mathbf{L_{i,t}} = \left[l_{i,t-1}, l_{i,t-2}, \ldots, l_{i,t-L} \right] \tag{24}$$

$$\mathbf{P_{i,t}^{re}} = \left[p_{i,t-1}^{re}, p_{i,t-2}^{re}, \ldots, p_{i,t-L}^{re} \right] \tag{25}$$

where $\boldsymbol{\lambda_t^d}$ is the retail price of the DSO at time step t, $l_{i,t-1}$ is the load in the MG i at time step $t-1$, and $p_{i,t-1}^{re}$ is the output of renewable energy in the MG i at time step $t-1$. The MG agent makes decisions based on the above local observation information, and its decisions at time step t are as follows:

$$a_t^{m,i} : \left\{ P_{i,j}^g, P_t^B, P_t^D \right\} \ \forall \mathrm{j} \in \mathbf{M} \tag{26}$$

where each component represents the output of each controllable unit, the power of energy storage, and the response power of the load block.

The reward of the MG agent A_i^m is the opposite of the operating cost. In addition, the penalty item of SOC is also taken into account.

$$r_t^m = -\cos t(t)/J_{base}^m - \mathbb{I}\Big(SOC(t+1) < SOC^{min} \mathrm{or} SOC(t+1) > SOC^{\max}\Big)/\varphi_{base}^m \tag{27}$$

where J_{base}^m and φ_{base}^m are the normalization factors of the cost item and penalty item, respectively.

4. Solution Based on Multi-Agent Reinforcement Learning

4.1. Multi-Agent Reinforcement Learning Algorithm

The MADDPG is used to solve the above Markov game problem in this paper. In order to reach equilibrium in the non-stationary game environment as soon as possible, the MADDPG adopts a centralized training mode, in which additional information can be introduced to improve the stability of the environment. As shown in Equation (28), additional information of other agents is introduced into the training process. Even if the strategies of other agents change ($\pi_i \neq \pi_i{}'$), the training environment is still stable.

$$\mathcal{P}\left(s'|s, a_1, \ldots, a_N, \pi_1, \ldots, \pi_N\right) = \mathcal{P}\left(s'|s, a_1, \ldots, a_N\right) = \mathcal{P}\left(s'|s, a_1, \ldots, a_N, \pi_1{}', \ldots, \pi_N{}'\right) \tag{28}$$

Agent i is composed of the online actor network μ_i and the online critic network Q_i, and their parameters are θ_i and η_i, respectively. The experience generated in the interaction with the environment is continuous and does not meet the requirements of an independent and identical distribution of sampling, so replay buffer D is used to store historical experience, with the capacity of N_e. During training, random sampling is conducted in the replay buffer to cut off the correlation between experiences. The objective of reinforcement learning is to find a strategy to maximize reward expectation J.

$$J(\mu_i) = E\Big[Q_i^{\mu}(o, a_1, \ldots, a_N)\Big] \tag{29}$$

Actor network μ_i updates its parameters based on the gradient ascent method. The gradient of the objective function is shown in Equation (30):

$$\nabla_{\theta_i} J(\mu_i) = E_{o,a\sim\mathcal{D}}\Big[\nabla_{\theta_i}\mu_i(o_i)\nabla_{a_i}Q_i^{\mu}(o, a_1, \ldots, a_N)\Big|_{a_i=\mu_i(o_i)}\Big] \tag{30}$$

Where the policy gradient is related to the gradient of the critic network, and the online actor network μ_i achieves improvement with the help of the critic network gradient.

In order to improve the stability of the learning process, the target actor network (with parameter $\theta_i{}'$) and the target critic network (with parameter $\eta_i{}'$) are introduced. The target network has the same structure as the online network, but the parameter update frequency is different. The online critic network is updated by minimizing the error L with the target value y_i:

$$L(\theta_i) = \mathbb{E}_{o,a,r,o'\sim\mathcal{D}}\left[\left(Q_i^{\mu}(\mathrm{o}, a_1, \ldots, a_L) - y_i\right)^2\right] \tag{31}$$

$$y_i = r_i + \gamma Q_i^{\mu'}\left(o', a'_1, \ldots, a'_L\right)\Big|_{a'_i=\mu'_i(o'_i)} \tag{32}$$

where o' is the state observed after the agent executes action a, and γ is the discount factor.

The soft update strategy is used to update the target actor network and target critic network, as shown in (33) and (34):

$$\theta' \leftarrow \rho\theta + (1-\rho)\theta' \tag{33}$$

$$\eta' \leftarrow \rho\eta + (1-\rho)\eta' \tag{34}$$

where $\rho << 1$ is the update parameter of the target network, and the soft update is relatively slow.

