In *Proceedings of the* 29^{th} *International Joint Conference on Artificial Intelligence* (IJCAI 2020), Yokohama, Japan July 2020

Balancing Individual Preferences and Shared Objectives in Multiagent Reinforcement Learning

Ishan Durugkar[1*], **Elad Liebman**[2*] and **Peter Stone**[1,3]
[1]University of Texas at Austin
[2]SparkCognition Research
[3]Sony AI
ishand@cs.utexas.edu, eliebman@sparkcognition.com, pstone@cs.utexas.edu

Abstract

In multiagent reinforcement learning scenarios, it is often the case that independent agents must jointly learn to perform a cooperative task. This paper focuses on such a scenario in which agents have individual preferences regarding how to accomplish the shared task. We consider a framework for this setting which balances individual preferences against task rewards using a linear mixing scheme. In our theoretical analysis we establish that agents can reach an equilibrium that leads to optimal shared task reward even when they consider individual preferences which are not fully aligned with this task. We then empirically show, somewhat counterintuitively, that there exist mixing schemes that outperform a purely task-oriented baseline. We further consider empirically how to optimize the mixing scheme.

1 Introduction

When independent agents jointly learn to perform a cooperative task, multiagent reinforcement learning (RL) methods can be brought to bear [Busoniu *et al.*, 2008; Hernandez-Leal *et al.*, 2019] . When they must do so without any prior coordination, it is referred to as ad hoc teamwork [Barrett *et al.*, 2013; Stone *et al.*, 2010]. Ad hoc teamwork typically assumes that the participating agents have fully aligned preferences – that they prioritize the shared task completely.

In contrast, in this paper we consider ad hoc teamwork in which agents working together on a shared task have individual preferences regarding how to accomplish it. For example, consider a (human) musical ensemble. Each musician learns to play their instrument independently, and typically has their own aesthetic preferences regarding what they'd like the shared music to sound like. However, when performing together, the musicians also have to harmonize and coordinate so that they produce music that is pleasing to their audience, who serve as a shared and extrinsic reward signal.

To incorporate the agents' individual preferences, we assume that each agent linearly blends their individual preference with the shared task reward. A natural assumption would be that the more weight the agents place on their individual preferences, i.e. the more selfishly they behave, the worse the team will perform. On the contrary, we find that a certain degree of selfishness can be beneficial to team performance.

The main contribution of this paper is a detailed analysis of this phenomenon, from both a theoretical and an empirical perspective. In particular:

- We theoretically analyze the conditions in which individual preferences can still lead to maximizing task reward.
- We empirically study how different blending proportions (mixing schemes) for individual and shared reward impact learning of the shared task.
- For a given set of preferences, we show a practical method to search over mixing schemes for the purpose of optimizing joint task performance.

We observe this effect in two different multiagent domains, the predator prey domain (a canonical multiagent RL environment also known as the Pursuit domain), and a novel music generation environment motivated by human musical ensembles. [1]

2 Background

Reinforcement Learning (RL) considers an agent acting in a Markov Decision Process (MDP), a mathematical framework for modeling sequential decision-making [Sutton and Barto, 2018]. An MDP is defined as a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, R, \gamma \rangle$, where $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of actions, and $\mathcal{T}$ is the transition probability, $p(s_{t+1}|s_t, a_t)$, where $s_t, s_{t+1} \in \mathcal{S}$ and $a_t \in \mathcal{A}$. $R(s_t, a_t, s_{t+1}) \in \mathbb{R}$ is the reward for taking action a_t in state s_t and transitioning to state s_{t+1}. The discount factor $\gamma \in [0, 1)$ specifies how much to discount future rewards. Reinforcement learning aims to maximize the return, i.e. the sum of expected discounted future rewards $\mathbb{E}_\pi \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1})$.

π is a stochastic policy that specifies the probability of taking an action $a_t \in \mathcal{A}$ in state $s_t \in \mathcal{S}$. To maximize the expected returns, one possible technique is to optimize the policy directly: $\pi^* = \text{argmax}_\pi \, \mathbb{E}_\pi \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1})$

[*]equal contribution

[1]Appendix at https://tinyurl.com/yb8hzx73.

In this study we use proximal policy gradient (PPO) [Schulman *et al.*, 2017], which is an algorithm to optimize the policy directly using the policy gradient technique [Sutton *et al.*, 2000].

2.1 Multiagent Reinforcement Learning

In the multiagent setting, we consider a fully observable environment with K agents acting simultaneously. Most of the MDP formulation for RL with a single agent carries over to this setting without notational modifications. The action space is modified to $\boldsymbol{\mathcal{A}} \equiv \mathcal{A}^K$. At each time step, all agents take their actions, and the joint action vector $\boldsymbol{a}_t = [a_{1,t}, a_{2,t}, \ldots, a_{K,t}]$ acts on the state to produce the transition to the next state and reward, $R_e(s_t, \boldsymbol{a}_t, s_{t+1})$.

