PROBABILISTIC RECURSIVE REASONING FOR MULTI-AGENT REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Humans are capable of attributing latent mental contents such as beliefs, or intentions to others. The social skill is critical in everyday life to reason about the potential consequences of their behaviors so as to plan ahead. It is known that humans use this reasoning ability recursively, i.e. considering what others believe about their own beliefs. In this paper, we introduce a probabilistic recursive reasoning (PR2) framework for multi-agent reinforcement learning. Our hypothesis is that it is beneficial for each agent to account for how the opponents would react to its future behaviors. Under the PR2 framework, we adopt variational Bayes methods to approximate the opponents' conditional policy, to which each agent finds the best response and then improve their own policy. We develop decentralized-training-decentralized-execution algorithms, PR2-Q and PR2-Actor-Critic, that are proved to converge in the self-play scenario when there is one Nash equilibrium. Our methods are tested on both the matrix game and the differential game, which have a non-trivial equilibrium where common gradient-based methods fail to converge. Our experiments show that it is critical to reason about how the opponents believe about what the agent believes. We expect our work to contribute a new idea of modeling the opponents to the multi-agent reinforcement learning community.

1 INTRODUCTION

In the long journey of creating artificial intelligent (AI) that mimics human intelligence, a hallmark of an AI agent is its capabilities of understanding and interacting with other agents (Lake et al., 2017). At the cognitive level, the real-world intelligent entities (e.g. rats, humans) are born to be able to reason about various properties of interests of others (Tolman, 1948; Pfeiffer & Foster, 2013). Those interests usually indicates unobservable mental state including desires, beliefs, and intentions (Premack & Woodruff, 1978; Gopnik & Wellman, 1992). In everyday life, people use this inborn ability to reason about others' behaviors (Gordon, 1986), plan effective interactions (Gallese & Goldman, 1998), or match with the folk psychology (Dennett, 1991). It is known that people can use this reasoning ability recursively; that is, they engage in considering what others believe about their own beliefs. A number of human social behaviors have been profiled by the recursion reasoning ability (Pynadath & Marsella, 2005). Behavioral game theorist and experimental psychologist believe that reasoning recursively is a tool of human cognition that is equipped with evolutionary advantage (Camerer et al., 2004; 2015; Goodie et al., 2012; Robalino & Robson, 2012).

Traditional approach of constructing the models of other agents, also known as opponent modeling, has a rich history in the multi-agent learning (Shoham et al., 2007; Albrecht & Stone, 2018). Even though equipped with modern machine learning methods that could enrich the representation of the opponent's behaviors (He et al., 2016), those algorithms tend to only work either under limited types of scenarios (e.g. mean-field games (Yang et al., 2018)), pre-defined opponent strategies (e.g. Tit-fot-Tat in iterated Prisoner's Dilemma (Foerster et al., 2018)), or in cases where opponents are assumed to constantly return to the same strategy (Da Silva et al., 2006). Recently, a promising methodology from game theory – recursive reasoning – has become popular in opponent modeling (Gmytrasiewicz & Durfee, 2000; Camerer et al., 2004; Gmytrasiewicz & Doshi, 2005; De Weerd et al., 2013b). Similar to the way of thinking from humans, recursive reasoning refers to the belief reasoning process where each agent considers the reasoning process of other agents, based on which it expects to make better decisions. Importantly, it allows an opponent to reason about the modeling agent rather than being a fixed type; the process can therefore be nested in a form as "I believe

that you believe that I believe ...". Despite some initial trails (Gmytrasiewicz & Doshi, 2005; Von Der Osten et al., 2017), there has been little work that tries to adopt this idea into the multi-agent deep reinforcement learning (DRL) setting. One main reason is that computing the optimal policy is prohibitively expensive (Doshi & Gmytrasiewicz, 2006; Seuken & Zilberstein, 2008).

In this paper, we introduce a probabilistic recursive reasoning (PR2) framework for multi-agent DRL tasks. Unlike previous work on opponent modeling, each agent here is to consider how the opponents would react to its potential behaviors, before it tries to find the best response for its own decision making. By employing variational Bayes methods to model the uncertainty of opponents' conditional policies, we develop decentralized-training-decentralized-execution algorithms, PR2-Q and PR2-Actor-Critic, and prove the convergence in the self-play scenario when there is only one Nash equilibrium. Our methods are tested on the matrix game and the differential game. The games come with a non-trivial equilibrium where conventional gradient-based methods find challenging. We compare against multiple strong baselines. The results justify the unique value provided by agent's recursive reasoning capability throughout the learning. We expect our work to offer a new angel on incorporating conditional opponent modeling into the multi-agent DRL context.

2 Related Work

Game theorists take initiatives in modeling the recursive reasoning procedures (Harsanyi, 1962; 1967). Since then, alternative approaches, including logics-based models (Bolander & Andersen, 2011; Muise et al., 2015) or graphical models (Doshi et al., 2009; Gal & Pfeffer, 2003; 2008), have been adopted. Recently, the idea of Theory of Mind (ToM) (Goldman et al., 2012) from cognitive science becomes popular. An example of ToM is the "Recursive Modeling Method" (RMM) (Gmytrasiewicz et al., 1991; Gmytrasiewicz & Durfee, 1995; 2000), which incorporates the agent's uncertainty about opponent's exact model, payoff, and recursion depth. However, these methods follow the decision-theoretic approaches, and are studied in the limited context of one-shot games. The environment is relatively simple and the opponents are not RL agents.

The Interactive POMDP (I-POMDP) (Gmytrasiewicz & Doshi, 2005) implements the idea of ToM to tackle the multi-agent RL problems. It extends the partially observed MDP (Sondik, 1971) by introducing an extra space of models of other agents into the MDP; as such, an agent can build belief models about how it believes other agents know and believe. Despite the added flexibility, I-POMDP has limitations in its solvability (Seuken & Zilberstein, 2008). Solving I-POMDP with N models in each recursive level with K maximum level equals to solving $\mathcal{O}(N^K)$ PODMPs. Such inherent complexity requires high precision on the approximation solution methods, including particle filtering (Doshi & Gmytrasiewicz, 2009), value iteration (Doshi & Perez, 2008), or policy iteration (Sonu & Doshi, 2015). Out work is different from I-POMDP in that we do not adjust the MDP; instead, we provide a probabilistic framework to implement the recursive reason in the MDP. We approximate the opponent's conditional policy through variational Bayes methods. The induced PR2-Q and PR2-AC algorithms are model-free and can practically be used as the replacement to other multi-agent RL algorithms such as MADDPG (Lowe et al., 2017).

Our work can also be tied into the study of opponent modeling (OM) Albrecht & Stone (2018). OM is all about shaping the anticipated movements of the other agents. Traditional OM can be regarded as level-0 recursive reasoning in that OM methods model how the opponent behaves based on the history, but not how the opponent would behave based on what I would behave. In general, OM methods have two major limitations. One is that OM tends to work with a pre-defined target of opponents; for example, fictitious play (Brown, 1951) and joint-action learners (Claus & Boutilier, 1998) require opponents play stationary strategies, Nash-Q (Hu & Wellman, 2003) require all agents play towards the Nash equilibrium, so do Correlated Q-learning (Greenwald et al., 2003), Minimax-Q (Littman, 1994), and Friend-or-foe Q (Littman, 2001). These algorithms become invalid if the opponents change their types of policy. The other major limitation is that OM algorithms require to know the exact (Nash) equilibrium policy of the opponent during training. Typical examples include the series of *WoLF* models (Bowling, 2005; Bowling & Veloso, 2001a; 2002) or the Nash-Q learning (Hu & Wellman, 2003), both of which require the Nash Equilibrium at each stage game to update the Q-function. By contrast, our proposed methods, PR2-Q & PR2-AC, do not need to pre-define the type of the opponents. Neither do our methods require to know the equilibrium beforehand.

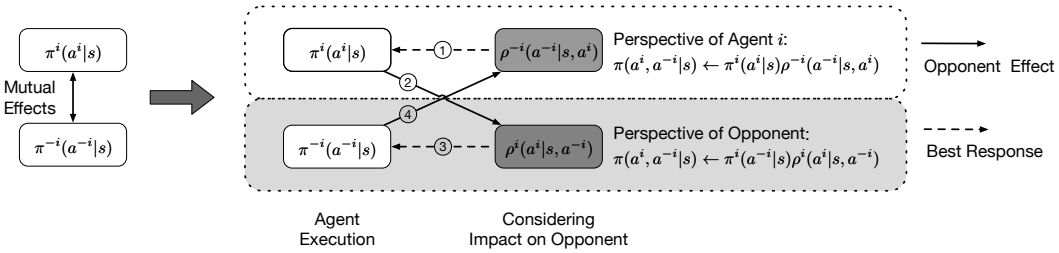


Figure 1: Probabilistic recursive reasoning framework. PR2 decouples the connections between agents by Eq. 3. ①: agent i takes the best response after considering all the potential consequences of opponents' actions given its own action a^i. ②: how agent i behaves in the environment serves as the prior for the opponents to learn how their actions would affect a^i. ③: similar to ①, opponents take the best response to agent i. ④: similar to ②, opponents' actions are the prior knowledge to agent i on estimating how a^i will affect the opponents. Looping from step 1 to 4 forms recursive reasoning.

Despite the recent success of applying deep RL algorithms on the single-agent discrete (Mnih et al., 2015) and continuous (Lillicrap et al., 2015) control problems, it is still challenging to transfer these methods into the multi-agent RL context. The reason is because learning independently while ignoring the others in the environment will simply break the theoretical guarantee of convergence (Tuyls & Weiss, 2012). A modern framework is to maintain a centralized critic (i.e. Q-network) during training, e.g. MADDPG (Lowe et al., 2017) and multi-agent soft Q-learning (Wei et al., 2018); however, they require strong assumptions that the parameters of agent policies are fully observable (so does LOLA by Foerster et al. (2018)), letting alone the centralized Q-network potentially prohibits the algorithms from scaling up. By contrast, our approach employs decentralized training with no need to maintain a central critic; neither does it require to know the exact opponents' policies.

