actuators

MDPI

Article

Multi-Agent Reinforcement Learning with Optimal Equivalent Action of Neighborhood

Haixing Wang [1], Yi Yang [1,*], Zhiwei Lin [2] and Tian Wang [3]

[1] Henan Key Laboratory of Intelligent Detection and Control of Coal Mine Equipment, School of Electrical Engineering and Automation, Henan Polytechnic University, Shiji Road, Jiaozuo 454003, China; wanghaixing@hpu.edu.cn
[2] School of Mathematics and Physics, Queen's University Belfast, University Road, 10587, Belfast BT7 1NN, UK; z.lin@qub.ac.uk
[3] Institute of Artificial Intelligence, Beihang University, Xueyuan Road, Beijing 100083, China; wangtian@buaa.edu.cn
* Correspondence: yangyi@hup.edu.cn

Abstract: In a multi-agent system, the complex interaction among agents is one of the difficulties in making the optimal decision. This paper proposes a new action value function and a learning mechanism based on the optimal equivalent action of the neighborhood (OEAN) of a multi-agent system, in order to obtain the optimal decision from the agents. In the new Q-value function, the OEAN is used to depict the equivalent interaction between the current agent and the others. To deal with the non-stationary environment when agents act, the OEAN of the current agent is inferred simultaneously by the maximum a posteriori based on the hidden Markov random field model. The convergence property of the proposed methodology proved that the Q-value function can approach the global Nash equilibrium value using the iteration mechanism. The effectiveness of the method is verified by the case study of the top-coal caving. The experiment results show that the OEAN can reduce the complexity of the agents' interaction description, meanwhile, the top-coal caving performance can be improved significantly.

Keywords: multi-agent reinforcement learning; optimal decision; hidden Markov random field; top-coal caving

Citation: Wang, H.; Yang, Y.; Lin, Z.; Wang, T. Multi-Agent Reinforcement Learning with Optimal Equivalent Action of Neighborhood. *Actuators* **2022**, *11*, 99. https://doi.org/10.3390/act11040099

Academic Editor: Zhuming Bi

Received: 1 March 2022
Accepted: 23 March 2022
Published: 25 March 2022

1. Introduction

Optimal decision-making in a multi-agent system with uncertainty [1–3] in the non-stationary environment [4–6] is a challenging problem. Reinforcement learning (RL) [7,8] is an effective method to yield the optimal decision of the multi-agent system based on the Markov decision-making process and dynamic programming [9–14]. Theoretically, each agent calculates its action based on the current state and the interaction with other agents. The calculation always retraces all the possible decision processes from the terminal state, and the complexity will become exponential with a higher number of agents [15]. The method to obtain the cooperative policy of each agent is based on the Q-value about state and joint actions of a multi-agent system [16,17]. Hence, how to establish the expression of Q-value to describe the interaction structure among the agents is one of the most important issues in a multi-agent system.

The existing approaches include recording all interactions among agents [15,18,19]. In these methods, each agent has its Q-value function to depict the joint actions of all the other agents. Hence, it can fully present the relationships of any agent pair. However, the computing complexity will rise dramatically as well as the space complexity. More importantly, it is even impossible to enumerate all the relationships if the number of agents is very large. Employ graph network is a new method to establish the relationship among the agent [20], establishes a graph network to describe attention. The target agents and

traffic participants are covered by the graph network [21] proposes a graph-based attention communication method to coordinate the interactions of scheduler and message processor.

Sharing neighbors' equivalent actions is another effective method to construct the interaction among agents. The parameters of the Q-value function include the action value of the current agent and the equivalent action of the agents in the neighbours. That reduces the number of Q-value functions and decreases the learning complexity significantly. These approaches [18,22] obtain the optimal decision of two competitor agents, in which the opponent action of the current agent could be considered to be a special kind of equivalent action. A computation rule to calculate joint action for Q-value function is proposed to reduce the compute complexity [23], and the Q-value function of the current agent is designed using the neighborhood equivalent action [24]. They do not consider the incidence relation of all the agents, hence the memory space for training the Q-value is smaller and the training process is faster.

In a multi-agent system, if the neighborhood agents choose optimal action, the current agent could obtain the optimal action more easily. Based on this, we develop a new Q-value function to depict the environment state, the current agent action, and the optimal equivalent action of the neighborhood agents.

However, it is well-known that the environment of a multi-agent system is non-stationary when the agents execute their policy [25,26]. Therefore, it is hard to obtain the optimal equivalent action from the neighbors. To address these issues, this paper proposes a multi-agent optimal decision method based on a new Q-value function that consists of the system state, the current agent action, and the optimal equivalent action of the neighborhood (OEAN). To obtain the OEAN, the hidden Markov random field (HMRF) is employed to establish the probability graphical model (PGM) [22] for the multi-agent system decision, then the OEAN is obtained by the maximum a posterior (MAP) estimation. The main contribution of this work includes:

(1) A new Q-value function based on OEAN is proposed to establish the interaction between the current agent and its neighbors, so that the current agent could obtain the optimal decision more easily.

(2) The PGM is used to infer the optimal actions of agents in the neighborhood by the MAP, based on the HMRF model, to avoid the issue due to non-stationary environment. The OEAN of the current agent is calculated based on the PGM inference result.

(3) The learning mechanism of the new Q-value function based on the OEAN is presented to guarantee the Q-value converging to the global Nash equilibrium.

The remainder of the paper is organized as follows. Section 2 presents related work in order to motivate our work. In Section 3, the new Q-value function is proposed for the multi-agent RL. In Section 4, the HMRF model is employed to estimate the OEAN, and the convergence of the method is proved. In Section 5, the experiment of top-coal caving demonstrates the effectiveness of the method. The conclusion is given in Section 6.

2. Related Work

This paper addresses the issue of multi-agents optimal decision by RL with PGM. Setting the independent Q-value function for each agent is the direct method [25,27], in which the interaction of each agent is depicted by the actions of the current agent and the other agents [28]. In 2003 [29], proposed the Nash Q-learning to the non-cooperative multi-agent, and the RL iteration can converge to the Nash equilibrium point. A similar convergence proof can be found in [30,31]. At present, Nash Q-learning is extended to electricity markets [32], interconnected multi-carrier systems [33], continuous control systems [34], etc. However, if the agent number is huge, the calculation and storage for depicting the relationship between each current is complex. To decrease the calculation and storage [24], defines a new Q-function, in which the neighbor's action is transformed into an equivalent action based on mean field theory. In this paper, we propose a new Q-value function based on the optimal OEAN along the way of above references .

The OEAN is inferred by PGM. Actually, PGM is one of the effective ways to describe the Markov decision problem by RL, in which the random field is used to formulate the relationship of agents by node and edge [35]. The implementation of PGM for Markov decision process often includes the Bayesian network [36–38] and the conditional random field [39,40] and they are classical model-based method, which means the ground truth is often needed to train the parameters of the model.

Nevertheless, the environment of the multi-agent system is non-stationary during the decision process [41], hence it is difficult or even impossible to obtain the ground truth. The Hidden Markov Model (HMM) [42] is an available method to deal with parameter learning without ground truth in RL, in which the unknown ground truth is action considered to be a hidden variable [43]. At present, to the best of our knowledge, HMM is employed to deal with the signal-agent systems due to the principle of HMM is restricted for a single Markov decision. Hidden Markov random field (HMRF) [44] extends the single hidden variable to a hidden random field. Despite the fact that it is proposed to deal with image segmentation problem [44,45], it provides an available method to infer the optimal decision without ground truth.

This paper follows Nash Q-learning and proposes a new Q-value function based on OEAN for the optimal decision of multi-agent based on HMM and HMRF.

