International Joint Conference on Artificial Intelligence (IJCAI-ECAI) 2022

Exploring the Benefits of Teams in Multiagent Learning

David Radke[1*], **Kate Larson**[1] and **Tim Brecht**[1]
[1]David R. Cheriton School of Computer Science, University of Waterloo
{dtradke, kate.larson, brecht}@uwaterloo.ca

Abstract

For problems requiring cooperation, many multiagent systems implement solutions among either individual agents or across an entire population towards a common goal. Multiagent teams are primarily studied when in conflict; however, organizational psychology (OP) highlights the benefits of teams among human populations for learning how to coordinate and cooperate. In this paper, we propose a new model of multiagent teams for reinforcement learning (RL) agents inspired by OP and early work on teams in artificial intelligence. We validate our model using complex social dilemmas that are popular in recent multiagent RL and find that agents divided into teams develop cooperative pro-social policies despite incentives to not cooperate. Furthermore, agents are better able to coordinate and learn emergent roles within their teams and achieve higher rewards compared to when the interests of all agents are aligned.

1 Introduction

Observed in both animal and human behavior, the ability to work in teams can magnify a group's abilities beyond the capability of individuals. Organizational psychology (OP) and biology have done extensive research on how "teams-of-teams" with a shared overall goal increase the collective efforts of all component teams and individuals [Zaccaro *et al.*, 2020]. Recently, there is increasing interest in making the study of cooperation central to the development of artificial intelligence (AI) and multiagent systems (MAS) [Dafoe *et al.*, 2021]. We propose that adapting a similar team structure as in OP to a population of AI agents serves as a middle ground between centralized and decentralized systems of coordination and cooperation that can benefit agents' learning processes.

In the context of teams, multiagent reinforcement learning (MARL) has achieved impressive results in competitive two-team zero-sum settings such as capture the flag [Jaderberg *et al.*, 2019], hide-and-seek [Baker *et al.*, 2019], and Robot Soccer (RoboCup) [Kitano *et al.*, 1997]. However, when agents are deployed into the real world, they will be faced with problems that are not zero-sum [Baker, 2020]. Therefore, it is essential to explore cooperation in mixed-motive domains, such as Sequential Social Dilemmas (SSDs) [Leibo *et al.*, 2017]. With the growing interest in exploring mixed-motive domains and investigating the impact of group size and structure on system stability [Nisioti and Moulin-Frier, 2020], we aim to model teams and understand their benefits on MARL agents' ability to learn.

Inspired by group structures in OP and early models of teams from the AI literature for task completion, we propose a general model of multiagent teams and validate it in the context of social dilemmas. It is well documented that individual RL agents fail to learn cooperation in social dilemmas while agents with common interest have more success [Anastassacos *et al.*, 2020; Baker *et al.*, 2019]. Our teams model is situated between these two extremes, where teammates are bound by common interest but mixed-motives exist between non-teammates. We show, across several games, that teams improve how agents learn and develop pro-social policies. This work makes the following contributions:

- We define a model of teams inspired by early work in multiagent systems and organizational psychology.
- We discuss the theoretical ramifications of our model in the context of social dilemmas regarding game theoretical incentives under different environmental conditions.
- Through an extensive empirical evaluation, we show how our model of teams helps agents develop globally beneficial pro-social behavior despite short-term incentives to defect. As a result, agents in these teams achieve higher rewards in complex domains than when the interests of all agents are aligned, and autonomously learn more efficient combinations of roles when compared with common interest scenarios.

2 Related Work

There is a long history of exploring how agents can coordinate their behavior via teamwork, typically for task completion domains. Early work by Pollack formally defines an agent's mental model for making collaborative plans between two agents [Pollack, 1990]. Extending Pollack's work, [Grosz and Sidner, 1988] construct SharedPlans, a model that includes more actions and agent beliefs, similar to shared

[*] Contact Author

mental models in human teamwork and modeled joint actions as non-additive [Grosz and Kraus, 1996]. Tambe uses a task allocation structure similar to SharedPlans in developing STEAM, a general model of teams where tasks can be completed by sub-teams of agents within a larger system [Tambe, 1997]. The idea of sub-teams within a population was novel to AI; however, SharedPlans and STEAM remained unable to broadly generalize due to limited agent capabilities.

Recent work with teams in MAS has focused on ad hoc teamwork [Macke *et al.*, 2021], teams in competition [Ryu *et al.*, 2021], or coordination problems [Jaques *et al.*, 2019]. When not in competition, teams in AI are typically designed to achieve a common objective and has been criticized for its disregard for adapting significant findings in OP [Andrejczuk *et al.*, 2016]. We evaluate our model of teams in other settings where teams are not directly competitive, complex mixed-motive social dilemmas, to explore how structures inspired by OP can be adapted to MAS.

Studying social dilemmas and how agents or people can overcome them has been a topic of research in game theory, economics, psychology, and more recently, AI. Fostering cooperation in social dilemmas with MARL agents often relies on reward sharing among the population [McKee *et al.*, 2020] or only a subset of agents with uncertainty [Baker, 2020]. We broadly classify the literature into centralized and decentralized approaches of fostering cooperation.

Centralized systems have taken various forms in the literature, ranging from centralized training for better coordination at test time [Kraemer and Banerjee, 2016] to explicitly providing access to another agent's internal state [Deka and Sycara, 2021]. While centralized systems are efficient and convergence is reliable, it is often unsafe to assume access to all agents and they are brittle to exogenous changes.

Inspired by the hypothesized emergence of cooperation in humans and evolutionary game theory, decentralized systems that promote cooperation are primarily implemented at the individual agent level. Giving agents the ability to punish and sanction others in response to a specific behavior has been shown to foster cooperation in MARL [Anastassacos *et al.*, 2021; Leibo *et al.*, 2017].

