A Policy Gradient Theorem for Learning to Learn in Multiagent Reinforcement Learning

Dong-Ki Kim[1,3] dkkim93@mit.edu
Miao Liu[2,3] miao.liu1@ibm.com
Matthew Riemer[2,3] mdriemer@us.ibm.com
Golnaz Habibi[1,3] golnaz@mit.edu
Sebastian Lopez-Cot[1,3] slcot@mit.edu
Samir Wadhwania[1] samirw@mit.edu
Gerald Tesauro[2,3] gtesauro@us.ibm.com
Jonathan P. How[1,3] jhow@mit.edu

[1]LIDS, MIT [2]IBM Research [3]MIT-IBM Watson AI Lab

Abstract

Learning optimal policies in the presence of non-stationary policies of other simultaneously learning agents is a major challenge in multiagent reinforcement learning. This paper proposes the first policy gradient theorem based on meta-learning that addresses this non-stationarity problem. The policy gradient theorem that we derive inherently includes both a *self-shaping* term that considers the impact of a meta-agent's initial policy on its adapted policy and an *opponent-shaping* term that exploits the learning dynamics of the other agents. We demonstrate that our meta-policy gradient provides the most all-encompassing approach for agents to meta-learn about different sources of non-stationarity in the environment to improve their learning performances. Furthermore, our approach enables fast adaptation (i.e., need only a few interactions) with respect to the non-stationary fellow agents.

Introduction

Learning in a multiagent setting is inherently more difficult than a single-agent learning problem because agents interact with both the environment and the other agents (Buşoniu, Babuška, and De Schutter 2010; Hernandez-Leal et al. 2017). Specifically, the fundamental challenge is the difficulty of learning optimal policies due to the presence of *non-stationary* policies of the other simultaneously learning agents whose actions jointly affect the performance (Tuyls and Weiss 2012; Omidshafiei et al. 2017). Hence, agents are required to adapt their strategic behaviors with respect to potentially large, unpredictable changes in the fellow agents' policies, and failure to do so may result in undesirable outcomes. The problem is further complicated by the fact that the non-stationarity limits agents to have only a few interactions with each other before the policies change, imposing the constraint of fast adaptation (Al-Shedivat et al. 2018).

A standard approach of addressing the non-stationarity problem is to consider information about the policies of the fellow agents and reason about the effect of joint actions (Papoudakis et al. 2019). The literature on opponent modeling, for example, learns to infer the opponents' behaviors and conditions policies on the inferred information (He et al. 2016; Raileanu et al. 2018; Grover et al. 2018). Approaches based on the centralized training with decentralized execution framework, which accounts for the other agents' behavior through the centralized critic, can be also classified into this category (Lowe et al. 2017; Foerster et al. 2017b; Yang et al. 2018; Wen et al. 2019). While these works alleviate the non-stationarity issue, they optimize a value function that assumes the other agents have stationary policies (see Equation (1)) and thus fail to consider the learning process of the others (Foerster et al. 2017a). Because fellow agents may have different policies in the future due to their learning process, the incorrect assumption in the value function can cause failure (see Remark 1).

Meta-learning (also referred to as learning to learn) has recently been proposed to address the non-stationarity problem. The framework by Al-Shedivat et al. (2018) proposes an optimization of initial policy parameters by which the meta-agent can anticipate the changes in an opponent's policy and adapt such that its updated policy performs better than the evolved opponent. The key in the optimization is the use of a *self-shaping* term (see Equation (4)) by which the agent accounts for the fact that its initial parameters will be updated in the future, and considers the impact of the initial parameters on its updated parameters and adaptation performance. The work has shown that the agent can successfully adapt against an evolving opponent in a complex environment and, furthermore, its optimization can find initial parameters that use only a few interactions to adapt. However, the optimization treats the evolving opponent as an external factor such that the agent cannot influence the opponent's future policies. Consequently, the framework fails to consider the unique property of multiagent settings: the opponent is also a learning agent that is changing its policy based on the collected trajectories, so the agent can influence the opponent's future policy by changing the distribution of current trajectories.