3 Preliminaries

For an n-agent stochastic game (Shapley, 1953), we define a tuple $(\mathcal{S}, \mathcal{A}^1, \ldots, \mathcal{A}^n, r^1, \ldots, r^n, p, \gamma)$, where $\mathcal{S}$ denotes the state space, p is the distribution of the initial state, γ is the discount factor for future rewards, $\mathcal{A}^i$ and $r^i = r^i(s, a^i, a^{-i})$ are the action space and the reward function for agent $i \in \{1, \ldots, n\}$ respectively. Agent i chooses its action $a^i \in \mathcal{A}^i$ according to the policy $\pi^i_{\theta^i}(a^i|s)$ parameterized by θ^i conditioning on some given state $s \in \mathcal{S}$. Let us define the joint policy as the collection of all agents' policies π_θ with θ representing the joint parameter. It is convenient to interpret the joint policy from the perspective of agent i such that $\pi_\theta = (\pi^i_{\theta^i}(a^i|s), \pi^{-i}_{\theta^{-i}}(a^{-i}|s))$, where $a^{-i} = (a^j)_{j\neq i}$, $\theta^{-i} = (\theta^j)_{j\neq i}$, and $\pi^{-i}_{\theta^{-i}}(a^{-i}|s)$ is a compact representation of the joint policy of all complementary agents of i. At each stage of the game, actions are taken simultaneously. Each agent is presumed to pursue the maximal cumulative reward (Sutton et al., 1998), expressed as

$$\max \;\; \eta^i(\pi_\theta) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t r^i(s_t, a^i_t, a^{-i}_t)\right], \tag{1}$$

with (a^i_t, a^{-i}_t) sample from $(\pi^i_{\theta^i}, \pi^{-i}_{\theta^{-i}})$. Correspondingly, for the game with (infinite) time horizon, we can define the state-action Q-function by $Q^i_{\pi_\theta}(s_t, a^i_t, a^{-i}_t) = \mathbb{E}\left[\sum_{l=0}^{\infty} \gamma^l r^i(s_{t+l}, a^i_{t+l}, a^{-i}_{t+l})\right]$.

3.1 Non-correlated Factorization on the Joint Policy

In the multi-agent learning tasks, each agent can only control its own action; however, the resulting reward value depends on other agents' actions. The Q-function of each agent, $Q^i_{\pi_\theta}$, is subject to the joint policy π_θ consisting of all agents' policies. One common approach is to decouple the joint policy assuming conditional independence of actions from different agents (Albrecht & Stone, 2018):

$$\pi_\theta(a^i, a^{-i}|s) = \pi^i_{\theta^i}(a^i|s)\pi^{-i}_{\theta^{-i}}(a^{-i}|s). \tag{2}$$

The study regarding the topic of "centralized training with decentralized execution" in the deep RL domain, including MADDPG (Lowe et al., 2017), COMA (Foerster et al., 2017), MF-AC (Yang et al., 2018), Multi-Agent Soft-Q (Wei et al., 2018), and LOLA (Foerster et al., 2018), can be classified into this category (see more clarifications in Appendix B). Although the non-correlated factorization of the joint policy simplifies the algorithm, this simplication is vulnerable because it ignores the agents' connections, e.g. impacts of one agent's action on other agents, and the subsequent reactions from other agents. One might argue that during training, the joint Q-function should potentially

guide each agent to learn to consider and act for the mutual interests of all the agents; nonetheless, a counter-example is that the non-correlated policy could not even solve the simplest two-player zero-sum differential game where two agents act in x and y with the reward functions defined by $(xy, -xy)$. In fact, by following Eq. 2, both agents are reinforced to trace a cyclic trajectory that never converge to the equilibrium (Mescheder et al., 2017).

It is worth clarifying that the idea of non-correlated policy is still markedly different from the independent learning (IL). IL is a naive method that completely ignore other agents' behaviors. The objective of agent i is simplified to $\eta^i(\pi_{\theta^i})$, depending only on i's own policy π_{θ^i} compared to Eq. 1. As Lowe et al. (2017) has pointed out, in IL, the probability of taking a gradient step in the correct direction decreases exponentially with the increasing number of agents, letting alone the major issue of the non-stationary environment due to the independence assumption (Tuyls & Weiss, 2012).

4 Multi-Agent Probabilistic Recursive Reasoning

In the previous section, we have shown the weakness of the learning algorithms that build on the non-correlated factorization on the joint policy. Here we introduce the probabilistic recursive reasoning approach that aims to capture how the opponents believe about what the agent believes. Under such setting, we devise a new multi-agent policy gradient theorem. We start from assuming the true opponent conditional policy $\pi^{-i}_{\theta^{-i}}$ is given, and then move onward to the practical case where it is approximated through variational inference.

4.1 Probabilistic Recursive Reasoning

The issue on the non-correlated factorization is that it fails to help each agent to consider the consequence of its action on others, which could lead to the ill-posed behaviors in the multi-agent learning tasks. On the contrary, people explicitly attribute contents such as beliefs, desires, and intentions to others in daily life. It is known that human beings are capable of using this ability recursively to make decisions. Inspired by this, here we integrate the concept of recursive reasoning into the joint policy modeling, and propose the new probabilistic recursive reasoning (PR2) framework. Specifically, we employ the nested process of belief reasoning where each agent simulates the reasoning process of other agents, thinking about how its action would affect others, and then make actions based on such predictions. The process can be nested in a form as "I believe [that you believe (that I believe)]". Here we start from considering the level-1 recursion, as psychologist have found that humans tend to reason on average at one or two level of recursion (Camerer et al., 2004), and levels higher than two do not provide significant benefits (De Weerd et al., 2013a;b; de Weerd et al., 2017). Based on this, we re-formulate the joint policy by

$$\pi_\theta(a^i, a^{-i}|s) = \underbrace{\pi^i_{\theta^i}(a^i|s)\pi^{-i}_{\theta^{-i}}(a^{-i}|s, a^i)}_{\text{Agent } i\text{'s perspective}} = \underbrace{\pi^{-i}_{\theta^{-i}}(a^{-i}|s)\pi^i_{\theta^i}(a^i|s, a^{-i})}_{\text{The opponents' perspective}}. \tag{3}$$

Similar ways of decomposition can also be found in dual learning (Xia et al., 2017) on machine translation. From the perspective of agent i, the first equality in Eq. 3 indicates that the joint policy can be essentially decomposed into two parts. The conditional part $\pi^{-i}_{\theta^{-i}}(a^{-i}|s, a^i)$ represents what actions would be taken by the opponents given the fact that the opponents know the current state of environment and agent i's action; this is based on what agent i believes other opponents might think about itself. Note that the way of thinking developed by agent i regarding how others would consider of itself is also shaped by opponents' original policy $\pi^{-i}_{\theta^{-i}}(a^{-i}|s)$, as this is also how the opponents actually act in the environment. Taking into account different potential actions that agent i thinks the opponents would take, agent i uses the marginal policy $\pi^i_{\theta^i}(a^i|s)$ to find the best response. To this end, a level-1 recursive procedure is established: $a^i \to a^{-i} \to a^i$. The same inference logic can be applied to the opponents from their perspectives, as shown in the second equality of Eq. 3.

Albeit intuitive, Eq. 3 may not be practical due to the requirement on the full knowledge regarding the actual conditional policy $\pi^{-i}_{\theta^{-i}}(a^{-i}|s, a^i)$. A natural solution is that one approximates the actual policy via a best-fit model from a family of distributions. We denote this family as $\rho^{-i}_{\phi^{-i}}(a^{-i}|s, a^i)$ with learnable parameter ϕ^{-i}. PR2 is probabilistic as it considers the uncertainty of modeling $\pi^{-i}_{\theta^{-i}}(a^{-i}|s, a^i)$. The reasoning structure is now established as shown in Fig. 1. With the recursive

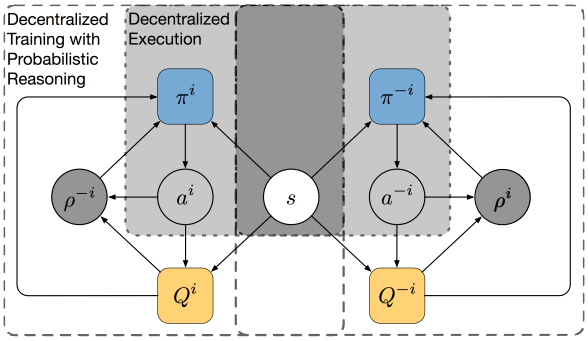


Figure 2: Diagram of multi-agent PR2 learning algorithms. It conducts decentralized training with decentralized execution. The light grey areas on two sides indicate decentralized execution for each agent. White areas give the decentralized learning procedures. All agents share the interaction experiences in the environment represented by dark area in the middle.

joint policy defined in Eq. 3, the n-agent learning task can therefore be formulated as

$$\arg\max_{\theta^i, \phi^{-i}} \; \eta^i \left(\pi^i_{\theta^i}(a^i|s) \rho^{-i}_{\phi^{-i}}(a^{-i}|s, a^i) \right), \tag{4}$$

$$\arg\max_{\theta^{-i}, \phi^i} \; \eta^{-i} \left(\pi^{-i}_{\theta^{-i}}(a^{-i}|s) \rho^i_{\phi^i}(a^i|s, a^{-i}) \right). \tag{5}$$

With the new learning protocol defined in Eq. 4 and 5, each agent now learns its own policy as well as the approximated conditional policy of other agents given its own actions. In such a way, both the agent and the opponents can keep track of the joint policy by $\pi^i_{\theta^i}(a^i|s)\rho^{-i}_{\phi^{-i}}(a^{-i}|s, a^i) \rightarrow \pi_\theta(a^i, a^{-i}|s) \leftarrow \pi^{-i}_{\theta^{-i}}(a^{-i}|s)\rho^i_{\phi^i}(a^i|s, a^{-i})$. Once converged, the resulting approximates satistfies: $\pi_\theta(a^i, a^{-i}|s) = \pi^i_{\theta^i}(a^i|s)\rho^{-i}_{\phi^{-i}}(a^{-i}|s, a^i) = \pi^{-i}_{\theta^{-i}}(a^{-i}|s)\rho^i_{\phi^i}(a^i|s, a^{-i})$, according to Eq. 3.

4.2 Probabilistic Recursive Reasoning Policy Gradient

Given the true opponent policy $\pi^{-i}_{\theta^{-i}}$ and that each agent tries to maximize its cumulative return in the stochastic game with the objective defined in Eq. 1, we establish the policy gradient theorem by accounting for the PR2 joint policy decomposition in Eq. 3.

Proposition 1. *In a stochastic game, under the recursive reasoning framework defined by Eq. 3, the update for the multi-agent recursive reasoning policy gradient method can be derived as follows:*

$$\nabla_{\theta^i} \eta^i = \mathbb{E}_{s \sim p, a^i \sim \pi^i} \left[\nabla_{\theta^i} \log \pi^i_{\theta^i}(a^i|s) \int_{a^{-i}} \pi^{-i}_{\theta^{-i}}(a^{-i}|s, a^i) Q^i(s, a^i, a^{-i}) \, \mathrm{d}a^{-i} \right]. \tag{6}$$

Proof. See Appendix B.2. ■

Proposition 1 states that each agent should improve its policy toward the direction of the best response after it takes into account all kinds of possibilities of how other agents would react if that action is taken. The term of $\pi^{-i}_{\theta^{-i}}(a^{-i}|s, a^i)$ can be regarded as the **posterior** estimation of agent i's belief about how the opponents would respond to his action a^i, given opponents' true policy $\pi^{-i}_{\theta^{-i}}(a^{-i}|s)$ serving as the **prior**. Note that compared to the direction of policy update in the conventional multi-agent policy gradient theorem (Wei et al., 2018), $\int_{a^{-i}} \pi^{-i}_{\theta^{-i}}(a^{-i}|s) Q^i(s, a^i, a^{-i}) \, \mathrm{d}a^{-i}$, the direction of the gradient update in PR2 is guided by the term $\int_{a^{-i}} \pi^{-i}_{\theta^{-i}}(a^{-i}|s, a^i) Q^i(s, a^i, a^{-i}) \, \mathrm{d}a^{-i}$.