3. Reinforcement Learning Based on the OEAN

3.1. Background

For the decision process of a multi-agent system, let the state space be $\mathcal{S}$, and the state of the multi-agent system be $s \in \mathcal{S}$, the action space be $\mathcal{A}$. For agent i, let its action be $a_i \in \mathcal{A}_i$, where $\mathcal{A}_i \subseteq \mathcal{A}$. The Markov decision process of the agents is defined as $\mathcal{M} \triangleq \{\mathcal{S}; \mathcal{A}_1, \ldots, \mathcal{A}_N; r_1, \ldots, r_N; p; \gamma\}$, where N is the agents number, the discount factor of reward is γ, $\gamma \in (0, 1)$. For agent i:

(1) the reward function is $r_i : \mathcal{S} \times \mathcal{A}_1 \times \ldots \times \mathcal{A}_N \to \mathbb{R}$;

(2) the transition probability is $p : \mathcal{S} \times \mathcal{A}_1 \times \ldots \times \mathcal{A}_N \to \Gamma(\mathcal{S})$, $\Gamma(\mathcal{S})$ describes the states' transition probability distribution over $\mathcal{S}$;

(3) the policy is defined as $\pi_i(a_i \mid s) : \mathcal{S} \to \Gamma_A(\mathcal{A}_i)$, $\Gamma_A(\mathcal{A}_i)$ characterizing the probability distribution over action space $\mathcal{A}_i$; the joint policy of all the agents is π, $\pi = [\pi_1, \ldots, \pi_N]$.

If the multi-agent system initial state is s, the value function of agent i under the joint policy π is formulized as

$$v_i^{\pi}(s) = \mathbb{E}_{\pi}\left(\sum_{t=0}^{\infty} \gamma^t r_i^t \mid s\right) \tag{1}$$

The action value function of agent i under the joint policy π is defined as $Q_i^{\pi} : \mathcal{S} \times \mathcal{A}_1 \times \ldots \times \mathcal{A}_N \to \mathbb{R}$, and

$$Q_i^{\pi}(s, a) = r_i(s, a) + \gamma \mathbb{E}_{s' \sim p}\left(v_i^{\pi}(s')\right) \tag{2}$$

where $a = [a_1, \ldots, a_N]$ is the action of each agent under the joint policy π, s' is the state of next step. By Equation (1), the action value function can be rewritten as

$$v_i^{\pi}(s) = \mathbb{E}_{\pi}(Q_i^{\pi}(s, a)) \tag{3}$$

In the multi-agent system, the input and output of the agents could be, respectively, considered to be a random field of states and a *Markov random field* (MRF) of the actions. In this paper, we suppose the input random field is the state of the multi-agent s, and the MRF of action is denoted by $a = \{a_1, \ldots, a_N\}$. In the multi-agent system, each action value function $Q(s, a)$ with the joint action of its neighborhood could be factored as follows [24].

$$Q_i(s, a) = \frac{1}{\lfloor N_i \rfloor} \sum_{k \in N_i} Q_i(s, a_i, a_k) \tag{4}$$

where N_i is the neighborhood of agent i, and $\lfloor N_i \rfloor$ is the neighborhood size.

For agent i, the Q-value function with joint actions can be estimated by an approximated function $Q_i(s, a_i, \bar{a}_i)$, $\bar{a}_i$ is the equivalent action of the neighborhood. This conclusion can be found in [24] shown as the following lemma.

Lemma 1 ([24]). *In a multi-agent system, for agent i, the neighborhood is N_i, in which the agent k belongs to N_i, $k \in N_i$; $\bar{a}_i$ is the* equivalent action *of N_i, $\bar{a}_i = \frac{1}{\lfloor N_i \rfloor} \sum_{k \in N_i} a_k$, such that $Q_i(s, a)$ can be expanded into Taylor series at the point $(s, a_i, \bar{a}_i)$, and $Q_i(s, a)$ can be approximated as $Q_i(s, a_i, \bar{a}_i)$, i.e.,*

$$Q_i(s, a) \approx Q_i(s, a_i, \bar{a}_i) \tag{5}$$

3.2. Multi-Agent Policy with the OEAN

When the current agent makes a decision in a multi-agent system, if the actions of neighborhood agent N_i are optimal, the current agent i could be easily to obtain the optimal action. Hence, we define the optimal equivalent action of the neighborhood (OEAN) as follows to describe the neighborhood condition:

$$\bar{a}_i^* = \frac{1}{\lfloor N_i \rfloor} \sum_{k \in N_i} a_k^* \tag{6}$$

where a_k^* is the optimal action of agent k in the neighborhood N_i. Based on the OEAN, this paper proposes a new action value function $Q_i\left(s, a_i, \bar{a}_i^*\right)$ to describe the relation of the environment, the current agent, and the equivalence of the neighbors.

It should be noted that $\bar{a}_i^*$ is result of picking the optimal equivalent action in the optimal action a_k^* of neighborhood, $k \in N_i$. We denote the equivalent policy of neighborhood agents as $\bar{\pi}_i$. That means if all the neighborhood agents obtain the optimal policy, $\bar{\pi}_i$ is an optimal equivalent policy, and the policy decided by Q-value function for the current agent would be more directly and feasibly. $\bar{\pi}_i$ is a hypothesis policy because it is difficult to directly calculate the optimal action of the neighbors on time in the non-stationary environment, and the estimating method of OEAN $\bar{a}_i^*$ will be given in Section 4.

According to Lemma 1 and Q-learning algorithm [7], we consider the OEAN $\bar{a}_i^*$ to substitute the equivalent action $\bar{a}_i$, and establish the learning mechanism for the Q-value function as follows:

$$Q_i^{t+1}(s, a_i, \bar{a}_i^*) = (1 - \alpha) Q_i^t(s, a_i, \bar{a}_i^*) + \alpha\left(r_i + \gamma v_i^t\left(s'\right)\right) \tag{7}$$

where

$$v_i^t\left(s'\right) = \mathbb{E}_{\bar{\pi}_i\left(\bar{a}_i^* | s'\right)} \mathbb{E}_{\pi_i^t\left(a_i | s', \bar{a}_i^*\right)} Q_i^t\left(s', a_i, \bar{a}_i^*\right) \tag{8}$$

in which $\gamma \in [0, 1)$ is the discount factor, α is the learning rate; π shown as follows is the policy of the agent i.

$$\pi_i^t(a_i \mid s, \bar{a}_i^*) = \frac{\exp\left(\beta Q_i^t\left(s, a_i, \bar{a}_i^*\right)\right)}{\sum_{a_i' \in \mathcal{A}_i} \exp\left(\beta Q_i^t\left(s, a_i', \bar{a}_i^*\right)\right)} \tag{9}$$

As we all know, training the Q-value function converging to optimal value is one of the keys in RL. In the dynamic environment, the system state depends on the executing action. Hence the Q-value function is difficult to train due to the fact that the state space is hard to be covered especially when the system is huge. This paper proposes a state extension training method for an especially kind RL in which the state and the corresponding reward meets the following condition:

$$r\left(a_k \mid s_j\right) \geq r\left(a_k \mid s_{j-1}\right) \tag{10}$$

where $a_k \in \mathcal{A}$, $s_j, s_{j-1} \in \mathcal{S}$, s_j is the next state of s_{j-1}. Equation (10)) means the reward function is a monotonous regarding to the same action. If the current state is s_j, the following state extension training method can accelerate the Q-value function training.

$$\begin{cases} \coprod\limits_{k=j+1,j+2,\ldots,L} M(k), & \text{if } r_i > 0 \\ \coprod\limits_{k=1,2,\ldots,j} M(k), & \text{if } r_i \leq 0 \end{cases} \tag{11}$$

where $M(k)$ is the learning mechanism

$$Q_i^{t+1}(s_j, a_i, \bar{a}_i^*) = (1-\alpha)Q_i^t(s_j, a_i, \bar{a}_i^*) + \alpha\left(r_i + \gamma \mathbb{E}_{\bar{\pi}_i}\mathbb{E}_{\pi_i^t} Q_i^t(s', a_i, \bar{a}_i^*)\right) \tag{12}$$

and $\coprod\limits_k M(k)$ means $M(k)$ execute the learning process.

