AgentMixer: Multi-Agent Correlated Policy Factorization

Zhiyuan Li[1], Wenshuai Zhao[1], Lijun Wu[2], Joni Pajarinen[1]

[1]Department of Electrical Engineering and Automation, Aalto University
[2]School of Computer Science and Engineering, University of Electronic Science and Technology of China
{zhiyuan.li, wenshuai.zhao, joni.pajarinen}@aalto.fi, wljuestc@sina.com

Abstract

In multi-agent reinforcement learning, centralized training with decentralized execution (CTDE) methods typically assume that agents make decisions based on their local observations independently, which may not lead to a correlated joint policy with coordination. Coordination can be explicitly encouraged during training and individual policies can be trained to imitate the correlated joint policy. However, this may lead to an *asymmetric learning failure* due to the observation mismatch between the joint and individual policies. Inspired by the concept of correlated equilibrium, we introduce a *strategy modification* called AgentMixer that allows agents to correlate their policies. AgentMixer combines individual partially observable policies into a joint fully observable policy non-linearly. To enable decentralized execution, we introduce *Individual-Global-Consistency* to guarantee mode consistency during joint training of the centralized and decentralized policies and prove that AgentMixer converges to an ϵ-approximate Correlated Equilibrium. In the Multi-Agent MuJoCo, SMAC-v2, Matrix Game, and Predator-Prey benchmarks, AgentMixer outperforms or matches state-of-the-art methods.

Code — https://github.com/LiZhYun/BackPropagationThroughAgents.git

Extended version — https://arxiv.org/abs/2401.08728

Introduction

Cooperative multi-agent reinforcement learning (MARL) has attracted substantial attention in recent years owing to its promise in solving many real-world tasks that naturally comprise multiple decision-makers interacting at the same time, such as multi-robot control (Gu et al. 2023), traffic signal control (Ma and Wu 2020), and autonomous driving (Shalev-Shwartz, Shammah, and Shashua 2016). In contrast to the single-agent RL settings, learning in multi-agent systems (MAS) poses two primary challenges: 1) coordination, that is, agents learning to work together in order to achieve a common goal, and 2) partial observability which limits each agent to her own local observations and actions. To address these challenges, the commonly used learning framework called Centralized Training Decentralized Execution (CTDE) (Lowe et al. 2017; Yu et al. 2022) allows agents to access global information during the training phase while during evaluation learned decentralized policies have access only to local information.

To enhance coordination, one line of research is to use value decomposition (VD). VDN (Sunehag et al. 2017) and QMIX (Rashid et al. 2020) learn a centralized joint action value function factorized by decentralized agent utility functions. With the structural constraint of Individual-Global-Max (IGM) (Son et al. 2019), this approach guarantees action consistency between the centralized and decentralized policies. On the other hand, multi-agent policy gradient (MAPG) methods, such as MADDPG (Lowe et al. 2017) and MAPPO (Yu et al. 2022), have achieved high performance. However, even when learning a centralized critic, previous works are still constrained by assuming independence among agents during exploration. Inspired by the Correlated Equilibrium (CE) (Maschler, Zamir, and Solan 2013) in game theory, MAVEN (Mahajan et al. 2019) and SIC (Chen et al. 2022) introduce a hierarchical control method with an external shared latent variable as additional information for agent coordination. Assuming a pre-defined execution order a few recent works further propose autoregressive policies to impose coordination among agents by allowing agents to observe other agents' actions, either explicitly (Wang, Ye, and Lu 2023; Li et al. 2024b) or implicitly (Li et al. 2023; Wen et al. 2022). Most existing correlated policy approaches violate the requirement for decentralized execution. This paper instead aims to achieve a correlated equilibrium in a fully decentralized way which is crucial for real-world applications such as wireless devices (Pajarinen, Hottinen, and Peltonen 2014), where agents are required to operate autonomously while collectively maximizing performance.

In order to mitigate the difficulty of learning under partial observability, CTDE exploits true state information, usually via a centralized critic, to train individual policies conditioned on the local observation-action history. While it is possible to first learn a centralized expert policy and then train the decentralized agents to follow it, it may result in suboptimal partially observable policies since the omniscient critic or agent has no knowledge of what the decentralized agents do not know, referred to as the *asymmetric learning failure* (Warrington et al. 2021). Consider a sce-

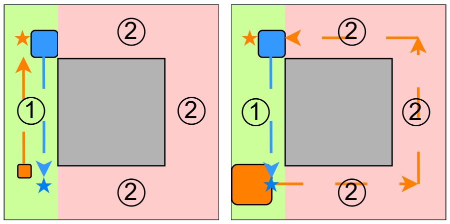

Figure 1: The partially observable bridge crossing task. Two agents (blue and orange boxes), with changing physiques (box sizes) in different episodes as shown in the left and right figures, must navigate to their destinations, marked with stars with corresponding colors, through passageways 1 or 2 while avoiding congestion. The expert is conditioned on an omniscient state indicating the physiques of all agents, while an agent cannot see the physique of another agent. Naively learning from the full observation expert policy, the agents would never stagger passageways, and instead cross the same passageway directly, incurring congestion.

nario where two agents of distinct physical shapes try to get to their opposite destinations through two possible paths 1 and 2, as shown in Figure 1. Successful policies should avoid collision as the body sizes of agents always change and each passage only permits two small agents or one big agent. In CTDE, we can learn the optimal fully observable joint policy conditioned on agents' physiques, which would select the shorter path 1 when both agents are small. However, naively learning from such a centralized agent could lead to agents jamming on the same passage, as the partially observable agents cannot access the other agent's body size. In contrast, the optimal partially observable policies should ideally ensure that each agent consistently selects distinct passageways to avoid collision. This *asymmetric learning failure* is a prevalent issue in MARL due to the partial observability nature of MAS. While a few works have studied similar challenges in the context of single-agent RL (Walsman et al. 2023), it is worth noting that this issue within the MARL domain has not been thoroughly investigated to the best of our knowledge.

In this paper, we propose *correlated policy factorization*, dubbed AgentMixer, to tackle the above two challenges and achieve CE among agents in a fully decentralized way. Firstly, we propose a novel framework, named *Policy Modifier* (PM), to model the correlated joint policy, which takes as input decentralized partially observable policies and the state information and outputs the modified policies. Consequently, PM acts as an *observer* from the CE perspective and the modified policies form a correlated joint policy. Further, to mitigate the *asymmetric learning failure* when learning decentralized partially observable policies from the correlated joint fully observable policy, we then introduce a novel mechanism called *Individual-Global-Consistency* (IGC), which keeps consistent modes between individual policies and joint policy while allowing correlated exploration in joint policy. Theoretically, we prove that AgentMixer converges to ϵ-approximate Correlated Equilibrium. Experimental results on various benchmarks confirm its strong empirical performance against current state-of-the-art MARL methods.