In practice, agent i might not have access to the opponents' actual policy parameters θ^{-i}, it is often needed to approximate $\pi^{-i}_{\theta^{-i}}(a^{-i}|s, a^i)$ by $\rho^{-i}_{\phi^{-i}}(a^{-i}|s, a^i)$, thereby we propose Proposition 2.

Proposition 2. *In a stochastic game, under the recursive reasoning framework defined by Eq. 3, with the opponent policy approximated by $\rho^{-i}_{\phi^{-i}}(a^{-i}|s, a^i)$, the update for the multi-agent recursive reasoning policy gradient method can be formulated as follows:*

$$\nabla_{\theta^i} \eta^i = \mathbb{E}_{s \sim p, a^i \sim \pi^i} \left[\nabla_{\theta^i} \log \pi^i_{\theta^i}(a^i|s) \cdot \mathbb{E}_{a^{-i} \sim \rho^{-i}_{\phi^{-i}}} \left[\frac{\pi^{-i}_{\theta^{-i}}(a^{-i}|s, a^i)}{\rho^{-i}_{\phi^{-i}}(a^{-i}|s, a^i)} Q^i(s, a^i, a^{-i}) \right] \right]. \tag{7}$$

Proof. Substituting the approximated model $\rho^{-i}_{\phi^{-i}}(a^{-i}|s, a^i)$ for the true policy $\pi^{-i}_{\theta^{-i}}$ in Eq. 6. ■

Proposition 2 raises an important point: the difference between decentralized training (algorithms that do not require the opponents' policies) with centralized learning (algorithms that require the opponents' policies) can in fact be quantified by a term of importance weights, similar to the connection between on-policy and off-policy methods. If we find a best-fit approximation such that $\rho^{-i}_{\phi^{-i}}(a^{-i}|s,a^i) \to \pi^{-i}_{\theta^{-i}}(a^{-i}|s,a^i)$, then Eq.7 collapses into Eq. 6.

Based on Proposition 2, we could provide multi-agent PR2 learning algorithm. As illustrated in Fig. 2, it is a decentralized-training-with-decentralized-execution algorithm. In this setting, agents share the experiences in the environment including state and historical joint actions, while each agent receive its rewards privately. Our method does not require the knowledge of other agents' policy parameters. We list the pseudo-code of PR2-AC and PR2-Q in Appendix A. Finally, one last piece missing is how to find the best-fit approximation of $\rho^{-i}_{\phi^{-i}}(a^{-i}|s,a^i)$.

4.3 Variational Inference on Opponent Conditional Policy

We adopt an optimization-based approximation to infer the unobservable $\rho^{-i}_{\phi^{-i}}(a^{-i}|s,a^i)$ via variational inference (Jordan et al., 1999). We first define the trajectory τ up to time t including the experiences of t consecutive time stages, i.e. $\tau = [(s_1, a^i_1, a^{-i}_1), \ldots, (s_t, a^i_t, a^{-i}_t)]$. In the probabilistic reinforcement learning (Levine, 2018), the probability of τ being generated can be derived as

$$p(\tau) = \left[p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a^i_t, a^{-i}_t)\right] \exp\left(\sum_{t=1}^{T} r^i(s_t, a_t, a^{-i}_t)\right). \tag{8}$$

Assuming the dynamics is fixed (i.e. the agent can not influence the environment transition probability), our goal is then to find the best approximation of $\pi^i_{\theta^i}(a^i_t|s_t)\rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t)$ such that the induced trajectory distribution $\hat{p}(\tau)$ can match with the true trajectory probability $p(\tau)$:

$$\hat{p}(\tau) = p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a^i_t, a^{-i}_t)\pi^i_{\theta^i}(a^i_t|s_t)\rho^{-i}_{\theta^{-i}}(a^{-i}_t|s_t,a^i_t). \tag{9}$$

In other words, we can optimize the opponents' policy $\rho^{-i}_{\phi^{-i}}$ via minimizing the KL-divergence, i.e.

$$\begin{aligned} D_{\mathrm{KL}}(\hat{p}(\tau)\|p(\tau)) &= -\mathbb{E}_{\tau\sim\hat{p}(\tau)}[\log p(\tau) - \log \hat{p}(\tau)] \\ &= -\sum_{t=1}^{t=T} E_{\tau\sim\hat{p}(\tau)}\left[r^i\left(s_t, a^i_t, a^{-i}_t\right) + \mathcal{H}\left(\pi^i_{\theta^i}\left(a^i_t|s_t\right)\rho^{-i}_{\phi^{-i}}\left(a^{-i}|s_t,a^i_t\right)\right)\right]. \end{aligned} \tag{10}$$

Besides the reward term, the objective introduces an additional term of the conditional entropy on the joint policy $\mathcal{H}\left(\pi^i_{\theta^i}\left(a^i_t|s_t\right)\rho^{-i}_{\phi^{-i}}\left(a^{-i}|s_t,a^i_t\right)\right)$ that potentially promotes the explorations for both the agent i's best response and the opponents' conditional policy. Note that the entropy here is conditioning not only on the state s_t but also on agent i's action. Minimizing Eq. 10 gives us:

Theorem 1. *The optimal Q-function for agent i that satisfies minimizing Eq. 10 is formulated as:*

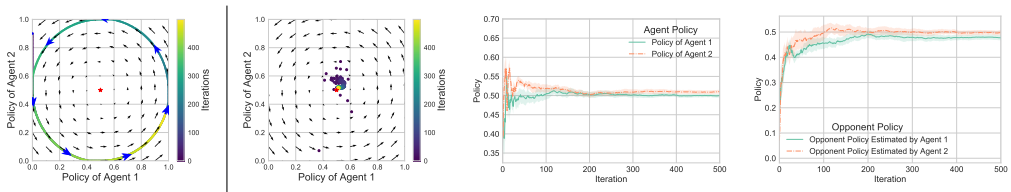
$$Q^i_{\pi_\theta}(s,a^i) = \log\int_{a^{-i}} \exp(Q^i_{\pi_\theta}(s,a^i,a^{-i}))\,\mathrm{d}a^{-i}. \tag{11}$$

And the corresponding optimal opponent conditional policy reads:

$$\rho^{-i}_{\phi^{-i}}(a^{-i}|s,a^i) = \exp(Q^i_{\pi_\theta}(s,a^i,a^{-i}) - Q^i_{\pi_\theta}(s,a^i)) \tag{12}$$

Proof. See Appendix C. ■

Theorem 1 states that the learning of $\rho^{-i}_{\phi^{-i}}(a^{-i}|s,a^i)$ can be further converted to minimizing the KL-divergence between the estimated policy $\rho^{-i}_{\phi^{-i}}$ and the *advantage* function: $D_{\mathrm{KL}}\left(\rho^{-i}_{\phi^{-i}}(a^{-i}|s,a^i)\|\exp(Q^i(s,a^i,a^{-i}) - Q^i(s,a^i))\right)$. We can obtain a solution to Eq. 12 by maintaining two Q-functions, and then iteratively update them. We prove the convergence under self-play when there is one equilibrium. This leads to a fixed-point iteration that resembles value iteration.



(a) IGA dynamics. (b) PR2-Q dynamics. (c) PR2-Q Agent Policies. (d) PR2-Q Opponent Policies

Figure 3: Learning paths on the iterated matrix game. a: IGA. b-d: PR2-Q.

Theorem 2. *In a symmetric game with only one equilibrium, and the equilibrium meets one of the conditions: 1) the global optimum, i.e.* $\mathbb{E}_{\pi_*}\left[Q_t^i(s)\right] \geq \mathbb{E}_{\pi}\left[Q_t^i(s)\right]$*; 2) a saddle point, i.e.* $\mathbb{E}_{\pi_*}\left[Q_t^i(s)\right] \geq \mathbb{E}_{\pi^i}\mathbb{E}_{\pi_*^{-i}}\left[Q_t^i(s)\right]$ *or* $\mathbb{E}_{\pi_*}\left[Q_t^i(s)\right] \geq \mathbb{E}_{\pi_*^i}\mathbb{E}_{\pi^{-i}}\left[Q_t^i(s)\right]$*; where* Q_* *and* π_* *are the equilibrium value function and policy, respectively. The PR2 soft value iteration operator defined by:*

$$\mathcal{T}Q^i(s,a^i,a^{-i}) \triangleq r^i(s,a^i,a^{-i}) + \gamma\mathbb{E}_{s',a^{i\prime}\sim p_s,\pi^i}\left[\log\int_{a^{-i\prime}}\exp(Q^i(s',a^{i\prime},a^{-i\prime}))\,\mathrm{d}a^{-i\prime}\right], \tag{13}$$

is a contraction mapping.

Proof. See Appendix D. ■

4.4 Sampling in continuous action space

In continuous controls, getting the actions from the opponent policy $\rho^{-i}_{\phi^{-i}}(a^{-i}|s,a^i)$ is challenging. In this work, we follow Haarnoja et al. (2017) to adopt the amortized Stein Variational Gradient Descent (SVGD) (Liu & Wang, 2016; Wang & Liu, 2016) in sampling from the soft Q-function. Compared to MCMC, Amortized SVGD is a computationally-efficient way to estimate $\rho^{-i}_{\phi^{-i}}(a^{-i}|s,a^i)$. Thanks to SVGD, agent i is able to reason about potential consequences of opponent bavhaviors $\int_{a^{-i}}\pi^{-i}_{\theta^{-i}}(a^{-i}|s,a^i)Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}$, and finally find the corresponding best response.

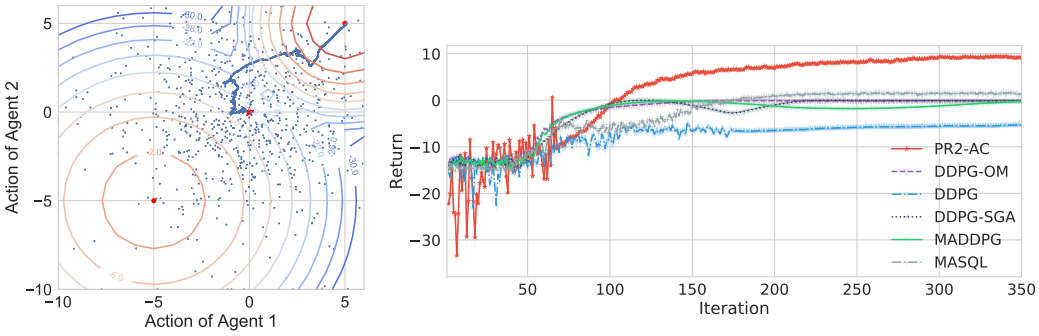
5 Experiments

We evaluate the performance of PR2 methods on the iterated matrix games, differential games, and particle world environment. Those games can by design have a non-trivial equilibrium that requires certain levels of intelligent reasonings between agents. We compared our algorithm with a series of baselines. In the matrix game, we compare against IGA (Infinitesimal Gradient Ascent) (Singh et al., 2000). In the differential games, the baselines from multi-agent learning algorithms are MASQL (Multi-Agent Soft-Q) (Wei et al., 2018) and MADDPG (Lowe et al., 2017). We also including independent learning algorithms implemented through DDPG (Lillicrap et al., 2015). To compare against traditional method of opponent modeling, we include one baseline that is based on DDPG but with one additional opponent modeling unit that is trained in an online and supervised way to learn the most recent opponent policy, which is then fed into the critic. Similar approach has been implemented by Rabinowitz et al. (2018) in realizing machine theory of mind. Besides, we applied centralized Symplectic Gradient Adjustment (SGA) (Balduzzi et al., 2018) optimization for DDPG agents (DDPG-SGA), which has recently been found to help converge to a local equilibrium quickly.