In this work, we adapt findings from OP to AI by constructing a general model of multiagent teams inspired by human teams, SharedPlans, and STEAM to analyze the benefits of teams on how MARL agents learn. We divide a population of MARL agents into teams not in direct competition and show how teams, as they have been found to do with humans, improve how RL agents co-evolve and learn in challenging domains. We position our work between centralized and decentralized systems. While our teams model does not assume any form of centralized control, the team structure itself provides a way for agents to better coordinate.

3 A Model for Multiagent Teams

We model our base environment as a stochastic game $\mathcal{G} = \langle N, S, \{A\}_{i \in N}, \{R\}_{i \in N}, P, \gamma, \Sigma \rangle$. N is our set of all agents that learn online from experience and S is the state space, observable by all agents, where s_i is a single state observed by agent i. $A = A_1 \times \ldots \times A_N$ is the joint action space for all agents where A_i is the action space of agent i. $R = R_1 \times \ldots \times R_N$ is the joint reward space for all agents where R_i is the reward function of agent i defined as $R_i : S \times A \times S \mapsto \mathbb{R}$, a real-numbered reward for taking an action in an initial state and resulting in the next state. $P : S \times A \mapsto \Delta(S)$ represents the transition function which maps a state and joint action into a next state with some probability and γ represents the discount factor so that $0 \leq \gamma < 1$. Σ represents the policy space of all agents, and the policy of agent i is represented as $\pi_i : S \mapsto A_i$ which specifies an action that the agent should take in an observed state.[1]

Our teams model consists of a stochastic game with teams $\langle \mathcal{G}, \mathcal{T} \rangle$, where $\mathcal{T}$ is a partition of the population of agents into disjoint teams, $\mathcal{T} = \{T_i | T_i \subseteq N, \cup T = N, T_i \cap T_j = \emptyset \forall i, j\}$. A team *structure* defines the composition of $\mathcal{T}$, such as the number of teams and agents on each team. Consistent with the original groundwork on multiagent teams [Tambe, 1997; Grosz and Sidner, 1988], we define a team of agents as being bounded together through common interest. Consistent with recent MARL work, we model common interest through reward sharing making the assumption agents value rewards equivalently [McKee *et al.*, 2020; Hughes *et al.*, 2018]. We define a new reward function for agents on a team as $TR_i : S \times A \times S \mapsto \mathbb{R}$ so that the reward for $i \in T_i$ is dependant on their own behavior and that of their teammates. Any function can be implemented to define TR_i. In our analysis and experiments, we use:

$$TR_i = \frac{\sum_{j \in T_i} R_j(S, A, S')}{|T_i|}, \qquad (1)$$

where teammates share their rewards equally to be consistent with past work [Wang *et al.*, 2019; Baker *et al.*, 2019].

Agents learn from their individual experience using RL. As is standard in many MARL problems, agents are trained to independently maximize their own rewards. In particular, at time t each agent i selects some action a_i which together form a joint action a^t. This action results in a transition from state s^t to state s^{t+1}, according to the transition function P, and provides each agent i with reward $R_{i,t}(s^t, a^t, s^{t+1})$. Agents seek to maximize their sum of discounted future rewards, $V_i = \sum_{t=0}^{\infty} \gamma^t R_{i,t}$. Our model replaces R_i with TR_i, reconfiguring the learning problem so that agents must simultaneously learn what individual behavior maximizes their team's expected discounted future reward.

4 Social Dilemmas

Social dilemmas are situations which present tension between short-term individual incentives and the long-term collective interest where agents must cooperate for higher rewards. All agents prefer the benefits of a cooperative equilibria; however, the short-term benefits of self-interested behavior outweigh those of cooperative behavior. For our analysis, we consider intertemporal social dilemmas with active provision defined as when cooperation carries an explicit cost [Hughes *et al.*, 2018]. We implement our model of teams in the Iterated Prisoner's Dilemma (IPD) [Rapoport, 1974] and the Cleanup domain game [Vinitsky *et al.*, 2019].

[1] We can also allow for randomized policies.

4.1 Environment 1: Iterated Prisoner's Dilemma

In the one-shot Prisoner's Dilemma, two agents interact and each must decide to either cooperate (C) with or defect (D) on each other. We assume there is a cost (c) and a benefit (b) to cooperating where $b > c > 0$. If an agent cooperates, it incurs the cost c. If both agents cooperate, they both also benefit, each receiving a reward of $b - c$. If one agent cooperates but the other defects, then we assume that the cooperating agent incurs the cost c, but the defecting agent reaps the benefit b (e.g., by stealing the contribution of the cooperator). If neither cooperate, neither benefit nor incur a cost, leading to a reward of zero for both. The unique Nash Equilibrium is obtained when both agents defect, represented by (D, D). Joint cooperation does not form an equilibrium, since if one agent cooperates, the other agent is strictly better off defecting and receiving b, instead of $b - c$.

In the IPD, this game is repeatedly played which adds a temporal component and allows agents to learn a policy over time. Instead of just two agents, we work with a population of agents that are divided into teams *a priori*. At each timestep, agents are randomly paired with another agent, a *counterpart*, that may or may not be a teammate. Agents are informed as to what team their counterpart belongs to through a numerical signal s_i, though additional identity information is not shared. Agents must decide to cooperate with or defect on their counterpart. Their payoff for the interaction is the team reward, TR_i, based on their own and other teammates' interactions. Agents update their strategies (i.e., learn) using their direct observation s_i, what action they chose a_i, and their team reward TR_i. Since only the team information of the counterpart is shared, the strategies of all agents on team T_i ultimately affects how agents learn to play any member of T_i.

Equilibrium Analysis

We are interested in understanding how the introduction of teams may help or hinder cooperation. As a first step towards addressing this question, we investigate the impact of teams on the *stage game* of the IPD. To provide a clear comparison with the standard IPD, we take an ex-ante approach, where agents are aware of their imminent interaction and the existence of other teams but not the actual team membership of their counterpart. For further details, refer to Appendix A[2].