3.3. Nash Equilibrium Policy Based on the OEAN

In the multi-agent system, the agents learn action value function Q^t converging to the optimal policy. In the learning process, the policy is denoted by $\pi^t = [\pi_1^t, \ldots, \pi_N^t]$. If the agent obtains its optimal policy non-cooperatively, the multi-agent system could achieve Nash equilibrium [46].

The Nash equilibrium policy of the multi-agent system is denoted by π^*. Actually, the policy is produced by Q, hence the Nash equilibrium of Q-value function Q^* is equivalent to the π^*.

For agent i, generally, the global Nash equilibrium policy is defined as follows:

$$\mathbb{E}_{\pi_{-i}^*}\mathbb{E}_{\pi_i^*} Q_i(s, a_i) \geq \mathbb{E}_{\pi_{-i}}\mathbb{E}_{\pi_i} Q_i(s, a_i) \tag{13}$$

and the Nash equilibrium saddle of the policy is defined as Equation (14)

$$\begin{aligned} \mathbb{E}_{\pi_{-i}^*}\mathbb{E}_{\pi_i^*} Q_i(s, a_i) &\geq \mathbb{E}_{\pi_{-i}^*}\mathbb{E}_{\pi_i} Q_i(s, a_i) \\ \mathbb{E}_{\pi_{-i}^*}\mathbb{E}_{\pi_i^*} Q_i(s, a_i) &\leq \mathbb{E}_{\pi_{-i}}\mathbb{E}_{\pi_i^*} Q_i(s, a_i) \end{aligned} \tag{14}$$

where $\pi_{-i} = [\pi_1, \ldots, \pi_{i-1}, \pi_{i+1}, \pi_N]$.

The OEAN of the current agent means the neighbourhood agents obtain the optimal policy; hence, we defined the Nash equilibrium policy based on the OEAN as follows:

$$\mathbb{E}_{\bar{\pi}_i}\mathbb{E}_{\pi_i^*} Q_i(s, a_i, \bar{a}_i^*) \geq \mathbb{E}_{\bar{\pi}_i}\mathbb{E}_{\pi_i} Q_i(s, a_i, \bar{a}_i^*) \tag{15}$$

We should note that $\bar{\pi}_i$ is an equivalent policy of the agents in neighbourhood, and the neighbor agents are supposed to obtain the optimal policy. Because $\bar{\pi}_i$ is an optimal equivalent policy, hence based on OEAN, the agent just has the global Nash equilibrium policy.

The proposed learning mechanism of $Q_i(s, a_i, \bar{a}_i^*)$ is shown in Equation (8). It can converge to the Nash equilibrium defined above. The convergence proof will be given in the next section.

4. OEAN Based on HMRF

4.1. HMRF for Multi-Agent System

According to the defined random field and MRF for multi-agent system in Section 3.1, the input states of the system $s = \{s_1, \ldots, s_N\}$ can be considered to be the observable random variables, and the output $a = \{a_1, .., a_N\}$ can be regarded as the latent random variables. Hence, we employ the *hidden Markov random field* (HMRF) [47] to estimate the optimal action of agent in the neighbourhood.

Suppose that the conditional probability distribution of each state $p(s_i \mid a_i)$ follows the same function $f(s;\theta_{a_i})$, i.e.,

$$p(s_i \mid a_i) = f(s;\theta_{a_i}) \tag{16}$$

This paper assumes that each element in the random fields of the state and the corresponding action is conditional independent, therefore we can obtain the following result:

$$p(s \mid a) = \prod_i p(s_i \mid a_i) \tag{17}$$

Define the neighborhood set of agent i is N_i, $i \notin N_i$ and the following result can be yielded [44]

$$\begin{aligned} p(s_i, a_i \mid N_i) &= p(s_i \mid a_i, N_i)p(a_i \mid N_i) \\ &= p(s_i \mid a_i)p(a_i \mid N_i) \end{aligned} \tag{18}$$

The marginal probability distribution of s_i under the condition of N_i is

$$\begin{aligned} p(s_i \mid N_i;\theta_{a_i}) &= \sum_{a_i \in \mathcal{A}_i} p(s_i, a_i \mid N_i) \\ &= \sum_{a_i \in \mathcal{A}_i} f(s;\theta_{a_i})p(a_i \mid N_i) \end{aligned} \tag{19}$$

where the prior probability $p(a_i \mid N_i)$ could be obtained as Equation (9), hence the HMRF model can be rewritten as follows:

$$p(s_i \mid N_i;\theta_{a_i}) = \sum_{a_i \in \mathcal{A}_i} f(s;\theta_{a_i}) \frac{\exp\left(\beta Q_i^t(s, a_i, \bar{a}_i^*)\right)}{\sum_{a_i' \in \mathcal{A}_i} \exp\left(\beta Q_i^t(s, a_i', \bar{a}_i^*)\right)} \tag{20}$$

4.2. Optimal Equivalent Action of the Neighborhood

Based on the above definition of HMRF, the optimal action of each agent denote by $\hat{a}$ can be estimated by *maximum a posterior* (MAP) estimation as follows:

$$\begin{aligned} &\hat{a} = \arg\max_a p(s \mid a)p(a) = \\ &\arg\max_{a_1,\ldots,a_N} \prod_i f(s \mid a_i, N_i) \frac{\exp\left(\beta Q_i^t(s, a_i, \bar{a}_i^*)\right)}{\sum_{a_i' \in \mathcal{A}_i} \exp\left(\beta Q_i^t(s, a_i', \bar{a}_i^*)\right)} \end{aligned} \tag{21}$$

The above problem can be considered to be maximizing the likelihood function as follows:

$$\begin{aligned} &L_2(\theta) = \\ &-\log\left(\prod_i f(s \mid a_i, N_i) \frac{\exp\left(\beta Q_i^t(s, a_i, \bar{a}_i^*)\right)}{\sum_{a_i' \in \mathcal{A}_i} \exp\left(\beta Q_i^t(s, a_i', \bar{a}_i^*)\right)}\right) \\ &= \sum_i \log f(s \mid a_i, N_i) + \beta Q_i^t(s, a_i, \bar{a}_i^*) + \log Z \end{aligned} \tag{22}$$

where $Z = \log\left(\sum_{a_i' \in \mathcal{A}_i} \exp\left(\beta Q_i^t(s, a_i', \bar{a}_i^*)\right)\right)$ is a constant respected to a_i. Hence the MAP can be transformed as

$$\hat{a} = \arg\min_{a_i,\ldots,a_N} \sum_i \left(-\log f(s \mid a_i, N_i) - \beta Q_i^t(s, a_i, \bar{a}_i^*)\right) \tag{23}$$

If the parameters of $f(s \mid a_i, N_i)$ are unknown, the *expectation maximization* (EM) algorithm [48] can be used to learn the parameters. It should be noted that Equation (24)

is a mathematical method to obtain MAP probability; however, in practice, the general approach is the iteration method [49] as follows: Suppose

$$T_\phi = \sum_i \left(-\log f(s \mid a_i, N_i) - \beta Q_i^t(s, a_i, \bar{a}_i^*)\right) \tag{24}$$

The convergence condition of the iteration is

$$T_\phi^{m+1} - T_\phi^{m+1} < \zeta \tag{25}$$

where ζ is a constant, m is the iteration counter. In Equation (23), the last term of the parentheses is derived from the prior probability, it is could be considered to be a constant in the certain iteration steps, hence the OEAN of agent i can be formulized as

$$\bar{a_i}^* = \bar{\pi}_i(\bar{a}_i^* \mid s) = \frac{1}{\lfloor N_i \rfloor} \sum_{k \in N_i} \hat{a}_k \tag{26}$$

Based on HMRF and OEAN, we can obtain the following lemma.