Figure 2 shows the interaction process between multiple agents and the environment. μ_i makes decisions based on the local observations. The action of all agents jointly affects the environment, and each agent also receives corresponding real-time rewards. This transition is stored in the replay buffer of each agent. During training, agents take batch samples from replay buffers. Q_i needs to obtain overall state and action information. μ_i only needs local observation information.

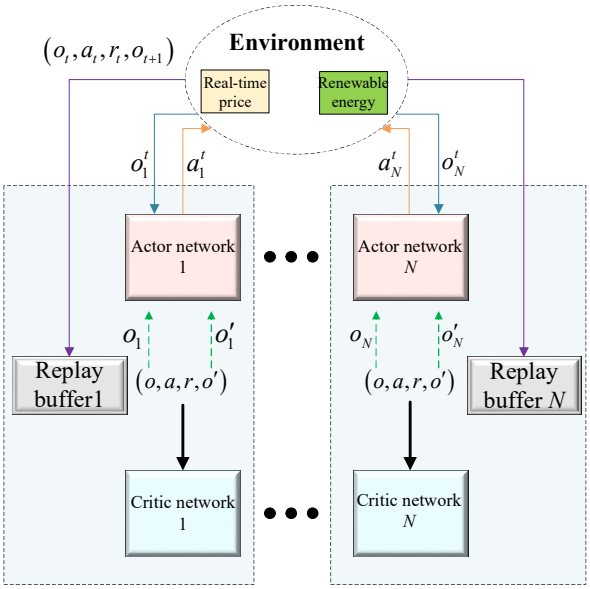

Figure 2. Framework of multi-agent centralized training.

4.2. The Prioritized Experience Replay Strategy

The experiences are sampled from the replay buffer during the training process, and the sampled experiences determine the learning effect of the neural network. In order to improve the learning efficiency, this paper proposes to use the prioritized-experience-replay strategy in the multi-agent training process. The value of experience in the replay buffer can be measured by the absolute value of TD error, as shown in Equation (35). A larger positive value of $\delta_{i.n}$ means that the action selection of the experience was relatively successful, and a large negative value means that there was a mistake in the action selection. These experiences can be used to quickly learn successful strategies and avoid major mistakes. Therefore, the sampling priority can be determined according to the absolute value of TD error to improve learning efficiency.

$$\delta_{i,n} = r_{i,n} + \gamma Q_{i,n}^{\mu'} - Q_{i,n}^{\mu} \tag{35}$$

The probability of experience being sampled is positively related to its priority. For agent i, when experience n is sampled and trained, its priority $p_{i,n}$ is updated as shown in (36):

$$p_{i,n} = |\delta_{i,n}| \tag{36}$$

When the newly generated experience is stored, its priority is equal to the current maximum priority, so as to avoid the newly generated experience not obtaining sampling opportunities. In order to effectively update and sample experience, as shown in Figure 3, a binary tree structure of sum tree is used to store experience and its priority. Each node is the sum of its child nodes. The priority of each experience is taken as the leaf node, and the root node is the sum of all experience priorities.

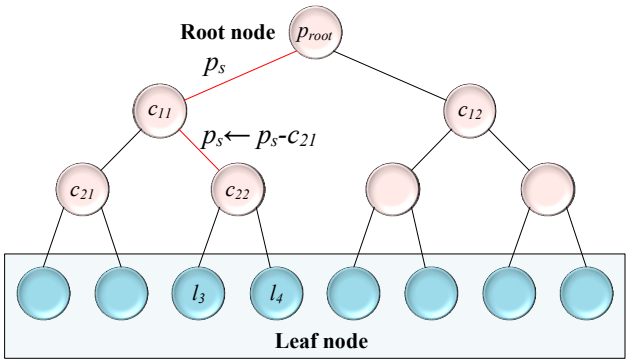

Figure 3. Structure diagram of the sum-tree.

Sample N_s samples in the replay buffer. The value of root node is divided into N_s intervals equally, and one randomly samples in each interval. For a certain sampling value p_s, if it is less than the value c_{11} of the left child node, the path on the left will be followed; otherwise, the path on the right will be followed. If $p_s < c_{11}$, p_s is compared with the values of the two child nodes in the lower layer on the left. If $p_s < c_{21}$, the left path will be followed. If $p_s > c_{21}$, the right path will be followed and p_s is updated. The new p_s is equal to the difference between the original value and the value of the left child node. It will continue according to the above rules until a leaf node is reached. Finally, the experience of the selected leaf node is trained. For the leaf node with priority of $p_{i,n}$, the probability of being selected is $P(n)$:

$$P(n) = \frac{p_{i,n}^{\beta_1}}{\sum_k p_{i,k}^{\beta_1}} \tag{37}$$

where β_1 is the priority factor, which changes the original priority in an exponential form. The value of β_1 is taken in the interval [0,1]. It is worth noting that compared with random sampling, there is a deviation between the state distribution obtained by priority sampling and the actual distribution. In order to correct this deviation, the importance sampling (IS) weight is introduced here ($\omega_{i,n}$):