Contribution. With this insight, we propose a new multiagent reinforcement learning framework based on meta-learning for addressing the non-stationarity problem in multiagent settings. In contrast to previous work by Al-Shedivat et al. (2018), our meta-agent additionally exploits the sequential dependency between the agent's current policy and the opponent's future policy. We begin by extending the meta-policy gradient theorem in Al-Shedivat et al. (2018) based on the multiagent stochastic policy gradient theorem (Wei

et al. 2018) to derive a new meta-policy gradient theorem. We pinpoint that our derivation inherently includes a new term, called an *opponent-shaping term*, in which the agent can consider the impact of the initial parameters on the opponent's future policy parameters. We then observe that the opponent-shaping term is closely related to the shaping of the opponent's learning dynamics in the work by Foerster et al. (2017a), concluding that our new framework unifies both Al-Shedivat et al. (2018) and Foerster et al. (2017a). Hence, our framework enjoys the following perceived benefits: 1) considers the impact of the initial parameters on its adapted parameters through the self-shaping term *while* actively influencing the opponent's future policy through the opponent-shaping term, and 2) fast adaptation (i.e., only needs a few interactions) to adapt to an evolving opponent.

Preliminary

Problem statement. Interactions between multiple agents can be represented by stochastic games (Shapley 1953). Formally, a stochastic game for n agents is defined as a tuple $\langle \mathcal{I}, \mathcal{S}, \boldsymbol{\mathcal{A}}, \mathcal{P}, \boldsymbol{\mathcal{R}}, \gamma \rangle$; $\mathcal{I} = \{1, \ldots, n\}$ is the set of n agents, $\mathcal{S}$ is the set of states, $\boldsymbol{\mathcal{A}} = \times_{i \in \mathcal{I}} \mathcal{A}^i$ is the set of joint action spaces, $\mathcal{P} : \mathcal{S} \times \boldsymbol{\mathcal{A}} \mapsto \mathcal{S}$ is the state transition probability function, $\boldsymbol{\mathcal{R}} = \times_{i \in \mathcal{I}} \mathcal{R}^i$ is the set of joint reward functions, and $\gamma \in [0, 1)$ is the discount factor. We typeset joint sets in bold for clarity. At each timestep t, each agent i executes an action according to its stochastic policy $a_t^i \sim \pi^i(a_t^i | s_t, \phi^i)$ parameterized by ϕ^i, where $s_t \in \mathcal{S}$. A joint action $\boldsymbol{a_t} = \{a_t^i, \boldsymbol{a_t^{-i}}\}$ yields a transition from current state s_t to next state $s_{t+1} \in \mathcal{S}$ with probability $\mathcal{P}(s_{t+1} | s_t, \boldsymbol{a_t})$, where the notation $\boldsymbol{-i}$ indicates all complementary agents of agent i. Agent i then obtains a reward according to its reward function $r_t^i = \mathcal{R}^i(s_t, \boldsymbol{a_t})$.

Learning with stationary assumption. Each agent has an objective to maximize its expected cumulative reward, represented by the value function (Sutton and Barto 1998). One standard approach to estimate the value is to assume the other agents have *stationary* policies (Hernandez-Leal et al. 2017):

$$\begin{aligned} V_{\boldsymbol{\phi}}^i(s_0) &= \mathbb{E}_{\tau_{\boldsymbol{\phi}} \sim P(\tau_{\boldsymbol{\phi}} | \boldsymbol{\phi})} \Big[\sum_{t=0}^{H} \gamma^t r_t^i | s_0 \Big] \\ &= \mathbb{E}_{\tau_{\boldsymbol{\phi}} \sim P(\tau_{\boldsymbol{\phi}} | \boldsymbol{\phi})} \Big[G_0^i(\tau_{\boldsymbol{\phi}}) \Big], \end{aligned} \tag{1}$$

where $\boldsymbol{\phi} = \{\phi^i, \boldsymbol{\phi^{-i}}\}$ is the set of parameters, s_0 is an initial state, $\tau_{\boldsymbol{\phi}}$ denotes trajectories sampled under the joint policy with parameters $\boldsymbol{\phi}$, H is the horizon, and G_0^i denotes agent i's discounted return from the beginning of an episode.

Meta-learning with self-shaping term. The work by Al-Shedivat et al. (2018) proposes to optimize initial policy parameters θ^i that maximize agent i's meta-value function:

$$V_{\boldsymbol{\theta}, \boldsymbol{\phi}}^i(s_0) = \mathbb{E}_{\tau_{\boldsymbol{\theta}} \sim P(\tau_{\boldsymbol{\theta}} | \boldsymbol{\theta})} \Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}} \sim P(\tau_{\boldsymbol{\phi}} | \boldsymbol{\phi})} \big[G_0^i(\tau_{\boldsymbol{\phi}}) \big] \Big] \tag{2}$$

where $\boldsymbol{\theta} = \{\theta^i, \boldsymbol{\theta^{-i}}\}$. We note $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ are the set of parameters before and after the update, respectively (see Figure 2).