We consider a cooperative setting where the environment's reward $\mathcal{R}_e$ is shared among all the agents. In this study we focus on the decentralized learning scenario, in which each agent's policy π_k (parameterized by θ_k) is learned and updated separately by each individual agent. To learn in this manner, we provide this shared reward to all the agents and the agents individually use PPO to improve their policies, with a small learning step to allow the PPO algorithm learning to proceed in a stable manner.

2.2 Individual Preferences

We consider an individual preference for agent k as the policy π_k^p that the agent prefers to execute while performing the shared task. The agent can be initialized to this policy or can use a dataset $\mathcal{D}_k$ of tuples (s_t, a_t) generated by executing π_k^p and recover the maximum likelihood policy via Behavioral Cloning. We then model this preference via an individual reward R_k that most likely induces this preferred policy. The individual reward can be inferred via inverse RL [Abbeel and Ng, 2004]. Specifically, we use GAIL [Ho and Ermon, 2016] (an adversarial form of inverse RL) to infer an individual reward via the dataset $\mathcal{D}_k$.

3 Balancing Preferences with Shared Task

Figure 1 illustrates the framework of individual preference reward balanced with a shared objective. At each time step, the environment presents a reward $R_e(s_t, \boldsymbol{a}_t, s_{t+1})$. But each agent also receives its individual reward or preference signal R_k, inferred according to Section 2.2. For each agent, we balance R_e and R_k using a mixing factor $\alpha_k \in [0, 1]$. We refer to the combination of α values for all agents ($\langle \alpha_1, \alpha_2, \ldots, \alpha_K \rangle$) as the mixing scheme. We train the agent policies to maximize this weighted joint reward $R_{k,e}(s_t, \boldsymbol{a}_t, s_{t+1})$, calculated for the k^{th} agent as:

$$R_{k,e}(s_t, \boldsymbol{a}_t, s_{t+1}) = \alpha_k R_e(s_t, \boldsymbol{a}_t, s_{t+1}) + (1 - \alpha_k) R_k(s_t, a_{k,t}, s_{t+1}) \quad (1)$$

Both our theoretical analysis in Section 4 and our empirical analysis in Section 5 lead to the somewhat counter-intuitive conclusion that blending in individual preferences (i.e. setting the α_k's to be < 1) can aid team learning.

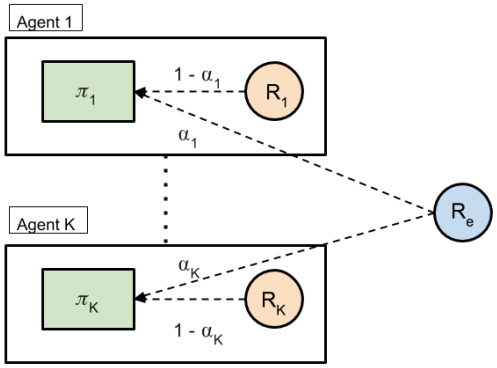

Figure 1: Multiagent Preference Balancing. Each Agent k balancing individual reward function R_k with a joint environment reward R_e using weights α_k.

4 Game-Theoretic Analysis

In this section we examine the interplay between individual preferences and task rewards. For simplicity, we look at a single step of a game, and show under which conditions individual preferences lead to maximizing the shared task reward. While we do not assume that the learning process converges to a Nash equilibrium, we show that if individual and environment returns are predicted correctly, they will lead to a Nash equilibrium for a certain range of α coefficients (i.e. mixing scheme). A priori, it seems intuitive that individual preferences can be substantially harmful to team performance if these preferences are not perfectly aligned with the shared task. However, this analysis establishes that there is a range of selfishness conditions under which the game does converge to an equilibrium which maximizes task reward even if selfish preferences are not perfectly aligned with the shared reward.

Let us assume there are two agents A and B interacting in a game [Leyton-Brown and Shoham, 2008] in which each agent can take one of two actions (this can be Bach-Stravinsky or Matching Pennies or any other game of this form). Agents A, B's preferences are denoted with the following payoff matrices, respectively:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \; B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

However, *unlike* a typical game, in this game there is also *an environment* which gives both agents the following reward according to their matching actions:

$$\textit{payoff}(\textit{environment}) = \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \end{bmatrix}$$