Related Work

Modeling complex correlations among agents has been attracting a growing amount of attention in recent years. The centralized training decentralized execution (CTDE) paradigm has demonstrated its success in cooperative multi-agent domain (Lowe et al. 2017; Yu et al. 2022). Centralized training with additional global information makes agents cooperate better while decentralized execution enables distributed deployment.

Value decomposition. Value decomposition methods decompose the joint Q-function into individual utility functions following different interpretations of Individual-Global-Maximum (IGM) (Son et al. 2019), i.e., the consistency between optimal local actions and optimal joint action. VDN (Sunehag et al. 2017) and QMIX (Rashid et al. 2020) decompose the joint action-value function by additivity and monotonicity respectively. QTRAN (Son et al. 2019) and QPLEX (Wang et al. 2021a) introduce additional components to enhance the expressive capability of value decomposition. To enhance coordination, MAVEN (Mahajan et al. 2019) introduces committed exploration among agents into QMIX. Recent works delve into applying value decomposition to actor-critic methods. FACMAC (Peng et al. 2021) and DOP (Wang et al. 2021b) combine value decomposition to compute policy gradient with a centralized but factored critic. FOP (Zhang et al. 2021) and MACPF (Wang, Ye, and Lu 2023) derive joint soft-Q-function decomposition according to independent and conditional policy factorization respectively.

Policy factorization. Existing approaches commonly assume the independence of agents' policies, modeling the joint policy as the Cartesian Product of each agent's fully independent policy (Yu et al. 2022; Zhang et al. 2021; Li et al. 2024a; Zhao et al. 2024). However, such an assumption lacks in modeling complex correlations as it constrains the expressiveness of the joint policy and limits the agents' capability to coordinate. In contrast, some recent works (Wang, Ye, and Lu 2023; Wen et al. 2022; Fu et al. 2022) explicitly take the dependency among agents by presenting the joint policy in an auto-regressive form based on the chain rule. MAT (Wen et al. 2022) casts MARL into a sequence modeling problem and introduces Transformer (Vaswani et al. 2017) to generate actions. Wang, Ye, and Lu (2023) extends FOP (Zhang et al. 2021) with auto-regressive policy factorization. ACE (Li et al. 2023) transforms multi-agent Markov Decision Process (MMDP) (Littman 1994) into a single-agent Markov Decision Process (MDP) (Feinberg and Shwartz 2012), which implicitly models the auto-regressive joint policy. Despite the merits of the auto-regressive model, the fixed execution order may struggle to generalize different cooperative paradigms. Inspired by Correlated Equilibrium (Maschler, Zamir, and Solan 2013), SIC (Chen et al. 2022) introduces a coordination signal to achieve richer classes of the joint policy and maximizes the mutual information between the signal and the joint policy, which is close to MAVEN. Correlated Q-learning (Greenwald and Hall 2003) generalizes Nash Q-learning (Hu and Wellman 2003) based on CE and proposes several variants to resolve the equilibrium selection problem. Similarly, Schroeder de Witt et al. (2019) learns a hierarchical policy tree based on a shared random seed. Sheng et al. (2023) learns coordinated behavior with recursive reasoning. However, most existing work focuses on fully observable settings or violates the decentralized execution requirement.

Moreover, existing approaches rarely study the issues arising from the use of asymmetric information (Warrington et al. 2021) in CTDE, that is, the joint fully observable critic or agent has access to information unavailable to the partially observable agents. In this paper, we study how to factorize the correlated joint fully observable policy into decentralized policies under partial observability.

Background

Decentralized Partially Observable Markov Decision Processes

In this work, we model a fully cooperative multi-agent game with N agents as a *decentralized partially observable Markov decision process* (Dec-POMDP) (Oliehoek and Amato 2016), which is formally defined as a tuple $\mathcal{G} = (\mathcal{N}, \mathcal{S}, \mathcal{O}, \mathbb{O}, \mathcal{B}, \mathcal{A}, \mathcal{T}, \Omega, R, \gamma, \rho_0)$. $\mathcal{N} = \{1, \dots, N\}$ is a set of agents, $s \in \mathcal{S}$ denotes the state of the environment and ρ_0 is the distribution of the initial state. $\mathcal{A} = \prod_{i=1}^{N} A^i$ is the joint action space, $\mathbb{O} = \prod_{i=1}^{N} O^i$ is the set of joint observations. At time step t, each agent i receives an individual partial observation $o_t^i \in O^i$ given by the observation function $\mathcal{O} : (a_t, s_{t+1}) \mapsto P(o_{t+1}|a_t, s_{t+1})$ where a_t, s_{t+1} and o_{t+1} are the joint actions, states and joint observations respectively. Each agent i uses a stochastic policy $\pi^i(a_t^i|h_t^i, \omega_t^i)$ conditioned on its action-observation history $h_t^i = (o_0^i, a_0^i, \dots, o_{t-1}^i, a_{t-1}^i)$ and a random seed $\omega_t^i \in \Omega_t$ to choose an action $a_t^i \in A^i$. A belief state b_t is a probability distribution over states at time t, where $b_t \in \mathcal{B}$, and $\mathcal{B}$ is the space of all probability distributions over the state space. Actions a_t drawn from joint policy $\pi(a_t|s_t, \omega_t)$ conditioned on state s_t and joint random seed $\omega_t = (\omega_t^1, \dots, \omega_t^N)$ change the state according to transition function $\mathcal{T} : (s_t, a_t^1, \dots, a_t^N) \mapsto P(s_{t+1}|s_t, a_t^1, \dots, a_t^N)$. All agents share the same reward $r_t = R(s_t, a_t^1, \dots, a_t^N)$ based on s_t and a_t. γ is the discount factor for future rewards. Agents try to maximize the expected total reward, $\mathcal{J}(\pi) = \mathbb{E}_{s_0, a_0, \dots}[\sum_{t=0}^{\infty} \gamma^t r_t]$, where $s_0 \sim \rho_0(s_0), a_t \sim \pi(a_t|s_t, \omega_t)$.