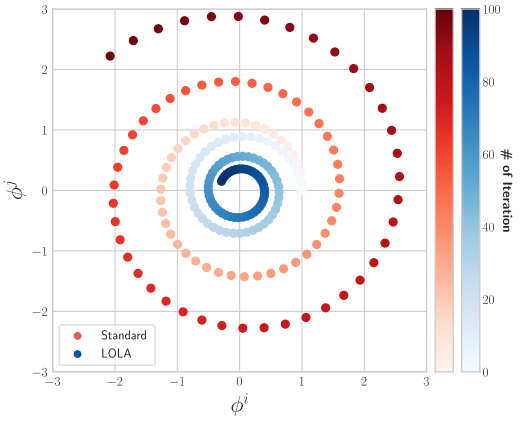

Figure 1: Learning paths on the zero-sum game. The standard approach that maximizes the value function in Equation (1) with the stationary assumption diverges, resulting in worse performance for both agents. In contrast, an approach that considers the learning process of the other agents, such as LOLA (Foerster et al. 2017a), converges to the equilibrium.

The following inner-loop optimization updates from $\boldsymbol{\theta}$ to $\boldsymbol{\phi}$:

$$\begin{aligned} \phi^i &:= \theta^i + \alpha \nabla_{\theta^i} \mathbb{E}_{\tau_{\boldsymbol{\theta}} \sim P(\tau_{\boldsymbol{\theta}} | \boldsymbol{\theta})} \Big[G_0^i(\tau_{\boldsymbol{\theta}}) \Big] \\ \boldsymbol{\phi^{-i}} &:= \boldsymbol{\theta^{-i}} + \alpha \nabla_{\boldsymbol{\theta^{-i}}} \mathbb{E}_{\tau_{\boldsymbol{\theta}} \sim P(\tau_{\boldsymbol{\theta}} | \boldsymbol{\theta})} \Big[\boldsymbol{G_0^{-i}}(\tau_{\boldsymbol{\theta}}) \Big], \end{aligned} \tag{3}$$

where α is a learning rate. Differentiating Equation (2) with respect to θ^i results in the meta-policy gradient theorem:

$$\begin{aligned} &\nabla_{\theta^i} V_{\boldsymbol{\theta}, \boldsymbol{\phi}}^i(s_0) = \\ &\mathbb{E}_{\tau_{\boldsymbol{\theta}} \sim P(\tau_{\boldsymbol{\theta}} | \boldsymbol{\theta})} \Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}} \sim P(\tau_{\boldsymbol{\phi}} | \boldsymbol{\phi})} \big[\nabla_{\theta^i} \log \pi(\tau_{\boldsymbol{\theta}} | \theta^i) G_0^i(\tau_{\boldsymbol{\phi}}) \big] \Big] + \\ &\mathbb{E}_{\tau_{\boldsymbol{\theta}} \sim P(\tau_{\boldsymbol{\theta}} | \boldsymbol{\theta})} \Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}} \sim P(\tau_{\boldsymbol{\phi}} | \boldsymbol{\phi})} \big[\underbrace{\nabla_{\theta^i} \log \pi(\tau_{\boldsymbol{\phi}} | \phi^i)}_{\text{Self-Shaping Term}} G_0^i(\tau_{\boldsymbol{\phi}}) \big] \Big] \end{aligned} \tag{4}$$

Intuitively, the meta-policy gradient is searching for initial parameters θ^i such that the inner-loop optimization based on $\tau_{\boldsymbol{\theta}}$ results in adapted parameters ϕ^i that can perform better than opponents with parameters $\boldsymbol{\phi^{-i}}$. In particular, the last term $\nabla_{\theta^i} \log \pi(\tau_{\boldsymbol{\phi}} | \phi^i)$ explicitly differentiates through $\log \pi(\tau_{\boldsymbol{\phi}} | \phi^i)$ with respect to θ^i, which meta-agent i can account for the impact of θ^i on its adapted parameters ϕ^i. As such, we name the term as the *self-shaping term*.

Learning with opponent shaping. Learning with Opponent Learning Awareness (LOLA) (Foerster et al. 2017a) optimizes the following value function:

$$V_{\{\phi^i, \boldsymbol{\phi^{-i}} + \boldsymbol{\Delta \phi^{-i}}(\tau_{\boldsymbol{\phi}})\}}^i(s_0), \tag{5}$$

where $\boldsymbol{\Delta \phi^{-i}}(\tau_{\boldsymbol{\phi}})$ is the predicted learning step in the other agents' policies. LOLA assumes the first-order Taylor expansion and focuses on the opponent shaping capability, resulting in the following LOLA policy gradient:

$$\begin{aligned} &\nabla_{\phi^i} V_{\{\phi^i, \boldsymbol{\phi^{-i}} + \boldsymbol{\Delta \phi^{-i}}(\tau_{\boldsymbol{\phi}})\}}^i(s_0) \approx \\ &\nabla_{\phi^i} V_{\boldsymbol{\phi}}^i(s_0) + \big(\nabla_{\boldsymbol{\phi^{-i}}} V_{\boldsymbol{\phi}}^i(s_0) \big) \big(\nabla_{\phi^i} \boldsymbol{\Delta \phi^{-i}}(\tau_{\boldsymbol{\phi}}) \big). \end{aligned} \tag{6}$$