Assume a pair of agents, i, j, have been selected to interact at some iteration of the IPD and agent i knows j will be a teammate with probability ν and a non-teammate with probability $(1 - \nu)$. Let $\sigma_{T_i} = (\sigma_{ji}, 1 - \sigma_{ji})$ represent j's strategy profile when $j \in T_i$, where σ_{ji} is the probability for cooperation (C). Likewise, let $\sigma_{T_j} = (\sigma_{jj}, 1 - \sigma_{jj})$ be j's strategy profile when $j \in T_j$, any other team.

If agent i decides to cooperate, its expected utility, subject to agent j's strategy, is:

$$\mathbb{E}(C, \sigma_T) = \frac{\nu(b - c)(\sigma_{ji} + 1)}{2} + (1 - \nu)(\sigma_{jj} b - c). \quad (2)$$

If agent i decides to defect, its expected utility, subject to agent j's strategy, is:

$$\mathbb{E}(D, \sigma_T) = \frac{\nu \sigma_{ji}(b - c)}{2} + (1 - \nu)\sigma_{jj} b. \quad (3)$$

We determine the conditions under which agent i has incentive to cooperate as when $\mathbb{E}(C, \sigma_T) \geq \mathbb{E}(D, \sigma_T)$. Substituting Equations 2 and 3, this simplifies to:

$$\nu \geq \frac{2c}{b + c}, \quad (4)$$

regardless of σ_{T_i} or σ_{T_j}. Since $b > c > 0$, this constraint is meaningful. Satisfying Equation 4 shifts the Nash Equilibrium of the stage game from (D, D) to (C, C) in expectation. This means that there exist circumstances where teams can support cooperative incentives, but these circumstances are not universal. Our experiments explore this in more detail.

4.2 Environment 2: Cleanup Markov Game

Cleanup [Vinitsky *et al.*, 2019] is a temporally and spatially extended Markov game representing a social dilemma. This domain allows us to examine if the benefits of teams generalize to more complex environments since agents must learn a cooperative policy through movement and decision actions instead of choosing an explicit *cooperation* action like in the IPD. Active provision is represented in Cleanup by agents choosing actions with no associated environmental reward that are necessary agents to achieve any rewards.

Figure 6 in Appendix B.2 shows the Cleanup environment and provides more details about the environment parameters. At each timestep, agents choose among nine actions: 5 movement (up, down, left, right, or stay), 2 turning (left or right), and a cleaning or punishing beam. One half of the map contains an aquifer, or river, and the other an apple orchard. Waste accumulates in the river with some probability at each timestep which must be cleaned by the agents. Once a cleanliness threshold is reached, apples spawn in the orchard proportional to the overall cleanliness of the river. Agents receive a reward of +1 for consuming an apple by moving on top of them. The dilemma exists in agents needing to spend time cleaning the river to spawn new apples and receive no exogenous reward versus just staying in the orchard and enjoying the fruits of another's labor. Agents have the incentive to stay in the orchard, however if all agents attempt this free-riding policy, no apples grow and no-one gets any reward. A successful group in Cleanup will balance the temptation to free-ride with the public obligation to clean the river.

5 Empirical Evaluation

In this section, we present the setup and results of experiments in both environments using MARL agents. While our teams model does not require it, we assume that for all $T_i, T_j \in \mathcal{T}$, $|T_i| = |T_j|$ (i.e., given a team model, the teams are the same size). This avoids complications that might arise with agent interactions if teams were significantly different sizes and to be consistent across our domains. Alternative interaction mechanisms and teams of different sizes are left for future work. We use the notation $|\mathcal{T}|/|T_i|$ to indicate the total number of teams and the size of each team. For example, $1/N$

[2] https://cs.uwaterloo.ca/~dtradke/pdfs/IJCAI22_Appendix.pdf

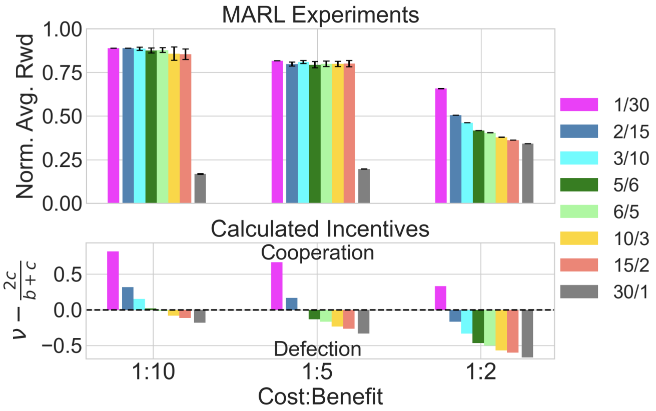

Figure 1: **IPD:** The top graph shows the normalized average population reward of MARL experiments with three different cost:benefit ratios when $N = 30$ with 95% confidence intervals. The bottom graph shows incentivized actions by Equation 4, where positive (or zero) is cooperation and negative is defection being incentivized. Team structures are labeled $|\mathcal{T}|/|T_i|$ and bookended with fully common interest (1/30) and fully mixed-motive (30/1). When $b \in \{5, 10\}$, every team structure besides the individualistic case (30/1) achieves about as much reward as 1/30 without requiring full common interest in the population.

indicates one team of N agents (fully common-interest) and $N/1$ represents N teams of one agent (fully mixed-motive). Of course, many scenarios may fall between these two extremes. Since fully mixed-motive has agents working as individuals (i.e., no teams), it serves as a benchmark against which we can compare the performance of team structures.