Equilibrium Notions

We first define a *joint (potentially correlated) policy* as $\pi = \pi^1 \odot \pi^2 \cdots \odot \pi^N$. We also denote $\pi^{-i} = \pi^1 \odot \cdots \pi^{i-1} \odot \pi^{i+1} \odot \cdots \odot \pi^N$ to be the joint policy excluding the i^{th} agent. A *product* policy is denoted as $\pi = \pi^1 \times \pi^2 \cdots \times \pi^N$ if the distribution of drawing each seed ω_t^i for different agents is independent. We define the value function $V^i_{\pi^i, \pi^{-i}}(s)$ as the expected returns under state s that i^{th} agent will receive if all agents follow joint policy $\pi = (\pi^i, \pi^{-i})$:

$$V^i_{\pi^i, \pi^{-i}}(s) = \mathbb{E}_{a^i_{0:\infty} \sim \pi^i, a^{-i}_{0:\infty} \sim \pi^{-i}, s_{1:\infty} \sim \mathcal{T}}[\Sigma_{t=0}^{\infty} \gamma^t r_t | s_0 = s]. \tag{1}$$

A *strategy modification* for the i^{th} agent is a map $f^i : A^i \mapsto A^i$, which maps from the action set to itself. We can define the resulting policy by applying the map on π^i as $f^i \diamond \pi^i$.

With the definition above, we can accordingly define the solution concepts.

Definition 1 (ϵ-approximate Nash Equilibrium)**.** A **product** policy π_* is an ϵ-approximate Nash Equilibrium (NE) if for all $i \in \mathcal{N}$ and any $\epsilon \geq 0$:

$$V^i_{\pi^i_*, \pi^{-i}_*}(s) \geq \max_{\pi^i} V^i_{\pi^i, \pi^{-i}_*}(s) - \epsilon. \tag{2}$$

Definition 2 (ϵ-approximate Coarse Correlated Equilibrium)**.** A **joint** policy π_* is an ϵ-approximate Coarse Correlated Equilibrium (CCE) if for all $i \in \mathcal{N}$ and any $\epsilon \geq 0$:

$$V^i_{\pi^i_*, \pi^{-i}_*}(s) \geq \max_{\pi^i} V^i_{\pi^i, \pi^{-i}_*}(s) - \epsilon. \tag{3}$$

The only difference between Definition 1 and Definition 2 is that an NE has to be a product policy while a CCE can be correlated.

Definition 3 (ϵ-approximate Correlated Equilibrium)**.** A *joint* policy π_* is an ϵ-approximate Correlated Equilibrium (CE) if for all $i \in \mathcal{N}$ and any $\epsilon \geq 0$:

$$V^i_{\pi^i_*, \pi^{-i}_*}(s) \geq \max_{f^i} V^i_{(f^i \diamond \pi^i_*) \odot \pi^{-i}_*}(s) - \epsilon. \tag{4}$$

CCE forms a larger set of distributions than CE, and CE is richer than NE (i.e., NE $\subseteq$ CE $\subseteq$ CCE).

AgentMixer

In this work, we propose AgentMixer to achieve correlated policy factorization. The proposed method consists of two main components: *Policy Modifier* that models correlated joint fully observable policy and *Individual-Global-Consistency* that leverages the resulting joint policy for learning the individual policies while mitigating the *asymmetric information issue*.

Policy Modifier

To efficiently introduce correlation among agents, we propose *Policy Modifier*, a novel framework based entirely on multi-layer perceptrons (MLPs) (see the Appendix), which contains two types of MLP layers (Tolstikhin et al. 2021): *agent-mixing MLPs* and *channel-mixing MLPs*. The agent-mixing MLPs allow inter-agent communication; they operate on each channel of the feature independently. The channel-mixing MLPs allow intra-agent information fusion; they operate on each agent independently. These two types of layers are interleaved to enable the interaction among agents and the correlated representation of the joint policy. Specifically, agent- and channel-mixing can be written as follows:

$$\begin{aligned} \mathrm{H}_{\text{agent}} &= \mathrm{H}_{\text{input}} + \mathrm{W}^{(2)}_{\text{agent}} \sigma(\mathrm{W}^{(1)}_{\text{agent}} \mathrm{LayerNorm}(\mathrm{H}_{\text{input}})), \\ \mathrm{H}_{\text{channel}} &= \mathrm{H}_{\text{agent}} + \sigma(\mathrm{W}^{(1)}_{\text{channel}} \mathrm{LayerNorm}(\mathrm{H}_{\text{agent}})) \mathrm{W}^{(2)}_{\text{channel}}, \end{aligned} \tag{5}$$

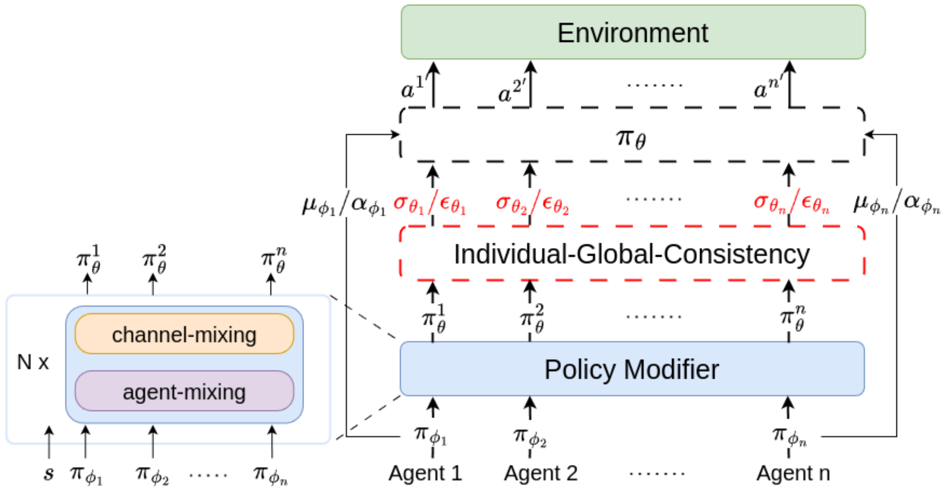

Figure 2: AgentMixer contains two components: 1) *Policy Modifier* takes the individual partially observable policies and state as inputs and produces correlated joint fully observable policy as outputs, and 2) *Individual-Global-Consistency* keeps the mode consistency among the joint policy and individual policies.

where $\mathrm{H}_{\text{input}}$ is a concatenation of state features and individual policies features and W denotes fully connected layers. The policy features are derived by compressing the policy parameters through one MLP layer. In the continuous action space, the policy parameters are the mean and standard deviation, while in the discrete action space, the policy parameters are the logits. Then, the output of PM will be combined with individual policies to generate the correlated joint policy, denoted as $\mathrm{PM}([\pi^i]_{i=1}^N) = ((f^1 \diamond \pi^1), \cdots, (f^N \diamond \pi^N)) = (f^1 \diamond \pi^1) \odot (f^2 \diamond \pi^2) \cdots \odot (f^N \diamond \pi^N)$, where f denotes a *strategy modification*. Consequently, PM maps the individual policies into a correlated joint policy by introducing dependencies among agents.