The LOLA agent learns to shape the learning dynamics of the other agents through the last term $\nabla_{\phi^i} \boldsymbol{\Delta \phi^{-i}}(\tau_{\boldsymbol{\phi}})$.

Remark 1. *Failure to consider the learning process of the other agents can result in divergence of learning objectives.*

For instance, consider a stateless zero-sum game playing between two agents. Agents i and j maximize simple value functions $V^i_{\boldsymbol{\phi}} = \phi^i\phi^j$ and $V^j_{\boldsymbol{\phi}} = -\phi^i\phi^j$ respectively, where $\phi^i, \phi^j \in \mathbb{R}$. In this game, there exists a unique Nash equilibrium at the origin (i.e., $\{\phi^i, \phi^j\} = \{0, 0\}$). We compare: 1) the standard approach that optimizes the value function in Equation (1) with the stationary assumption and 2) an approach that considers the learning process of others, such as the LOLA method in Equation (6). As Figure 1 shows, the standard approach diverges further from the equilibrium, resulting in worse results for both agents. The cause of the failure is the stationary assumption that each agent assumes its opponent has the same behavior in the future. In contrast, by considering the learning process of the opponent, the LOLA approach converges to the equilibrium.

Learning to Learn in Multiagent Reinforcement Learning

We propose a new multiagent reinforcement learning framework based on meta-learning. Our meta-agent learns to consider the impact of its initial parameters on its adapted parameters *while* actively influence the opponents' future policies. It is important to account for both of its and the other agents' adaptation to better address the non-stationarity problem since not only the meta-agent but also its fellow agents are simultaneously learning in multiagent settings. In this section, we first devise a meta-multiagent policy gradient theorem and show that both the self-shaping and opponent-shaping terms are inherently included in our policy gradient. We then present a probabilistic model view to explain an underlying methodology in our framework.

Theorem 1. *(Meta-multiagent policy gradient theorem) Policy gradient for meta-agent i learning in multiagent settings has the following form:*

$$\begin{aligned}
&\nabla_{\theta^i} V^i_{\boldsymbol{\theta},\boldsymbol{\phi}}(s_0) = \\
&\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}}\sim P(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi})}\big[\nabla_{\theta^i}\log\pi(\tau_{\boldsymbol{\theta}}|\theta^i)G^i_0(\tau_{\boldsymbol{\phi}})\big]\Big] + \\
&\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}}\sim P(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi})}\big[\underbrace{\nabla_{\theta^i}\log\pi(\tau_{\boldsymbol{\phi}}|\phi^i)}_{\textit{Self-Shaping Term}}G^i_0(\tau_{\boldsymbol{\phi}})\big]\Big] + \\
&\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}}\sim P(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi})}\big[\underbrace{\nabla_{\theta^i}\log\boldsymbol{\pi}(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi^{-i}})}_{\textit{Opponent-Shaping Term}}G^i_0(\tau_{\boldsymbol{\phi}})\big]\Big].
\end{aligned} \tag{7}$$

Proof. We begin our derivation from the meta-value function defined in Equation (2). We expand the meta-value function with the state-action value and joint actions, assuming the conditional independence between agents' actions:

$$\begin{aligned}
&V^i_{\boldsymbol{\theta},\boldsymbol{\phi}}(s_0) \\
&= \mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}}\sim P(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi})}\big[G^i_0(\tau_{\boldsymbol{\phi}})\big]\Big] \\
&= \mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[V^i_{\boldsymbol{\phi}}(s_0)\Big] \\
&= \mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\sum_{a^i}\pi(a^i|s_0,\phi^i)\sum_{\boldsymbol{a^{-i}}}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s_0,\boldsymbol{\phi^{-i}})Q^i_{\boldsymbol{\phi}}(s_0,\boldsymbol{a})\Big],
\end{aligned} \tag{8}$$

where $\boldsymbol{a} = \{a^i, \boldsymbol{a^{-i}}\}$. Taking the gradient of Equation (8) with respect to θ^i results in:

$$\begin{aligned}
&\nabla_{\theta^i} V^i_{\boldsymbol{\theta},\boldsymbol{\phi}}(s_0) \\
&= \nabla_{\theta^i}\Big[\sum_{\tau_{\boldsymbol{\theta}}} P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})\sum_{a^i}\pi(a^i|s_0,\phi^i)\sum_{\boldsymbol{a^{-i}}}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s_0,\boldsymbol{\phi^{-i}})Q^i_{\boldsymbol{\phi}}(s_0,\boldsymbol{a})\Big].
\end{aligned} \tag{9}$$