For the experiment settings, all the policies and Q-functions are parameterized by the MLP with 2 hidden layers, each with 100 units ReLU activation. The sampling network ξ for the $\rho^{-i}_{\phi^{-i}}$ in SGVD follows the standard normal distribution. In the iterated matrix game, we trained all the methods including the baselines for 500 iterations. In the differential game, we trained the agents for 350 iterations with 25 steps per iteration. For the actor-critic methods, we set the exploration noise to 0.1 in first 1000 steps, and the annealing parameters for PR2-AC and MASQL are set to 0.5 to balance between the exploration and acting as the best response.

5.1 Iterated Matrix Game

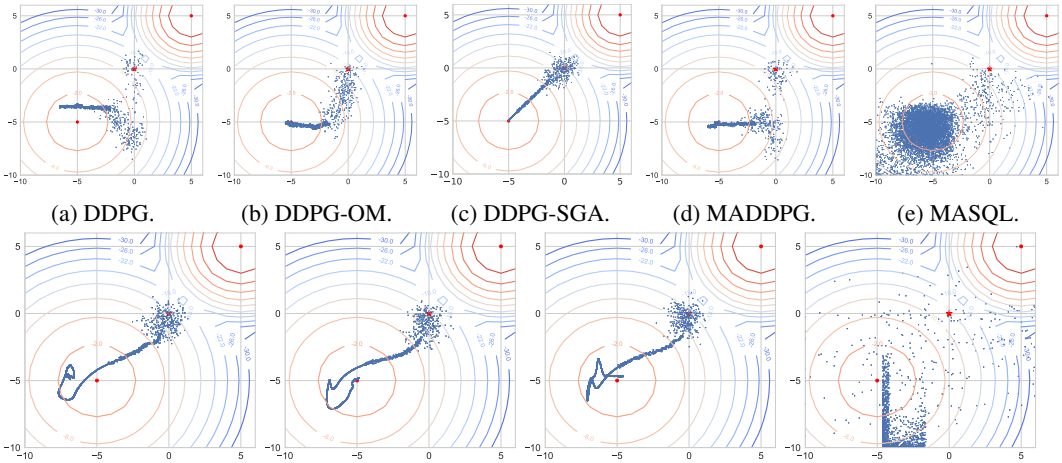
In the matrix game, the payoffs are defined by: $R^1 = \begin{bmatrix} 0 & 3 \\ 1 & 2 \end{bmatrix}$, and $R^2 = \begin{bmatrix} 3 & 2 \\ 0 & 1 \end{bmatrix}$. These exists the only Nash Equilibrium at $(0.5, 0.5)$. This game has been intensively investigated in multi-agent studies (Bowling & Veloso, 2001a;b). One reason is that in solving the Nash Equilibrium for this game, simply taking simultaneous gradient steps on both agent's value functions will present



(a) The learning path of PR2-AC vs. PR2-AC.

(b) The learning curves.

Figure 4: Max of Two Quadratic Game.



(a) DDPG. (b) DDPG-OM. (c) DDPG-SGA. (d) MADDPG. (e) MASQL.

(f) PR2-AC / DDPG. (g) PR2-AC / DDPG-OM. (h) PR2-AC / MADDPG. (i) PR2-AC / MASQL.

Figure 5: The learning path of Agent 1 (x-axis) vs. Agent 2 (y-axis).

the rotational behaviors on the gradient vector field; this leads to an endlessly iterative change of behaviors. Without considering the consequence of one agent's action on the other agent beforehand, it is challenging for both players to find the equilibrium. Similar issue has been found on training the GANs (Goodfellow et al., 2014; Mescheder et al., 2017)

The results are shown in Fig. 3. As expected, IGA fails to converge to the equilibrium but rotate around the equilibrium point. On the contrary, our method can find precisely the central equilibrium with a fully distributed fashion (see Fig. 3b). The convergence can also be justified by the agents' policies in Fig. 3c, and the opponent's policy that is maintained by each agent in Fig. 3d.

5.2 Differential Game

We adopt the same differential game, the Max of Two Quadratic Game, as Panait et al. (2006); Wei et al. (2018). The agents have continuous action space of $[-10, 10]$. Each agent's reward depends on the joint action following the equations: $r^1\left(a^1, a^2\right) = r^2\left(a^1, a^2\right) = \max\left(f_1, f_2\right)$, where $f_1 = 0.8 \times [-(\frac{a^1+5}{3})^2 - (\frac{a^2+5}{3})^2]$, $f_2 = 1.0 \times [-(\frac{a^1-5}{1})^2 - (\frac{a^2-5}{1})^2] + 10$. The task poses a great challenge to general gradient-based algorithms because gradient tends to points to the sub-optimal solution. The reward surface is shown in Fig. 4a; there is a local maximum 0 at $(-5, -5)$ and a global maximum 10 at $(5, 5)$, with a deep valley staying in the middle. If the agents' policies are initialized to $(0, 0)$ (the red starred point) that lies within the basin of the left local maximum, the gradient based methods would tend to fail to find the global maximum equilibrium point due to the valley blocking the upper right area. The pathology of finding a suboptimal Nash equilibrium is also called *relative over-generalization* (Wei & Luke, 2016).

We present the results in Fig. 4b, PR2-AC shows superior performance that manages to converge to the global equilibrium, while all the other baselines fall into the local basin on the left, except that the MASQL has small chance to find the optimal point. On top of the convergence result, it is worth noting that as the temperature annealing is required for energy-based RL methods, the learning

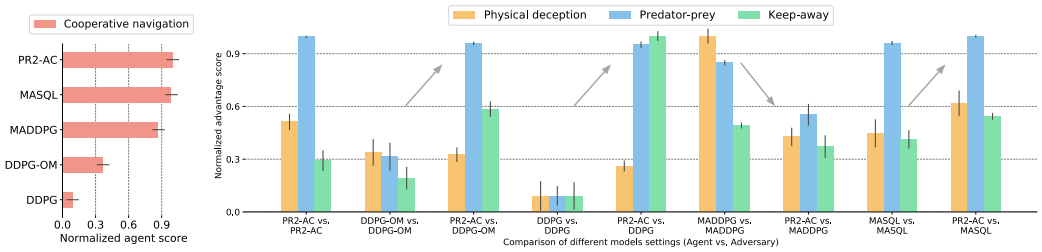


Figure 6: Performance of PR2-AC on the Particle World environment. Each bar shows the $0-1$ normalized score for agent in cooperative navigation task and the normalized advantage score (*agent reward - adversary reward*) in a set of competitive tasks. Higher score is better.

outcomes of MASQL are extremely sensitive to the way of annealing, i.e. when and how to anneal the temperature to a small value during training is non-trivial. However, our method does not need to tune the the annealing parameter at all because the each agent is acting the best response to the approximated conditional policy, considering all potential consequences of the opponent's response.

Interestingly, by comparing the learning path in Fig. 4a against Fig. 5(a-e) where the scattered blue dots are the exploration trails at the beginning, we can tell that if the PR2-AC model finds the peak point in joint action space, the agents can quickly go through the shortcut out of the local basin in a *clever* way, while other algorithms just converge to the local equilibrium. This further justifies the effectiveness and benefits of conducting recursive reasoning with opponents. Apart from testing in the self-play setting, we also test the scenario when the opponent type is different. We pair PR2-AC with all four baseline algorithms in Fig. 5(f-i). Similar result can be found, that is, algorithm that has the function of taking into account the opponents (i.e. DDPG_OM & MADDPG) can converge to the local equilibrium even though not global, while DDPG and MASQL completely fails due to the inborn defect from the independent learning methods.

5.3 Particle World Environments

We further test our method on the multi-state multi-player Particle World Environments (Lowe et al., 2017). This includes four testing scenarios: 1) *Cooperative Navigation* with 3 agents and 3 landmarks. Agents are collectively rewarded based on the proximity of any agent to each landmark while avoiding collisions; 2) *Physical Deception* with 1 adversary, 2 good agents, and 2 landmarks. All agents observe the positions of landmarks and other agents. Only one landmark is the true target landmark. Good agents are rewarded based on how close any of them is to the target landmark, and how well they deceive the adversary; 3) *Keep-away* with 1 agent, 1 adversary, and 1 landmark. Agent is rewarded based on distance to landmark. Adversary is rewarded if it push away the agent from the landmark; 4) *Predator-prey* with 1 prey agent who moves faster try to run away from 3 adversary predator who move slower but are motivated to catch the prey cooperatively.

The PR2 methods are compared against a series of the centralized MARL methods in Fig. 6. Under the fully-cooperative setting (the left plot), PR2AC achieves the best performance over all baselines, even though it is a decentralized algorithm that does not have access to the exact opponent policies. Under the competitive settings where PR2AC rivals against the a set of adversary baselines, we find that PR2AC learners can beat all the baselines, including DDPG, DDPG-OM, and MASQL. The only exception is MADDPG, as it is suggested by the drop-down arrow. PR2AC performs particularly bad on the physical deception task. We believe it is mainly because the centralized critic can access the full knowledge of the exact policies of PR2-AC, but PR2-AC cannot access the models of its opponents in the reversed way; this could place PR2-AC in an inferior position during testing time as its deceptive strategy has been found out by the opponents already during training.

6 Conclusion

Inspired by the recursive reasoning capability of human intelligent, in this paper, we introduce a probabilistic recursive reasoning framework for multi-agent RL that follows "I believe that you believe that I believe". We adopt variational Bayes methods to approximate the opponents' conditional policy, to which each agent finds the best response and then improve their own policy. The training and execution is full decentralized and the resulting algorithms, PR2-Q and PR2-AC, converge in self-play when there is one Nash equilibrium. Our results on three kinds of testing beds with increasing complexity justify the advantages of learning to reason about the opponents in a recursive manner. In the future, we plan to investigate other approximation methods for the PR2 framework.

REFERENCES

Stefano V Albrecht and Peter Stone. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence*, 258:66–95, 2018.

David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. The mechanics of n-player differentiable games. *arXiv preprint arXiv:1802.05642*, 2018.

Dipyaman Banerjee and Sandip Sen. Reaching pareto-optimality in prisoner's dilemma using conditional joint action learning. *Autonomous Agents and Multi-Agent Systems*, 15(1):91–108, 2007.