$$\omega_{i,n} = (N_e P(n))^{-\beta_2} / \max_k \omega_{i,k} \tag{38}$$

where β_2 controls the correction degree of deviation. When the value is zero, it represents no deviation correction. When the value is one, it represents the complete correction of the deviation. At the end of the training, unbiased sampling of experience is very important, so β_2 increases linearly from the initial value to one in steps of $\Delta\beta$. At the beginning of training, more priority sampling is used to improve learning efficiency. Additionally, at the end of training, sampling is closer to unbiased sampling. For stability reasons, we adopted $1/\max_k \omega_{i,k}$ to normalize the IS weight, and Equation (31) is rewritten as follows:

$$L(\theta_i) = \frac{1}{N_s} \sum_{n=1}^{N_s} \omega_{i,n} \delta_{i,n}^2 \tag{39}$$

Finally, a prioritized multi-agent deep deterministic policy gradient (PMADDPG) algorithm is formed.

5. Results

5.1. Simulation Setup

The proposed method in this paper was implemented in Python with TensorFlow. The simulation experiment was implemented on Baidu AI Cloud server, and the configuration was as follows: Intel Xeon Platinum 6271 processor, 32-core CPU, 64 GB memory. The real-time electricity price data used in training were from the PJM electricity market, and the resolution of the data was 15 min. PV output and wind power output data were from power plants in China. The rated output of wind power and photovoltaic unit were 45 and 65 kW, respectively. Seven weeks of historical data were selected as the training set.

The case included three MG agents and one DSO agent. MG1 and MG2 contained wind power units and photovoltaic units, respectively, and MG3 contained both. In addition, traditional units, energy storage, and demand response resources existed in each MG. The specific parameters of the system are shown in Table 1.

Table 1. Parameters.

Parameter	Value	Parameter	Value
ε_m	1.1	J^d_{base}/¥	1
φ^d_{base}/kW	100	P^{st}/kW	200
P_{thr}/kW	50	γ	0.9
a_j/(¥/kW^{2}h)	[0.0015,0.003]	b_j/(¥/kWh)	[0.001,0.026]
c_j/(¥/h)	[1,1.8]	$P^{g,\min}_j$/kW	{0}
$P^{g,\max}_j$/kW	{60,80}	k_B/(¥/kWh)	0.02
cap/kWh	80	$P^{B,min}$/kW	−50
$P^{B,max}$/kW	50	SOC^{min}	0.2
SOC^{max}	0.8	$\varepsilon_c/\varepsilon_{disc}$	0.9
α_1/(¥/kWh)	0.14	$\Delta\alpha$/(¥/kWh)	0.0035
K_d	0.3	J^m_{base}/¥	2
φ^m_{base}	1	β_1	0.6
β^0_2	0.4	–	–

5.2. Network Structure

The structures of the actor network and critic network designed in this paper are shown in Figure 4. In order to prevent overfitting of the network, layer normalization (LN) is applied to the output vector of the hidden layer. The activation function of the hidden layer of the actor network was a rectified linear unit (ReLU), and the activation function of the output layer was a hyperbolic tangent function (tanh). The activation function of the output layer of the critic network was still ReLU. The optimizer used in the gradient descent process was the adaptive moment estimation optimizer (Adam).

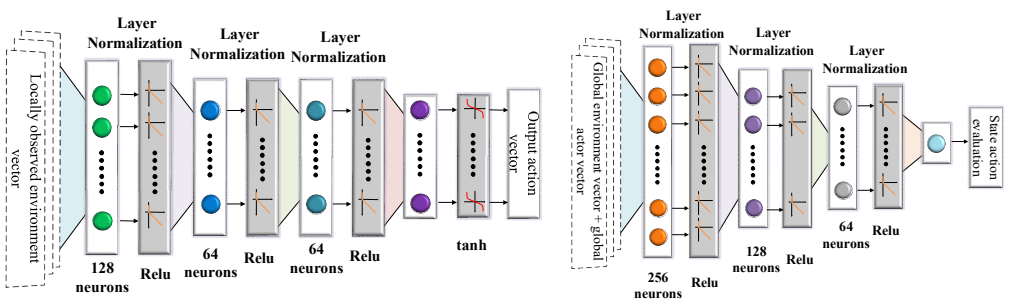

Figure 4. The structures of actor and critic networks: (**a**) actor network; (**b**) critic network.