Individual Global Consistency

With the resulting correlated joint fully observable policy generated by PM, we can easily adopt different single-agent algorithms to get a (sub-)optimal correlated joint fully observable policy. To fulfill decentralized execution, we further ask a question:

Question 1: Can we just derive the decentralized partially observable policies by distilling the learned (sub-)optimal correlated joint fully observable policy?

In this section, we take several steps to provide a negative answer to the above research question. We begin by defining the joint policy and product policy as $\pi_\theta(a|s)$ and $\pi_\phi(a|b)$ respectively. Let the joint occupancy, $\rho^\pi(s, b)$, as the (improper) marginal state-belief distribution induced by a policy π: $\rho^\pi(s, b) = \sum_{t=0}^{\infty} \gamma^t P(s_t = s, b_t = b|\pi)$. Then, the marginal state distribution and marginal belief distribution induced by π are denoted as $\rho^\pi(s) = \int_b \rho^\pi(s, b)db$ and $\rho^\pi(b) = \int_s \rho^\pi(s, b)ds$ respectively. To distill the joint policy $\pi_\theta(a|s)$ into the product policy $\pi_\phi(a|b)$, previous work (Ye et al. 2022) leverage imitation learning, i.e., optimizing the asymmetric distillation objective:

$$\mathbb{E}_{\rho^{\pi_\beta}(s,b)} \left[D_{\mathrm{KL}} \left(\pi_\theta(a|s) \parallel \pi_\phi(a|b) \right) \right], \tag{6}$$

where $\pi_\beta(s, b) = \beta\pi_\theta(a|s) + (1-\beta)\pi_\phi(a|b)$, π_β is a mixture of the joint policy $\pi_\theta(a|s)$, and the product policy $\pi_\phi(a|b)$. The coefficient β is annealed to zero during training. This avoids compounding error which grows with time horizon.

We then show that the optimal product policy defined by this objective can be expressed as posterior inference over state conditioned on the joint policy:

Definition 4 (Implicit product policy)**.** For any correlated joint fully observable policy π_θ, product partially observable behavioral policy π_ψ, belief state b, and conditional occupancy ρ^{π_ψ}, we define $\hat{\pi}_\theta^\psi$ as the implicit product policy of π_θ under π_ψ as:

$$\hat{\pi}_\theta^\psi = \mathbb{E}_{\rho^{\pi_\psi}(s|b)} \left[\pi_\theta(a|s) \right]. \tag{7}$$

When $\pi_\psi = \hat{\pi}_\theta^\psi$, we refer to this product policy as the implicit product policy of π_θ, denoted as $\hat{\pi}_\theta$.

Implicit product policy is defined as a posterior inference procedure, marginalizing the conditional occupancy $\rho^{\pi_\psi}(s|b)$. Since the observations/belief may not contain information to distinguish two different latent states, the $\rho^{\pi_\psi}(s|b)$ is a stochastic distribution, and the implicit product policy is the average of the fully observable policy. Suppose a scenario where the agent learns to cross the ice while avoiding the pits in the middle of the ice. The fully observable policy which can observe the location of the pits will choose safer routes that avoid the pits, i.e., both sides of the ice. However, according to 7, the implicit policy that is not informed of the pit locations will take an average path of those safe routes, despite the danger of pits. The key insight is that directly imitating the fully observable policy will cause *asymmetric learning failure*. We show that the solution to the asymmetric distillation objective in 6 is equivalent to the implicit product policy 7 in the Appendix.

However, the implicit product policy requires marginalizing the conditional occupancy $\rho^\pi(s|b)$, which is intractable.

Therefore, we can introduce a variational implicit product policy, π_η, as a proxy to the implicit product policy, which can be learned by minimizing the following objective:

$$\mathbb{E}_{\rho^{\pi_\psi}(s,b)}\left[D_{\mathrm{KL}}\left(\pi_\theta(a|s) \parallel \pi_\eta(a|b)\right)\right]. \tag{8}$$

Under sufficient expressiveness and exact updates assumptions, by setting $\pi_\psi = \pi_\eta$, updating 8 converges to the fixed point, i.e., the implicit product policy (see the Appendix).

We now reason about the *asymmetric learning failure*. To guarantee the optimal product partially observable policy, the divergence between the joint policy and product policy should be strictly zero, which we denote as *identifiability*:

Definition 5 (Identifiable policy pair)**.** Given a correlated joint fully observable policy π_θ and a product partially observable policy π_ϕ, we define $\{\pi_\theta, \pi_\phi\}$ as an identifiable policy pair if and only if $\mathbb{E}_{\rho^{\pi_\phi}(s,b)}\left[D_{\mathrm{KL}}\left(\pi_\theta(a|s) \parallel \pi_\phi(a|b)\right)\right] = 0$.

Identifiable policy pairs require that the product partially observable policy can exactly recover the correlated joint fully observable policy. *Identifiability* then requires the optimal correlated joint fully observable policy and the corresponding implicit product policy to form an identifiable policy pair. Using *identifiability*, we can prove that given an optimal correlated joint fully observable policy, optimizing the asymmetric distillation objective is guaranteed to recover an optimal product partially observable policy:

Theorem 1 (Convergence of asymmetric distillation)**.** *Given an optimal correlated joint fully observable policy π_{θ^*} being identifiability, the iteration defined by:*

$$\eta_{k+1} = \arg\min_{\eta} \mathbb{E}_{\rho^{\pi_{\eta_k}}(s,b)}\left[D_{\mathrm{KL}}\left(\pi_{\theta^*}(a|s) \parallel \pi_\eta(a|b)\right)\right] \tag{9}$$

converges to $\pi_{\eta^*}(a|b)$ that defines an optimal product partially observable policy, as $k \to \infty$.