We note both ϕ^i and $\boldsymbol{\phi^{-i}}$ depend on θ^i because the inner-loop optimizations in Equation (3), which output $\boldsymbol{\phi}$, are a function of trajectories $\tau_{\boldsymbol{\theta}}$ affected by θ^i. Specifically, we can write the gradients in the inner-loop optimizations based on the multiagent stochastic policy gradient (Wei et al. 2018):

$$\begin{aligned}
&\nabla_{\theta^i}\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[G^i_0(\tau_{\boldsymbol{\theta}})\Big] = \nabla_{\theta^i}V^i_{\boldsymbol{\theta}}(s_0) \\
&= \sum_s \rho_{\boldsymbol{\theta}}(s)\sum_{a^i}\nabla_{\theta^i}\pi(a^i|s,\theta^i)\sum_{\boldsymbol{a^{-i}}}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s,\boldsymbol{\theta^{-i}})Q^i_{\boldsymbol{\theta}}(s,\boldsymbol{a}) \\
&\nabla_{\boldsymbol{\theta^{-i}}}\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\boldsymbol{G^{-i}_0}(\tau_{\boldsymbol{\theta}})\Big] = \nabla_{\boldsymbol{\theta^{-i}}}\boldsymbol{V^{-i}_{\boldsymbol{\theta}}}(s_0) \\
&= \sum_s \rho_{\boldsymbol{\theta}}(s)\sum_{\boldsymbol{a^{-i}}}\nabla_{\boldsymbol{\theta^{-i}}}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s,\boldsymbol{\theta^{-i}})\sum_{a^i}\pi^i(a^i|s,\theta^i)\boldsymbol{Q^{-i}_{\boldsymbol{\theta}}}(s,\boldsymbol{a}),
\end{aligned} \tag{10}$$

where $\rho_{\boldsymbol{\theta}}$ is the stationary distribution under the joint policy with parameters $\boldsymbol{\theta}$. Because the other agents' inner-loop gradients $\nabla_{\boldsymbol{\theta^{-i}}}\boldsymbol{V^{-i}_{\boldsymbol{\theta}}}(s_0)$ are a function of θ^i, there exists the dependency between θ^i and $\boldsymbol{\phi^{-i}}$, which our approach aims to additionally exploit, unlike Al-Shedivat et al. (2018).

Continuing from Equation (9) and applying the product rule based on the dependencies (see Equation (10)):

$$\begin{aligned}
&\nabla_{\theta^i} V^i_{\boldsymbol{\theta},\boldsymbol{\phi}}(s_0) = \\
&\sum_{\tau_{\boldsymbol{\theta}}}\tfrac{\partial P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}{\partial\theta^i}\sum_{a^i}\pi(a^i|s_0,\phi^i)\sum_{\boldsymbol{a^{-i}}}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s_0,\boldsymbol{\phi^{-i}})Q^i_{\boldsymbol{\phi}}(s_0,\boldsymbol{a}) + \\
&\sum_{\tau_{\boldsymbol{\theta}}}P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})\sum_{a^i}\tfrac{\partial\pi(a^i|s_0,\phi^i)}{\partial\theta^i}\sum_{\boldsymbol{a^{-i}}}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s_0,\boldsymbol{\phi^{-i}})Q^i_{\boldsymbol{\phi}}(s_0,\boldsymbol{a}) + \\
&\sum_{\tau_{\boldsymbol{\theta}}}P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})\sum_{a^i}\pi(a^i|s_0,\phi^i)\sum_{\boldsymbol{a^{-i}}}\tfrac{\partial\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s_0,\boldsymbol{\phi^{-i}})}{\partial\theta^i}Q^i_{\boldsymbol{\phi}}(s_0,\boldsymbol{a}) + \\
&\sum_{\tau_{\boldsymbol{\theta}}}P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})\sum_{a^i}\pi(a^i|s_0,\phi^i)\sum_{\boldsymbol{a^{-i}}}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s_0,\boldsymbol{\phi^{-i}})\tfrac{\partial Q^i_{\boldsymbol{\phi}}(s_0,\boldsymbol{a})}{\partial\theta^i}.
\end{aligned} \tag{11}$$

