Coordination Between Individual Agents in Multi-Agent Reinforcement Learning

Yang Zhang,[1] **Qingyu Yang,**[1,2,3*] **Dou An,**[1,2,3] **Chengwei Zhang**[4]

[1]Faculty of Electronics and Information, Xi'an Jiaotong University, Xi'an, China
[2]State Key Laboratory for Manufacturing System Engineering (SKLMSE), Xi'an Jiaotong University, Xi'an, China
[3]Ministry of Education (MOE) Key Laboratory for Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an, China
[4]Dalian Maritime University, Dalian, China
zy804703098@stu.xjtu.edu.cn, yangqingyu@mail.xjtu.edu.cn, douan2017@xjtu.edu.cn, chenvy@dlmu.edu.cn

Abstract

The existing multi-agent reinforcement learning methods (MARL) for determining the coordination between agents focus on either global-level or neighborhood-level coordination between agents. However the problem of coordination between individual agents is remain to be solved. It is crucial for learning an optimal coordinated policy in unknown multi-agent environments to analyze the agent's roles and the correlation between individual agents. To this end, in this paper we propose an agent-level coordination based MARL method. Specifically, it includes two parts in our method. The first is correlation analysis between individual agents based on the Pearson, Spearman, and Kendall correlation coefficients; And the second is an agent-level coordinated training framework where the communication message between weakly correlated agents is dropped out, and a correlation based reward function is built. The proposed method is verified in four mixed cooperative-competitive environments. The experimental results show that the proposed method outperforms the state-of-the-art MARL methods and can measure the correlation between individual agents accurately.

Introduction

In recent years, the Multi-Agent Reinforcement Learning (MARL) has gained more and more attention with the development of single-agent reinforcement learning (Sutton and Barto 2018), deep learning (LeCun, Bengio, and Hinton 2015), and multi-agent systems (Wooldridge 2009). Many successful single-agent reinforcement learning methods, including the DQN (Mnih et al. 2015) and DDPG (Lillicrap et al. 2015), have been extended to the multi-agent systems. However, simple extending of the single-agent reinforcement learning methods to the multi-agents environment has faced big challenges, i.e., the coordination between agents (Hernandez-Leal, Kartal, and Taylor 2019).

Based on the coordination structure between agents, existing studies on the MARL coordination algorithms can be divided into two groups: global-level coordination based methods (Sunehag et al. 2018; Rashid et al. 2018; Son et al. 2019; Wen et al. 2020), and neighborhood-level coordination based methods (Yang et al. 2018; Ganapathi Subramanian et al. 2020). In the global-level coordination based methods, all agents share observations and actions with each other and a virtual agent is integrated to learn a centralized value function for all agents. In contrast, in the neighborhood-level coordination based methods, the coordination exists only in the neighborhood of agents, meaning that agents share their actions and observations only with their neighbors. In the training process, agents in the same neighborhood are integrated into a virtual agent.

Both global-level and neighborhood-level methods attempt to solve the problem of coordination between agents as a whole, while the coordination between individual agents is ignored. In many real scenarios, different agents play different roles in the environment, and they cannot be simply integrated into a virtual agent. For example, in soccer game, a forward player is in a cooperative relationship with his teammates, and they all have the same aim to kick the ball into the other team's goal; In addition, the forward player is in a competitive relationship with the opponent players who try to prevent him from the goal. The two teams of players cannot be analyzed as an agent because the aim of them are completely opposite. In the same team, the forward player also has different correlations with others. Namely, he has a stronger cooperative relationship with players who are mainly involved in the offense, such as centers, but a weaker cooperative relationship with others who are mainly responsible for defense, such as guards. Thus, it is important to analyze the correlation between individual agents for learning a coordinated strategy. The three coordination structures of MARL methods (global-level, neighborhood-level, and agent-level based methods) are illustrated in Figure 1.

Unfortunately, in unknown environments, the correlation between individual agents is usually not given. During the interaction with the environment, the information an agent obtains directly are the state of environment, joint action, and received rewards. Based on the change in the reward of different agents, we can analyze the correlation between individual agents preliminarily. If the rewards of two agents increase or decrease simultaneously, the two agents are much more likely in a cooperative relationship. Conversely, if the reward of one agent increases while the reward of other agent decreases, the two agents are much more likely in a competitive relationship. However, this method is too simple to identify the correlation between agents accurately.

[*]Corresponding author.

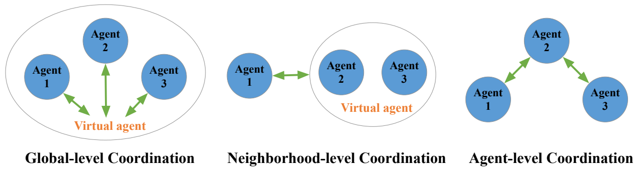


Figure 1: Three coordination structures of MARL methods. Left: The global-level based methods which integrate all agents as a virtual agent. Middle: The neighborhood-level based methods which integrate neighbors of one agent as a virtual agent. Right: The agent-level based methods which studies the coordination between each individual agent.

To tackle with above issues, we introduce the idea of correlation coefficient to measure the correlation between agents, and further present an agent-level coordinated (ALC) training framework. Extending ALC training framework with MADDPG (Lowe et al. 2017), we propose an agent-level coordination based MADDPG method (ALC-MADDPG) to learn optimal coordinated policy for agents. Specifically, our contribution is summarized as follows.