Thomas Bolander and Mikkel Birkegaard Andersen. Epistemic planning for single-and multi-agent systems. *Journal of Applied Non-Classical Logics*, 21(1):9–34, 2011.

Michael Bowling. Convergence and no-regret in multiagent learning. In *Advances in neural information processing systems*, pp. 209–216, 2005.

Michael Bowling and Manuela Veloso. Convergence of gradient dynamics with a variable learning rate. In *ICML*, pp. 27–34, 2001a.

Michael Bowling and Manuela Veloso. Rational and convergent learning in stochastic games. In *International joint conference on artificial intelligence*, volume 17, pp. 1021–1026. Lawrence Erlbaum Associates Ltd, 2001b.

Michael Bowling and Manuela Veloso. Multiagent learning using a variable learning rate. *Artificial Intelligence*, 136(2):215–250, 2002.

George W Brown. Iterative solution of games by fictitious play. *Activity analysis of production and allocation*, 13(1):374–376, 1951.

Colin F Camerer, Teck-Hua Ho, and Juin-Kuan Chong. A cognitive hierarchy model of games. *The Quarterly Journal of Economics*, 119(3):861–898, 2004.

Colin F Camerer, Teck-Hua Ho, and Juin Kuan Chong. A psychological approach to strategic thinking in games. *Current Opinion in Behavioral Sciences*, 3:157–162, 2015.

Caroline Claus and Craig Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. *AAAI/IAAI*, 1998:746–752, 1998.

Bruno C Da Silva, Eduardo W Basso, Ana LC Bazzan, and Paulo M Engel. Dealing with non-stationary environments using context detection. In *Proceedings of the 23rd international conference on Machine learning*, pp. 217–224. ACM, 2006.

Harmen De Weerd, Rineke Verbrugge, and Bart Verheij. Higher-order theory of mind in negotiations under incomplete information. In *International Conference on Principles and Practice of Multi-Agent Systems*, pp. 101–116. Springer, 2013a.

Harmen De Weerd, Rineke Verbrugge, and Bart Verheij. How much does it help to know what she knows you know? an agent-based simulation study. *Artificial Intelligence*, 199:67–92, 2013b.

Harmen de Weerd, Rineke Verbrugge, and Bart Verheij. Negotiating with other minds: the role of recursive theory of mind in negotiation with incomplete information. *Autonomous Agents and Multi-Agent Systems*, 31(2):250–287, 2017.

Daniel C Dennett. Two contrasts: folk craft versus folk science, and belief versus opinion. *The future of folk psychology: Intentionality and cognitive science*, pp. 135–148, 1991.

Prashant Doshi and Piotr J Gmytrasiewicz. On the difficulty of achieving equilibrium in interactive pomdps. In *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, volume 21, pp. 1131. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006.

Prashant Doshi and Piotr J Gmytrasiewicz. Monte carlo sampling methods for approximating interactive pomdps. *Journal of Artificial Intelligence Research*, 34:297–337, 2009.

Prashant Doshi and Dennis Perez. Generalized point based value iteration for interactive pomdps. In *AAAI*, pp. 63–68, 2008.

Prashant Doshi, Yifeng Zeng, and Qiongyu Chen. Graphical models for interactive pomdps: representations and solutions. *Autonomous Agents and Multi-Agent Systems*, 18(3):376, 2009.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. *arXiv preprint arXiv:1705.08926*, 2017.

Jakob Foerster, Richard Y Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 122–130. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

Roy Fox, Ari Pakman, and Naftali Tishby. Taming the noise in reinforcement learning via soft updates. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, pp. 202–211. AUAI Press, 2016.

Ya'akov Gal and Avi Pfeffer. A language for modeling agents' decision making processes in games. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pp. 265–272. ACM, 2003.

Ya'akov Gal and Avi Pfeffer. Networks of influence diagrams: a formalism for representing agents' beliefs and decision-making processes. *Journal of Artificial Intelligence Research*, 33:109–147, 2008.

Vittorio Gallese and Alvin Goldman. Mirror neurons and the simulation theory of mind-reading. *Trends in cognitive sciences*, 2(12):493–501, 1998.

Piotr J Gmytrasiewicz and Prashant Doshi. A framework for sequential planning in multi-agent settings. *Journal of Artificial Intelligence Research*, 24:49–79, 2005.

Piotr J Gmytrasiewicz and Edmund H Durfee. A rigorous, operational formalization of recursive modeling. In *ICMAS*, pp. 125–132, 1995.

Piotr J Gmytrasiewicz and Edmund H Durfee. Rational coordination in multi-agent environments. *Autonomous Agents and Multi-Agent Systems*, 3(4):319–350, 2000.

Piotr J Gmytrasiewicz, Edmund H Durfee, and David K Wehe. A decision-theoretic approach to coordinating multi-agent interactions. In *IJCAI*, volume 91, pp. 63–68, 1991.

Alvin I Goldman et al. Theory of mind. *The Oxford handbook of philosophy of cognitive science*, pp. 402–424, 2012.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Adam S Goodie, Prashant Doshi, and Diana L Young. Levels of theory-of-mind reasoning in competitive games. *Journal of Behavioral Decision Making*, 25(1):95–108, 2012.

Alison Gopnik and Henry M Wellman. Why the child's theory of mind really is a theory. *Mind & Language*, 7(1-2):145–171, 1992.

Robert M Gordon. Folk psychology as simulation. *Mind & Language*, 1(2):158–171, 1986.

Amy Greenwald, Keith Hall, and Roberto Serrano. Correlated q-learning. In *ICML*, volume 3, pp. 242–249, 2003.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. *arXiv preprint arXiv:1702.08165*, 2017.

John C Harsanyi. Bargaining in ignorance of the opponent's utility function. *Journal of Conflict Resolution*, 6(1):29–38, 1962.

John C Harsanyi. Games with incomplete information played by bayesian players, i–iii part i. the basic model. *Management science*, 14(3):159–182, 1967.

He He, Jordan Boyd-Graber, Kevin Kwok, and Hal Daumé III. Opponent modeling in deep reinforcement learning. In *International Conference on Machine Learning*, pp. 1804–1813, 2016.

Junling Hu and Michael P Wellman. Nash q-learning for general-sum stochastic games. *Journal of machine learning research*, 4(Nov):1039–1069, 2003.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40, 2017.

Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*, 2018.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Michael L Littman. Markov games as a framework for multi-agent reinforcement learning. In *Machine Learning Proceedings 1994*, pp. 157–163. Elsevier, 1994.

Michael L Littman. Friend-or-foe q-learning in general-sum games. In *ICML*, volume 1, pp. 322–328, 2001.

Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose bayesian inference algorithm. In *Advances In Neural Information Processing Systems*, pp. 2378–2386, 2016.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems*, pp. 6379–6390, 2017.

Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of gans. In *Advances in Neural Information Processing Systems*, pp. 1825–1835, 2017.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Christian J Muise, Vaishak Belle, Paolo Felli, Sheila A McIlraith, Tim Miller, Adrian R Pearce, and Liz Sonenberg. Planning over multi-agent epistemic states: A classical planning approach. In *AAAI*, pp. 3327–3334, 2015.

Liviu Panait, Sean Luke, and R Paul Wiegand. Biasing coevolutionary search for optimal multiagent behaviors. *IEEE Transactions on Evolutionary Computation*, 10(6):629–645, 2006.

Brad E Pfeiffer and David J Foster. Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, 497(7447):74, 2013.

David Premack and Guy Woodruff. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences*, 1(4):515–526, 1978.

David V Pynadath and Stacy C Marsella. Psychsim: Modeling theory of mind with decision-theoretic agents. In *IJCAI*, volume 5, pp. 1181–1186, 2005.

Neil C Rabinowitz, Frank Perbet, H Francis Song, Chiyuan Zhang, SM Eslami, and Matthew Botvinick. Machine theory of mind. *arXiv preprint arXiv:1802.07740*, 2018.

Nikolaus Robalino and Arthur Robson. The economic approach to 'theory of mind'. *Phil. Trans. R. Soc. B*, 367(1599):2224–2233, 2012.

Sven Seuken and Shlomo Zilberstein. Formal models and algorithms for decentralized decision making under uncertainty. *Autonomous Agents and Multi-Agent Systems*, 17(2):190–250, 2008.

Lloyd S Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 39(10): 1095–1100, 1953.

Yoav Shoham, Rob Powers, Trond Grenager, et al. If multi-agent learning is the answer, what is the question? *Artificial Intelligence*, 171(7):365–377, 2007.

Satinder Singh, Michael Kearns, and Yishay Mansour. Nash convergence of gradient dynamics in general-sum games. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pp. 541–548. Morgan Kaufmann Publishers Inc., 2000.

Edward Jay Sondik. The optimal control of partially observable markov processes. Technical report, STANFORD UNIV CALIF STANFORD ELECTRONICS LABS, 1971.

Ekhlas Sonu and Prashant Doshi. Scalable solutions of interactive pomdps using generalized and bounded policy iteration. *Autonomous Agents and Multi-Agent Systems*, 29(3):455–494, 2015.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*. MIT press, 1998.

Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pp. 1057–1063, 2000.

Edward C Tolman. Cognitive maps in rats and men. *Psychological review*, 55(4):189, 1948.

Karl Tuyls and Gerhard Weiss. Multiagent learning: Basics, challenges, and prospects. *Ai Magazine*, 33(3):41, 2012.

Friedrich Burkhard Von Der Osten, Michael Kirley, and Tim Miller. The minds of many: opponent modelling in a stochastic game. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI), AAAI Press*, pp. 3845–3851, 2017.

Dilin Wang and Qiang Liu. Learning to draw samples: With application to amortized mle for generative adversarial learning. *arXiv preprint arXiv:1611.01722*, 2016.

Ermo Wei and Sean Luke. Lenient learning in independent-learner stochastic cooperative games. *The Journal of Machine Learning Research*, 17(1):2914–2955, 2016.

Ermo Wei, Drew Wicke, David Freelan, and Sean Luke. Multiagent soft q-learning. *AAAI*, 2018.

Yingce Xia, Tao Qin, Wei Chen, Jiang Bian, Nenghai Yu, and Tie-Yan Liu. Dual supervised learning. *arXiv preprint arXiv:1707.00415*, 2017.

Yaodong Yang, Rui Luo, Minne Li, Ming Zhou, Weinan Zhang, and Jun Wang. Mean field multi-agent reinforcement learning. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5571–5580, Stockholmsmassan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

APPENDIX

A DECENTRALIZED MULTI-AGENT PROBABILISTIC RECURSIVE REASONING ALGORITHMS

Algorithm 1 gives the step by step learning procedures for PR2-AC algorithm.

Algorithm 1: Multi-Agent Probabilistic Recursive Reasoning Actor Critic (PR2-AC).

Result: Policy:π^i, Opponent Recursive Reasoning: $\rho^{-i}(a^{-i}|s,a^i)$.