Let α_a, α_b be the mixing scheme for agents A, B respectively, determining their extent of blending environment and individual reward signals. The resulting game payoff matrices for A,B are then:

$$A_e = \begin{bmatrix} (1-\alpha_a)a_{11} + \alpha_a e_{11} & (1-\alpha_a)a_{12} + \alpha_a e_{12} \\ (1-\alpha_a)a_{21} + \alpha_a e_{21} & (1-\alpha_a)a_{22} + \alpha_a e_{22} \end{bmatrix}$$

$$B_e = \begin{bmatrix} (1-\alpha_b)b_{11} + \alpha_b e_{11} & (1-\alpha_b)b_{12} + \alpha_b e_{12} \\ (1-\alpha_b)b_{21} + \alpha_b e_{21} & (1-\alpha_b)b_{22} + \alpha_b e_{22} \end{bmatrix}$$

Assume w.l.o.g. that e_{11} yields the highest environment reward. Let us consider which α values could lead to e_{11}

being an equilibrium.[2] The combination of actions leading to e_{11} is a Nash equilibrium [Osborne and Rubinstein, 1994] if: $(1-\alpha_a)\cdot a_{11}+\alpha_a\cdot e_{11} > (1-\alpha_a)\cdot a_{21}+\alpha_a\cdot e_{21}$ and $(1-\alpha_b)\cdot b_{11}+\alpha_b\cdot e_{11} > (1-\alpha_b)\cdot b_{12}+\alpha_b\cdot e_{12}$.

Simplifying the math leads to: $\alpha_a > \frac{a_{21}-a_{11}}{e_{11}-e_{21}+a_{21}-a_{11}}$, and analogously, $\alpha_b > \frac{b_{12}-b_{11}}{e_{11}-e_{12}+b_{12}-b_{11}}$.

Let us define $\delta(a) = a_{21}-a_{11}$ and $\delta(e) = e_{21}-e_{11}$. Then $\delta(a)$ and $\delta(e)$ denote the utility in defecting from action a_{11} to action a_{21} in terms of selfish reward and environment reward, respectively. Observe that $\delta(e) \leq 0$ since by definition $\forall e_{ij}, e_{11} \geq e_{ij}$. Then $\alpha_a > \frac{\delta(a)}{\delta(a)-\delta(e)}$.

If we consider $\delta(a)$ as the selfish improvement by defecting to the suboptimal action, and $\delta(a)-\delta(e)$ as the impact of defecting combined with the global relative payoff of *not* defecting, then this means that α needs to be greater than the ratio between the agent's selfish improvement and this combined global payoff impact.

If we also consider that $1 \geq \alpha_a \geq 0$ and $1 \geq \alpha_b \geq 0$, we get a simple set of linear inequalities – if it has a feasible solution, then there exists a mixing scheme for the agents that would lead to the environment-wise ideal solution becoming a game-theoretic equilibrium. Observe that this analysis holds even for arbitrary values of e_{ij}, which implies that we can exhaustively study how multiple mixing schemes can lead to multiple equilibria. This observation can be straightforwardly generalized to k agents, leading to this set of inequalities:

$$\forall i \forall j \neq l . \alpha_i > \frac{a^i_{j\neq l} - a^i_l}{e^i_l - e^i_{j\neq l} + a^i_{j\neq l} - a^i_l}$$

With action l to be the best response for agent A_i.

Let us denote $\delta(a^i_j) = a^i_{j\neq l} - a^i_l$ and $\delta(e^i_j) = e_{j\neq l} - e^i_l$ (recall a^i_l and e^i_l are assuming the other agents' actions are fixed). Then we obtain a generalized set of inequalities:

$$\forall i \forall j . \alpha_i > \frac{\delta(a^i_j)}{\delta(a^i_j) - \delta(e^i_j)}$$

These inequalities, combined with the constraints that $\forall i . 1 \geq \alpha_i \geq 0$, can all be solved efficiently, and if satisfied, it means the vector of α_i values again leads to a Nash equilibrium. Refer to Appendix D for a detailed derivation. Interestingly, this result bears some resemblance to the findings of Peterson [Peterson, 2009], who studied cooperation in a generalization of coalition games.

We note that it is straightforward to extend this analysis to a multi-step scenario as a repeated stage game with the above conditions repeated at each step.

While this analysis is conducted for a single step k-agent game, it nonetheless provides formal support for the idea that optimal shared task performance can be reached as a stable equilibrium when taking individual preferences into account, even when these preferences are not aligned perfectly with the shared task reward. This analysis also specifies the concrete conditions the mixing scheme needs to satisfy in order to maximize environment reward.

[2] There may be other equilibria as well. The purpose of this analysis is to show that there are non-trivial α values (i.e. $\alpha < 1$) that lead to *some* equilibrium

5 Experimental Analysis

In this section we investigate empirically whether the above conclusion holds in more realistic, more complex settings.The experimental methodology is detailed in Algorithm 1. In Section 5.1 we detail two domains set in the multiagent MDP framework from section 2.1. In our first experiment (Section 5.2), we vary the different preference signals and the mixing schemes and compare the effect on learning the shared task. Somewhat surprisingly, we find that preferences accelerate improvement in the task performance in both these environments. Further, in Section 5.3 we demonstrate a method to find a mixing scheme that outperforms purely task-reward-based learning in both domains.