1) Three types of correlation coefficients, including Pearson (Benesty et al. 2009), Spearman (Myers and Sirois 2004) and Kendall correlation coefficient (Abdi 2007) are introduced in the MARL to measure the correlation between agents accurately. Based on different agents' reward trajectories, the correlation coefficient of each pair of agents is calculated. Next, the negative, positive, strong, and weak correlations between agents are defined.

2) Based on the correlation coefficient between agents, ALC training framework is presented. In ALC training framework, strong correlated agents learn a coordinated policy by exchanging their individual information when learning, and weak correlated agents does not communicate with each other. By discarding the communication information between agents who are weakly correlated with each other, the presented training framework solves the problem of the high dimensionality of input space of the state-action value network. In addition, a correlation based reward function is presented to overcome the problem of inefficiency of the individual reward function in learning coordinated policy.

3) Finally, the effectiveness of proposed method is verified in four mixed cooperative-competitive environments, and results show that the proposed method can achieve better performance than the state-of-the-arts MARL methods.

Related Work

Based on the framework of centralized training and decentralized execution (CTDE) (Foerster et al. 2017, 2018), the existing MARL studies about the coordination of agents can be categorized into two categories.

The first category considers the coordination problem from a global perspective and mainly studies the relationship between the virtual agent who integrates all agents and each individual agent. The mainstream methods of this category include the VDN (Sunehag et al. 2018), QMIX (Rashid et al. 2018), QTRAN (Son et al. 2019) and SMIX(λ) (Wen et al. 2020). Based on the structural constraint of different size of the hypothesis space, the VDN, QMIX, QTRAN, and SMIX(λ) methods all assume that the optimal joint action represents the set of optimal action of each of the agents. Unfortunately, the structural constraints lead to the limited scalability of these four algorithms.

The second category considers the coordination problem from a perspective of neighborhood (Yang et al. 2018; Subramanian and Mahajan 2019; Ganapathi Subramanian et al. 2020; Wang, Yang, and Wang 2020). For each agent in the multi-agent environment, his neighboring agents are integrated into an virtual agent. Based on the theory of mean field approximation, the coordination between agent and his neighboring agents is equivalent to the mean coordination between agent and his neighboring virtual agent.

Unlike global-level or neighborhood-level coordination based MARL methods, this paper aims to explore the correlation of individual agents and proposes an agent-level coordination based method. Our method mainly contains two parts: analysis of the correlation between individual agents and an agent-level coordinated training framework.

The ideas of the study presented in (Kim, Cho, and Sung 2019) and (Kim et al. 2019) are similar to ours, we all address the problem of increased input dimensionality of state-action network by discarding some agents' communication information in policy training. In (Kim, Cho, and Sung 2019), the communication message was randomly dropout. By contrast, the communication information is dropped based on the correlation type between individual agents in our work, thus the method proposed in this paper enhances the coordination of different agents in a multi-agent environment. In the proposed centralized communication schedule mechanism of (Kim et al. 2019), the message that different agents receive are same, thus the roles of different agents cannot be differentiated. In this paper, a message schedule such that each agent receives a communication message only from his own strongly correlated agents. And the correlation analysis between each agents provides a new perspective to identify the roles of agents in multi-agent environment.

Preliminaries

Partially Observable Markov Game

This paper considers a mixed cooperative-competitive multi-agent system that can be described as a partially observable Markov Game (POMG). The POMG is defined as $G =< \mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{P}, r, \gamma >$, where $\mathcal{N} = \{1, 2, \ldots, N\}$ denotes a set of agents, $s \in \mathcal{S}$ represents the state of the environment. Each agent $i \in \mathcal{N}$, at each time slot first observes the environment and draws partial observation o^i from the observation space O^i, and the joint observation of agents is $o = \{o^1, o^2, \ldots, o^N\}$. Then, agent $i \in \mathcal{N}$ executes the action a^i from the action space A^i, and the joint action space is $\mathcal{A} = \prod_{i=1}^{n} A^i$. With the joint action of agents $a = \{a^1, a^2, \ldots, a^N\}$, the environment goes to the next state s' and the state transition function is $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$. Each agent $i \in \mathcal{N}$ receives a reward r^i based on the reward function $\mathcal{S} \times \mathcal{A} \to \mathbb{R}$ and joint reward $r = \{r^1, r^2, \ldots, r^N\}$. In an episode containing T time slots, the objective of each agent i is to develop an optimal action policy π^{i^*} to maximizes the cumulative discounted reward $R_i = \sum_{t=1}^{T} \gamma^{t-1} r_t^i$ where $\gamma \in [0, 1)$ is a discounted factor.