5.3. Simulation Result

The training was in episode form. There were 1000 episodes in total. Each episode contained 1344 steps, and the reward was the average daily reward of each episode. The DDPG algorithm was used as the control group, which has been used to solve the problems of multi-agent decision making in many papers [10,27]. The same network structure was used by the PMADDPG algorithm and DDPG algorithm, and their convergence paths are shown in Figures 5 and 6, respectively:

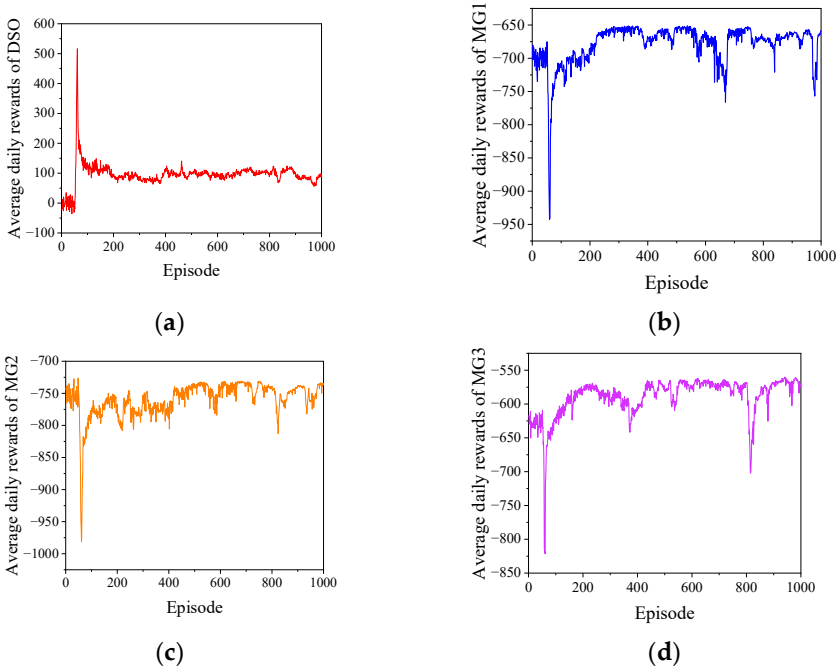

Figure 5. Average daily rewards of multiple agents under PMADDPG algorithm: (**a**) DSO; (**b**) MG1; (**c**) MG2; (**d**) MG3.

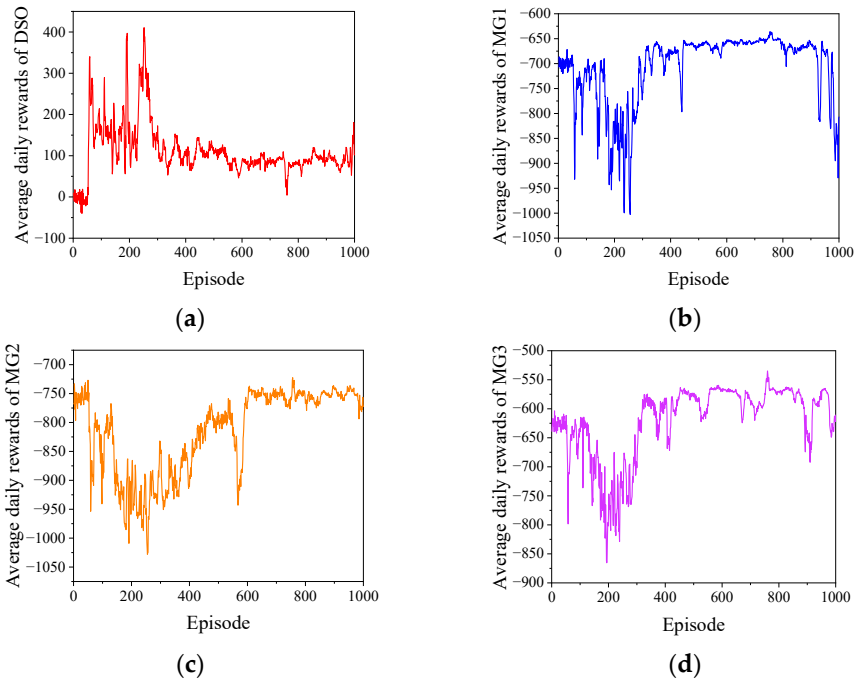

Figure 6. Average daily rewards of multiple agents under the DDPG algorithm: (**a**) DSO; (**b**) MG1; (**c**) MG2; (**d**) MG3.

It can be seen in Figure 5 that after 200 episodes of training, all agents could quickly learn their optimal strategies and reach the equilibrium point. As shown in Figure 5a, the reward of the DSO agent was zero at the initial stage, but it could be stabilized to about 100 after 150 episodes. At the initial stage, the reward of the DSO agent rose sharply, and the reward of the MG agents dropped rapidly. This is because DSO makes full use of its dominant position before MG agents learn the best strategy. Once all MG agents learn their optimal strategies, the dominant position of the DSO will be restrained. Compared with the other MGs, the reward of MG3 is larger, which is due to its higher proportion of renewable energy. Differently from the agents in Figure 5, the agents in Figure 6 experienced a long period of fluctuation and did not reach a stable stage until 400 rounds. However, the fluctuations were still large. This fully shows that the environment of multi-agent game is unstable. The introduction of centralized training and priority sampling strategy can significantly improve the learning efficiency of multiple agents.