Lemma 2. *For agent i, the OEAN $\bar{a}_i^*$ obtained by Equation (26), such that $Q_i(s, a)$ can be approximated to $Q_i\left(s, a_i, \bar{a}_i^*\right)$.*

Proof. According to Equation (26), $\hat{a}_k$ is obtained by Equation (23), hence $\hat{a}_k \in \mathcal{A}_k \subseteq \mathcal{A}$, and $\hat{a}^k$ belongs to the domain of Q-value function at the right hand of Equation (5). Therefore, based on Lemma 1, $Q_i(s, a)$ can be expanded into Taylor series at the point $\left(s, a_i, \bar{a}_i^*\right)$, and $Q_i(s, a)$ can be approximated to $Q_i\left(s, a_i, \bar{a}_i^*\right)$. □

4.3. Convergence Proof

The action value function for iteration is denoted by $Q^t = \left[Q_1^t, \ldots, Q_N^t\right]$, and the Nash equilibrium defined in Section 3.3 is denoted by $Q^* = \left[Q_1^*, \ldots, Q_N^*\right]$. Furthermore, the following assumptions should be held:

Assumption 1. *The learning rate α^t in Equation (7) is time variable, $0 \leq \alpha^t(s, a) \leq 1$ meets the following condition [50]*

(1) $\sum_{t=0}^{\infty} \alpha^t(s, a) = \infty$, $\sum_{t=0}^{\infty} \left(\alpha^t(s, a)\right)^2 = \infty$ holds uniformly with probability 1.
(2) $\alpha^t(s, a) = 0$ if $(s, a) \neq \left(s^t, a^t\right)$.

The Q converges to the optimal value by iteration, hence there is a value space of Q-value, denote by $\mathcal{Q}$, for agent i, $Q_i \in \mathcal{Q}$. According to references [29,50], we can obtain the following lemma

Lemma 3 ([29,50])**.** *Define the following iteration*

$$Q^{t+1} = \left(1 - \alpha^t\right) Q^t + \alpha^t \Psi\left(Q^t\right) \tag{27}$$

If the learning rate α meets Assumption 1, for all $Q \in \mathcal{Q}$, the mapping $\Psi : \mathcal{Q} \to \mathcal{Q}$ meets the following condition:

(1) There exists a constant $0 < \eta < 1$ and a sequence $\lambda^t \geq 0$ converging to zero with probability 1;
(2)$\|\Psi\left(Q^t\right) - \Psi(Q^*)\| \leq \eta \|Q^t - Q^*\| + \lambda^t$
(3) $Q^* = \mathbb{E}(\Psi(Q^*))$
Such that the iteration of Equation (27) converges to Q^* with probability 1.

Based on the above assumptions and the Lemma 3, the convergence of Equation (7) can be guaranteed by the following theorem.

Theorem 1. *If* $Q^* = \left[Q_1^*, \ldots, Q_N^*\right]$ *is a global Nash equilibrium, and Q-value of the multi-agent is updated by Equation* (7) *with the Assumption 1, such that* $Q^t = \left[Q_1^t, \ldots, Q_N^t\right]$ *converges to the Nash equilibrium Q-value* $Q^* = \left[Q_1^*, \ldots, Q_N^*\right]$.

Proof. According to Lemma 3 and Equation (7), the mapping Ψ for agent i can be formalized as follows:

$$\Psi\left(Q_i^t\right) = r_i(s, a_i, \bar{a}_i^*) + \gamma \mathbb{E}_{\bar{\pi}_i} \mathbb{E}_{\pi_i^t} Q_i^t\left(s', a_i, \bar{a}_i^*\right) \tag{28}$$

Now we need to proof $\Psi\left(Q_i^t\right)$ meet the condition of Lemma 3.

(1) According Equation (2), the Nash equilibrium Q-value meet the following equation

$$\begin{aligned} Q_i^* &= r_i(s, a_i, \bar{a}_i^*) + \gamma \mathbb{E}_{s' \sim p}\left(v_i^{\pi^*}\left(s'\right)\right) \\ &= r_i(s, a_i, \bar{a}_i^*) + \gamma \sum_{s' \in \mathcal{S}} p_{s,s'}^{a_i, \bar{a}_i^*} v_i^{\pi^*}\left(s'\right) \\ &= \sum_{s' \in \mathcal{S}} p_{s,s'}^{a_i, \bar{a}_i}\left(r_i(s, a_i, \bar{a}_i^*) + \gamma \mathbb{E}_{\bar{\pi}_i} \mathbb{E}_{\pi_i^*} Q_i^*\left(s', a_i, \bar{a}_i^*\right)\right) \\ &= \sum_{s' \in \mathcal{S}} p_{s,s'}^{a_i, \bar{a}_i} \Psi(Q^*) \\ &= \mathbb{E}(\Psi(Q_i^*)) \end{aligned} \tag{29}$$

Hence the condition (3) in Lemma 3 is meted by iterating of Equation (7).

(2) To prove $\Psi\left(Q^t\right)$ meets the condition 2 of Lemma 3, we should define the following metric operator at first

$$\begin{aligned} &\|Q^t - Q^*\| = \max_i \left\{\|Q_i^t - Q_i^*\|\right\} \\ &= \max_i \left\{\max_s \left\{\|Q_i^t(s) - Q_i^*(s)\|\right\}\right\} \\ &= \max_i \left\{\max_s \left\{\max_{a_i, \bar{a}_i^*} \left\{|Q_i^t(s, a_i, \bar{a}_i^*) - Q_i^*(s, a_i, \bar{a}_i^*)|\right\}\right\}\right\} \end{aligned} \tag{30}$$

It should be noted that Q is a tensor and the dimension could be write as $N \times L \times D \times D$, N is the number of the agent, L and D are, respectively, the dimension of state space and action space. The metric operator of Equation (30) is to define a distance between Q and Q^*. Hence, we can obtain the following deduction on current step

$$\begin{aligned} &\|\Psi\left(Q^t\right) - \Psi(Q^*)\| \\ &= \max_i \|\Psi\left(Q_i^t\left(s', a_i, \bar{a}_i^*\right)\right) - \Psi\left(Q_i^*\left(s', a_i, \bar{a}_i^*\right)\right)\| \\ &= \max_i \max_{s'} |\Psi\left(Q_i^t\left(s', a_i, \bar{a}_i^*\right)\right) - \Psi\left(Q_i^*\left(s', a_i, \bar{a}_i^*\right)\right)| \\ &= \gamma \max_i \max_{s'} |\mathbb{E}_{\bar{\pi}_i} \mathbb{E}_{\pi_i^t} Q_i^t\left(s', a_i, \bar{a}_i^*\right) - \mathbb{E}_{\bar{\pi}_i} \mathbb{E}_{\pi_i^*} Q_i^*\left(s', a_i, \bar{a}_i^*\right)| \end{aligned} \tag{31}$$

Because of the definition of theNash equilibrium policy by Equation (15), we can obtain:

$$\begin{aligned} &|\mathbb{E}_{\bar{\pi}_i} \mathbb{E}_{\pi_i^t} Q_i^t\left(s', a_i, \bar{a}_i^*\right) - \mathbb{E}_{\bar{\pi}_i} \mathbb{E}_{\pi_i^*} Q_i^*\left(s', a_i, \bar{a}_i^*\right)| \\ \leq &|\mathbb{E}_{\bar{\pi}_i} \mathbb{E}_{\pi_i^*} Q_i^t\left(s', a_i, \bar{a}_i^*\right) - \mathbb{E}_{\bar{\pi}_i} \mathbb{E}_{\pi_i^*} Q_i^*\left(s', a_i, \bar{a}_i^*\right)| \end{aligned} \tag{32}$$