MADDPG

Under the framework of centralized training and decentralized execution (CTDE), MADDPG extends DDPG into multi-agent environment. During the execution of policy, each actor i has the knowledge only about his own observation and executes the decentralized action based on the action policy network μ^i. During the training, each critic i shares the information on observation and action taken with other critics and learns a centralized joint state-action value $Q^i(s, a)$ with parameters θ^{Q^i}. The state-action value $Q^i(s, a)$ of agent i is updated such that to minimize the loss function given by:

$$L^i = E_{s,a,s' \sim D}\left[y^i - Q^i\left(s, a; \theta^{Q^i}\right)\right] \tag{1}$$

$$y^i = r^i + \gamma Q^{i'}\left(s', a'; \theta^{Q^{i'}}\right)|_{a^{i\prime}=\mu^{i\prime}(o^{i\prime})} \tag{2}$$

where D denotes the experience replay memory where transition tuples $\{(s, a, r, s')\}$ are stored; $\mu^{i'}$ represents the target action policy with parameters $\theta^{\mu^{i'}}$, $Q^{i'}$ is the value of target critic network of agent i with parameters $\theta^{Q^{i'}}$.

In order to maximize the cumulative reward, the action policy network μ^i is updated using the following gradient:

$$\nabla J = E_{s,a \sim D}\left[\nabla_{\theta^{\mu^i}} \mu^i\left(o^i\right) \nabla_{a^i} Q^i\left(s, a\right)|_{a^i=\mu^i(o^i)}\right] \tag{3}$$

The target networks are updated by tracking learned networks slowly as follows:

$$\theta^{\mu^{i'}} = \tau \theta^{\mu^i} + (1 - \tau)\, \theta^{\mu^{i'}} \tag{4}$$

$$\theta^{Q^{i'}} = \tau \theta^{Q^i} + (1 - \tau)\, \theta^{Q^{i'}} \tag{5}$$

where τ denotes the soft updating parameter, and $\tau \ll 1$.

Method

In this section, the proposed ALC-MADDPG method is presented in detail. First, the Pearson, Spearman, and Kendall correlation coefficients are introduced in the MARL to measure the correlation between agents accurately. Then, an agent-level coordinated (ALC) training framework is developed to train the joint policy for agents.

Correlation Between Individual Agents

In an unknown competitive-cooperative environment, initially, the agent has no knowledge about other agents and does not know which of the agents are his teammates and which are competitors. Further, in a team, the agent does not know which agents have a closer cooperative relationship with him and which have a weaker cooperative relationship with him. The unknown roles of other agents makes it very difficult for agent to learn the coordinated policy.

Based on the reward trajectory, there is a simple inference technique to analyze the correlation between agents. In a training episode of the interaction between agents and environment, a reward trajectory $r^i = \{r_1^i, r_2^i, \ldots, r_T^i\}$ for each agent i is obtained. The changes in the received rewards of agents reflect the correlation between agents to a certain extent. A simple inference is that if the rewards of two agents increase or decrease simultaneously, the two agents are more likely to be teammates with cooperative relationship. Conversely, if the reward of one agent increases while the reward of other agent decreases, the two agents are more likely in a competitive relationship. Obviously, this inference technique is too simple to identify the correlation between agents accurately. To solve this problem, in this work, we attempt to obtain the correlation between agents based on the Pearson, Spearman's, and Kendall correlation coefficients.

The Pearson correlation coefficient is the most commonly used statistical estimator to measure the correlation between variables. For two sampled data arrays $X = \{x_i, 1 \leq i \leq n\}$ and $Y = \{y_i, 1 \leq i \leq n\}$, the Pearson correlation coefficient r_p is given by:

$$r_p = \frac{\sum\limits_{i=1}^{N} \left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)}{\sqrt{\sum\limits_{i=1}^{n} \left(x_i - \bar{x}\right)^2} \sqrt{\sum\limits_{i=1}^{n} \left(y_i - \bar{y}\right)^2}} \tag{6}$$

where $\bar{x}$ and $\bar{y}$ denote the means of X and Y, respectively.

Although the Pearson correlation coefficient has great robustness, it can be used to measure only linear correlation between variables. The Spearman's rank order correlation coefficient r_s is used to measure more complex nonlinear correlation between two variables. For arrays X and Y, $X' = \{x_i', 1 \leq i \leq n\}$ and $Y' = \{y_i', 1 \leq i \leq n\}$ are ranked arrays where x_i' and y_i' denote the ranks of data in the array X and Y, respectively, the Spearman's correlation coefficient is calculated by:

$$r_s = \frac{\sum\limits_{i=1}^{N} \left({x'}_i - \bar{x}'\right)\left({y'}_i - \bar{y}'\right)}{\sqrt{\sum\limits_{i=1}^{n} \left({x_i}' - \bar{x}'\right)^2} \sqrt{\sum\limits_{i=1}^{n} \left({y_i}' - \bar{y}'\right)^2}} \tag{7}$$

where $\bar{x}'$ and $\bar{y}'$ denote the mean ranks of sampled data.