It can be seen in Equation (21) that the weight factor α in the DSO;s reward is used to maintain the balance between the DSO's profits and the stationary degree of power exchange at the PCC (the stationary degree is the opposite of Equation (3)). The retail electricity price strategy and reward of the DSO agent will be impacted by α. Figure 7 shows the optimal retail electricity price under different α values.

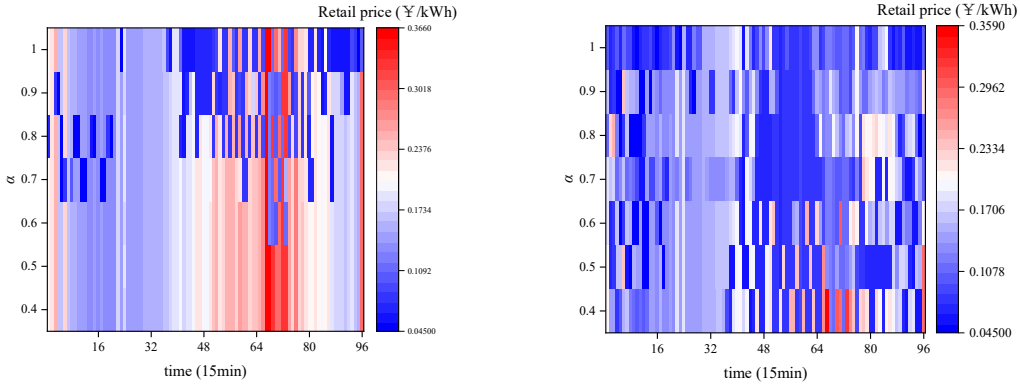

Figure 7. The retail electricity prices under different α values: (**a**) PMADDPG; (**b**) DDPG.

It can be seen in Figure 7a that the DSO tends to reduce retail electricity price as the weight factor α increases. At time step 16, the retail electricity price changed from light blue, representing the lowest prices, to dark blue, representing low prices, and then to light blue, as α increased from 0.4 to 1. At time step 48, the retail electricity price changed from white to dark blue, representing low prices. At time step 70, the retail electricity price changed from red, representing high prices, to blue, representing low prices. This is because the DSO will pay less attention to the power-selling profit when α is increasing. Compared with the PMADDPG model, the DSO agent based on the DDPG model starts to reduce the retail electricity price when the value of α is not high, leading to less profit. This is because the DDPG lacks a centralized training mechanism, which makes it difficult for multiple agents to explore an optimal strategy in a non-stationary environment.

5.4. Sensitive Analysis

Figures 8 and 9 show average daily profit of the DSO and the stationary degree of power exchange at the PCC under different α.

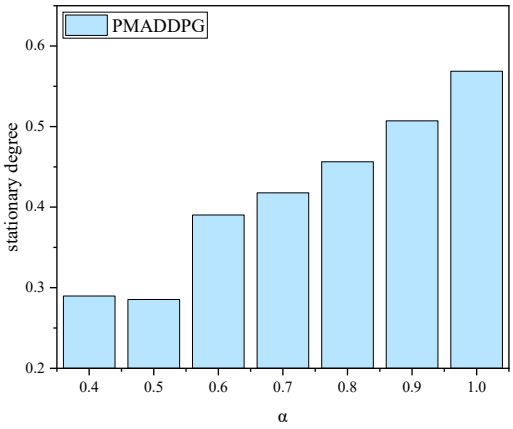

Figure 8. Stationary degrees under different values of α.

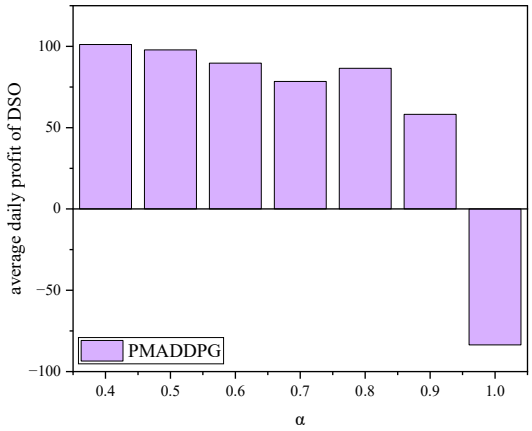

Figure 9. Average daily profit of the DSO under different values of α.