5.1 Environments

We study the above framework on two multiagent cooperative domains: the well known *predator prey* domain [Barrett *et al.*, 2013], and a new *chord generation* domain. We consider these domains as they are suited to different policy combinations by the agents (homogeneous policies for predator prey and heterogeneous for chord generation).

The first domain, predator prey, has 4 predators and one prey in a 10×10 toroidal grid world (i.e. wrapped around on all sides), and they prey is caught only if the predators surround the prey on all four sides. Episodes are 100 steps long with a reward of -1 at the end if the prey is not caught. If the prey is caught, the episode terminates with a reward of $+1$. All other steps have a reward of 0 and we use a discount factor $\gamma = 0.99$. The evaluation metric used is the number of steps it takes to capture the prey (lower is better). For preferences, we use three policies from the literature that can catch the prey and one policy that cannot (random actions). These policies also form the baselines we compare to. Refer to Appendix A.1 for further details.

For the second domain, we introduce the chord domain to learn how to generate a sequence of chords. A motivation for this setting is the musical ensemble scenario from Section 1.

Algorithm 1 Experimental Methodology

Input: K, preferences $R_1, R_2, \ldots R_K$, dataset $\mathcal{D}_1, \mathcal{D}_2, \ldots \mathcal{D}_K$, mixing values $\alpha_1, \alpha_2, \ldots, \alpha_k$, shared reward R_e, number of training episodes H, episode length T

Initialize policies $\pi_1, \pi_2, \ldots \pi_K$
for $i \leftarrow 0$ **to** K **do**
  pretrain(π_i, $\mathcal{D}_i$) using Behavioral Cloning
for $h = 0;\ h < H;\ h+ = 1$ **do**
  for $t = 0;\ t < T;\ t+ = 1$ **do**
    get joint action $\boldsymbol{a}_t$
    act in environment
    compute mixed reward $R_{k,e}$ (Equation 1)
    optimize policies

In this setting each agent plays a single note at each time step out of the 12 possible pitch classes, simply denoted with integer values $\{0, \ldots, 11\}$. [3] The agents must generate a valid chord (which is a possible $4 \cdot 12 = 48$ note configurations out of $12^4 = 20736$ possible 4-note configurations), as well as valid transitions (no repetitions or inversions).

The reward signal is designed to reward generating valid note configurations, and penalizing repetitions and inversions, and repeated intervals (same chord configuration with base note changed). We model different reward functions by changing the relative weighting between these two penalties (w_1) and between the above penalties and the rewards for valid chords (w_2). These criteria are heuristics designed by us, but with solid grounding in music theory, as surprise and novelty are basic driving forces in Western music [Cook, 1994]. It is important to note that while musical performance is subjective, we have designed this particular domain to focus on chord generation, which can be evaluated objectively through our reward function. The task is trained in a continuing manner (with no termination), over 30000 steps, with $\gamma = 0.99$. The individual preferences are policies that prefer certain chord sets, and are further differentiated with variations on the actual reward function they maximize (by setting w_1 and w_2 to $\{0.2, 0.8\}$ and $\{0.8, 0.2\}$).

5.2 Varying Mixing Schemes and Agent Configurations

As we've established in Section 4, under reasonable conditions there exists a mixing scheme, i.e. a blend of selfishness and selflessness for all agents, which ensures that the individual preferences do not take precedence over the shared task. To verify this effect in practice, in this section we study the performance of agents with different preferences and mixing schemes in the two aforementioned environments.

Given 4 possible agent preference models in each domain, we considered 23 different mixes of different agent types, sampled from the overall $4^4 = 256$ possible agent configurations. Similarly, to better study the trade offs between different agent configurations and different mixing schemes, we canvased a wide range of configurations. A technical discussion on how preferences were constructed in each environment is presented in Appendices A.1 and A.3.

Figure 2 presents the effects of different preferences and different mixing schemes at the end of training in the predator prey domain. Figure 3 presents the effects of different preferences and different mixing schemes at the end of training in the chord generation domain. Scores for these configurations are averaged over 8 independent runs. The red asterisk in each column marks the mixing scheme that works best for those preferences at that time step.

A surprising observation arising from Figures 2 and 3 is that the optimal configuration is not the baseline (four selfless agents attempting to maximize their reward w.r.t. shared task reward only). Rather, in each environment a mixing scheme exists which performs the best on a variety of agent mixes. Interestingly, even in the case of random preferences (preference #0 in Figure 2) in the predator-prey domain, despite overall reduced performance, an intermediate mixing scheme still does better than a selfless mixing scheme ($\alpha = 1$).