Initialize parameters $\theta^i, \phi^{-i}, \omega^i$ for each agent i, and the random process $\mathcal{N}$ for action exploration.
Assign target parameters of joint action Q-function: $\omega^{i\prime} \leftarrow \omega^i$, and target policy parameter: $\theta^{i\prime} \leftarrow \theta^i$
$D^i \leftarrow$ empty replay buffer for each agent.
for *each episode* **do**
  Initialize random process $\mathcal{N}$ for action exploration.
  for *each step t* **do**
    Given the current s, for each agent i, select action $a^i = \mu^i_{\theta^i}(s) + \mathcal{N}_t$;
    Take the joint action (a^i, a^{-i}) and observe own reward r^i and new state s';
    Add the tuple $(s, a^i, a^{-i}, r^i, s')$ in corresponding replay buffer D^i;
    $s \leftarrow s'$;
    for *each agent i* **do**
      Sample a random mini-batch $\{(s, a^i_j, a^{-i}_j, r^i_j, s'_j)\}_{j=0}^N$ from D^i;
      Get $a^{i\prime}_j = \mu^{i\prime}_{\theta^i}$ for each state s'_j ;
      Sample $\{a^{-i\prime}_{k,j}\}_{k=0}^M \sim \rho^{-i}_{\phi^{-i}}(\cdot|s'_j, a^{i\prime}_j)$ for each $a^{i\prime}_j$ and s'_j ;
      Set $y^i_j = r^i_j + \gamma \frac{1}{M} \sum_{k=0}^M Q^i_{\mu^{i\prime}}(s', a^{i\prime}, a^{-i\prime}_{k,j})$;
      Update the critic by minimizing the loss $\mathcal{L}(\omega^i) = \frac{1}{N}\sum_{j=0}^N \left(y_j - Q^i_{\mu^i}(s_j, a^i_j, a^{-i}_j)\right)^2$;
      Update the actor using the sampled policy gradient:

$$\nabla_{\theta^i}\eta^i \approx \frac{1}{N}\sum_{j=0}^{N} \nabla_{\theta^i}\mu^i\left(s_j\right)\nabla_{a^i}\frac{1}{M}\sum_{k=0}^{M} Q^i_{\mu^i}(s_j, a^i_j, a^{-i}_{k,j});$$

      Compute $\Delta\rho^{-i}_{\phi^{-i}}$ using empirical estimation:

$$\begin{aligned}\Delta\rho^{-i}_{\phi^{-i}}(\cdot|s,a^i) =& \mathbb{E}_{a^{-i}_t \sim \rho^{-i}_{\phi^{-i}}}\Big[\kappa\left(a^{-i}_t, \rho^{-i}_{\phi^{-i}}\left(\cdot; s_t, a^i_t\right)\right)\nabla_{\tilde{a}^{-i}} Q^i\left(s_t, a^i_t, \tilde{a}^{-i}\right)\big|_{\tilde{a}^{-i}=a^{-i}_t} \\ &+ \kappa\left(\tilde{a}^{-i}, \rho^{-i}_{\phi^{-i}}\left(\cdot; s_t, a^i_t\right)\right)\nabla_{\tilde{a}^{-i}}\big|_{\tilde{a}^{-i}=a^{-i}_t}\Big],\end{aligned}$$

      where κ is a kernel function;
      Compute empirical gradient $\hat{\nabla}_{\phi^{-i}} J_{\rho^{-i}}$;
      Update ϕ^{-i} according to $\hat{\nabla}_{\phi^{-i}} J_{\rho^{-i}}$;
    end
    Update target network parameters for each agent i:

$$\theta^{i\prime} \leftarrow \lambda\theta^i + (1-\lambda)\theta^{i\prime};$$
$$\omega^{i\prime} \leftarrow \lambda\omega^i + (1-\lambda)\omega^{i\prime};$$

  end
end

The Algorithm 2 shows the variant of Decentralized Multi-Agent Probabilistic Recursive Reasoning. We can simply approximate the $\rho^{-i}(a^{-i}|s,a^i)$ by counting:$\rho^{-i}(a^{-i}|s,a^i) = C(a^i, a^{-i}, s)/C(a^i, s)$ in tabular if the state-action space is small, where C is the counting function. It this case, an agent

Algorithm 2: Multi-Agent Probabilistic Recursive Reasoning Q-Learning (PR2-Q).

Result: Policy: π^i, Opponent Recursive Reasoning: $\rho^{-i}(a^{-i}|s,a^i)$.

Initialize $Q^i(s,a^i,a^{-i})$ arbitrarily, set α as the learning rate, γ as discount factor;

while *not converge* **do**

Given the current s, calculate the opponent best response $\rho^{-i}(a^{-i}|s,a^i)$ according to:

$$\rho^{-i}(a^{-i}|s,a^i) = \exp(Q^i(s,a^i,a^{-i}) - Q^i(s,a^i))$$

Select and sample action a^i based on the Recursive Reasoning $\rho^{-i}(a^{-i}|s,a^i)$;

$$\mathrm{softmax}(\int_{a^{-i}} \phi^{-i}(a^{-i}|s,a^i) Q^i(s,a^i,a^{-i}))$$

Observing joint-action (a^i,a^{-i}), reward r^i, and next state s';

$$Q^i(s,a^i,a^{-i}) \leftarrow (1-\alpha)Q^i(s,a^i,a^{-i}) + \alpha(r^i + \gamma V^i(s'))$$
$$Q^i(s,a^i) \leftarrow (1-\alpha)Q^i(s,a^i) + \alpha(r^i + \gamma V^i(s'))$$

where,

$$V^i(s) = \max_{a^i} \int_{a^{-i}} \rho^{-i}(a^{-i}|s,a^i) Q^i(s,a^i,a^{-i})$$

end

only needs to learn a joint action Q-function, and if the game is static, our method would degenerate to Conditional Joint Action Learning (CJAL) (Banerjee & Sen, 2007).

B Multi-Agent Policy Gradient

B.1 Multi-Agent Non-correlated Policy Gradient

Since $\pi_\theta\left(a^i,a^{-i}|s\right) = \pi^i_{\theta^i}\left(a^i\right)\pi^{-i}_{\theta^{-i}}\left(a^{-i}|,a^i\right) = \pi^{-i}_{\theta^{-i}}\left(a^{-i}|s\right)\pi^i_{\theta^i}\left(a^i|s,a^{-i}\right)$, $\pi_\theta\left(a^i,a^{-i}|s\right)$ can be factorized as $\pi^i_{\theta^i}(a^i|s)\pi^{-i}_{\theta^{-i}}(a^{-i}|s)$ if a^i and a^{-i} are non-correlated. We follow the policy gradient formulation (Sutton et al., 2000; Wei et al., 2018) using Leibniz integral rule and Fubini's theorem which can give us Multi-Agent Non-correlated Policy Gradient:

$$\begin{aligned}
\eta^i &= \int_s \int_{a^i} \int_{a^{-i}} \pi(a^i,a^{-i}|s) Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}\,\mathrm{d}a^i\,\mathrm{d}s \\
&= \int_s \int_{a^i} \int_{a^{-i}} \pi^i(a^i|s)\pi^{-i}(a^{-i}|s) Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}\,\mathrm{d}a^i\,\mathrm{d}s \\
&= \int_s \int_{a^i} \pi^i(a^i|s) \int_{a^{-i}} \pi^{-i}(a^{-i}|s) Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}\,\mathrm{d}a^i\,\mathrm{d}s.
\end{aligned} \tag{14}$$

Suppose the $\pi^i(a^i)$ is parameterized by θ^i, and we apply the gradient over the η^i:

$$\begin{aligned}
\nabla_{\theta^i}\eta^i &= \int_s \int_{a^i} \nabla_{\theta^i}\pi^i_{\theta_i}(a^i|s) \int_{a^{-i}} \pi^{-i}(a^{-i}|s) Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}\,\mathrm{d}a^i\,\mathrm{d}s \\
&= \mathbb{E}_{s\sim p, a^i\sim\pi^i}[\nabla_{\theta^i}\log\pi^i(a^i|s) \int_{a^{-i}} \pi^{-i}(a^{-i}|s) Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}].
\end{aligned} \tag{15}$$

In practice, off-policy is more data-efficient. In MADDPG (Lowe et al., 2017) and COMA (Foerster et al., 2017), the replay buffer is introduced in a centralized deterministic actor-critic method for off-policy training. They apply batch sampling to the centralized critic which gives the joint-action Q-values:

$$\nabla_{\theta^i}\eta^i = \mathbb{E}_{s,a^i,a^{-i}\sim D}[\nabla_{\theta^i}\mu^i_{\theta^i}(a^i|s)\nabla_{a^i}Q^i(s,a^i,a^{-i})|_{a^i=\mu^i(s)}]. \tag{16}$$

B.2 MULTI-AGENT RECURSIVE REASONING POLICY GRADIENT

Proposition 1. *In a stochastic game, under the recursive reasoning framework defined by Eq. 3, the update rule for the multi-agent recursive reasoning policy gradient method can be devised as follows:*

$$\nabla_{\theta^i}\eta^i = \mathbb{E}_{s\sim p, a^i\sim\pi^i}\left[\nabla_{\theta^i}\log\pi^i_{\theta^i}(a^i|s)\int_{a^{-i}}\pi^{-i}_{\theta^{-i}}(a^{-i}|s,a^i)Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}\right]. \tag{17}$$

Proof: As following.

If we apply the chain rule to factorize the joint policy to: $\pi_\theta(a^i,a^{-i}|s) = \pi^i_{\theta^i}(a^i|s)\pi^{-i}_{\theta^{-i}}(a^{-i}|s,a^i)$. Then, we can have multi-agent recursive reasoning objective function as:

$$\begin{aligned}\eta^i &= \int_s\int_{a^i}\int_{a^{-i}}\pi(a^i,a^{-i}|s)Q^i(a^i,a^{-i})\,\mathrm{d}a^{-i}\,\mathrm{d}a^i\,\mathrm{d}s\\ &= \int_s\int_{a^i}\pi^i(a^i|s)\int_{a^{-i}}\pi^{-i}(a^{-i}|s,a^i)Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}\,\mathrm{d}a^i\,\mathrm{d}s.\end{aligned} \tag{18}$$

Compare to Eq. 14, a^{-i} in Eq. 18 is additionally conditioned on a^i. We introduce agent i'a action a^i into other agents's policies, leading to $\pi^{-i}(a^{-i}|s,a^i)$. We now compute the policy gradient analytically. Following the single agent Policy Gradient Theorem with Leibniz integral rule and Fubini's theorem, we get the multi-Agent Recursive Reasoning Policy Gradient:

$$\nabla_{\theta^i}\eta^i = \mathbb{E}_{s\sim p, a^i\sim\pi^i}[\nabla_{\theta^i}\log\pi^i(a^i|s)\int_{a^{-i}}\pi^{-i}(a^{-i}|s,a^i)Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}]. \tag{19}$$