For the scenario where some values of variables are the same, spearman correlation coefficient cannot work well. To address this problem, we retort to kendall correlation coefficient τ_b. For any pair (x_i, y_i) and (x_j, y_j), there are three cases: 1) concordant pair: if ($x_i > x_j$ and $y_i > y_j$) or ($x_i < x_j$ and $y_i < y_j$); 2) discordant pair: if ($x_i > x_j$ and $y_i < y_j$) or ($x_i < x_j$ and $y_i > y_j$); 3) neither concordant nor discordant pair. We denote n_c as the number of concordant pairs, n_d as the number of discordant pairs and n_0 as the number of unordered pairs where $n_0 = 0.5n(n-1)$. Besides, n_1 denotes the number of variable with the same values in X, for instance, $x_i = x_j$. Similarly, n_2 denotes the number of variable with the same values in Y. The Kendall correlation coefficient is calculated as follows:

$$\tau_b = \frac{n_c - n_d}{\sqrt{(n_0 - n_1)}\sqrt{(n_0 - n_2)}} \tag{8}$$

The value ranges of the Pearson, Spearman's, and Kendall correlation coefficients are all $[-1,1]$. The positive value of correlation coefficient suggests that two variables X and Y have a positive correlation, i.e., when the value of X increases, the value of Y also increases; And the negative value suggests that two variables have a negative correlation, i.e., the value of X increases, the value of Y decreases. Further, the higher the absolute value of the correlation coefficient is, the stronger correlation between the two variables.

In the following, the Kendall correlation coefficient is used as an example, and the difference of the results with three correlation coefficients is shown in the experiment. For any pairwise agent (i,j), the correlation coefficient $c_{i,j}$ between agents i and j is defined as follows:

$$c_{i,j} = \frac{n_c - n_d}{\sqrt{(n_0 - n_1)}\sqrt{(n_0 - n_2)}} \tag{9}$$

where n_c, n_d, n_0, n_1, and n_2 denote the relevant values introduced in the Kendall correlation coefficient when the sampled data X and Y are replaced with the corresponding reward trajectories $r^i = \{r_1^i, r_2^i, \ldots, r_T^i\}$ and $r^j = \{r_1^j, r_2^j, \ldots, r_T^j\}$, respectively.

Based on the calculated correlation between each pair of agents, positive, negative, strong, and weak correlations between agents are defined as follows.

Definition 1: Positive correlation between agents: For any pair of agents i and j, if the correlation coefficient $c_{i,j}$ is greater than zero, then the two agents are in a positive correlation with each other.

Definition 2: Negative correlation between agents: For any pair of agents i and j, if the correlation coefficient $c_{i,j}$ is smaller than zero, then the two agents are in a negative correlation with each other.

Definition 3: Strong correlation between agents: For any pair of agents i and j, if the absolute value of correlation coefficient $c_{i,j}$ is greater than a threshold G, then the two agents are in a strong correlation with each other.

Definition 4: Weak correlation between agents: For any pair of agents i and j, if the absolute value of correlation coefficient $c_{i,j}$ is smaller than a threshold G, then the two agents are in a weak correlation with each other.

The threshold G is set as 0.5, and the performance of different values of threshold G is analyzed in the experiment.

By using the proposed method, agents can identify which of the agents are their teammates and which are opponents. Further, agents can identify which agents are their strong correlated teammates, which agents are their weak correlated teammates. In the following, based on the identified correlation between individual agents, we propose an agent-level coordinated training framework to learn optimal coordinated policy for agents.

ALC Training Framework

In the existing coordinated MARL methods, such as MADDPG, VDN, Qmix, the input space of the state-action value network increases exponentially with the number of agents. The increased input greatly increases communication cost between agents and reduces the speed of policy learning.

Algorithm 1 ALC-MADDPG algorithm

```
Initialize correlation coefficient $c_{i,j}$, replay memory $D$,
  actor network $\mu^i$ and critic network $Q^i$ for each agent $i$
for $episode = 1, 2, \ldots, E$ do
    for $t \leq T$ and not terminal do
        Each agent $i$ observes $o_t^i$
        Each agent $i$ executes action $a_t^i = \mu^i\left(o_t^i\right)$ with
          noise $N_t$
        Receive the full state of environment $s_t$, action $a_t$,
          reward $r_t$, and the next state $s_{t+1}$
        Each agent $i$ sends the communication message to
          his strongly correlated agents
        Store transition $(s_t, a_t, r_t, s_{t+1})$ in $D$
        Each agent $i$ calculates the correlated reward function $\tilde{r}^i$ using Eq. (12)
        Update networks by Eq. (13-15)
    end for
    Each pair of agents $(i,j)$ calculates $c_{i,j}$ using Eq. (9)
    for Each pair of agents $(i,j)$ do
        if $|c_{i,j}| > G$ then
            agent $i$ and $j$ are strongly correlated, $w_{i,j} = 1$
        else
            agent $i$ and $j$ are weakly correlated, $w_{i,j} = 0$
        end if
        if $c_{i,j} < 0$ then
            agent $i$ is negative correlated with agent $j$
        else
            agent $i$ is positive correlated with agent $j$
        end if
    end for
end for
```

To address this problem, Kim *et al.* proposed a message dropout method (Kim, Cho, and Sung 2019). They drop out the communication message (e.g., observation and action taken) between agents randomly to reduce the dimension of the input space of the state-action network. However, the randomness of message dropping out may cause the lost of useful communication information between agents, further makes agents unable to learn optimal policy. In order to address this problem, ALC training framework is developed in this paper. In the policy learning of agents, the communication information of agents who are weakly correlated to the agent is dropped out. And the communication information of strongly correlated agents is used to develop the reward function and then to learn the optimal coordinated policy.