As can be seen in Figures 8 and 9, the increasing α will lead to an increase in the stationary degree and a decrease in the average daily profit of the DSO. This is because with α increasing, the DSO expands the peak–valley difference in retail electricity price to guide the MMG toward more stationary power exchange at the PCC. Figure 10 shows the rewards of MG agents under different values of α.

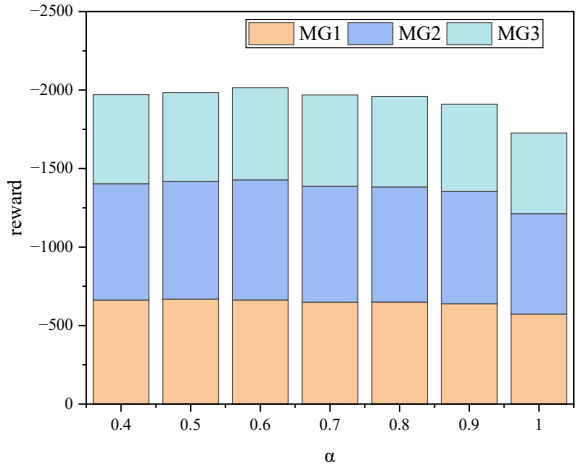

Figure 10. Reward of each MG agent under different values of α.

It can be seen in Figure 10 that the impact of α on MGs is small, since the MGs do not assume the responsibility for power smoothing. Next, the power smoothing responsibility of MG agents will be studied. At this time, the penalty item needs to be applied to the reward of MGs. The new reward of MG is shown in Equation (40):

$$\widetilde{r}_t^m = (1-\beta) r_t^m - \beta \varphi_t^d \ / \ \varphi_{base}^m \tag{40}$$

where β refers to the weight factor of penalty item for MG agents, and its value is in [0,1]. The higher the value of β is, the greater the MG's responsibility for stationary degree of power exchange. The expressions of the penalty factor φ_t^d and the original reward r_t^m of the MG m are shown in Equations (3) and (27), respectively. φ_{base}^m represents the normalization factors of the penalty item of MG.

Figure 11 shows the stationary degrees of power exchange at the PCC under different combinations of α and β. Figure 12 shows the average daily profits of the DSO under different combinations of α and β.

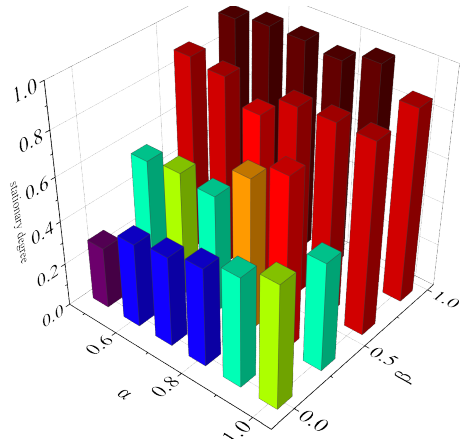

Figure 11. Stationary degrees under different values of α and β.

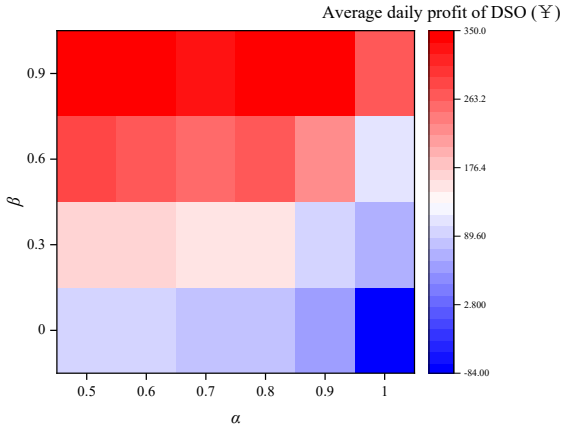

Figure 12. Average daily profit of the DSO under different values of α and β.

It can be seen in Figure 11 that as α and β increase, the stationary degrees show an upward trend. Compared with α, an increase in β has a greater impact on the improvement of the stationary degrees. However, when β is large, a change in α has little effect on the stationary degree. It can be seen in Figure 12 that the average daily profit of the DSO will increase significantly if the MGs assume certain responsibility for power smoothing.

5.5. Adjustment of Energy Storage and Demand Response

Finally, this paper shows the utilization of energy storage and demand resources by each MG for one week, as shown in Figures 13 and 14, respectively. It can be seen that the energy storage utilization rates of MG1 and MG3 were high. This is because many distributed wind-power resources existed in MG1 and MG3, in which large volatility of wind power output improved the utilization of energy storage system. In addition, demand–response resources were fully utilized in every microgrid, which also reflects the importance of load regulation to economic operation.