5.1 IPD Evaluation

In the IPD, each experiment lasts 1.0×10^6 episodes where $N = 30$ agents learn using Deep Q-Learning [Mnih *et al.*, 2015]. An episode is defined by a set of agent interactions where each agent is paired with another agent and plays an instance of the Prisoner's Dilemma. Agent pairings are assigned using a uniform random distribution over each team so agents are unable to explicitly modify who they interact with, shown as a challenging scenario for cooperation to arise without additional infrastructure [Anastassacos *et al.*, 2020]. Each experiment is repeated five times. In Appendix B.1, we prove how this configuration ensures that each agent has the same number of expected interactions to learn from. Further implementation details are provided in Appendix B.1[3].

Reward

In our first set of experiments, we explore the degree to which team structures support cooperation. We fix the cost (c) at 1, and let the benefit (b) be 2, 5, or 10. To capture the behavior of agents after they have converged to a policy, the top graph of Figure 1 shows the normalized average global reward of the last 25% of the episodes using MARL agents. We normalize the average global reward of each experiment in the interval $[0 - c, 0 + b]$ and calculate 95% confidence intervals to compare different cost and benefit ratios. To show the corresponding incentives of each experiment, we include the bottom graph which displays the calculated action incentive by a modified Equation 4, $\nu - \frac{2c}{b+c}$. Each bar in this graph corresponds with the experiment above so that positive (or zero) bars represent cooperation being incentivized and negative bars represent defection. Cost and benefit ratios are arranged from highest benefit (left) to lowest benefit (right).

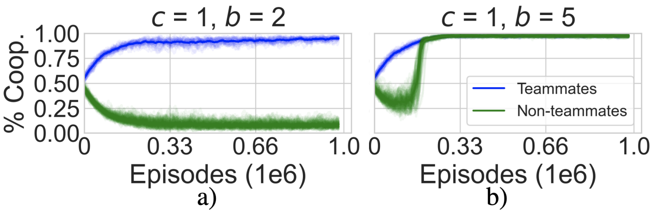

Figure 2: **IPD:** The 5/6 team composition showing the percent of cooperation towards teammates and non-teammates when $c = 1$ and $b \in \{2, 5\}$. When benefit is greater, agents develop pro-social policies towards non-teammates despite the incentive to defect.

By Equation 4, team structures present agents with the incentive to defect in 13 of 18 scenarios (72%) in Figure 1 (not including 1/30 and 30/1). Our results show teams always achieve more reward than individual agents (30/1); however, this reward depends on the cost and benefit ratio. When $b = 2$, the experiment results for average population reward a follow trend to the incentives of each scenario in the bottom graph. Our main finding in Figure 1 is how, when the benefit increases, MARL agents achieve high average population reward despite the incentive to defect as shown in the bottom graph. When $b \in \{5, 10\}$, every team structure, other than the individualistic 30/1 scenario, achieves basically the same reward as 1/30 even though there exists mixed-motive environments. Defection is the incentivized action in seven of 12 (58%) of these experiments which would produce low global reward if agents actually learned defection. Instead, we observe agents develop reciprocally pro-social policies that achieve high rewards in every scenario with teams of multiple agents when $b \in \{5, 10\}$. To analyze how high rewards are achieved in environmental conditions which promote defection, we study agents' behavior over time.

Policy

In evolutionary biology, fostering cooperation at various *levels* has been found to depend on the size of the cooperative return [Schnell *et al.*, 2021]. Different types of cooperation, or levels of cooperation, have yet to be explicitly explored in MARL. We identify two levels of cooperation in our IPD environment: with teammates and with non-teammates. Figure 2 shows the percent of cooperative actions over time with the 5/6 team structure when $b \in \{2, 5\}$. By Equation 4, agents have the incentive to defect in both scenarios. The x-axis shows time and the y-axis shows the percent of an agent's actions that are cooperation (2,000 episode averages).

Both graphs in Figure 2 show that agents immediately learn to cooperate with teammates regardless of b. When $b = 2$, agents defect on non-teammates; however, when $b = 5$, agents learn to cooperate with both teammates and non-teammates. We observe similar behavior with every other

[3] Code: https://github.com/Dtradke/Teams_IPD

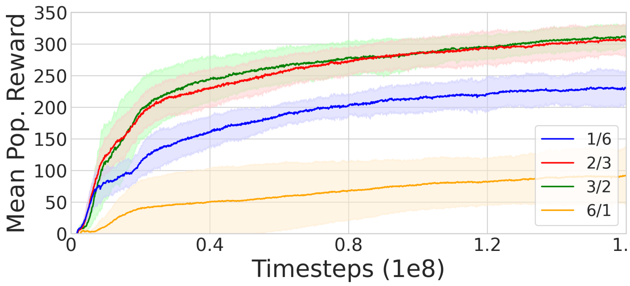

Figure 3: **Cleanup:** Mean population reward for each team structure with 95% confidence intervals. 6/1 represents individualistic agents and 1/6 represents when all agents have common interest. Both 2/3 and 3/2 team structures achieve more reward than 1/6 and 6/1.

team structure (not including 30/1) when $b \in \{5, 10\}$. That is, cooperation emerges with teammates and non-teammates despite incentives to defect. While other work requires strong assumptions of agent behavior to foster cooperation, our results indicate teams allow agents to learn an emergent cooperative convention at multiple levels of a system.

5.2 Cleanup Evaluation

In Cleanup, similar to previous work [Hughes *et al.*, 2018; McKee *et al.*, 2020; Jaques *et al.*, 2019], we experiment with $N = 6$ agents. Our agents use the Proximal Policy Optimization (PPO) [Schulman *et al.*, 2017] RL algorithm for 1.6×10^8 environmental timesteps (each episode is 1,000 timesteps). Agent observability is limited to a 15 × 15 RGB window. Teammates share the same color and optimize for TR_i calculated at each environmental timestep. Each experiment is repeated for eight trials. Further details are in Appendix B.2.