For each agent i, the critic network is denoted as $Q^i(o,a) = Q^i\left(o^1, o^2, \ldots, o^n, a^1, a^2, \ldots a^n\right)$ in the original MARL method. And the message that agent i receives from all other agents is given by:

$$\boxed{m_i = (o_{-i}, a_{-i})} \tag{10}$$

where o_{-i} is the observation of all agents excepts for agent i and a_{-i} is the action of all agents excepts for agent i

In multi-agent environment, for any agent i, some agents are weakly correlated with him and the messages of these agents are not important for agent i to learn the policy.

Therefore, the information of these agents can be discarded to reduce the communication cost and improve the learning speed of joint policy. The definition of weakly correlated agents has been introduced in Definition 4. For each agent i, we define a strong-weak correlation array $w_i = \{w_{i,1}, \ldots, w_{i,i-1}, w_{i,i+1}, \ldots, w_{i,N}\}$ where $w_{i,j} = 0$ if agent j is weakly correlated with agent i, and $w_{i,j} = 1$ if agent j is strongly correlated with agent i. Thus, the new message $\tilde{m}_i$ of agent i is expressed as:

$$\boxed{\tilde{m}_i = (w_i \odot o_{-i}, w_i \odot a_{-i})} \tag{11}$$

where $\odot$ denotes the element-wise product of two arrays.

Further, the new state-action value is expressed as $\tilde{Q}^i(s,a) = Q^i\left(\tilde{m}_i, o^i, a^i\right)$.

In the multi-agent environment, another problem is the inefficiency of the individual reward function. As presented in the previous studies (Foerster et al. 2018), the individual reward function often fails to encourage agents to make sacrifices for a greater benefit of the group. This problem often makes the existing MARL method to learn the sub-optimal joint policy. In order to address this problem, a correlation based reward function is proposed.

Our aim is to make strongly correlated agents in a team learn coordinated policy in the direction of maximizing the mutual benefits and minimizing their opponents' benefits. For each pair of agent i and j, we set $\tilde{w}_{i,j} = 1$ if agent i and j are strongly and positively correlated with each other; $\tilde{w}_{i,j} = -1$ if agent i and j are strongly but negatively correlated with each other, and $\tilde{w}_{i,j} = 0$ if agent i and j are weakly correlated with each other. The correlation based reward function is defined as follows:

$$\boxed{\tilde{r}^i = r^i + \tilde{w}_i * r_{-i}} \tag{12}$$

where r_{-i} is the reward of all agents except for agent i and $\tilde{w}_i = \{\tilde{w}_{i,1}, \ldots, \tilde{w}_{i,i-1}, \tilde{w}_{i,i+1}, \ldots, \tilde{w}_{i,N}\}$.

With the correlation based reward function, the aim of each agent is not only to maximize his own cumulative reward but also to maximize the reward of strongly correlated agents in a team and minimize the reward of opponents. Each agent may slightly sacrifice his individual benefit if the benefit of the team will be increased. In order to reduce the communication cost and improve the speed of policy learning, the correlation based reward function incorporates only the individual rewards of strongly correlated agents.

Under the proposed ALC training framework, the actor network μ^i is updated as following gradient:

$$\begin{aligned} \nabla J\left(\mu^i\right) = E_{o,a,\tilde{m}\sim D}[\nabla_{\theta^{\mu^i}} \mu^i\left(o^i\right) \\ \nabla_{a^i} Q^i\left(\tilde{m}^i, o^i, a^i\right)|_{a^i=\mu^i(o^i)} \end{aligned} \tag{13}$$

The critic network Q^i is trained to minimize the loss:

$$L^i = E_{o,a,\tilde{m},\tilde{m}'\sim D}\left[y^i - Q^i\left(\tilde{m}^i, o^i, a^i; \theta^{Q^i}\right)\right] \tag{14}$$

$$y^i = r^i + \gamma Q^{i'}\left(\tilde{m^i}', o^{i'}, a^{i'}; \theta^{Q^{i'}}\right)|_{a^{i'}=\mu^{i'}(o^{i'})} \tag{15}$$

where $Q^{i'}$ and $\mu^{i'}$ are the value of target critic network, target actor network, respectively. Both target networks are soft updated according to Eq. (4-5).

Extending the presented ALC training framework with MADDPG method, we refer our proposed method as ALC-MADDPG, which is summarized in Algorithm 1.

Experiments

In this section, the experimental environment and implementation details are presented. Then the experimental results and ablation studies are given.

Evaluation Methodology

The experiments were performed in four mixed competitive-cooperative environments, which can be found in (Lowe et al. 2017; Mordatch and Abbeel 2018).

Physical Deception: This environment consists of two good agents, one adversary and two landmarks. Both agents and adversary desire to approach the landmark. The closer the agent and the landmark are, the more reward that the agent receives. Besides, if the adversary is closer to the landmark, agents will receive a negative reward.

Covert Communication: This environment consists of two good agents (Alice and Bob) and one adversary. Both agents have a random private key and learn how to use the private key to encrypt messages. Alice sends an encrypted message to Bob, and the two agents are rewarded based on how well Bob can reconstruct the received message. Besides, the two agents will receive a negative reward if the adversary reconstructs the message.