Proof. See the Appendix for a detailed proof. □

Theorem 1 shows that *identifiability* of the optimal joint policy defines a sufficient condition to guarantee the thorough distillation of the optimal joint fully observable policy into product partially observable policies. Unfortunately, the *identifiability* imposes a strong limitation on the applicability of asymmetric distillation. Hereby, we can conclude a negative answer to the **Question 1**. Therefore, we propose to modify the joint policy online by imposing a constraint on the mode between the joint policy and the decentralized policies to form an (approximately) identifiable policy pair. Furthermore, instead of naively applying distillation on the learned joint policy, we simultaneously learn the correlated joint fully observable policy and its product partially observable counterpart. We will show that the interleaving of the two learning processes moves the product partially observable policy closer to Correlated Equilibrium, i.e., the optimal product partially observable policy.

We now use the insight from Theorem 1 and the definition of *identifiability* to define *Individual-Global-Consistency* (IGC), which keeps consistent modes between product partially observable policy and correlated joint fully observable policy.

Definition 6 (IGC)**.** For a correlated joint fully observable policy $\pi_\theta(a|s)$, if there exists a product partially observable policy $\pi_\phi(a|b) = \pi_{\phi^1}(a|h^1) \times \pi_{\phi^2}(a|h^2) \cdots \times \pi_{\phi^N}(a|h^N)$, such that

$$\mathrm{Mo}(\pi_\theta) = \left(\mathrm{Mo}(\pi_{\phi^1}), \ldots, \mathrm{Mo}(\pi_{\phi^N})\right), \tag{10}$$

where $Mo(\cdot)$ denotes the mode of distribution. Then, we say that $\pi_\phi(a|b)$ satisfy IGC.

IGC enables the actions that occur most frequently in the joint policy and the product policy to be equivalent. Crucially, IGC minimizes the divergence between the two policies while allowing correlated exploration in the joint policy. One may find that IGC and IGM share some similarities. IGM ensures value-based individual optimal actions constitute the optimal joint action, while IGC maintains policy-based consistency with explicit consideration of partial observability. See the Appendix for a detailed clarification.

Implementation of IGC To preserve IGC, we adopt the method of disentanglement between exploration and exploitation to decompose the joint policy into two components: one for the mode (exploitation) and the other for the deviation (exploration). Then, IGC can be enforced through an equality constraint on the mode of joint policy and individual policies. Based on this disentanglement, agents are able to coordinate their exploration through the centralized policy. In practice, we divide the implementation of IGC into two categories: continuous action space and discrete action space.

Continuous Case: In this case, we assume the continuous action policy of agent i as a Gaussian distribution with mean μ_{ϕ^i} and standard deviation σ_{ϕ^i}: $\pi_{\phi^i}(a|h^i) = \mathcal{N}(\mu_{\phi^i}(h^i), \sigma^2_{\phi^i}(h^i))$. Since the mode of a Gaussian distribution is equal to the mean, we set the mean of joint policy as the collection of individual policies while the standard deviation is generated by PM: $\pi_\theta(a|s) = \mathcal{N}(([\mu_{\phi^i}]_{i=1}^N), \sigma^2_\theta(s))$.

Discrete Case: In this case, we denote the discrete action policy of agent i as a categorical distribution parameterized by probabilities α_{ϕ^i}:

$$\pi_{\phi^i}(a|h^i) = Cat(\alpha_{\phi^i}(h^i)) = \mathrm{softmax}(\alpha_{\phi^i}(h^i)), \tag{11}$$

where $\sum_{k=1}^K \alpha^k_{\phi^i}(h^i) = 1$. The mode of a categorical distribution is the category with the highest frequency. However, it is tricky to promote cooperative exploration while preserving the mode consistency. Fortunately, Gumbel-Softmax distribution (Jang, Gu, and Poole 2017) provides another perspective, where we explicitly disentangle exploration and mode. Specifically, we define the joint policy as:

$$\pi_\theta = \begin{pmatrix} \mathrm{softmax}((\epsilon^1_\theta + \log\alpha_{\phi^1})/\tau^1) \\ \vdots \\ \mathrm{softmax}((\epsilon^N_\theta + \log\alpha_{\phi^N})/\tau^N) \end{pmatrix}, \tag{12}$$

where τ is a temperature hyperparameter and ϵ_θ is sampled using inverse transform sampling by generating $u_\theta \in (0, 1)$ with sigmoid function and computing $\epsilon_\theta = -\log(-\log(u_\theta))$. Note that when the temperature approaches 0, the joint policy degrades to the collection of individual policies.

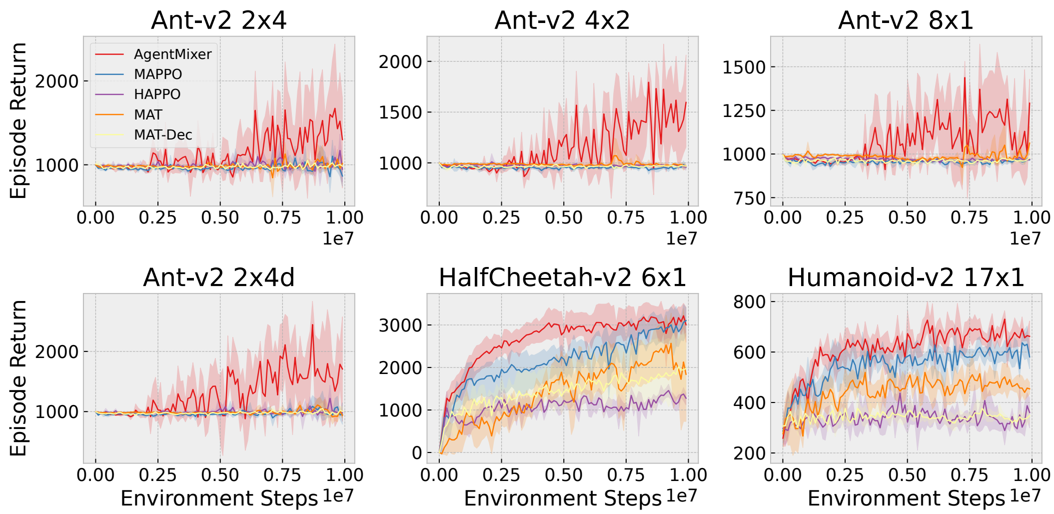

Figure 3: AgentMixer outperforms comparison methods on multiple Multi-Agent MuJoCo tasks. Please, see the statistical significance tests in the Appendix for further evidence. Partial observability in MA-MuJoCo is a critical challenge to most baselines.