However, in practice, the agent may not get access to other agents' policies. We need to infer the other agents' policies. We let $\rho^{-i}_{\phi_{-i}}(a^{-i}|s,a^i)$ denotes the parameterized opponent conditional policy of agent i to approximate other agents policies, i.e, $\pi^{-i}(a^{-i}|s,a^i)$. Then we have Decentralized Multi-Agent Recursive Reasoning Policy Gradient comes as:

$$\begin{aligned}\nabla_{\theta^i}\eta^i \approx& \mathbb{E}_{s\sim p, a^i\sim\pi^i}[\nabla_{\theta^i}\log\pi^i_{\theta^i}(a^i|s)\int_{a^{-i}}\rho^{-i}_{\phi_{-i}}(a^{-i}|s,a^i)Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}]\\ =& \mathbb{E}_{s\sim p, a^i\sim\pi^i}[\nabla_{\theta^i}\log\pi^i_{\theta^i}(a^i|s)Q^i_{\rho^{-i}_{\phi_{-i}}}(s,a^i)].\end{aligned} \tag{20}$$

In Eq. 20, the gradient for agent i is scaled by $Q^i_{\rho^{-i}_{\phi_{-i}}}(s,a^i) = \int_{a^{-i}}\rho^{-i}_{\phi_{-i}}(a^{-i}|s,a^i)Q^i(s,a^i,a^{-i})\,\mathrm{d}a^{-i}$. The trajectories generated by updated policy would help to train $\rho^{-i}_{\phi_{-i}}(a^{-i}|s,a^i)$ and $Q^i(s,a^i,a^{-i})$. These steps form a Expectation-Maximization style learning procedures: first, fix $\rho^{-i}_{\phi_{-i}}$ and $Q^i(s,a^i,a^{-i})$ to improve $\pi^i_{\theta^i}(a^i|s)$; then, improve $\rho^{-i}_{\phi_{-i}}$ and $Q^i(s,a^i,a^{-i})$ by the trajectories generated by $\pi^i_{\theta^i}(a^i|s)$. Furthermore, since PR2 method do not require opponents' actual private policies, Decentralized Multi-Agent Recursive Reasoning Policy Gradient can be decoupled from other agents' on-policies or target policies. In other words, the training can be conducted in an off-policy fashion by sampling mini-batches from the memory buffer D with the help of the learned $\rho^{-i}_{\phi_{-i}}(a^{-i}|s,a^i)$ from $Q^i(s,a^i,a^{-i})$. ■

C OPPONENT CONDITIONAL POLICY INFERENCE VIA OPTIMAL TRAJECTORY

Theorem 1. *The optimal Q-function for agent i that satisfies minimizing Eq. 10 is formulated as:*

$$Q^i_{\pi_\theta}(s,a^i) = \log\int_{a^{-i}}\exp(Q^i_{\pi_\theta}(s,a^i,a^{-i}))\,\mathrm{d}a^{-i}. \tag{21}$$

And the corresponding optimal opponent conditional policy reads:

$$\rho^{-i}_{\phi^{-i}}(a^{-i}|s,a^i) = \exp(Q^i_{\pi_\theta}(s,a^i,a^{-i}) - Q^i_{\pi_\theta}(s,a^i)) \tag{22}$$

Proof. As following.

Follow the proof in Levine (2018); Haarnoja et al. (2017), we first give the overall distribution by:

$$p(\tau) = [p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a_t^i, a_t^{-i})]\exp(\sum_{t=1}^{T} r^i(s_t, a_t, a_t^{-i})). \tag{23}$$

We can adopt an optimization-based approach to approximate the opponent conditional policy, in which case the goal is to fit an approximation $\pi(a_t^i, a_t^{-i}|s_t) \approx \pi^i(a_t^i|s_t)\rho^{-i}(a_t^{-i}|s_t, a_t^i)$ such that the trajectory distribution,

$$\hat{p}(\tau) = p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a_t^i, a_t^{-i})\pi_{\theta^i}^i(a_t^i|s_t)\rho_{\theta^{-i}}^{-i}(a_t^{-i}|s_t, a_t^i), \tag{24}$$

has high likelihood to be observed. In the case of exact inference, as derived in the previous section, $D_{\mathrm{KL}}(\hat{p}(\tau)\|p(\tau)) = 0$. We can therefore view the inference process as minimizing the KL-divergence:

$$D_{\mathrm{KL}}(\hat{p}(\tau)\|p(\tau)) = -\mathbb{E}_{\tau\sim\hat{p}(\tau)}[\log p(\tau) - \log \hat{p}(\tau)]. \tag{25}$$

Negating both sides and substituting, we get:

$$\begin{aligned}
-D_{\mathrm{KL}}(\hat{p}(\tau)\|p(\tau)) &= \mathbb{E}_{\tau\sim\hat{p}(\tau)}[\log p(s_1) + \sum_{t=1}^{T}(\log p(s_{t+1}|s_t, a_t, a_t^{-i}) + r^i(s_t, a_t^i, a_t^{-i})) \\
&\quad - \log p(s_1) - \sum_{t=1}^{T}(\log p(s_{t+1}|s_t, a_t^i, a_t^{-i}) + \log \pi(a_t^i, a_t^{-i}|s_t))] \\
&= \mathbb{E}_{\tau\sim\hat{p}(\tau)}[\sum_{t=1}^{T} r^i(s_t, a_t^i, a_t^{-i}) - \log \pi(a_t^i, a_t^{-i}|s_t)] \\
&= \sum_{t=1}^{T}\mathbb{E}_{(s_t, a_t^i, a_t^{-i})\sim\hat{p}(s_t, a_t^i, a_t^{-i}))}[r^i(s_t, a_t^i, a_t^{-i}) - \log \pi(a_t^i, a_t^{-i}|s_t)] \\
&= \sum_{t=1}^{T}\mathbb{E}_{(s_t, a_t^i, a_t^{-i})\sim\hat{p}(s_t, a_t^i, a_t^{-i}))}[r^i(s_t, a_t^i, a_t^{-i})] \\
&\quad + \mathbb{E}_{s_t, a_t^i\sim\hat{p}(s_t)}[\mathcal{H}(\rho^{-i}(a_t^{-i}|s_t, a_t^i))] + \mathbb{E}_{s_t\sim\hat{p}(s_t)}[\mathcal{H}(\pi^i(a_t^i|s_t))],
\end{aligned} \tag{26}$$

where $\mathcal{H}$ is the entropy term. In the recursive case, we can rewrite the objective as follows:

$$Q^i(s, a^i) = \log \int_{a^{-i}} \exp(Q^i(s, a^i, a^{-i}))\, \mathrm{d}a^{-i}. \tag{27}$$

This corresponds to a standard bellman backup with a soft maximization for the value function. choosing optimal opponent recursive reasoning policy

$$\rho^{-i}(a^{-i}|s, a^i) = \exp(Q^i(s, a^i, a^{-i}) - Q^i(s, a^i)). \tag{28}$$

Then we can have the objective function:

$$\begin{aligned}
J^i(\phi^{-i}) = \sum_{t=1}^{T}\mathbb{E}_{(s_t, a_t^i, a_t^{-i})\sim\hat{p}(s_t, a_t^i, a_t^{-i})}[&r^i(s_t, a_t^i, a_t^{-i}) \\
&+ \mathcal{H}(\rho_{\phi^{-i}}^{-i}(a_t^{-i}|s_t, a_t^i)) + \mathcal{H}(\pi_{\theta^i}^i(a_t^i|s_t))].
\end{aligned} \tag{29}$$

Then the gradient is then given by:

$$\begin{aligned}
\nabla_{\phi^{-i}} J^i(\phi^{-i}) = &\sum_{t=1}^{T}\mathbb{E}_{(s_t, a_t^i, a_t^{-i})\sim p(s_t, a_t^i, a_t^{-i})}[\nabla_{\phi^{-i}} \log \rho_{\phi^{-i}}^{-i}(a_t^{-i}|s_t, a_t^i)(\sum_{t'=t}^{T} r^i(s_{t'}, a_{t'}^i, a_{t'}^{-i})] \\
&+ \nabla_{\phi^{-i}}\sum_{t=1}^{T}\mathbb{E}_{(s_t, a_t^i, a_t^{-i})\sim p(s_t, a_t^i, a_t^{-i})}[\mathcal{H}(\rho_{\phi^{-i}}^{-i}(a_t^{-i}|s_t, a_t^i)) + \mathcal{H}(\pi_{\theta^i}^i(a_t^i|s_t))].
\end{aligned} \tag{30}$$

The gradient of the entropy terms is given by:

$$\begin{aligned}\nabla_{\phi^{-i}}\mathcal{H}(\rho^{-i}_{\phi^{-i}}) &= -\nabla_\phi \mathbb{E}_{(s_t,a^i_t)\sim p(s_t,a^i_t,a^{-i}_t)}[\mathbb{E}_{a^{-i}_t\sim\rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t)}[\log \rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t)]] \\ &= -\mathbb{E}_{(s_t,a^i_t,a^{-i}_t)\sim p(s_t,a^i_t,a^{-i}_t)}[\nabla_\phi \log \rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t)(1+\log \rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t)].\end{aligned} \tag{31}$$

We can do the same for $\nabla_{\phi^{-i}}\mathcal{H}(\pi^i_{\theta^i})$, and substitute these back we have:

$$\begin{aligned}\nabla_{\phi^{-i}} J^i(\phi^{-i}) = \sum_{t=1}^{T}\mathbb{E}_{(s_t,a^i_t,a^{-i}_t)\sim p(s_t,a^i_t,a^{-i}_t)}[&\nabla_{\phi^{-i}}\log \rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t) \\ &(\sum_{t'=t}^{T} r^i(s_{t'},a^i_{t'},a^{-i}_{t'}) - \log \rho^{-i}_{\phi^{-i}}(a^{-i}_{t'}|s_t,a^i_{t'}) - \log \pi^i_{\theta^i}(a^i_t|s_t) - 1)].\end{aligned} \tag{32}$$