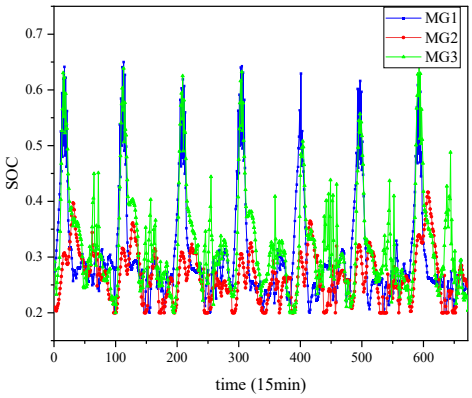

Figure 13. Profiles of SOC of the energy storage unit in each MG.

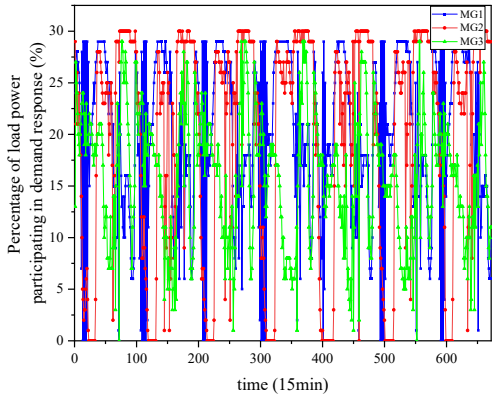

Figure 14. Participation of demand–response block in each MG.

6. Conclusions

In this paper, the energy management of MMG with partial observation information was studied, and the management model composed of the DSO agent and MG agents was established. The MADDPG was used to learn the optimal strategy of each agent, and the learning efficiency was improved by the prioritized-experience-replay strategy. This method improves the management of MMGs in the following ways: Firstly, the multi-agent model in this paper can fully consider the interests of each agent and simulate the game process of the agents. The trained agent can make decisions with only local information, which reduces the communication requirements. Secondly, the data-driven algorithm used in this paper can better handle the random variables that are difficult to model accurately and can learn from the historical electricity price, wind power, and photovoltaic output data. In addition, the prioritized experience replay strategy is adopted to improve the learning efficiency, and the agents can quickly reach the equilibrium. Simulation results show that compared to the traditional DDPG, the proposed PMADDPG achieves fast training convergence and stationary training. All agents can make optimal decisions based

on local observation, and real-time decision making can be achieved thanks to the inference of the neural network.

Author Contributions: Conceptualization, G.G. and Y.G.; methodology, G.G. and Y.G.; investigation, G.G. and Y.G.; writing—original draft preparation, G.G.; writing—review and editing, Y.G.; supervision, G.G. and Y.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data included in this study are available upon request by contact with the corresponding author.