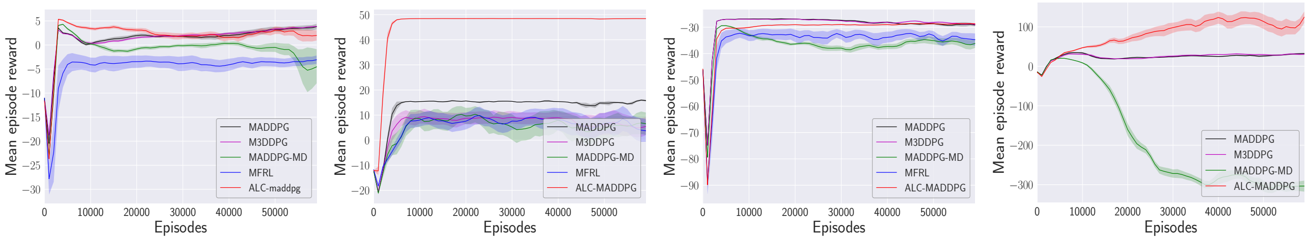
Keep-away: This environment consists of three good agents, one adversary and two landmarks. The environment setup is similar to physical deception where agents are rewarded based on the distance between agents and landmark. Differently, the adversary has the ability to physically push agents to keep away from landmarks. The adversary does not have the knowledge about the locations of landmarks and can infer this information from the movements of agents.

Complex Predator-prey: This environment consists of two good agents, four adversaries and one landmark. The environment setup is similar to physical deception. Differently, adversaries with a slow movement speed can hit agents, and agents must learn to avoid being hit. Besides, there are 'forests' where agents can hide from being seen in the environment. For the adversaries, there is a leader that can see all agents' locations which are sent to other adversaries.

The mean received reward of agents in an episode was used as the evaluation metric. Besides, the same network structure as MADDPG was used. In each actor and critic network, there were two fully-connected layers, and both of them consisted of 64 neurons. The parameters τ, γ, learning rate, and the size of replay memory were set as 0.1, 0.95, 0.01 and $10,0000$, respectively.

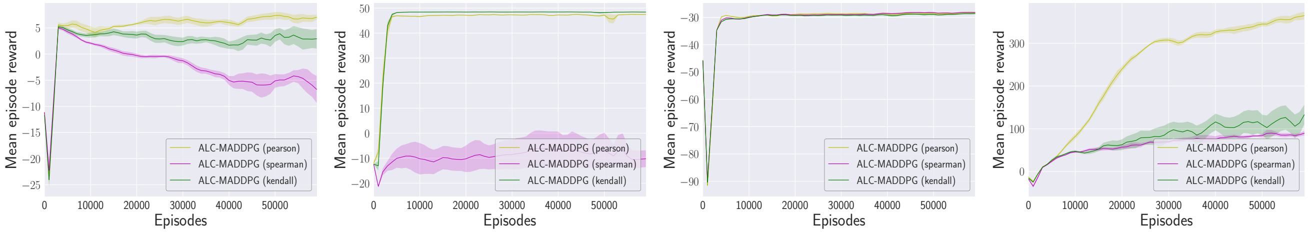
Comparison Results

The proposed ALC-MADDPG method was compared with four state-of-the-art MARL algorithm: MFRL (Yang et al. 2018), MADDPG (Lowe et al. 2017), M3DDPG (Li et al.



(a) Physical deception (b) Covert communication (c) Keep-away (d) Complex predator-prey

Figure 2: The mean received reward of agents obtained by the proposed ALC-MADDPG method and benchmarks. The solid and dashed lines show the mean and standard deviation of results over ten runs, respectively. In the complex predator-prey environment, the observation dimension is different for agents, and MFRL method cannot be applied to this environment.



(a) Physical deception (b) Covert communication (c) Keep-away (d) Complex predator-prey

Figure 3: Sensitivity of the ALC-MADDPG to selected correlation coefficient in four environments.

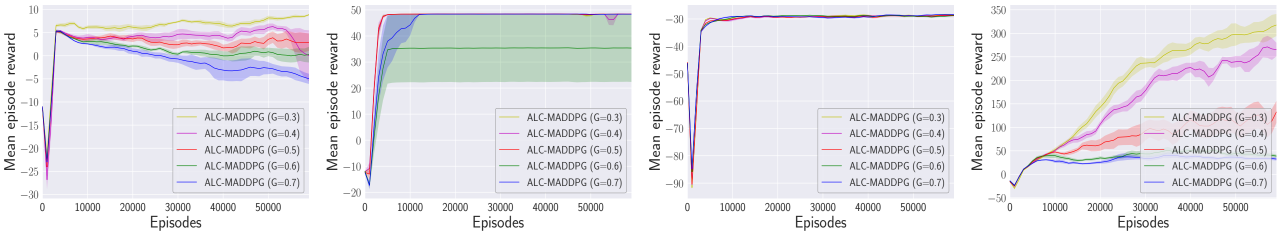
2019) and MADDPG-MD (Kim, Cho, and Sung 2019). MFRL is a popular neighborhood-level MARL method, and MADDPG is one of the most popular MARL methods. M3DDPG and MADDPG-MD are two MADDPG based algorithms that introduce the minimax theory and message dropout technique in the MADDPG, respectively.