Equation (31) can be rewritten as follows:

$$\begin{aligned} &\|\Psi(Q^t) - \Psi(Q^*)\| \\ \leq &\gamma \max_i \max_{s'} |\mathbb{E}_{\bar{\pi}_i} \mathbb{E}_{\pi_i^*} \left(Q_i^t(s', a_i, \bar{a}_i^*) - Q_i^*(s', a_i, \bar{a}_i^*)\right)| \\ \leq &\gamma \max_i \max_{s'} \max_{a_i, \bar{a}_i^*} |\left(Q_i^t(s', a_i, \bar{a}_i^*) - Q_i^*(s', a_i, \bar{a}_i^*)\right)| \\ = &\gamma \|Q^t - Q^*\| \end{aligned} \tag{33}$$

Since $\gamma \in (0, 1)$, hence the mapping Ψ meets the condition (2) in Lemma 3. That means the Q-value update mechanism Equation (7) could make the Q-value converging to the Nash equilibrium value $Q^* = [Q_1^*, \ldots, Q_N^*]$. □

5. Top-Coal Caving Experiment

5.1. Top-Coal Caving Simulation Platform

Coal is one of the most important energy sources at present. Currently, top-coal caving is the most efficient method for mining the thick coal seam underground, as shown in Figure 1, the hundreds of *hydraulic supports* (HSs) are the key equipment for roof supporting and top-coal mining.

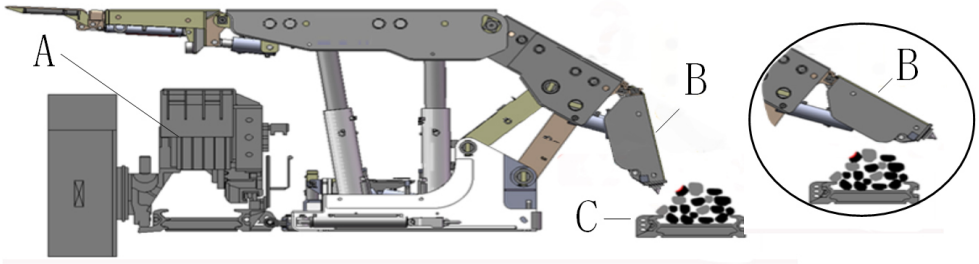
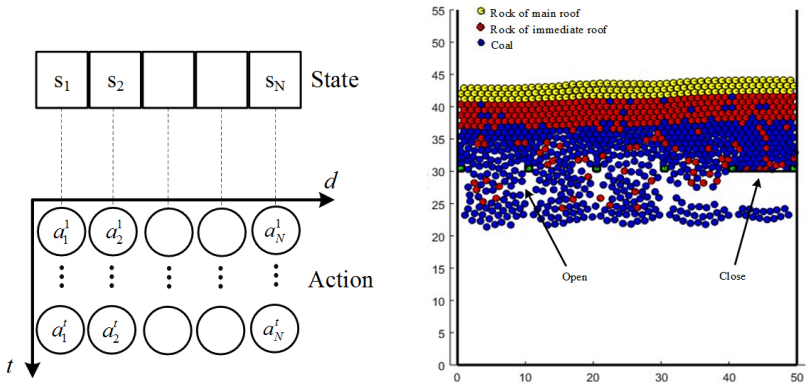


Figure 1. Top-coal caving process. A: Shearer, B: Tail boom of a hydraulic support, it acts as a window open and close. C: Drag conveyor. When the window is opened, the top-coal will collapse and be captured by the drag conveyor. In the sub-graph, the window is closed to prevent the rock falling down.

The top-coal mining sequence functions as follows: the shearer cuts coal in the coal wall, then the tail boom of HSs opens to captures the falling coal. In this process, the tail boom action is the key of HSs to exploit the maximum top-coal with minimum rocks. Hence, the tail boom acts as a window, it will open when the top-coal falling, while they will close to prevent the rock falling into the drag conveyor if all the top-coal has been captured [47,51,52]. However, it is hard to obtain a perfect performance by operating HSs individually [52]. Hence, considering the window of HSs as a multi-agent system is a direct choice to improve the top-coal mining performance.

In this paper, we employ the simulation platform developed by ourselves based on DICE [53] to validate the proposed method. The DICE system is an open source system to simulate a complicated dynamic process and interaction of discrete elements [54]. Our code [47] can be found in github (https://github.com/YangYi-HPU/Reinforcement-learning-simulation-environment-for-top-coal-caving accessed on 2 December 2019).

The Markov process of top-coal caving is shown in Figure 2a. Based on the Markov model, the top-coal caving dynamic is shown in Figure 2b. In this platform, there are five windows opening and closing to obtain and prevent the particles falling. The particles above the windows consist of three kinds: coal, rock from the immediate-roof, and rock from the main-roof [47].



(**a**) Simulation process (**b**) Sectional view of the workspace

Figure 2. simulation platform of top-coal caving. In (**a**), The s_i, a_i^t, respectively, denote the state and action of agent i on the time point t . In (**b**), a number of rock and coal particles distribute randomly in the boundary of rock and coal. Our aim is to obtain the maximum coal with minimum rock by opening and closing the windows.

5.2. Top-Coal Caving Decision Experiment Based on OEAN

In this experiment, the action space of HSs is set as $\mathcal{A} = \{\widetilde{a}_1, \widetilde{a}_2\}$, $\widetilde{a}_1$ and $\widetilde{a}_2$, respectively, which denote the opened and closed state of the windows. The condition of agent i by $\vec{s}_i$, $i = \{1, 2 \ldots, 5\}$, if the coal ration near the window is great than 0.5, set $\vec{s}_i = 1$, otherwise $\vec{s}_i = 0$. Hence, we define the state space of the multi-agent system as $\mathcal{S} = \{s_1, \ldots, s_{32}\}$, shown in Table 1.

Table 1. State space of the top-coal caving.

State	$\vec{s}_5$	$\vec{s}_4$	$\vec{s}_3$	$\vec{s}_2$	$\vec{s}_1$	State	$\vec{s}_5$	$\vec{s}_4$	$\vec{s}_3$	$\vec{s}_2$	$\vec{s}_1$
s_1	0	0	0	0	0	s_{17}	0	0	1	1	1
s_2	0	0	0	0	1	s_{18}	0	1	1	1	0
s_3	0	0	0	1	0	s_{19}	1	1	1	0	0
s_4	0	0	1	0	0	s_{20}	0	1	0	1	1
s_5	0	1	0	0	0	s_{21}	1	0	1	1	0
s_6	1	0	0	0	0	s_{22}	1	0	0	1	1
s_7	0	0	0	1	1	s_{23}	0	1	1	0	1
s_8	0	0	1	1	0	s_{24}	1	1	0	1	0
s_9	0	1	1	0	0	s_{25}	1	1	0	0	1
s_{10}	1	1	0	0	0	s_{26}	1	0	1	0	1
s_{11}	0	0	1	0	1	s_{27}	0	1	1	1	1
s_{12}	0	1	0	1	0	s_{28}	1	1	1	1	0
s_{13}	1	0	1	0	0	s_{29}	1	1	1	0	1
s_{14}	0	1	0	0	1	s_{30}	1	1	0	1	1
s_{15}	1	0	0	1	0	s_{31}	1	0	1	1	1
s_{16}	1	0	0	0	1	s_{32}	1	1	1	1	1

According to the states defined above, the function of HMRF shown in Equation (20) is given as follows:

$$f(s \mid a_i, N_i) = \exp\left(-\sigma_{a_i} \frac{(s - N^{\mu_{a_i}})}{\omega}\right) \tag{34}$$

where ω is a positive constant; μ_{a_i}, σ_{a_i} are variables and formalized as follows:

$$\mu_{a_i} = \begin{cases} 1, a_i = \widetilde{a}_1 \\ 0, a_i = \widetilde{a}_2 \end{cases} \tag{35}$$

$$\sigma_{a_i} = \begin{cases} -1, a_i = \widetilde{a}_1 \\ 1, a_i = \widetilde{a}_2 \end{cases} \tag{36}$$

By Equation (23), the optimal action is

$$\hat{a} = \arg\min_{a_i,\ldots,a_N} \sum_i \left(\sigma_{a_i} \frac{(s - N^{\mu_{a_i}})}{\omega} - \beta Q_i^t(s, a_i, \bar{a}_i^*) \right) \tag{37}$$

and the OEAN can be calculated by Equation (26).