Repeatedly unrolling the derivative of the Q-function $\partial Q^i_{\boldsymbol{\phi}}(s_0,\boldsymbol{a})/\partial\theta^i$ yields (proof omitted for clarity):

$$\begin{aligned}
&\nabla_{\theta^i} V^i_{\boldsymbol{\theta},\boldsymbol{\phi}}(s_0) = \\
&\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\nabla_{\theta^i}\log\pi(\tau_{\boldsymbol{\theta}}|\theta^i)\sum_{a^i}\pi(a^i|s_0,\phi^i)\times \\
&\qquad\qquad\sum_{\boldsymbol{a^{-i}}}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s_0,\boldsymbol{\phi^{-i}})Q^i_{\boldsymbol{\phi}}(s_0,\boldsymbol{a})\Big] + \\
&\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\sum_s\rho_{\boldsymbol{\phi}}(s)\sum_{a^i}\nabla_{\theta^i}\pi(a^i|s,\phi^i)\times \\
&\qquad\qquad\sum_{\boldsymbol{a^{-i}}}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s,\boldsymbol{\phi^{-i}})Q^i_{\boldsymbol{\phi}}(s,\boldsymbol{a})\Big] + \\
&\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\sum_s\rho_{\boldsymbol{\phi}}(s)\sum_{\boldsymbol{a^{-i}}}\nabla_{\theta^i}\boldsymbol{\pi}(\boldsymbol{a^{-i}}|s,\boldsymbol{\phi^{-i}})\times \\
&\qquad\qquad\sum_{a^i}\pi(a^i|s,\phi^i)Q^i_{\boldsymbol{\phi}}(s,\boldsymbol{a})\Big].
\end{aligned} \tag{12}$$

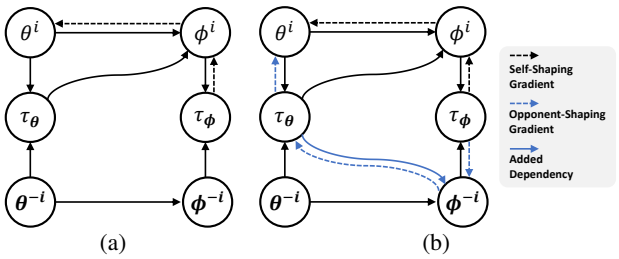

Figure 2: **(a)** Probabilistic graph in Al-Shedivat et al. (2018). **(b)** Probabilistic graph in ours. Unlike Al-Shedivat et al. (2018), our framework considers to actively change the future distribution of the opponent policies.

Finally, we summarize and express in expectations:

$$\nabla_{\theta^i} V^i_{\boldsymbol{\theta},\boldsymbol{\phi}}(s_0) =$$

$$\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}}\sim P(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi})}\big[\nabla_{\theta^i}\log\pi(\tau_{\boldsymbol{\theta}}|\theta^i)G^i_0(\tau_{\boldsymbol{\phi}})\big]\Big]+$$

$$\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}}\sim P(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi})}\big[\underbrace{\nabla_{\theta^i}\log\pi(\tau_{\boldsymbol{\phi}}|\phi^i)}_{\text{Self-Shaping Term}}G^i_0(\tau_{\boldsymbol{\phi}})\big]\Big]+$$

$$\mathbb{E}_{\tau_{\boldsymbol{\theta}}\sim P(\tau_{\boldsymbol{\theta}}|\boldsymbol{\theta})}\Big[\mathbb{E}_{\tau_{\boldsymbol{\phi}}\sim P(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi})}\big[\underbrace{\nabla_{\theta^i}\log\boldsymbol{\pi}(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi}^{-\boldsymbol{i}})}_{\text{Opponent-Shaping Term}}G^i_0(\tau_{\boldsymbol{\phi}})\big]\Big].\ \square$$