The -1 comes from the derivative of the entropy terms, and replacing -1 with a state and self-action dependent baseline $b(s_{t'},a^i_{t'})$ we can obtain the approximated gradient for ϕ:

$$\begin{aligned}\nabla_{\phi^{-i}} J^i(\phi^{-i}) &= \sum_{t=1}^{T}\mathbb{E}_{(s_t,a^i_t,a^{-i}_t)\sim p(s_t,a^i_t,a^{-i}_t)}[\nabla_{\phi}\log \rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t) \\ &\quad (\sum_{t'=t}^{T} r^i(s_{t'},a^i_{t'},a^{-i}_{t'}) - \log \rho^{-i}_{\phi^{-i}}(a^{-i}_{t'}|s_{t'},a^i_{t'}) - \log \pi^i_{\theta^i}(a^i_{t'}|s_{t'}) - \underbrace{1}_{\text{baseline ignore}})] \\ &\approx \sum_{t=1}^{T}\mathbb{E}_{(s_t,a^i_t,a^{-i}_t)\sim p(s_t,a^i_t,a^{-i}_t)}[\nabla_{\phi^{-i}}\log \rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t) \\ &\quad (r^i(s_t,a^i_t,a^{-i}_t) - \underbrace{\log \pi^i_{\theta^i}(a^i_t|s_t)}_{Q^i_t(s_t,a^i_t)-V^i_t(s_t)} - \underbrace{\log \rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t)}_{Q^i_t(s_t,a^i_t,a^{-i}_t)-Q^i_t(s_t,a^i_t)} \\ &\quad + \underbrace{\sum_{t'=t+1}^{T} r^i(s_{t'},a^i_{t'},a^{-i}_{t'}) - \log \rho^{-i}_{\phi^{-i}}(a^{-i}_{t'}|s_{t'},a^i_{t'}) - \log \pi^i_{\theta^i}(a^i_{t'}|s_{t'})}_{\approx Q^i_t(s_{t+1},a^i_{t+1},a^{-i}_{t+1})})] \\ &= \sum_{t=1}^{T}\mathbb{E}_{(s_t,a^i_t,a^{-i}_t)\sim p(s_t,a^i_t,a^{-i}_t)}[\nabla_{\phi}\log \rho^{-i}_{\phi^{-i}}(a^{-i}_t|s_t,a^i_t) \\ &\quad (r^i(s_{t'},a^i_{t'},a^{-i}_{t'}) + Q^i_t(s_{t+1},a^i_{t+1},a^{-i}_{t+1}) - Q^i_t(s_t,a^i_t,a^{-i}_t) + \underbrace{V^i_t(s_t)}_{\text{ignore}})] \\ &= \sum_{t=1}^{T}\mathbb{E}_{(s_t,a^i_t,a^{-i}_t)\sim p(s_t,a^i_t,a^{-i}_t)}[(\nabla_{\phi^{-i}}Q^i_t(s_t,a^i_t,a^{-i}_t) - \nabla_{\phi^{-i}}Q^i_t(s_t,a^i_t)) \\ &\quad (r^i(s_{t'},a^i_{t'},a^{-i}_{t'}) + Q^i_t(s_{t+1},a^i_{t+1},a^{-i}_{t+1}) - Q^i_t(s_t,a^i_t,a^{-i}_t) + \underbrace{V^i_t(s_t)}_{\text{ignore}})] \\ &= \sum_{t=1}^{T}\mathbb{E}_{(s_t,a^i_t,a^{-i}_t)\sim p(s_t,a^i_t,a^{-i}_t)}[(\nabla_{\phi^{-i}}Q^i_t(s_t,a^i_t,a^{-i}_t) - \nabla_{\phi^{-i}}Q^i_t(s_t,a^i_t)) \\ &\quad (\hat{Q}^i_t(s_t,a^i_t,a^{-i}_t) - Q^i_t(s_t,a^i_t,a^{-i}_t))],\end{aligned} \tag{33}$$

where $\hat{Q}^i_t(s_t,a^i_t,a^{-i}_t)$ is is an empirical estimate of the Q-value of the policy. ■

D Soft Bellman Equation and Soft Value Iteration

Theorem 2. *In a symmetric game with only one equilibrium, and the equilibrium meets one of the conditions: 1) the global optimum, i.e.* $\mathbb{E}_{\pi_*}\left[Q^i_t(s)\right] \geq \mathbb{E}_{\pi}\left[Q^i_t(s)\right]$*; 2) a saddle point, i.e.*

$\mathbb{E}_{\pi_*}\left[Q_t^i(s)\right] \geq \mathbb{E}_{\pi^i}\mathbb{E}_{\pi_*^{-i}}\left[Q_t^i(s)\right]$ *or* $\mathbb{E}_{\pi_*}\left[Q_t^i(s)\right] \geq \mathbb{E}_{\pi_*^i}\mathbb{E}_{\pi^{-i}}\left[Q_t^i(s)\right]$*; where* Q_* *and* π_* *are the equilibrium value function and policy, respectively. The PR2 soft value iteration operator defined by:*

$$\mathcal{T}Q^i(s,a^i,a^{-i}) \triangleq r^i(s,a^i,a^{-i}) + \gamma\mathbb{E}_{s',a^{i\prime}\sim p_s,\pi^i}\left[\log\int_{a^{-i\prime}}\exp(Q^i(s',a^{i\prime},a^{-i\prime}))\,\mathrm{d}a^{-i\prime}\right], \tag{34}$$

is a contraction mapping.

Proof. As following:

Based on Eq. 11 & 12 in Theorem 1, we can have the PR2 soft value iteration rules shown as:

$$\begin{aligned} Q_\pi^i(s,a^i,a^{-i}) &= r^i(s,a^i,a^{-i}) + \gamma\mathbb{E}_{s'\sim p_s}\left[\mathscr{H}(\pi^i(a^i|s)\pi^{-i}(a^{-i}|s,a^i)) + \mathbb{E}_{a^{-i\prime}\sim\pi^{-i}(\cdot|s',a^{i\prime})}[Q_\pi^i(s',a^{i\prime},a^{-i\prime})]\right] \\ &= r^i(s,a^i,a^{-i}) + \gamma\mathbb{E}_{s'\sim p_s}\left[Q_\pi^i(s',a^{i\prime})\right]. \end{aligned} \tag{35}$$

Correspondingly, we define the soft value iteration operator $\mathcal{T}$:

$$\mathcal{T}Q^i(s,a^i,a^{-i}) \triangleq r^i(s,a^i,a^{-i}) + \gamma\mathbb{E}_{s',a^{i\prime}\sim p_s,\pi^i}\left[\log\int_{a^{-i\prime}}\exp(Q^i(s',a^{i\prime},a^{-i\prime}))\,\mathrm{d}a^{-i\prime}\right]. \tag{36}$$

In a symmetric game with either one global equilibrium or saddle equilibrium, it has been shown by Yang et al. (2018) (see condition 1&2 in Theorem 1) that the payoff at the equilibrium point is unique. This validates applying the similar idea in proving the contraction mapping of soft-value iteration operator in the single agent case (see Lemma 1 in Fox et al. (2016)). We include it here to stay self-contained.

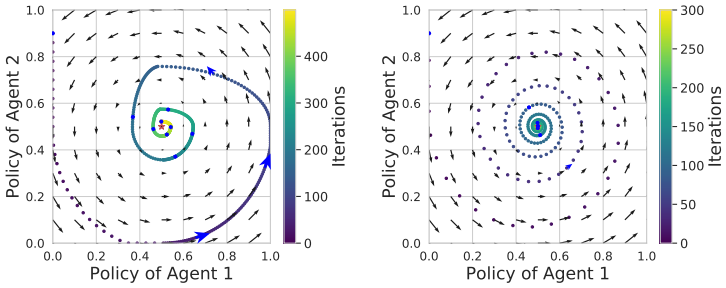
We first define a norm on Q-values as $\|Q_1^i - Q_2^i\| \triangleq \max_{s,a^i,a^{-i}}|Q_1^i(s,a^i,a^{-i}) - Q_2^i(s,a^i,a^{-i})|$. Suppose $\varepsilon = \|Q_1^i - Q_2^i\|$, then

$$\begin{aligned} \log\int_{a^{-i\prime}}\exp(Q_1^i(s',a^{i\prime},a^{-i\prime}))\,\mathrm{d}a^{-i\prime} &\leq \log\int_{a^{-i\prime}}\exp(Q_2^i(s',a^{i\prime},a^{-i\prime}) + \varepsilon)\,\mathrm{d}a^{-i\prime} \\ &= \log\int_{a^{-i\prime}}\exp(\varepsilon)\exp(Q_2^i(s',a^{i\prime},a^{-i\prime}))\,\mathrm{d}a^{-i\prime} \\ &= \varepsilon + \log\int_{a^{-i\prime}}\exp(Q_2^i(s',a^{i\prime},a^{-i\prime}))\,\mathrm{d}a^{-i\prime} \end{aligned} \tag{37}$$

Similarly, $\log\int_{a^{-i\prime}}\exp(Q_1^i(s',a^{i\prime},a^{-i\prime}))\,\mathrm{d}a^{-i\prime} \leq -\varepsilon + \log\int_{a^{-i\prime}}\exp(Q_2^i(s',a^{i\prime},a^{-i\prime}))\,\mathrm{d}a^{-i\prime}$. Therefore $\|\mathcal{T}Q_1^i - \mathcal{T}Q_2^i\| \leq \gamma\varepsilon = \gamma\|Q_1^i - Q_2^i\|$. ■

E SUPPLEMENTARY MATERIAL FOR REBUTTAL

E.1 ADDITIONAL BASELINES ON MATRIX GAME

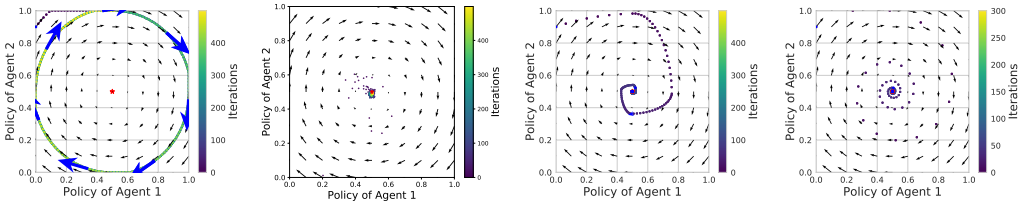


(a) WoLF-IGA dynamics. (b) LOLA-Ex dynamics.

Figure 7: Learning paths on the iterated matrix game in Sec. 5.1. a: WolF-IGA. b: LOLA-Ex.

We provided additional results of baselines (i.e. WolF-IGA and LOLA) on the matrix game in sec 5.1. The results in Fig. 7 show that both WolF-IGA and LOLA can converge to the Nash equilibrium. However, we would like to highlight that, despite the convergence, both WolF-IGA and LOLA require additional information, such as the exact the equilibrium/payoff information, or the exact opponent's value function and its gradient. This matrix game is simple; we believe it can help us understand the learning dynamics of the conditional opponent policy (see Fig. 3), especially under the condition that it can converge to non-trivial equilibrium with only the information of historical actions.

E.2 MATCHING PENNIES GAME



(a) IGA dynamics. (b) PR2-Q dynamics. (c) WoLF-IGA dynamics. (d) LOLA-Ex dynamics.

Figure 8: Learning paths on the iterated matrix game Matching Pennies Game.

Matching Pennies game is a zero-sum game with the reward defined by $R^1 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$, $R^2 = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}$. It also has one single Nash Equilibrium $(0.5, 0.5)$.

E.3 GRAPHICAL MODEL OF PR2 FRAMEWORK

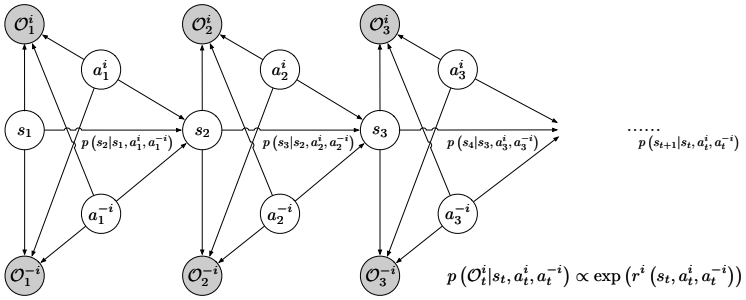


Figure 9: Probabilistic Graphical Model of Multi-Agent Decision Making.