[3] Relating the scale to actual notes, 0 denotes C, 1 denotes $C\#$, 2 denotes D and so forth up to 11 = B.

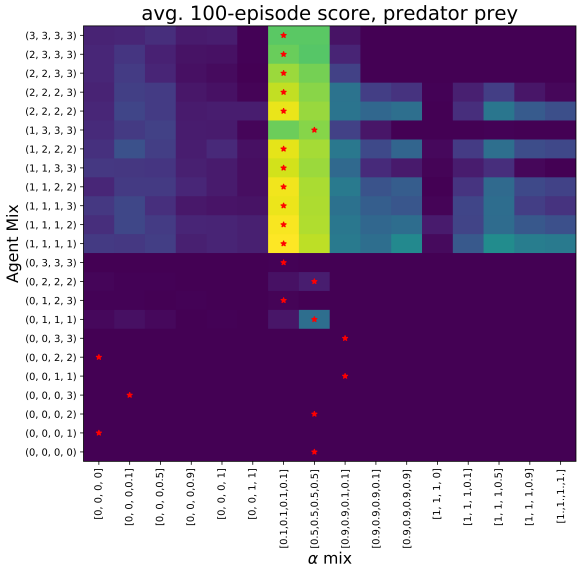

Figure 2: Heatmap of average score in the predator prey domain, smoothed using a 100 episode window. Score is number of steps taken by predators to catch the prey. The brighter the color, the better the score. X axis represents α mixing schemes. Y axis represents agent configurations. Results are averaged over $n = 9$ repetitions per agent mixture and α mixing scheme. A red star denotes the maximum over α mixing schemes for each given agent configuration.

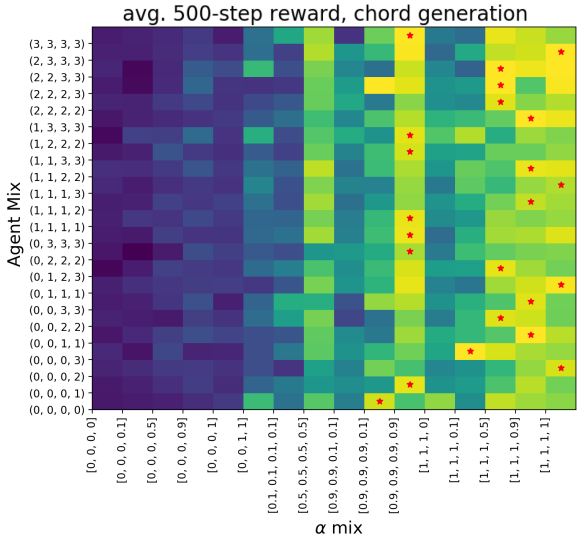

Figure 3: Heatmap of average rewards at the 29500-th step, smoothed using a 500 step window. The brighter the color, the higher the reward. X axis represents α mixing schemes, ordered lexically. Y axis represents agent configurations. Results are averaged over $n = 8$ repetitions per agent mixture and α mixing schemes. A red star denotes the maximum over α mixing schemes for each given agent configuration.

On a side note, we verify that this effect is not due to the reduced search space induced due to more directed policies (Appendix C).

5.3 Optimizing the Mixing Scheme

In the previous section we have established that some mixing schemes lead to accelerated improvement and better asymptotic performance on the shared task under certain conditions. This observation raises the question of how difficult it is to directly *optimize* these values in order to find a mixing scheme which leads to maximization of task performance. We find below that given a set of reasonable preferences, it is possible to find a mixing scheme that outperforms the purely task-reward-based approach rather handily.

In order to optimize the mixing scheme, we frame the problem as a constrained global Bayesian optimization task using Gaussian processes as described in [Snoek *et al.*, 2012]. Subsequently, we use the gradual posterior inference procedure to guide the optimization towards increasingly more beneficial mixing schemes. The actual evaluation of each mixing scheme sampled is done by running a trial on the shared task with these values and observing the resultant team performance. To evaluate the utility of a given mixing scheme, we run a trial with that scheme and consider the evaluation score.

The results for such a procedure in the predator-prey domain are presented in Figure 4. As can be observed, with a total of only 34 samples (i.e.runs) in total, we were able to find a configuration that outperforms the "preference-free" default of $\alpha = \langle 1, 1, 1, 1 \rangle$ (i.e., completely selfless agents with no previous preferences). More precisely, the mixing scheme in the neighborhood of $\alpha = \langle 0.15, 0.09, 0.19, 0.35 \rangle$ does best, leaning more towards preferences than the shared task reward. This mixing scheme resulted in a joint policy with an average 100 episode score of 17.68, more than twice as good as our strongest baseline in this domain.