Probabilistic model perspective. Probabilistic models for Al-Shedivat et al. (2018) and ours are shown in Figure 2a and Figure 2b, respectively. As shown by the self-shaping term's gradient direction in Figure 2a, meta-agent i can optimize its initial policy parameters θ^i by accounting for the impact of θ^i on its updated parameters ϕ^i and adaptation performance $G^i_0(\tau_{\boldsymbol{\phi}})$. However, the approach considers the evolving opponent as an external factor that cannot be influenced by the meta-agent, as indicated by the absence of the dependence between $\tau_{\boldsymbol{\theta}}$ and $\boldsymbol{\phi}^{-\boldsymbol{i}}$ in Figure 2a. As a result, the meta-agent loses an opportunity to learn to influence the opponents' future policies. By contrast, our framework aims to additionally exploit the sequential dependency between the agent's current policy θ^i and the opponents' future policies $\boldsymbol{\phi}^{-\boldsymbol{i}}$ through $\tau_{\boldsymbol{\theta}}$. In a probabilistic model view, our objective corresponds to considering an additional dependency between $\tau_{\boldsymbol{\theta}}$ and $\boldsymbol{\phi}^{-\boldsymbol{i}}$ in Figure 2b. Thanks to the added dependency, meta-agent i now can optimize θ^i by considering both the self-shaping and opponent-shaping gradients.

Remark 2. *The opponent-shaping term $\nabla_{\theta^i}\log\boldsymbol{\pi}(\tau_{\boldsymbol{\phi}}|\boldsymbol{\phi}^{-\boldsymbol{i}})$ in Equation (7) is closely related to the term $\nabla_{\phi^i}\boldsymbol{\Delta}\boldsymbol{\phi}^{-\boldsymbol{i}}(\tau_{\boldsymbol{\phi}})$ in the LOLA policy gradient (Equation (6)).*

Both terms consider the impact of the agent's current behaviors on its opponents' future policies. We note main differences between our approach and LOLA: 1) while LOLA only focuses on the opponent's adaptation, our optimization additionally includes the self-shaping term that the meta-agent learns to consider its adaptation and 2) our approach can adapt quickly compared to LOLA (i.e., needs a fewer number of episodes) thanks to the meta-learning optimization. Hence, our framework unifies the benefits of both works by Al-Shedivat et al. (2018) and Foerster et al. (2017a).

Table 1: Comparisons between related approaches

Approach	Self Shaping	Best Response Adaptation	Opponent Shaping
Zhang and Lesser (2010)	✗	✓	✗
Foerster et al. (2017a)	✗	✗	✓
Letcher et al. (2019)	✗	✓	✓
Foerster et al. (2018)	✗	✓	✓
Al-Shedivat et al. (2018)	✓	✓	✗
Ours	✓	✓	✓

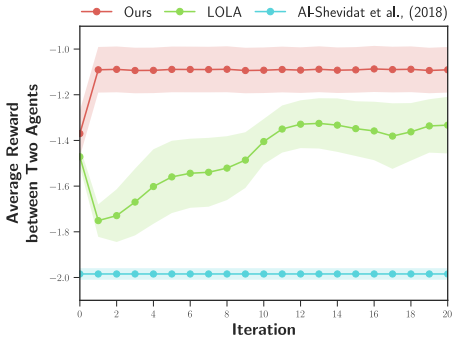

Figure 3: Results in IPD during meta-testing. Mean and 95% confidence interval computed for 20 random seeds are shown.

Related Work

Other related approaches also consider the learning process of the other agents. The work by Zhang and Lesser (2010) learns the best response adaptation to the anticipated future policy of a fellow agent. However, the work fails to consider the shaping of the opponent's learning dynamics. Another work by Letcher et al. (2019) interpolates between Zhang and Lesser (2010) and LOLA to guarantee convergence while shaping the opponent's future policy. In a different direction, the work by Foerster et al. (2018) extends LOLA and directly computes accurate higher-order derivatives based on the Differentiable Monte-Carlo Estimator. However, these approaches only account for the learning process of the other agents and fail to consider the impact on its adaptation, as in the self-shaping term. We summarize main differences between related works in Table 1.

Evaluation

Setup. We evaluate the performance of our approach on the iterative prisoner dilemma (IPD) (Myerson 1991). We construct a distribution of tasks, where each task in multiagent settings corresponds to a stochastic game. Specifically, we differ the payoff table between tasks by adding a uniform noise to the IPD payoff table. We then train two meta-agents that optimize Equation (7) with meta-training tasks and use the learned initial parameters to adapt to meta-testing tasks.
Results. Figure 3 shows that the baseline of Al-Shedivat et al. (2018) fails to cooperate (i.e., average reward of -2) due to the absence of the opponent shaping. By contrast, our approach that considers both the self and opponent shaping succeeds to cooperate (i.e., average reward of -1). Also, our framework enables fast adaptation such that agents need only

one iteration to cooperate, unlike the non-meta learning baseline of LOLA, which requires many iterations to cooperate.

Conclusion

In this paper, we introduce the meta-multiagent policy gradient theorem that a meta-agent inherently considers the impact of its initial policy on its adapted policy through the self-shaping term while exploits the learning dynamics of the other agents through the opponent-shaping term. Future works include empirical evaluations and convergence proofs.