Reward

Figure 3 shows the mean population reward for each scenario in Cleanup with 95% confidence intervals. It has been previously shown that the setting that achieves the most population reward is when agents are altruistic and optimize for the collective rewards of the entire group [Wang *et al.*, 2019; McKee *et al.*, 2020], similar to our 1/6 configuration. However, teams introduce a new dynamic to the environment and we find the 2/3 and 3/2 team structures both achieve higher reward than 1/6 despite the interests of all agents not being aligned. As expected, the 6/1 scenario fails to achieve significant reward since agents succumb to the incentive to free ride and few apples grow. [McKee *et al.*, 2020] has shown that only evaluating a system for mean reward masks other dynamics such as high levels of reward inequality among agents.

Equality

It is important to consider the process of how teams achieve higher reward and the potential side effects on population equality, such as the reward distributed among agents. We model population reward equality as the inverse Gini index, similar to past work [McKee *et al.*, 2020], calculated as:

$$Equality = 1 - \frac{\sum_{i=0}^{N} \sum_{j=0}^{N} |R_i - R_j|}{2|N|^2 \overline{R_N}}, \tag{5}$$

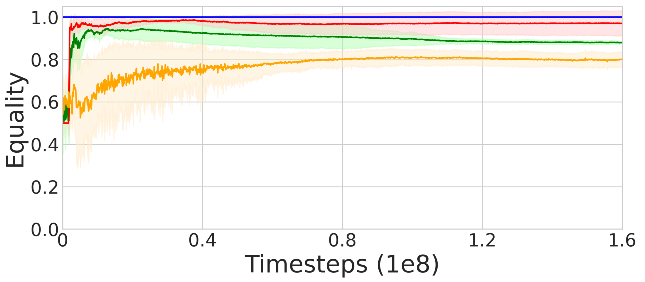

Figure 4: **Cleanup:** Inverse Gini index (equality) for each team structure with 95% confidence intervals. Higher values represent more equality. Both 2/3 and 3/2 team structures have high equality despite the interests of all agents not being aligned.

where $\overline{R_N}$ is the mean population reward. Figure 4 shows our results for equality where higher values represent more equality. The 1/6 scenario is, by definition, always 1 since there is only one team. Despite earning high reward, both 2/3 and 3/2 team structures also achieve high equality and always have greater equality than 6/1. Success in Cleanup relies on agents coordinating to form an effective joint policy instead of simply choosing an explicit cooperation action (as in the IPD). To further understand how team structures achieve the highest rewards while also maintaining high equality, we analyze agents' policies and division of labor among teammates.

Division of Labor

While agents consistently learn to divide labor among them, the same numbered agent does not always learn the same behavior across different trials of our experiments which makes aggregating multiple trials difficult. Therefore, Figure 5 shows the mean apples picked (top) and cleaning beams (bottom) for each agent in one trial of our evaluation. The behavior in this trial represents the most common division of labor for each team structure. Agents on the same team in each plot are presented as different shades of the same color. The y-axis shows the number of apples collected or cleaning actions taken and the x-axis represents time. Agents rarely punish, thus we omit it from our analysis.

In the 1/6 configuration (Figure 5, left), two agents learn to mostly pick apples while four agents clean the river. While this represents the most common division of labor with 1/6, we do observe two trials where three agents learn to pick apples and three agents learn to clean the river. These strategies achieve high mean reward but is not the best division of labor and consistently achieves less reward than the 2/3 and 3/2 team structures. When analyzing both team structures of 2/3 and 3/2 (Figure 5, middle columns), agents tend to divide themselves into four apple pickers and two river cleaners. This division of labor consistently achieves the highest reward in our evaluation. The 3/2 team structure tends to learn this division slightly quicker, although on average both configurations eventually achieve basically the same reward as shown in Figure 3. Independent agents in 6/1 fail to significantly clean the river, therefore few apples grow which leads to low rewards. In the example shown in Figure 5, five agents free-ride on the labor of only one river cleaning agent.

In summary, our results show how agents in team struc-

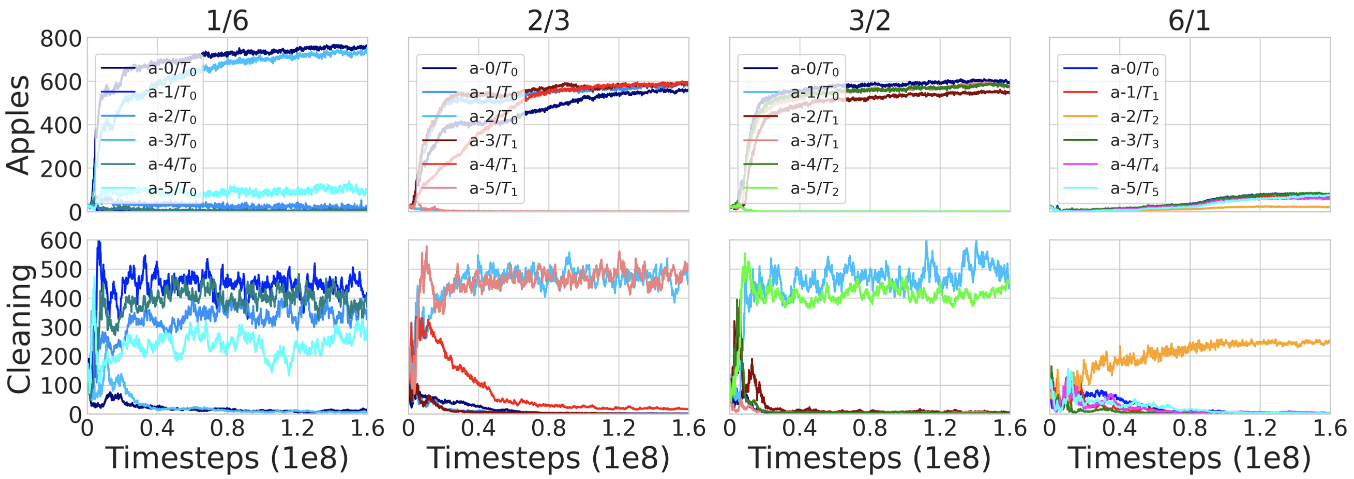

Figure 5: **Cleanup:** Mean number of apples (top) and cleaning beams (bottom) per-episode for each of 6 agents in different team structures.

tures learn better task specialization among the population by autonomously learning *roles* within their team. This allows populations in the 2/3 and 3/2 team structures to keep the river clean while most agents collect the spawning apples. This causes 2/3 and 3/2 to achieve high population reward and equality across teams even though agents on different teams optimize for their own team's reward.