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

References

1. Energy & Climate Intelligence Unit. Net Zero Scorecard. Available online: https://eciu.net/netzerotracker (accessed on 1 December 2022).
2. Zhu, Y.; Li, G.; Guo, Y.; Li, D.; Bohlooli, N. Modeling Optimal Energy Exchange Operation of Microgrids Considering Renewable Energy Resources, Risk-based Strategies, and Reliability Aspect Using Multi-objective Adolescent Identity Search Algorithm. *Sustain. Cities Soc.* **2022**, *91*, 104380. [CrossRef]
3. Shang, Y.; Wu, W.; Guo, J.; Ma, Z.; Sheng, W.; Lv, Z.; Fu, C. Stochastic dispatch of energy storage in microgrids: An augmented reinforcement learning approach. *Appl. Energy* **2020**, *261*, 114423. [CrossRef]
4. Zhang, H.; Zhou, S.; Gu, W.; Zhu, C. Optimized operation of micro-energy grids considering the shared energy storage systems and balanced profit allocation. *CSEE J. Power Energy Syst.* **2023**, *9*, 254–271.
5. Foruzan, E.; Soh, L.K.; Asgarpoor, S. Reinforcement learning approach for optimal distributed energy management in a microgrid. *IEEE Trans. Power Syst.* **2018**, *33*, 5749–5758. [CrossRef]
6. Jendoubi, I.; Bouffard, F. Data-driven sustainable distributed energy resources' control based on multi-agent deep reinforcement learning. *Sustain. Energy Grids Netw.* **2022**, *32*, 100919. [CrossRef]
7. Samadi, E.; Badri, A.; Ebrahimpour, R. Decentralized multi-agent based energy management of microgrid using reinforcement learning. *Int. J. Electr. Power Energy Syst.* **2020**, *122*, 106211. [CrossRef]
8. Li, X.; Wang, J.; Lu, Z.; Cai, Y. A cloud edge computing method for economic dispatch of active distribution network with multi-microgrids. *Electr. Power Syst. Res.* **2023**, *214*, 108806. [CrossRef]
9. Vera, E.G.; Cañizares, C.A.; Pirnia, M.; Guedes, T.P.; Melo, J.D. Two-Stage Stochastic Optimization Model for Multi-Microgrid Planning. *IEEE Trans. Smart Grid* **2022**. [CrossRef]
10. Liang, Y.; Guo, C.; Ding, Z.; Hua, H. Agent-Based Modeling in Electricity Market Using Deep Deterministic Policy Gradient Algorithm. *IEEE Trans. Power Syst.* **2020**, *35*, 4180–4192. [CrossRef]
11. Zheng, Y.; Song, Y.; Hill, D.J.; Zhang, Y. Multiagent system based microgrid energy management via asynchronous consensus ADMM. *IEEE Trans. Energy Convers.* **2018**, *33*, 886–888. [CrossRef]
12. Khan, B.; Singh, P. Economic operation of smart micro-grid: A meta-heuristic approach. In *Research Anthology on Smart Grid and Microgrid Development*; IGI Global: Hershey, PL, USA, 2022; pp. 1213–1230.
13. Du, Y.; Li, F. Intelligent multi-microgrid energy management based on deep neural network and model-free reinforcement learning. *IEEE Trans. Smart Grid* **2019**, *11*, 1066–1076. [CrossRef]
14. Hua, H.; Qin, Y.; Hao, C.; Cao, J. Optimal energy management strategies for energy Internet via deep reinforcement learning approach. *Appl. Energy* **2019**, *239*, 598–609. [CrossRef]
15. Ye, Y.; Qiu, D.; Wu, X.; Strbac, G.; Ward, J. Model-Free Real-Time Autonomous Control for A Residential Multi-Energy System Using Deep Reinforcement Learning. *IEEE Trans. Smart Grid* **2020**, *11*, 3068–3082. [CrossRef]
16. Tan, Y.; Shen, Y.; Yu, X.; Lu, X. Low-carbon economic dispatch of the combined heat and power-virtual power plants: A improved deep reinforcement learning-based approach. *IET Renew. Power Gener.* **2022**. [CrossRef]
17. Zhang, Y.; Mou, Z.; Gao, F.; Jiang, J.; Ding, R.; Han, Z. UAV-Enabled Secure Communications by Multi-Agent Deep Reinforcement Learning. *IEEE Trans. Veh. Technol.* **2020**, *69*, 11599–11611. [CrossRef]
18. Lowe, R.; Wu, Y.I.; Tamar, A.; Harb, J.; Abbeel, O.I.P.; Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 6379–6390.
19. Gu, B.; Yang, X.; Lin, Z.; Hu, W.; Alazab, M.; Kharel, R. Multi-Agent Actor-Critic Network-based Incentive Mechanism for Mobile Crowdsensing in Industrial Systems. *IEEE Trans. Ind. Inform.* **2020**, *17*, 6182–6191. [CrossRef]
20. Hu, D.; Ye, Z.; Gao, Y.; Ye, Z.; Peng, Y.; Yu, N. Multi-Agent Deep Reinforcement Learning for Voltage Control With Coordinated Active and Reactive Power Optimization. *IEEE Trans. Smart Grid* **2022**, *13*, 4873–4886. [CrossRef]
21. Schaul, T.; Quan, J.; Antonoglou, I.; Silver, D. Prioritized experience replay. *arXiv* **2015**, arXiv:1511.05952.

22. Tomin, N.; Shakirov, V.; Kozlov, A.; Sidorov, D.; Kurbatsky, V.; Rehtanz, C.; Lora, E.E.S. Design and optimal energy management of community microgrids with flexible renewable energy sources. *Renew. Energy* **2022**, *183*, 903–921. [CrossRef]
23. Yao, M.; Molzahn, D.K.; Mathieu, J.L. An optimal power-flow approach to improve power system voltage stability using demand response. *IEEE Trans. Control. Netw. Syst.* **2019**, *6*, 1015–1025. [CrossRef]
24. Scarabaggio, P.; Carli, R.; Dotoli, M. Noncooperative Equilibrium-Seeking in Distributed Energy Systems Under AC Power Flow Nonlinear Constraints. *IEEE Trans. Control. Netw. Syst.* **2022**, *9*, 1731–1742. [CrossRef]
25. Mignoni, N.; Scarabaggio, P.; Carli, R.; Dotoli, M. Control frameworks for transactive energy storage services in energy communities. *Control. Eng. Pract.* **2023**, *130*, 105364. [CrossRef]
26. Venkatesan, K.; Govindarajan, U. Optimal power flow control of hybrid renewable energy system with energy storage: A WOANN strategy. *J. Renew. Sustain. Energy* **2019**, *11*, 015501. [CrossRef]
27. Ye, Y.; Qiu, D.; Sun, M.; Papadaskalopoulos, D.; Strbac, G. Deep reinforcement learning for strategic bidding in electricity markets. *IEEE Trans. Smart Grid* **2020**, *11*, 1343–1355. [CrossRef]