Similarly, the optimization results in the chord generation domain (Figure 5) are based on 29 samples (trials) and radial basis function interpolation. The best mixing scheme found through the optimization procedure ($\langle 0.98, 0.69, 0.95, 0.49 \rangle$) leads to results more than three times better than the default values of pure selflessness ($\alpha = \{1, 1, 1, 1\}$). The above mixing scheme is interesting, indicating that two agents that are mostly selfless and two agents that balance selflessness and selfishness perform best in this domain. Such a result is somewhat reflective of observations made by Colman et al. [Colman *et al.*, 2008] and Sugden et al. [Sugden, 2008], who analyzed empirically observed team behaviors as a decomposition of selfish actors vs. team-reasoners.

These encouraging outcomes suggest a practical method to efficiently tune mixing schemes, making the framework proposed in this paper more immediately useful in cases in which adjusting the mixing scheme or selecting the agent configuration is feasible.

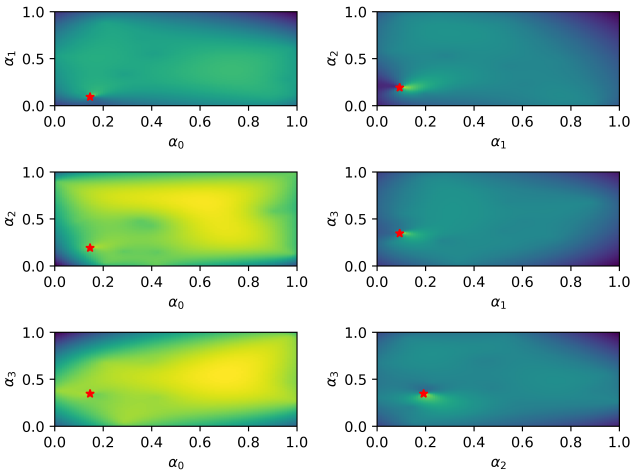

Figure 4: Results for optimizing the mixing scheme in predator prey domain. The search over 4 values is shown by comparing the heatmap of performance over groups of two mixing factors at a time. The asterisk indicates the optimal configuration found. It is demonstrably better than the baseline value of completely selfless agents.

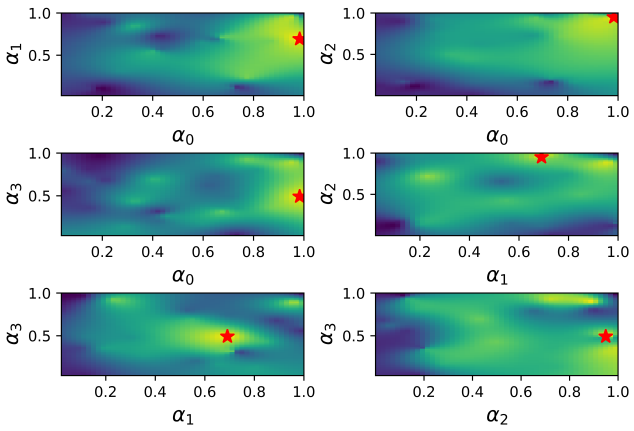

Figure 5: Results for optimizing the mixing scheme in the chord generation domain. The search over 4 values is shown by comparing the heatmap of performance over groups of two mixing factors at a time. The asterisk indicates the optimal configuration found. It is demonstrably better than the baseline value of completely selfless agents.

6 Related Work

Multiagent RL has been studied and applied in a variety of settings [Busoniu *et al.*, 2008], whereas the issue of cooperation and selfishness in team games has been of interest in the game theory literature for several decades [Aumann, 1961; Peterson, 2009]. There has been considerable work in scaling multiagent RL to more complex domains and studying multiagent behavior using Deep Reinforcement Learning [Tampuu *et al.*, 2017; Hernandez-Leal *et al.*, 2019; Foerster *et al.*, 2017], including human-level performance in multiplayer games [Jaderberg *et al.*, 2019]. In multiagent RL, MADDPG [Lowe *et al.*, 2017], QMIX [Rashid *et al.*, 2018] and COMA [Foerster *et al.*, 2018] have been recently proposed as algorithms to train agents. However, they assume a strong coordinating structure in learning, such as a central critic, while our approach considers completely decentralized policies.

Ad hoc teamwork [Barrett *et al.*, 2013; Stone *et al.*, 2010] is a setting in multiagent RL where individual agents come together to accomplish a shared task with no prior knowledge of the policies of the other agents. It is conceivable that in this setting, each agent has some previous policy that they prefer. Our work considers how to leverage such previously existing policies towards accomplishing the shared task.

Our approach uses GAIL as a way to generate the individual preferences which the agents use as their personal reward signals. The GAIL reward can be considered to induce directed exploration, and has been used in single agent RL scenarios to accelerate learning [Kang *et al.*, 2018; Zhu *et al.*, 2018].