6 Discussion and Future Work

Across multiple domains, we have shown that our model of teams has a significant impact on how agents learn to develop pro-social policies and coordinate their behavior. In the IPD, we show how teams allow agents to immediately identify and cooperate with their teammates, which may be similar to kin selection in human behavior [Muthukrishna, 2017]. Interestingly, we find that RL agents develop a prosocial convention and adapt this cooperative behavior towards non-teammates, even if defection has greater expected value. This behavior may be comparable with different levels of cooperation in humans, similar to increasing cooperation from only kin selection to notions of direct reciprocity with other groups [Muthukrishna, 2017].

While it was previously thought that optimizing for signals from all agents achieves the highest reward in Cleanup [Wang *et al.*, 2019; McKee *et al.*, 2020], our results show that agents optimizing for only a subset of the population (i.e., a team) achieves higher reward. Agent specialization in Cleanup is identified in [McKee *et al.*, 2020]. In that work, specialization is viewed as a negative result which causes high labor inequality. However, the context of teams should change how task specialization is viewed in MARL. In the literature on Team Forming and Coalition Structure Generation, teams are often explicitly constructed to fill necessary roles [Andrejczuk *et al.*, 2016]. We view task specialization as the agents autonomously learning these roles with only the feedback of their team's reward. This reinforces our hypothesis that teams can help improve how MARL agents learn to coordinate, and may be of specific interest to the emergent behavior community.

However, certain side effects may occur among teams. While our 3/2 team structure achieves high reward in Cleanup, there is higher inequality than 2/3. To achieve the four picker/two cleaner division of labor, one team (T_1 (red) in Figure 5) must free-ride on the labor of the other two teams. In practice, systems should consider potential side effects if slight inequality is detrimental to its welfare in the long-run, despite short-term stability. Furthermore, while we explore teams of AI agents, teams may also consist of humans or hybrid populations of both AI and humans. Exploring alternative team reward functions may lead to interesting results and future research, particularly in the context of hybrid teams.

We see many interesting open questions with multiagent teams. For example, constructing richer reward structures by individuals optimizing for various types of goals [Radke *et al.*, 2022]. Regarding levels of cooperation, exploring teams of unequal size and conditions under which low-level cooperation (i.e., nepotism or bribery) undermines global progress may be of interest. Our model is constructed to easily allow the adaptation of additional infrastructure among the agents. Thus, longer term questions include analyzing how features such as communication, negotiation, trust, and sanctions impact our model and introduce new challenges. We hope that this work will reinvigorate the study of multiagent teams with RL agents to further understand how findings in organizational psychology and AI can complement each other.

Acknowledgements

This research is funded by the Natural Sciences and Engineering Research Council of Canada (NSERC), an Ontario Graduate Scholarship, a Cheriton Scholarship, and the University of Waterloo President's Graduate Scholarship. We also thank Jesse Hoey, Robin Cohen, Alexi Orchard, Sriram Subramanian, Valerie Platsko, and Kanav Mehra for their feedback and useful discussion on earlier drafts of this work.

References

[Anastassacos *et al.*, 2020] N. Anastassacos, S. Hailes, and M. Musolesi. Partner selection for the emergence of cooperation in multi-agent systems using reinforcement learning. In *AAAI*, 2020.

[Anastassacos *et al.*, 2021] N. Anastassacos, J. Garcia, S. Hailes, and M. Musolesi. Cooperation and reputation dynamics with reinforcement learning. In *AAMAS*, 2021.

[Andrejczuk *et al.*, 2016] E. Andrejczuk, J. A. Rodriguez-Aguilar, and C. Sierra. A concise review on multiagent teams: contributions and research opportunities. *Multi-Agent Systems and Agreement Technologies*, 2016.

[Baker *et al.*, 2019] B. Baker, I. Kanitscheider, T. Markov, Y. Wu, G. Powell, B. McGrew, and I. Mordatch. Emergent tool use from multi-agent autocurricula. In *ICLR*, 2019.

[Baker, 2020] B. Baker. Emergent reciprocity and team formation from randomized uncertain social preferences. *NeurIPS*, 2020.

[Dafoe *et al.*, 2021] A. Dafoe, Y. Bachrach, G. Hadfield, E. Horvitz, K. Larson, and T. Graepel. Cooperative AI: machines must learn to find common ground. *Nature*, 2021.

[Deka and Sycara, 2021] A. Deka and K. Sycara. Natural emergence of heterogeneous strategies in artificially intelligent competitive teams. In *Int. Conf. on Swarm Intelligence*, 2021.

[Grosz and Kraus, 1996] B. Grosz and S. Kraus. Collaborative plans for complex group action. *Artif. Intell.*, 1996.

[Grosz and Sidner, 1988] B. J. Grosz and C. L. Sidner. Plans for discourse. Technical report, BBN Labs, 1988.

[Hughes *et al.*, 2018] E. Hughes, J. Z. Leibo, M. Phillips, K. Tuyls, E. A. Duéñez-Guzmán, A. Castañeda, I. Dunning, T. Zhu, K. R. McKee, R. Koster, H. Roff, and T. Graepel. Inequity aversion improves cooperation in intertemporal social dilemmas. In *NeurIPS*, 2018.