Convergence of AgentMixer

Together with PM, we can treat the learning of the correlated joint fully observable policy as a single-agent RL problem where abundant single-agent methods with theoretical guarantees of convergence and performance exist:

$$\mathcal{J}(\pi_\theta) = \mathbb{E}_{s\sim\rho^{\pi_\theta}, a\sim\pi_\theta}\left[\sum_{t=0}^{\infty}\gamma^t r_t\right], \text{subject to IGC}, \quad (13)$$

where $\pi_\theta = \mathrm{PM}([\pi_{\phi^i}]_{i=1}^N)$.

Theorem 2 (Convergence of AgentMixer)**.** *The product partially observable policy generated by AgentMixer is a ϵ-CE.*

Proof. For proof see the Appendix. □

With Theorem 2, we are ready to present the learning framework of AgentMixer, as illustrated in Figure 2, which consists of two main components: *Policy Modifier* and *Individual-Global-Consistency*. Specifically, PM acts as an observer who takes a holistic view and recommends that each agent follow her instructions. IGC then requires the agents to be obligated to follow the recommendations they receive. We provide the pseudo-code for AgentMixer in the Appendix. AgentMixer can benefit from a variety of strong single-agent algorithms, such as PPO (Schulman et al. 2017) and SAC (Haarnoja et al. 2019). In this work, our implementation of AgentMixer follows PPO (Schulman et al. 2017).

Experiments

We compare our method AgentMixer with MAPPO (Yu et al. 2022), HAPPO (Kuba et al. 2021), MAVEN (Mahajan et al. 2019), ARMAPPO (Fu et al. 2022), MAT (Wen et al. 2022), and MAT with decentralized actor MAT-Dec. Note that our method and all comparison methods except MAT and MAT-Dec have access to only local information during evaluation. We evaluate on several continuous action space Multi-Agent MuJoCo (Peng et al. 2021) (*MA-MuJoCo*) benchmark tasks and discrete action space *SMAC-v2* (Ellis et al. 2022) benchmark tasks. We include more experimental details and results on Climbing Matrix Game (Lauer and Riedmiller 2000) and Predator-Prey (Li et al. 2020) in the Appendix.

Continuous Actions Spaces: *MA-MuJoCo*

As in the full observation setting, previous methods have shown near-optimal performance in the *MA-MuJoCo* tasks (Kuba et al. 2021; Wen et al. 2022), we instead limit the set of observable elements for each agent to themselves, aiming to satisfy better the partial observability nature in MARL. We show the performance comparison against the baselines in Figure 3. We can see that AgentMixer enjoys superior performance over those baselines. The superiority of our method is highlighted especially in Ant-v2 tasks, where partial observability poses a critical challenge as the local observations of each agent (leg) of the ant are quite similar and make it hard to estimate the necessary state information for coordination. In these tasks, while other algorithms, even the *centralized* MAT, fail to learn any meaningful joint policies, AgentMixer outperforms the baselines by a large margin. These results show that AgentMixer can effectively exploit asymmetric information to mitigate the challenges incurred by severe partial observability.

Discrete Action Spaces: *SMAC-v2*

Compared to the StarCraft Multi-Agent Challenge (*SMAC*), we instead evaluate our method on the more challenging *SMAC-v2* benchmark which is designed with higher randomness. As shown in Table 1, AgentMixer outperforms

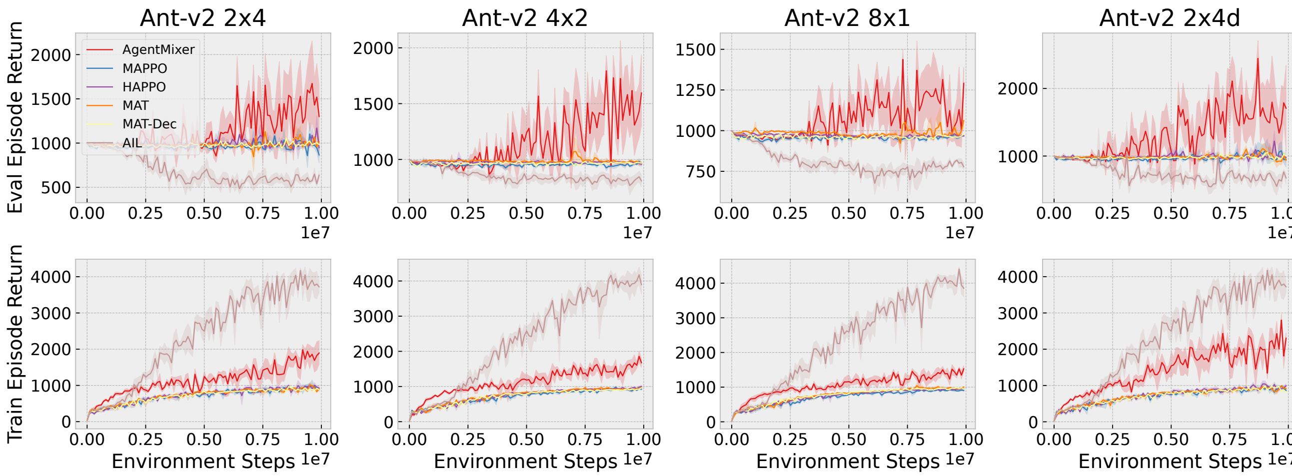

Figure 4: Ablations on Ant-v2. The large performance gap can be seen between training and testing on AIL, which is caused by *asymmetric learning failure*. Other baselines fail to learn any effective policies, while AgentMixer obtains superior performance.