Acknowledgements

Research funded by IBM (as part of the MIT-IBM Watson AI Lab initiative) and computational support through Amazon Web Services. Dong-Ki Kim was also supported by a Kwanjeong Educational Foundation Fellowship.

References

Al-Shedivat, M.; Bansal, T.; Burda, Y.; Sutskever, I.; Mordatch, I.; and Abbeel, P. 2018. Continuous adaptation via meta-learning in nonstationary and competitive environments. In *International Conference on Learning Representations*.

Buşoniu, L.; Babuška, R.; and De Schutter, B. 2010. *Multi-agent Reinforcement Learning: An Overview*. Berlin, Heidelberg: Springer Berlin Heidelberg. 183–221.

Foerster, J. N.; Chen, R. Y.; Al-Shedivat, M.; Whiteson, S.; Abbeel, P.; and Mordatch, I. 2017a. Learning with opponent-learning awareness. *CoRR* abs/1709.04326.

Foerster, J. N.; Farquhar, G.; Afouras, T.; Nardelli, N.; and Whiteson, S. 2017b. Counterfactual multi-agent policy gradients. *CoRR* abs/1705.08926.

Foerster, J.; Farquhar, G.; Al-Shedivat, M.; Rocktäschel, T.; Xing, E.; and Whiteson, S. 2018. DiCE: The infinitely differentiable Monte Carlo estimator. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 1524–1533. Stockholmsmässan, Stockholm Sweden: PMLR.

Grover, A.; Al-Shedivat, M.; Gupta, J.; Burda, Y.; and Edwards, H. 2018. Learning policy representations in multiagent systems. In Dy, J., and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 1802–1811. Stockholmsmässan, Stockholm Sweden: PMLR.

He, H.; Boyd-Graber, J.; Kwok, K.; and III, H. D. 2016. Opponent modeling in deep reinforcement learning. In Balcan, M. F., and Weinberger, K. Q., eds., *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, 1804–1813. New York, New York, USA: PMLR.

Hernandez-Leal, P.; Kaisers, M.; Baarslag, T.; and de Cote, E. M. 2017. A survey of learning in multiagent environments: Dealing with non-stationarity. *CoRR* abs/1707.09183.

Letcher, A.; Foerster, J.; Balduzzi, D.; Rocktäschel, T.; and Whiteson, S. 2019. Stable opponent shaping in differentiable games. In *International Conference on Learning Representations*.

Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, O. P.; and Mordatch, I. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Neural Information Processing Systems (NIPS)*, 6382–6393.

Myerson, R. B. 1991. *Game theory: Analysis of conflict*.

Omidshafiei, S.; Pazis, J.; Amato, C.; How, J. P.; and Vian, J. 2017. Deep decentralized multi-task multi-agent reinforcement learning under partial observability. In *International Conference on Machine Learning*, 2681–2690.

Papoudakis, G.; Christianos, F.; Rahman, A.; and Albrecht, S. V. 2019. Dealing with non-stationarity in multi-agent deep reinforcement learning. *CoRR* abs/1906.04737.

Raileanu, R.; Denton, E.; Szlam, A.; and Fergus, R. 2018. Modeling others using oneself in multi-agent reinforcement learning. In Dy, J., and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 4257–4266. Stockholmsmässan, Stockholm Sweden: PMLR.

Shapley, L. S. 1953. Stochastic games. *Proceedings of the National Academy of Sciences* 39(10):1095–1100.

Sutton, R. S., and Barto, A. G. 1998. *Introduction to Reinforcement Learning*. Cambridge, MA, USA: MIT Press, 1st edition.

Tuyls, K., and Weiss, G. 2012. Multiagent learning: Basics, challenges, and prospects. *Ai Magazine* 33:41–52.

Wei, E.; Wicke, D.; Freelan, D.; and Luke, S. 2018. Multiagent soft q-learning. *CoRR* abs/1804.09817.

Wen, Y.; Yang, Y.; Luo, R.; Wang, J.; and Pan, W. 2019. Probabilistic recursive reasoning for multi-agent reinforcement learning. In *International Conference on Learning Representations (ICLR)*.

Yang, Y.; Luo, R.; Li, M.; Zhou, M.; Zhang, W.; and Wang, J. 2018. Mean field multi-agent reinforcement learning. In Dy, J., and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 5571–5580. Stockholmsmässan, Stockholm Sweden: PMLR.

Zhang, C., and Lesser, V. R. 2010. Multi-agent learning with policy prediction. In *AAAI*.