[Jaderberg *et al.*, 2019] M. Jaderberg, W. Czarnecki, I. Dunning, L. Marris, G. Lever, A. Castañeda, C. Beattie, N. C. Rabinowitz, A. S. Morcos, A. Ruderman, N. Sonnerat, T. Green, L. Deason, J. Z. Leibo, D. Silver, D. Hassabis, K. Kavukcuoglu, and T. Graepel. Human-level performance in 3d multiplayer games with population-based reinforcement learning. *Science*, 364:859 – 865, 2019.

[Jaques *et al.*, 2019] N. Jaques, A. Lazaridou, E. Hughes, Ç. Gülçehre, P. A. Ortega, D. Strouse, J. Z. Leibo, and N. D. Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *ICML*, 2019.

[Kitano *et al.*, 1997] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa. Robocup: The robot world cup initiative. In *AGENTS '97*, 1997.

[Kraemer and Banerjee, 2016] L. Kraemer and B. Banerjee. Multi-agent reinforcement learning as a rehearsal for decentralized planning. *Neurocomputing*, 190:82–94, 2016.

[Leibo *et al.*, 2017] J. Leibo, V. Zambaldi, M. Lanctot, J. Marecki, and T. Graepel. Multiagent reinforcement learning in sequential social dilemmas. *AAMAS*, 2017.

[Macke *et al.*, 2021] W. Macke, R. Mirsky, and P. Stone. Expected value of communication for planning in ad hoc teamwork. In *AAAI-21*, 2021.

[McKee *et al.*, 2020] K.R. McKee, I. Gemp, B. McWilliams, E.A. Duéñez-Guzmán, E. Hughes, and J. Z. Leibo. Social diversity and social preferences in mixed-motive reinforcement learning. *AAMAS*, 2020.

[Mnih *et al.*, 2015] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. A. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

[Muthukrishna, 2017] M. Muthukrishna. Corruption, cooperation, and the evolution of prosocial institutions. *Evonomics*, 2017.

[Nisioti and Moulin-Frier, 2020] E. Nisioti and C. Moulin-Frier. Grounding artificial intelligence in the origins of human behavior. *ArXiv*, abs/2012.08564, 2020.

[Pollack, 1990] M. E. Pollack. Plans as complex mental attitudes. In *Intentions in Communication*, 1990.

[Radke *et al.*, 2022] D. Radke, K. Larson, and T. Brecht. The importance of credo in multiagent learning. *Adaptive and Learning Agents Workshop at AAMAS*, 2022.

[Rapoport, 1974] A. Rapoport. Prisoner's dilemma — recollections and observations. In *Game Theory as a Theory of a Conflict Resolution*, pages 17–34. Springer, 1974.

[Ryu *et al.*, 2021] H. Ryu, H. Shin, and J. Park. Cooperative and competitive biases for multi-agent reinforcement learning. In *AAMAS*, 2021.

[Schnell *et al.*, 2021] E. Schnell, R. Schimmelpfennig, and M. Muthukrishna. The size of the stag determines the level of cooperation. *bioRxiv*, 2021.

[Schulman *et al.*, 2017] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017.

[Tambe, 1997] M. Tambe. Towards flexible teamwork. *J. Artif. Intell. Res.*, 7:83–124, 1997.

[Vinitsky *et al.*, 2019] E. Vinitsky, N. Jaques, J. Leibo, A. Castenada, and E. Hughes. An open source implementation of sequential social dilemma games. https://github.com/eugenevinitsky/sequential_social_dilemma_games/issues/182, 2019.

[Wang *et al.*, 2019] J. X. Wang, E. Hughes, C. Fernando, W. M. Czarnecki, E. A. Duéñez-Guzmán, and J. Z. Leibo. Evolving intrinsic motivations for altruistic behavior. In *AAMAS-19*, pages 683–692, 2019.

[Zaccaro *et al.*, 2020] S. J. Zaccaro, S. Dubrow, E. M. Torres, and L. Campbell. Multiteam systems: An integrated review and comparison of different forms. *Annual Review of Organizational Psychology and Organizational Behavior*, 2020.

A Equilibrium Analysis of IPD

A.1 Expected Utilities

The expected utility of choosing to cooperate (C) or defect (D) can be derived using on Table 1, Table 2, ν, and the strategy profile of j, σ_T. As a first step towards addressing this question, we investigate the impact of teams on the *stage game* of the IPD. To provide a clear comparison with the standard IPD, we take an ex-ante approach, where agents are aware of their imminent interaction and the existence of other teams but not the actual team membership of their counterpart.

Assume a pair of agents, i, j, have been selected to interact at some iteration of the IPD and agent i knows j will be a teammate with probability ν and a non-teammate with probability $(1 - \nu)$. Also assume agent j is playing some strategy summarized by the probability that agent j selects action C conditioned on if they are a teammate or non-teammate. Let $\sigma_{T_i} = (\sigma_{ji}, 1 - \sigma_{ji})$ where σ_{ji} is the probability for action C represent when $j \in T_i$ and $\sigma_{T_j} = (\sigma_{jj}, 1 - \sigma_{jj})$ when $j \in T_j$, any other team. First we show the derivation for i's expected utility for choosing C subject to j's strategy:

$$\begin{aligned}
\mathbb{E}(C, \sigma_T) &= \nu \left[\sigma_{ji}(b - c) + (1 - \sigma_{ji})\frac{b - c}{2} \right] + \\
&\quad (1 - \nu) \left[\sigma_{jj}(b - c) + (1 - \sigma_{jj})(-c) \right] \\
&= \nu \left[\frac{2\sigma_{ji}(b - c)}{2} + \frac{b - c}{2} - \frac{\sigma_{ji}(b - c)}{2} \right] + \\
&\quad (1 - \nu) \left[\sigma_{jj}b - \sigma_{jj}c - c + \sigma_{jj}c \right] \\
&= \nu \left[\frac{\sigma_{ji}b - \sigma_{ji}c}{2} + \frac{b - c}{2} \right] + (1 - \nu) \left[\sigma_{jj}b - c \right] \\
&= \nu \left[\frac{(b - c)(\sigma_{ji} + 1)}{2} \right] + (1 - \nu) \left[\sigma_{jj}b - c \right] \\
&= \frac{\nu(b - c)(\sigma_{ji} + 1)}{2} + (1 - \nu)(\sigma_{jj}b - c)
\end{aligned}$$