The individual agent preferences we propose can also be considered as an approach to Curriculum Learning [Narvekar *et al.*, 2020]. These preferences can be seen as simpler tasks that can prime the agent policies for the shared task. They can also be considered an approach to multi-task Reinforcement Learning [Wilson *et al.*, 2007; Ammar *et al.*, 2014] where the agent attempts to learn a policy that can satisfy multiple objectives, or the individual reward - social choice setting in the multi-objective multi-agent taxonomy [Rădulescu *et al.*, 2020]. Another perspective on the individual preferences is as a sort of intrinsic motivation [Barto, 2013; Sequeira *et al.*, 2011], that uses it to encourage behaviors like exploration or better transfer [Barto *et al.*, 2004]. Previous work has also shown the effectiveness of learning a separate reward function for different agents [Jaderberg *et al.*, 2019], but does so from scratch using population based techniques. Our approach leverages known preferences to improve task performance. CM3 [Yang *et al.*, 2020] learns individual agent policies in isolation before introducing the multi-agent setting. However, they do so for tasks that each individual has to accomplish individually, as opposed to the more general coordinated tasks we consider here.

7 Discussion and Future Work

This paper analyzes the cooperative multiagent learning scenario where each agent has their own individual preference about how the shared task should be accomplished. We model this scenario with a linear mixing scheme that trades off the task reward with each agent's individual preference. Even though this model is fairly straightforward, we find it leads to the interesting outcomes where partial selfishness leads to better performance compared to purely selfless task based learning. This is a surprising result since our game theoretic analysis suggests that at best the individual preferences do not hurt the task performance. This gap between the empirical outcomes and theoretic analysis deserves more attention in future work.

Studying the impact of individual preferences on task performance is necessary in an environment where heterogeneous agents come together to accomplish a task (ad-hoc teamwork). While we do not address the question of where preferences come from, we show that a wide variety of preferences can be used in this scenario. On the other hand, our experiments with a preference that hurts task performance (random actions in predator prey) show that the exact conditions under which a preference is useful needs to be characterized further.

We also show that it is feasible to find a good mixing scheme for a given set of preferences using basic Bayesian Optimization. However, an interesting extension to this work would be to find a way to optimize the mixing scheme within the agent's lifetime.

Acknowledgements

This work has taken place in the Learning Agents Research Group (LARG) at UT Austin. LARG research is supported in part by NSF (CPS-1739964, IIS-1724157, NRI-1925082), ONR (N00014-18-2243), FLI (RFP2-000), ARO (W911NF-19-2-0333), DARPA, Lockheed Martin, GM, and Bosch. Peter Stone serves as the Executive Director of Sony AI America and receives financial compensation for this work. The terms of this arrangement have been reviewed and approved by the University of Texas at Austin in accordance with its policy on objectivity in research.

References

[Abbeel and Ng, 2004] Pieter Abbeel and Andrew Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04, pages 1–, New York, NY, USA, 2004. ACM.

[Ammar *et al.*, 2014] Haitham Bou Ammar, Eric Eaton, Paul Ruvolo, and Matthew Taylor. Online multi-task learning for policy gradient methods. In *International Conference on Machine Learning*, pages 1206–1214, 2014.

[Aumann, 1961] Robert J Aumann. The core of a cooperative game without side payments. *Transactions of the American Mathematical Society*, 98(3):539–552, 1961.

[Barrett *et al.*, 2013] Samuel Barrett, Peter Stone, Sarit Kraus, and Avi Rosenfeld. Teamwork with limited knowledge of teammates. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence (AAAI)*, July 2013.

[Barto *et al.*, 2004] Andrew G Barto, Satinder Singh, and Nuttapong Chentanez. Intrinsically motivated learning of hierarchical collections of skills. In *Proceedings of the 3rd International Conference on Development and Learning*, pages 112–19, 2004.

[Barto, 2013] Andrew G Barto. Intrinsic motivation and reinforcement learning. In *Intrinsically motivated learning in natural and artificial systems*, pages 17–47. Springer, 2013.

[Busoniu *et al.*, 2008] Lucian Busoniu, Robert Babuska, and Bart De Schutter. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, And Cybernetics-Part C: Applications and Reviews, 38 (2), 2008*, 2008.

[Colman *et al.*, 2008] Andrew M Colman, Briony D Pulford, and Jo Rose. Team reasoning and collective rationality: Piercing the veil of obviousness. *Acta psychologica*, 128(2):409–412, 2008.

[Cook, 1994] Nicholas Cook. *A guide to musical analysis*. Oxford University Press, 1994.