The comparison results were shown in Figure 2. As shown in Figure 2, our proposed ALC-MADDPG method performed better than other algorithms in four experiments. Specifically, in the environment of physical deception, the proposed ALC-MADDPG, MADDPG, and M3DDPG methods performed similarly and achieved nearly 5 rewards, while other two methods achieved almost 0 rewards. In the environment of convert communication, the proposed method performed the best among all methods and gained nearly 50 rewards, while the second-best method was MADDPG method that gained nearly 15 rewards. In the environment of keep-away, MFRL method achieved the worst performance among all methods, and it gained -35 rewards. In the environment of complex predator-prey, the ALC-MADDPG achieved the best performance among all methods and gained more than 100 rewards, while the second-best methods were MADDPG and M3DDPG methods that gained nearly 30 rewards. The comparison results show that the proposed method achieved the best performance in four complex competitive-cooperative environments.

It is worth mentioning that compared to the benchmark of MADDPG and two other MADDPG-based methods, the proposed ALC-MADDPG has different training framework. Namely, the proposed method focuses on the coordination between individual agents and adds the strongly correlated agents' individual reward in the reward function. The results in Figure 2 show that the proposed method achieved better performance than the MADDPG, M3DDPG that combines the minimax theory with the MADDPG, and MADDPG-MD that combines the message dropout technique with the MADDPG. Consequently, we can conclude that the agent-level coordination is more important than other techniques for MARL methods to improve performance in complex competitive-cooperative environments.

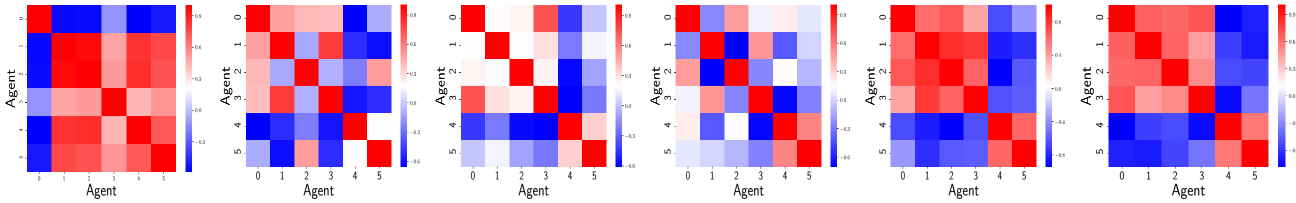
Ablation Studies

We investigated the performance of the proposed method with the Pearson, Spearman, and Kendall correlation coefficient. The results of the method with different correlation coefficients in four environments were shown in Figure 3. As presented in Figure 3, the proposed ALC-MADDPG with the Pearson correlation coefficient performed best in different environments. Though the Pearson correlation coefficient is the first proposed and the simplest among the three correlation coefficients, it has the best robustness and can be applied in most environments. By adopting the Pearson correlation coefficient, the proposed method can learn an optimal coordinated policy for agents. Besides, the method with the Spearman's rank achieved the worst performance in all four different environments. This may because the Spearman correlation coefficient is based on the rank order, which shows only the relative size of source data, but not absolute size, and the rank order cannot accurately reflect the real difference in the correlation between different agents.



(a) Physical deception (b) Covert communication (c) Keep-away (d) Complex predator-prey

Figure 4: Sensitivity of the ALC-MADDPG for different threshold G values in four environments.



(a) Episode 0 (b) Episode 10000 (c) Episode 20000 (d) Episode 30000 (e) Episode 40000 (f) Episode 50000

Figure 5: The correlation between each pair of agents in the complex predator-prey environment at training episode 0, 10000, 20000, 30000, 40000, and 50000. Each integer point on the *x*-axis and *y*-axis represents an agent. The grid color shows the correlation between a pair of agents: the red grid shows two agents have a positive correlation, and the blue grid shows two agents have a negative correlation. The darker the grid color is, the stronger correlation between this pair of agents have.

To investigate the performance of the proposed method at different values of threshold G, the ALC-MADDPG with the threshold of G = 0.3, 0.4, 0.5, 0.6, and 0.7, were tested. The results were presented in Figure 4, where it can be seen that ALC-MADDPG with $G = 0.3$ achieved the best performance in the experiments of physical deception and complex predator-prey environments, and the proposed method with $G = 0.6$ or 0.7 performed the worst in the two environments. According to the definition of hyperparameter G, it holds that when G is smaller, more agents are likely to be judged to be strongly correlated with each other, and there will be more communication messages that agents can receive from more agents. Therefore, the proposed method gained higher reward when the value of G was 0.3 than that when the value G was 0.6 or 0.7. However, more communication message leads to more communication cost. And the ALC-MADDPG with a smaller value of G did not perform better than that with higher value of G in all the environments. In the covert communication and keep-away environments, the performances of the ALC-MADDPG with different values of G were similar. In practice, a given bandwidth of a communication line can limit the volume of communication messages, and the ALC-MADDPG with a big value of G can be applied in more scenarios.

Correlation Between Agents

In this section, the correlation between agents studied in this paper was shown by the heat map. We randomly chose a training episode in the complex predator-prey environment as an example and the results were presented in Figure 5. It can be seen that at the beginning of the training, i.e., $episode = 0$, agent 0 had a negative correlation with all other agents, and agents 1–5 had a positive correlation with each other. At the end of the training, i.e., $episode = 50000$, agents 4 and 5 both had a negative correlation with agents 0–3. Also, agent 4 had a positive correlation with agent 5. Besides, we can observe that as the training process continued, the grid color became darker and darker, indicating that the correlation between agents became stronger and stronger. The experiment results are consistent with the setup of the environment where there are four adversaries (agent 0, 1, 2 and 3) and two good agents (agent 4 and 5). These results show that ALC-MADDPG can measure the correlation between agents accurately.