Map	MAPPO	HAPPO	MAT	MAT-Dec	MAVEN	ARMAPPO	Ours
protoss 5v5	$0.51_{0.10}$	$0.38_{0.08}$	$\mathbf{0.55_{0.11}}$	$0.53_{0.04}$	$0.49_{0.11}$	$0.50_{0.09}$	$\mathbf{0.68_{0.04}}$
terran 5v5	$0.48_{0.11}$	$0.39_{0.12}$	$\mathbf{0.63_{0.11}}$	$0.45_{0.06}$	$0.47_{0.03}$	$0.41_{0.04}$	$\mathbf{0.71_{0.10}}$
zerg 5v5	$0.48_{0.05}$	$0.39_{0.05}$	$0.48_{0.10}$	$0.51_{0.10}$	$0.46_{0.08}$	$0.44_{0.06}$	$\mathbf{0.66_{0.03}}$
protoss 10v10	$0.40_{0.08}$	$0.13_{0.02}$	$0.32_{0.09}$	$\mathbf{0.52_{0.02}}$	$0.36_{0.05}$	$0.36_{0.04}$	$0.47_{0.02}$
terran 10v10	$0.29_{0.05}$	$0.12_{0.09}$	$\mathbf{0.46_{0.14}}$	$0.33_{0.03}$	$0.26_{0.02}$	$0.20_{0.02}$	$\mathbf{0.50_{0.11}}$
zerg 10v10	$\mathbf{0.38_{0.12}}$	$0.29_{0.07}$	$\mathbf{0.52_{0.09}}$	$0.36_{0.05}$	$\mathbf{0.46_{0.05}}$	$0.36_{0.06}$	$\mathbf{0.52_{0.06}}$

Table 1: Median evaluate winning rate and standard deviation on six SMACv2 maps for different methods (steps=1e7). All values within 1 standard deviation of the maximum score rate are marked in bold.

other strong baselines in 5 out of 6 scenarios and achieves over 50% win rates in most maps, even over 70% in the terran 5v5 scenarios. Notably, in the 5v5 scenarios, we observe that AgentMixer improves win rates compared to the other six methods by a large margin, i.e. by 8% to 32%.

Ablation Results

To examine the effectiveness of AgentMixer in addressing *asymmetric learning failure*, we perform ablation experiments by adding an imitation learning baseline, asymmetric imitation learning (AIL) (Warrington et al. 2021), which uses PPO, conditioned on full state information, to supervise learning decentralized policies, conditioned on partial information. As shown in Figure 4, due to *asymmetric learning failure*, AIL performs poorly in evaluation, although it achieves superior performance in training. In contrast, AgentMixer couples the learning of the centralized policy and decentralized policies such that partially observed policies can perform consistently with the fully observed policy.

Conclusion and Future Work

To achieve coordination among partially observable agents, this paper presents a novel framework named AgentMixer which enables *correlated policy factorization* and provably converges to ϵ-approximate Correlated Equilibrium. AgentMixer consists of two key components: 1) the *Policy Modifier* that takes all the initial decisions from individual agents and composes them into a correlated joint policy based on the full state information; 2) the *Individual-Global-Consistency* which mitigates the *asymmetric learning failure* by preserving the consistency between individual and joint policy. Surprisingly, IGC and IGM can be considered as parallel works of policy gradient-based and value-based methods respectively. We will study the transformation between IGC and IGM in future work. We extensively evaluate the proposed method in Multi-Agent MuJoCo, SMAC-v2, Matrix Game, and Predator-Prey. The experiments demonstrate that our method outperforms strong baselines in most tasks and achieves comparable performance in the rest.

Acknowledgments

This work was supported by the Research Council of Finland project 357301 and Flagship programme: Finnish Center for Artificial Intelligence FCAI. We acknowledge CSC – IT Center for Science, Finland, for awarding this project access to the LUMI supercomputer, owned by the EuroHPC Joint Undertaking, hosted by CSC (Finland) and the LUMI consortium through CSC. We acknowledge the computational resources provided by the Aalto Science-IT project.

References

Chen, L.; Guo, H.; Du, Y.; Fang, F.; Zhang, H.; Zhang, W.; and Yu, Y. 2022. Signal Instructed Coordination in Cooperative Multi-agent Reinforcement Learning. In Chen, J.; Lang, J.; Amato, C.; and Zhao, D., eds., *Distributed Artificial Intelligence*, 185–205. Cham: Springer International Publishing. ISBN 978-3-030-94662-3.

Ellis, B.; Moalla, S.; Samvelyan, M.; Sun, M.; Mahajan, A.; Foerster, J. N.; and Whiteson, S. 2022. SMACv2: An Improved Benchmark for Cooperative Multi-Agent Reinforcement Learning. arXiv:2212.07489.

Feinberg, E.; and Shwartz, A. 2012. *Handbook of Markov Decision Processes: Methods and Applications*. International Series in Operations Research & Management Science. Springer US. ISBN 9781461508052.

Fu, W.; Yu, C.; Xu, Z.; Yang, J.; and Wu, Y. 2022. Revisiting Some Common Practices in Cooperative Multi-Agent Reinforcement Learning. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvari, C.; Niu, G.; and Sabato, S., eds., *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 6863–6877. PMLR.

Greenwald, A.; and Hall, K. 2003. Correlated-Q Learning. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, 242–249. AAAI Press. ISBN 1577351894.

Gu, S.; Grudzien Kuba, J.; Chen, Y.; Du, Y.; Yang, L.; Knoll, A.; and Yang, Y. 2023. Safe multi-agent reinforcement learning for multi-robot control. *Artificial Intelligence*, 319: 103905.

Haarnoja, T.; Zhou, A.; Hartikainen, K.; Tucker, G.; Ha, S.; Tan, J.; Kumar, V.; Zhu, H.; Gupta, A.; Abbeel, P.; and Levine, S. 2019. Soft Actor-Critic Algorithms and Applications. arXiv:1812.05905.

Hu, J.; and Wellman, M. P. 2003. Nash Q-Learning for General-Sum Stochastic Games. *J. Mach. Learn. Res.*, 4(null): 1039–1069.

Jang, E.; Gu, S.; and Poole, B. 2017. Categorical Reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*.

Kuba, J. G.; Chen, R.; Wen, M.; Wen, Y.; Sun, F.; Wang, J.; and Yang, Y. 2021. Trust Region Policy Optimisation in Multi-Agent Reinforcement Learning. *CoRR*, abs/2109.11251.

Lauer, M.; and Riedmiller, M. A. 2000. An Algorithm for Distributed Reinforcement Learning in Cooperative Multi-Agent Systems. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, 535–542. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. ISBN 1558607072.

Li, C.; Liu, J.; Zhang, Y.; Wei, Y.; Niu, Y.; Yang, Y.; Liu, Y.; and Ouyang, W. 2023. ACE: Cooperative Multi-Agent Q-learning with Bidirectional Action-Dependency. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(7): 8536–8544.

Li, S.; Gupta, J. K.; Morales, P.; Allen, R.; and Kochenderfer, M. J. 2020. Deep implicit coordination graphs for multi-agent reinforcement learning. *arXiv preprint arXiv:2006.11438*.