The reward architecture for reinforcement learning is

$$R = n_r r_r + n_c r_c - \tau \tag{38}$$

where n_r is the obtained rock number, r_r is the reward of each rock; n_c is the obtained coal number, r_c is the reward of each coal, τ is the time constant. The performance indices of top-coal caving experiment are focus on *total reward* (TR), shown as Equation (38), *coal recall* (CR) and the *rock ratio in all mining* (RR) [47].

$$\begin{aligned} CR &= \frac{n_c}{n_{c\_total}} \\ RR &= \frac{n_r}{n_c + n_r} \end{aligned} \tag{39}$$

where n_{c_total} is the number of all coal particle. In this experiment, we set the parameters as $r_r = -3$, $r_c = 1$, $\alpha = 0.2$, $\beta = 0.1$, $\omega = 1$, $\zeta = 0.0001$.

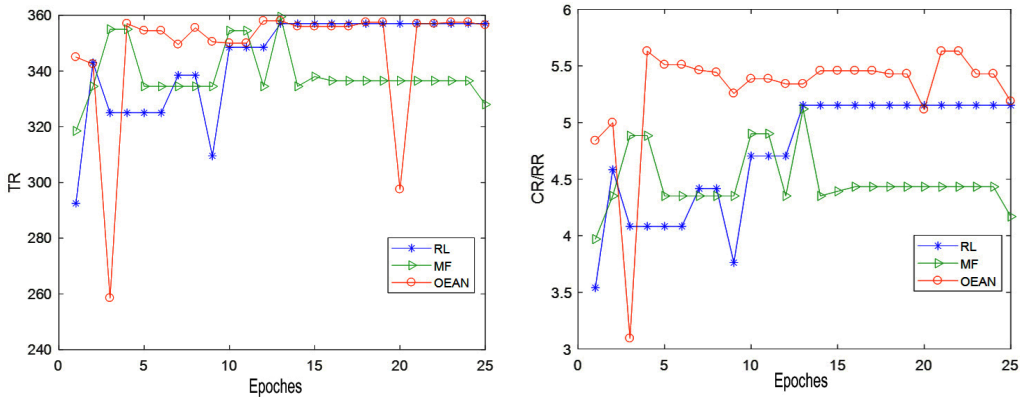
5.3. Experiment Result Analysis

The comparative experiments are carried out in this section. The three methods of multi-agent controlling are employed: independent RL [25], RL with mean field theory [24] and the method proposed in this paper. In the following sections, they are denoted by RL, MF, and OEAN, respectively.

The training and testing processes are alternatively carried out during the model learning. To make the states of each step covering the state space as much as possible, the location of rock particle in the coal layer is set as random.

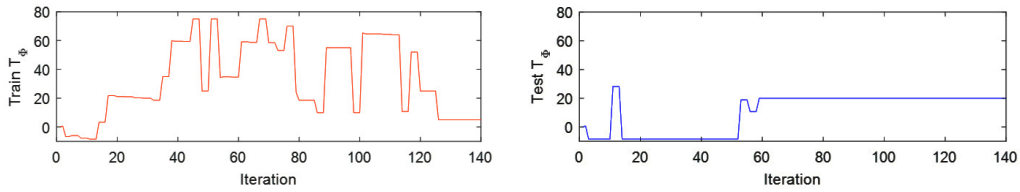
The TR of test during the model learning are shown in Figure 3a. As we can find out that the TRs increase with the learning process. The RL and OEAN can obtain the highest TR after 15 epochs learning. The OEAN swaying at the end, especially, in 20 epoch, there is a singular point. Although the imperfection of the swaying, the OEAN can obtain the perfect performance in top-coal caving, shown in Figure 3b, and the RR is lower with greater CR.

The dynamic process of MAP obtaining optimal decision is shown in Figure 4. We should note that the environment is changed during the top-coal mining, and T_Φ converges to the minimum. It is obvious that in the training process, the action decision is produced by ϵ-greedy algorithm, and most of them is random, hence it makes the frequent changes in dynamic process of T_Φ. While in the test process, the action is optimal decision by the agent, hence there are few changes in the dynamic process of T_Φ.



(**a**) Simulation process (**b**) Sectional view of the workspace

Figure 3. Performance indies of test process during the learning process. (**a**) is the total reward of the three methods. (**b**) is the ration of the ratio of CR to RR. As we can find out, the convergent tendency of the three methods is obvious in the two sub-figures. The TR is increased with the training process of the three method, and HMRF obtain best TR and the performance of the top-coal caving.



(**a**) AMP iteration in training (**b**) AMP iteration in the test

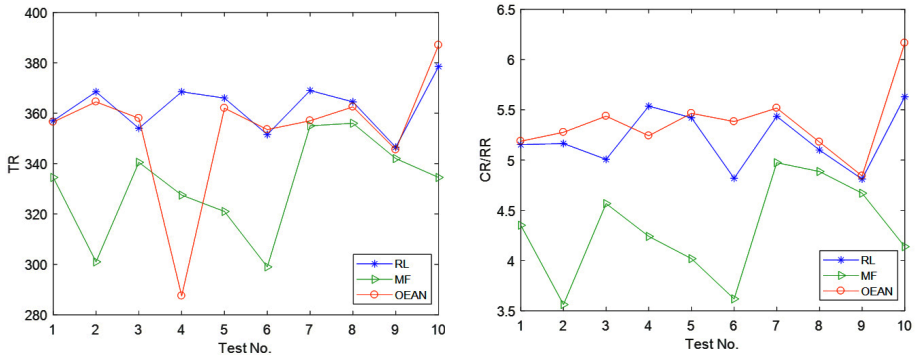
Figure 4. T_Φ dynamic during AMP iterating. It indicts the process of optimal decision making for the neighbourhood by MAP. If the change rate of T_Φ closes to zero, MAP obtains the OEAN.

After completing the training of Q-value, 10 tests are carried out for validating the method effect. In the tests, the rock particles location in the coal layer are given randomly. The performance indices are shown in Table 2 and Figure 5.

Table 2. Performance index of top-coal caving.

No.	CR			RR		
	RL	MF	OEAN	RL	MF	OEAN
1	0.90	0.94	0.91	0.17	0.23	0.18
2	0.94	0.96	0.92	0.18	0.27	0.17
3	0.91	0.92	0.89	0.18	0.19	0.16
4	0.90	0.93	0.66	0.16	0.19	0.13
5	0.93	0.95	0.91	0.17	0.24	0.17
6	0.92	0.94	0.89	0.19	0.23	0.17
7	0.93	0.92	0.89	0.17	0.17	0.16
8	0.94	0.94	0.92	0.18	0.19	0.18
9	0.92	0.93	0.92	0.19	0.2	0.19
10	0.93	0.94	0.93	0.17	0.19	0.15