Conclusion

In this paper, we propose an agent-level coordination based MARL method to address the problem of coordination in multi-agent environment. Specifically, we introduce three correlation coefficients to measure the correlation of agents. Then, based on the correlation of agents, we present an agent-level coordinated training framework. The communication information of agents who are weakly correlated with each other is dropped out to reduce the communication cost, and a correlation based reward function is designed to learn the optimal coordinated policy. The experiment results show that our proposed method achieves superior performance than the state-of-the-art MARL methods. Last but not least, the results also show our method has the ability to measure the correlation between agents accurately.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 61973247, 61673315, 61906027, 61803295 and 61906135, National Postdoctoral Program for Innovative Talent under grant BX2020272, China Postdoctoral Science Foundation Funded Project under Grant 2019M661080.

References

Abdi, H. 2007. The Kendall rank correlation coefficient. *Encyclopedia of Measurement and Statistics. Sage, Thousand Oaks, CA* 508–510.

Benesty, J.; Chen, J.; Huang, Y.; and Cohen, I. 2009. Pearson correlation coefficient. In *Noise reduction in speech processing*, 1–4. Springer.

Foerster, J.; Nardelli, N.; Farquhar, G.; Afouras, T.; Torr, P. H.; Kohli, P.; and Whiteson, S. 2017. Stabilising experience replay for deep multi-agent reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 1146–1155.

Foerster, J. N.; Farquhar, G.; Afouras, T.; Nardelli, N.; and Whiteson, S. 2018. Counterfactual multi-agent policy gradients. In *Thirty-second AAAI conference on artificial intelligence*.

Ganapathi Subramanian, S.; Poupart, P.; Taylor, M. E.; and Hegde, N. 2020. Multi Type Mean Field Reinforcement Learning. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, 411–419.

Hernandez-Leal, P.; Kartal, B.; and Taylor, M. E. 2019. A survey and critique of multiagent deep reinforcement learning. *Autonomous Agents and Multi-Agent Systems* 33(6): 750–797.

Kim, D.; Moon, S.; Hostallero, D.; Kang, W. J.; Lee, T.; Son, K.; and Yi, Y. 2019. Learning to schedule communication in multi-agent reinforcement learning. *arXiv preprint arXiv:1902.01554* .

Kim, W.; Cho, M.; and Sung, Y. 2019. Message-dropout: An efficient training method for multi-agent deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 6079–6086.

LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *nature* 521(7553): 436–444.

Li, S.; Wu, Y.; Cui, X.; Dong, H.; Fang, F.; and Russell, S. 2019. Robust multi-agent reinforcement learning via minimax deep deterministic policy gradient. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 4213–4220.

Lillicrap, T. P.; Hunt, J. J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; and Wierstra, D. 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971* .

Lowe, R.; Wu, Y. I.; Tamar, A.; Harb, J.; Abbeel, O. P.; and Mordatch, I. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in neural information processing systems*, 6379–6390.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *nature* 518(7540): 529–533.

Mordatch, I.; and Abbeel, P. 2018. Emergence of grounded compositional language in multi-agent populations. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Myers, L.; and Sirois, M. J. 2004. S pearman Correlation Coefficients, Differences between. *Encyclopedia of statistical sciences* .

Rashid, T.; Samvelyan, M.; Schroeder, C.; Farquhar, G.; Foerster, J.; and Whiteson, S. 2018. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning*, 4295–4304.

Son, K.; Kim, D.; Kang, W. J.; Hostallero, D. E.; and Yi, Y. 2019. QTRAN: Learning to Factorize with Transformation for Cooperative Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning*, 5887–5896.

Subramanian, J.; and Mahajan, A. 2019. Reinforcement learning in stationary mean-field games. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, 251–259.

Sunehag, P.; Lever, G.; Gruslys, A.; Czarnecki, W. M.; Zambaldi, V. F.; Jaderberg, M.; Lanctot, M.; Sonnerat, N.; Leibo, J. Z.; Tuyls, K.; et al. 2018. Value-Decomposition Networks For Cooperative Multi-Agent Learning Based On Team Reward. In *AAMAS*, 2085–2087.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Wang, L.; Yang, Z.; and Wang, Z. 2020. Breaking the Curse of Many Agents: Provable Mean Embedding Q-Iteration for Mean-Field Reinforcement Learning. *arXiv preprint arXiv:2006.11917* .

Wen, C.; Yao, X.; Wang, Y.; and Tan, X. 2020. SMIX (λ): Enhancing Centralized Value Functions for Cooperative Multi-Agent Reinforcement Learning. In *AAAI*, 7301–7308.

Wooldridge, M. 2009. *An introduction to multiagent systems*. John Wiley &amp; Sons.

Yang, Y.; Luo, R.; Li, M.; Zhou, M.; Zhang, W.; and Wang, J. 2018. Mean Field Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning*, 5571–5580.