Li, Z.; Wu, L.; Su, K.; Wu, W.; Jing, Y.; Wu, T.; Duan, W.; Yue, X.; Tong, X.; and Han, Y. 2024a. Coordination as inference in multi-agent reinforcement learning. *Neural Networks*, 172: 106101.

Li, Z.; Zhao, W.; Wu, L.; and Pajarinen, J. 2024b. Backpropagation Through Agents. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(12): 13718–13726.

Littman, M. L. 1994. Markov games as a framework for multi-agent reinforcement learning. In Cohen, W. W.; and Hirsh, H., eds., *Machine Learning Proceedings 1994*, 157–163. San Francisco (CA): Morgan Kaufmann. ISBN 978-1-55860-335-6.

Lowe, R.; WU, Y.; Tamar, A.; Harb, J.; Pieter Abbeel, O.; and Mordatch, I. 2017. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ma, J.; and Wu, F. 2020. Feudal Multi-Agent Deep Reinforcement Learning for Traffic Signal Control. In Seghrouchni, A. E. F.; Sukthankar, G.; An, B.; and Yorke-Smith, N., eds., *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, AAMAS '20, Auckland, New Zealand, May 9-13, 2020*, 816–824. International Foundation for Autonomous Agents and Multiagent Systems.

Mahajan, A.; Rashid, T.; Samvelyan, M.; and Whiteson, S. 2019. MAVEN: Multi-Agent Variational Exploration. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Maschler, M.; Zamir, S.; and Solan, E. 2013. *Game Theory*. Cambridge University Press. ISBN 9781107005488.

Oliehoek, F. A.; and Amato, C. 2016. *A Concise Introduction to Decentralized POMDPs*. Springer Publishing Company, Incorporated, 1st edition. ISBN 3319289276.

Pajarinen, J.; Hottinen, A.; and Peltonen, J. 2014. Optimizing Spatial and Temporal Reuse in Wireless Networks by Decentralized Partially Observable Markov Decision Processes. *IEEE Transactions on Mobile Computing*, 13(4): 866–879.

Peng, B.; Rashid, T.; Schroeder de Witt, C.; Kamienny, P.-A.; Torr, P.; Boehmer, W.; and Whiteson, S. 2021. FACMAC: Factored Multi-Agent Centralised Policy Gradients. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 12208–12221. Curran Associates, Inc.

Rashid, T.; Samvelyan, M.; De Witt, C. S.; Farquhar, G.; Foerster, J.; and Whiteson, S. 2020. Monotonic Value Function

Factorisation for Deep Multi-Agent Reinforcement Learning. *J. Mach. Learn. Res.*, 21(1).

Schroeder de Witt, C.; Foerster, J.; Farquhar, G.; Torr, P.; Boehmer, W.; and Whiteson, S. 2019. Multi-Agent Common Knowledge Reinforcement Learning. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347.

Shalev-Shwartz, S.; Shammah, S.; and Shashua, A. 2016. Safe, Multi-Agent, Reinforcement Learning for Autonomous Driving. *CoRR*, abs/1610.03295.

Sheng, J.; Li, W.; Jin, B.; Zha, H.; Wang, J.; and Wang, X. 2023. Negotiated Reasoning: On Provably Addressing Relative Over-Generalization. arXiv:2306.05353.

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 Chaudhuri, K.; and Salakhutdinov, R., eds., *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 5887–5896. PMLR.

Sunehag, P.; Lever, G.; Gruslys, A.; Czarnecki, W. M.; Zambaldi, V.; Jaderberg, M.; Lanctot, M.; Sonnerat, N.; Leibo, J. Z.; Tuyls, K.; and Graepel, T. 2017. Value-Decomposition Networks For Cooperative Multi-Agent Learning. arXiv:1706.05296.

Tolstikhin, I. O.; Houlsby, N.; Kolesnikov, A.; Beyer, L.; Zhai, X.; Unterthiner, T.; Yung, J.; Steiner, A.; Keysers, D.; Uszkoreit, J.; Lucic, M.; and Dosovitskiy, A. 2021. MLP-Mixer: An all-MLP Architecture for Vision. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 24261–24272. Curran Associates, Inc.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Walsman, A.; Zhang, M.; Choudhury, S.; Fox, D.; and Farhadi, A. 2023. Impossibly Good Experts and How to Follow Them. In *The Eleventh International Conference on Learning Representations*.

Wang, J.; Ren, Z.; Liu, T.; Yu, Y.; and Zhang, C. 2021a. {QPLEX}: Duplex Dueling Multi-Agent Q-Learning. In *International Conference on Learning Representations*.

Wang, J.; Ye, D.; and Lu, Z. 2023. More Centralized Training, Still Decentralized Execution: Multi-Agent Conditional Policy Factorization. In *The Eleventh International Conference on Learning Representations*.

Wang, Y.; Han, B.; Wang, T.; Dong, H.; and Zhang, C. 2021b. {DOP}: Off-Policy Multi-Agent Decomposed Policy Gradients. In *International Conference on Learning Representations*.

Warrington, A.; Lavington, J. W.; Scibior, A.; Schmidt, M.; and Wood, F. 2021. Robust Asymmetric Learning in POMDPs. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 11013–11023. PMLR.

Wen, M.; Kuba, J.; Lin, R.; Zhang, W.; Wen, Y.; Wang, J.; and Yang, Y. 2022. Multi-Agent Reinforcement Learning is a Sequence Modeling Problem. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 16509–16521. Curran Associates, Inc.

Ye, J.; Li, C.; Wang, J.; and Zhang, C. 2022. Towards global optimality in cooperative marl with sequential transformation. *arXiv preprint arXiv:2207.11143*.

Yu, C.; Velu, A.; Vinitsky, E.; Gao, J.; Wang, Y.; Bayen, A.; and WU, Y. 2022. The Surprising Effectiveness of PPO in Cooperative Multi-Agent Games. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 24611–24624. Curran Associates, Inc.

Zhang, T.; Li, Y.; Wang, C.; Xie, G.; and Lu, Z. 2021. FOP: Factorizing Optimal Joint Policy of Maximum-Entropy Multi-Agent Reinforcement Learning. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 12491–12500. PMLR.

Zhao, W.; Zhao, Y.; Li, Z.; Kannala, J.; and Pajarinen, J. 2024. Optimistic Multi-Agent Policy Gradient. In Salakhutdinov, R.; Kolter, Z.; Heller, K.; Weller, A.; Oliver, N.; Scarlett, J.; and Berkenkamp, F., eds., *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, 61186–61202. PMLR.