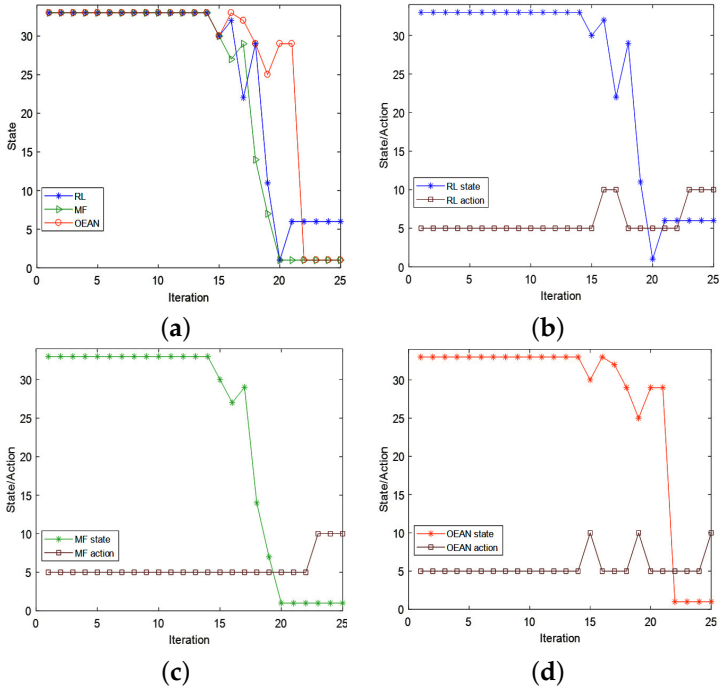
According to Figure 5a the TR of RL and OEAN can approach a high level, and in Figure 5b, the rate of CR to RR shows that OEAN could obtain the most performance of top-coal caving. Especially in the Tests No. 1, 3, 6, 9, and 10, the total reward and rate of CR to RR are the best if the OEAN is employed. That indicates in those tests, the OEAN method could obtain the global optimal decision. In the other tests of adopting the OEAN, although the total reward cannot approach the optimum level, the rate of CR to RR is the best. That means the OEAN method can obtain a better pose between the coal recall and rock ratio in the top-coal caving.



(**a**) Total reward (**b**) Rate of CR/RR

Figure 5. Tests result with the random location of the rock particles in the coal layer. (**a**) is the total reward of the three methods. (**b**) is the rate of CR to RR.

To analyze the details of optimal decision making, we chose the middle window in Test 10 to show the states and actions in the complete process of top-coal caving. The results are shown in Figure 6. Before the 15th iteration, the actions and states of the three methods are the same. Subsequently, the shearer approached the boundary of coal layer and rock layer. In this stage, the RL and MF method just close the window, while the OEAN method regulates the window switching close and open. Therefore the system state changes slowly, and the agent can obtain better performance of the top-coal caving.



(**a**) (**b**) (**c**) (**d**)

Figure 6. Tests result with the random location of the rock particles in the coal layer. (**a**) is the states of three methods. (**b**) is the states and actions of RL. (**c**) is the states and actions of MF. (**d**) is the states and actions of OEAN.

6. Conclusions

This paper proposes a new action value function of reinforcement learning-based OEAN for multi-agent system to approach the optimal decision. The new Q-value function contains the relationship between the current agent and its OEAN, and the proposed OEAN makes the communication between the agents simple and direct. The effectiveness of this method is validated by a case study of top-coal caving. The experiment results show that our method improves the training process of RL and obtains a better reward compared

with the other two methods. For the top-coal caving, our method can decrease the rock ratio in the mining and relieves the conflict between RR and CR.

In future, we will extend to the following two research topics to improve the performance of top-coal caving.

(1) The agent state of this paper depicts the global environment, hence the dimension of state space is great. If the number of multi-agents is huge, there needs to be a vast state space to describe the environment more detail. In the future, the local state of environment will be researched to decrease the state space dimension and improve the performance of the RL for multi-agent.

(2) The relationship between current agent and its neighbors is depicted by the HMRF model, and an explicit formula is used to establish the HMRF model. Hence, the generalization of the HMRF model is imperfect. In future work, the graph network based on OEAN will be employed to depict the relationship of multi-agent.

Author Contributions: Conceptualization, Y.Y.; methodology, H.W.; validation, Z.L. and T.W.; writing—original draft preparation, Y.Y.; writing—review and editing, Z.L., T.W.; supervision, H.W.; project administration, H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the national key research and development program of China grant number 2018YFC0604500; Henan Province Scientific and Technological Project of China grant number 212102210390, 222102210230.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Qian, W.; Xing, W.; Fei, S. H_∞ infinity state estimation for neural networks with general activation function and mixed time-varying delays. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 3909–3918. [CrossRef] [PubMed]
2. Ren, Y.; Zhao, Z.; Zhang, C.; Yang, Q.; Hong, K.S. Adaptive neural-network boundary control for a flexible manipulator with input constraints and model uncertainties. *IEEE Trans. Cybern.* **2020**, *51*, 4796–4807. [CrossRef] [PubMed]
3. Liu, C.; Wen, G.; Zhao, Z.; Sedaghati, R. Neural-network-based sliding-mode control of an uncertain robot using dynamic model approximated switching gain. *IEEE Trans. Cybern.* **2020**, *51*, 2339–2346. [CrossRef]
4. Zhao, Z.; Ren, Y.; Mu, C.; Zou, T.; Hong, K.S. Adaptive neural-network-based fault-tolerant control for a flexible string with composite disturbance observer and input constraints. *IEEE Trans. Cybern.* **2021**, 1–11. [CrossRef] [PubMed]
5. Jiang, Y.; Wang, Y.; Miao, Z.; Na, J.; Zhao, Z.; Yang, C. Composite-learning-based adaptive neural control for dual-arm robots with relative motion. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *33*, 1010–1021. [CrossRef]
6. Qian, W.; Li, Y.; Chen, Y.; Liu, W. L_2-L_∞ infinity filtering for stochastic delayed systems with randomly occurring nonlinearities and sensor saturation. *Int. J. Syst. Sci.* **2020**, *51*, 2360–2377. [CrossRef]
7. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 2018.
8. Lan, X.; Liu, Y.; Zhao, Z. Cooperative control for swarming systems based on reinforcement learning in unknown dynamic environment. *Neurocomputing* **2020**, *410*, 410–418. [CrossRef]
9. Bellman, R. Dynamic programming. *Science* **1966**, *153*, 34–37. [CrossRef]
10. Luo, B.; Liu, D.; Huang, T.; Wang, D. Model-Free Optimal Tracking Control via Critic-Only Q-Learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *27*, 2134–2144. [CrossRef]
11. Qian, W.; Gao, Y.; Yang, Y. Global consensus of multiagent systems with internal delays and communication delays. *IEEE Trans. Syst. Man Cybern. Syst.* **2018**, *49*, 1961–1970. [CrossRef]
12. Wei, Q.; Kasabov, N.; Polycarpou, M.; Zeng, Z. Deep learning neural networks: Methods, systems, and applications. *Neurocomputing* **2020**, *396*, 130–132. [CrossRef]
13. Liu, Z.; Shi, J.; Zhao, X.; Zhao, Z.; Li, H.X. Adaptive Fuzzy Event-triggered Control of Aerial Refueling Hose System with Actuator Failures. *IEEE Trans. Fuzzy Syst.* **2021**, *1*. [CrossRef]
14. Qian, W.; Li, Y.; Zhao, Y.; Chen, Y. New optimal method for L_2-L_∞ infinity state estimation of delayed neural networks. *Neurocomputing* **2020**, *415*, 258–265. [CrossRef]

15. Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, O.P.; Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–8 December 2017; pp. 6379–6390.
16. Matta, M.; Cardarilli, G.C.; Di Nunzio, L.; Fazzolari, R.; Giardino, D.; Re, M.; Silvestri, F.; Spanò, S. Q-RTS: A real-time swarm intelligence based on multi-agent Q-learning. *Electron. Lett.* **2019**, *55*, 589–591. [CrossRef]
17. Sadhu, A.K.; Konar, A. Improving the speed of convergence of multi-agent Q-learning for cooperative task-planning by a robot-team. *Robot. Auton. Syst.* **2017**, *92*, 66–80. [CrossRef]
18. Ni, Z.; Paul, S. A Multistage Game in Smart Grid Security: A Reinforcement Learning Solution. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 2684–2695. [CrossRef]
19. Sun, C.; Karlsson, P.; Wu, J.; Tenenbaum, J.B.; Murphy, K. Stochastic prediction of multi-agent interactions from partial observations. *arXiv* **2019**, arXiv:1902.09641.
20. Mo, X.; Huang, Z.; Xing, Y.; Lv, C. Multi-agent trajectory prediction with heterogeneous edge-enhanced graph attention network. *IEEE Trans. Intell. Transp. Syst.* **2022**, 1–14. [CrossRef]
21. Niu, Y.; Paleja, R.; Gombolay, M. Multi-agent graph-attention communication and teaming. In Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems, Online, 3–7 May 2021; pp. 964–973.
22. Koller, D.; Friedman, N.; Bach, F. *Probabilistic Graphical Models: Principles and Techniques*; MIT Press: Cambridge, MA, USA, 2009.
23. Sadhu, A.K.; Konar, A. An Efficient Computing of Correlated Equilibrium for Cooperative Q-Learning-Based Multi-Robot Planning. *IEEE Trans. Syst. Man, Cybern. Syst.* **2018**, *8*, 2779–2794. [CrossRef]
24. Yang, Y.; Rui, L.; Li, M.; Ming, Z.; Wang, J. Mean Field Multi-Agent Reinforcement Learning. *arXiv* **2018**, arXiv:1802.05438.
25. Matignon, L.; Laurent, G.J.; Le Fort-Piat, N. Independent reinforcement learners in cooperative markov games: A survey regarding coordination problems. *Knowl. Eng. Rev.* **2012**, *27*, 1–31. [CrossRef]
26. Buşoniu, L.; Babuška, R.; De Schutter, B. Multi-agent reinforcement learning: An overview. In *Innovations in Multi-Agent Systems and Applications-1*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 183–221.
27. Shoham, Y.; Powers, R.; Grenager, T. If multi-agent learning is the answer, what is the question? *Artif. Intell.* **2007**, *171*, 365–377. [CrossRef]
28. Littman, M.L. Value-function reinforcement learning in Markov games. *Cogn. Syst. Res.* **2001**, *2*, 55–66. [CrossRef]
29. Hu, J.; Wellman, M.P. Nash Q-Learning for General-Sum Stochastic Games. *J. Mach. Learn. Res.* **2003**, *4*, 1039–1069.
30. Hu, J.; Wellman, M.P. Multiagent reinforcement learning: Theoretical framework and an algorithm. In Proceedings of the ICML, Madison, WI, USA , 24–27 July 1998; Volume 98, pp. 242–250.
31. Jaakkola, T.; Jordan, M.I.; Singh, S.P. Convergence of Stochastic Iterative Dynamic Programming Algorithms. *Neural Comput.* **1993**, *6*, 1185–1201. [CrossRef]
32. Molina, J.P.; Zolezzi, J.M.; Contreras, J.; Rudnick, H.; Reveco, M.J. Nash-Cournot Equilibria in Hydrothermal Electricity Markets. *IEEE Trans. Power Syst.* **2011**, *26*, 1089–1101. [CrossRef]
33. Yang, L.; Sun, Q.; Ma, D.; Wei, Q. Nash Q-learning based equilibrium transfer for integrated energy management game with We-Energy. *Neurocomputing* **2019**, *396*, 216–223. [CrossRef]
34. Vamvoudakis, K.G. Non-zero sum Nash Q-learning for unknown deterministic continuous-time linear systems. *Automatica* **2015**, *61*, 274–281. [CrossRef]
35. Levine, S. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv* **2018**, arXiv:1805.00909.
36. Chalkiadakis, G.; Boutilier, C. Coordination in multiagent reinforcement learning: A Bayesian approach. In Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems, Melbourne, Australia, 14–18 July 2003; pp. 709–716.
37. Teacy, W.L.; Chalkiadakis, G.; Farinelli, A.; Rogers, A.; Jennings, N.R.; McClean, S.; Parr, G. Decentralized Bayesian reinforcement learning for online agent collaboration. In Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1. International Foundation for Autonomous Agents and Multiagent Systems, Valencia, Spain, 4–8 June 2012; pp. 417–424.
38. Chalkiadakis, G. *A Bayesian Approach to Multiagent Reinforcement Learning and Coalition Formation under Uncertainty*; University of Toronto: Toronto, ON, Canda, 2007.
39. Zhang, X.; Aberdeen, D.; Vishwanathan, S. Conditional random fields for multi-agent reinforcement learning. In Proceedings of the 24th International Conference on Machine Learning, Corvalis, OR, USA, 20–24 June 2007; pp. 1143–1150.
40. Handa, H. EDA-RL: estimation of distribution algorithms for reinforcement learning problems. In Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation, Montreal, QC, Canada, 8–12 July 2009; pp. 405–412.
41. Daniel, C.; Van Hoof, H.; Peters, J.; Neumann, G. Probabilistic inference for determining options in reinforcement learning. *Mach. Learn.* **2016**, *104*, 337–357. [CrossRef]
42. Dethlefs, N.; Cuayáhuitl, H. Hierarchical reinforcement learning and hidden Markov models for task-oriented natural language generation. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers-Volume 2, Association for Computational Linguistics, Portland, OR, USA, 19–24 June 2011; pp. 654–659.
43. Sallans, B.; Hinton, G.E. Reinforcement learning with factored states and actions. *J. Mach. Learn. Res.* **2004**, *5*, 1063–1088.

44. Zhang, Y.; Brady, M.; Smith, S. Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm. *IEEE Trans. Med. Imaging* **2001**, *20*, 45–57. [CrossRef] [PubMed]
45. Chatzis, S.P.; Tsechpenakis, G. The infinite hidden Markov random field model. *IEEE Trans. Neural Networks* **2010**, *21*, 1004–1014. [CrossRef] [PubMed]
46. Littman, M.L.; Stone, P. A polynomial-time Nash equilibrium algorithm for repeated games. *Decis. Support Syst.* **2005**, *39*, 55–66. [CrossRef]
47. Yang, Y.; Lin, Z.; Li, B.; Li, X.; Cui, L.; Wang, K. Hidden Markov random field for multi-agent optimal decision in top-coal caving. *IEEE Access* **2020**, *8*, 76596–76609. [CrossRef]
48. Moon, T.K. The expectation-maximization algorithm. *IEEE Signal Process. Mag.* **1996**, *13*, 47–60. [CrossRef]
49. Wang, Q. HMRF-EM-image: Implementation of the Hidden Markov Random Field Model and its Expectation-Maximization Algorithm. *Comput. Sci.* **2012**, *94*, 222–233.
50. Szepesvári, C.; Littman, M.L. A unified analysis of value-function-based reinforcement- learning algorithms. *Neural Comput.* **1999**, *11*, 2017–2059. [CrossRef]
51. Vakili, A.; Hebblewhite, B.K. A new cavability assessment criterion for Longwall Top Coal Caving. *Int. J. Rock Mech. Min. Sci.* **2010**, *47*, 1317–1329. [CrossRef]
52. Liu, C.; Li, H.; Mitri, H. Effect of Strata Conditions on Shield Pressure and Surface Subsidence at a Longwall Top Coal Caving Working Face. *Rock Mech. Rock Eng.* **2019**, *52*, 1523–1537. [CrossRef]
53. Zhao, G. DICE2D an Open Source DEM. Available online: http://www.dembox.org/ (accessed on 15 June 2017).
54. Sapuppo, F.; Schembri, F.; Fortuna, L.; Llobera, A.; Bucolo, M. A polymeric micro-optical system for the spatial monitoring in two-phase microfluidics. *Microfluid. Nanofluidics* **2012**, *12*, 165–174. [CrossRef]