[Foerster *et al.*, 2017] Jakob Foerster, Nantas Nardelli, Gregory Farquhar, Triantafyllos Afouras, Philip HS Torr, Pushmeet Kohli, and Shimon Whiteson. Stabilising experience replay for deep multi-agent reinforcement learning. *arXiv preprint arXiv:1702.08887*, 2017.

[Foerster *et al.*, 2018] Jakob N Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[Hernandez-Leal *et al.*, 2019] Pablo Hernandez-Leal, Bilal Kartal, and Matthew E Taylor. A survey and critique of multiagent deep reinforcement learning. *Autonomous Agents and Multi-Agent Systems*, 33(6):750–797, 2019.

[Ho and Ermon, 2016] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4565–4573. Curran Associates, Inc., 2016.

[Jaderberg *et al.*, 2019] Max Jaderberg, Wojciech M Czarnecki, Iain Dunning, Luke Marris, Guy Lever, Antonio Garcia Castaneda, Charles Beattie, Neil C Rabinowitz, Ari S Morcos, Avraham Ruderman, et al. Human-level performance in 3d multiplayer games with population-based reinforcement learning. *Science*, 364(6443):859–865, 2019.

[Kang *et al.*, 2018] Bingyi Kang, Zequn Jie, and Jiashi Feng. Policy optimization with demonstrations. In *International Conference on Machine Learning*, pages 2474–2483, 2018.

[Leyton-Brown and Shoham, 2008] Kevin Leyton-Brown and Yoav Shoham. Essentials of game theory: A concise multidisciplinary introduction. *Synthesis lectures on artificial intelligence and machine learning*, 2(1):1–88, 2008.

[Lowe *et al.*, 2017] 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*, pages 6379–6390, 2017.

[Narvekar *et al.*, 2020] Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E. Taylor, and Peter Stone. Curriculum learning for reinforcement learning domains: A framework and survey, 2020.

[Osborne and Rubinstein, 1994] Martin J Osborne and Ariel Rubinstein. *A course in game theory*. MIT press, 1994.

[Peterson, 2009] Elisha Peterson. Cooperation in subset team games: altruism and selfishness. *arXiv preprint arXiv:0907.2376*, 2009.

[Rădulescu *et al.*, 2020] Roxana Rădulescu, Patrick Mannion, Diederik M Roijers, and Ann Nowé. Multi-objective multi-agent decision making: a utility-based analysis and survey. *Autonomous Agents and Multi-Agent Systems*, 34(1):10, 2020.

[Rashid *et al.*, 2018] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder De Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Qmix: monotonic value function factorisation for deep multi-agent reinforcement learning. *arXiv preprint arXiv:1803.11485*, 2018.

[Schulman *et al.*, 2017] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[Sequeira *et al.*, 2011] Pedro Sequeira, Francisco S Melo, Rui Prada, and Ana Paiva. Emerging social awareness: Exploring intrinsic motivation in multiagent learning. In *2011 IEEE International Conference on Development and Learning (ICDL)*, volume 2, pages 1–6. IEEE, 2011.

[Snoek *et al.*, 2012] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012.

[Stone *et al.*, 2010] Peter Stone, Gal A Kaminka, Sarit Kraus, Jeffrey S Rosenschein, et al. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *AAAI*, 2010.

[Sugden, 2008] Robert Sugden. Nash equilibrium, team reasoning and cognitive hierarchy theory. *Acta Psychologica*, 128(2):402–404, 2008.

[Sutton and Barto, 2018] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[Sutton *et al.*, 2000] 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*, pages 1057–1063, 2000.

[Tampuu *et al.*, 2017] Ardi Tampuu, Tambet Matiisen, Dorian Kodelja, Ilya Kuzovkin, Kristjan Korjus, Juhan Aru, Jaan Aru, and Raul Vicente. Multiagent cooperation and competition with deep reinforcement learning. *PloS one*, 12(4):e0172395, 2017.

[Wilson *et al.*, 2007] Aaron Wilson, Alan Fern, Soumya Ray, and Prasad Tadepalli. Multi-task reinforcement learning: a hierarchical bayesian approach. In *Proceedings of the 24th international conference on Machine learning*, pages 1015–1022. ACM, 2007.

[Yang *et al.*, 2020] Jiachen Yang, Alireza Nakhaei, David Isele, Kikuo Fujimura, and Hongyuan Zha. Cm3: Cooperative multi-goal multi-stage multi-agent reinforcement learning. In *International Conference on Learning Representations*, 2020.

[Zhu *et al.*, 2018] Yuke Zhu, Ziyu Wang, Josh Merel, Andrei A. Rusu, Tom Erez, Serkan Cabi, Saran Tunyasuvunakool, János Kramár, Raia Hadsell, Nando de Freitas, and Nicolas Heess. Reinforcement and imitation learning for diverse visuomotor skills. *CoRR*, abs/1802.09564, 2018.