Now we show the derivation for i's expected utility for choosing D subject to j's strategy:

$$\begin{aligned}
\mathbb{E}(D, \sigma_T) &= \nu \left[\sigma_{ji}\frac{(b - c)}{2} \right] + (1 - \nu) \left[\sigma_{jj}b \right] \\
&= \frac{\nu\sigma_{ji}(b - c)}{2} + (1 - \nu)\sigma_{jj}b
\end{aligned}$$

The terms for playing defection with a counterpart who also defects is zero, therefore omitted above.

A.2 Action Incentives

The moment when agents have the incentive to cooperate given the expected utilities of cooperation and defection is presented in Constraint 6.

$$\mathbb{E}(C, \sigma_T) \geq \mathbb{E}(D, \sigma_T) \tag{6}$$

We calculate this scenario by substituting $\mathbb{E}(C, \sigma_T)$ and $\mathbb{E}(D, \sigma_T)$ from above.

$$\begin{aligned}
&\frac{\nu(b - c)(\sigma_{ji} + 1)}{2} + (1 - \nu)(\sigma_{jj}b - c) \geq \\
&\qquad \frac{\nu\sigma_{ji}(b - c)}{2} + (1 - \nu)\sigma_{jj}b
\end{aligned}$$

$$\frac{\nu(b - c)(\sigma_{ji} + 1)}{2} - c + \nu c \geq \frac{\nu\sigma_{ji}(b - c)}{2}$$

$$\frac{\nu(b - c)}{2} - c + \nu c \geq 0$$

$$\nu(b - c) + 2\nu c \geq 2c$$

$$\nu b + \nu c \geq 2c$$

$$\nu \geq \frac{2c}{b + c} \tag{7}$$

The above derivation simplifies Constraint 6 to calculate the point at which agents have incentives to cooperate in our environment. The incentives for each team structure in our IPD environment can be visualized in the bottom graph of Figure 1.

In the regular IPD without teams, agents have no common interest making (D, D) the unique Nash Equilibrium and (C, C), (C, D), and (D, C) the three Pareto Efficient strategies. Since teammates share rewards, the degree of common interest is ultimately determined by the amount they interact with their team, ν. Therefore if Equation 7 is satisfied, the game-theoretical properties of the IPD transform so that (C, C) is the unique Nash Equilibrium and Pareto Efficient strategy.

B Environment Setups

B.1 Iterated Prisoner's Dilemma (IPD)

IPD Payoff Scheme

Table 1 shows the Prisoner's Dilemma matrix game used for our IPD experiments and equilibrium analysis. This parameterization of the IPD considers the cost (c) and benefit (b) of cooperation where mutual defection yields a reward of 0. Agents interact with their counterpart and receive the reward from this matrix corresponding with their action and the action of their counterpart. The team reward TR_i is calculated after all agents in the population interact and is used by agents when learning.

Table 2 shows how the payoffs of the Prisoner's Dilemma change when two teammates are chosen to interact and fully share rewards. In this scenario, we observe the unique Nash Equilibrium shift from mutual defection to mutual cooperation. Agents receive the explicit payoff from Table 1 from the interaction, though share their rewards with their teammates. Thus, Table 2 represents their payoff after TR_i is calculated when interacting with a teammate.

Matching Algorithm

At each instance of the IPD, agent i is given a counterpart, j to play the game in Table 1. Agent pairings are assigned using a uniform random distribution from each team, meaning the probability of a counterpart being chosen from T_i is the same as any other team T_j. For example, if $|\mathcal{T}| = 5$ an agent has a 20% chance of being paired with a teammate. Each episode, we construct N pairings by matching each agent $i \in N$ to a partner $j \in N \setminus \{i\}$, with the constraint that the probability of j being on any team is equally likely. Each agent observes the team their counterpart belongs to through a numerical signal $s_i \in S$, but not their actual individual identity.

Team Size and the Number of Interactions

In each episode, agents are given a counterpart and could also be chosen to be the counterpart of another agent for a total of N pairings per-episode. Since agents learn only through their own direct interactions, we must ensure that the particular matching process we use does not bias the results. In particular, we need to be confidant that the underlying team structure in which agents are embedded in no way influences the agent training through under- or over-sampling or providing disproportionate opportunities to be matched and play an iteration of the IPD.

Proposition 1. *If $|T_i| = |T_j|$ $\forall i, j \in N$ and agents are randomly paired from any team with uniform probability, each agent will have the same expected number of IPD interactions for any value of $|T|$ or N.*

Proof. Let a population of N agents be split up into $|\mathcal{T}|$ teams of size n, so that $N = |\mathcal{T}|n$. Since agents are paired with an agent from any team with equal probability, $Pr(IN) = 1 - \frac{1}{|\mathcal{T}|(n-1)}$ and $Pr(OUT) = 1 - \frac{1}{|\mathcal{T}|n}$ represents the probability of **not** being matched with a teammate or non-teammate respectively. These are different since an agent is unable to be paired with themselves, leaving $n - 1$ agents to possibly be paired with from their own team. The probability of agent i not being chosen as the matching agent is defined as:

$$Pr(\bar{i})_{|\mathcal{T}|n} = Pr(IN)^{n-1} + Pr(OUT)^{n(|\mathcal{T}|-1)}.$$

Suppose m agents are added to each team so that $N' = |\mathcal{T}|n + |\mathcal{T}|m$ and $n := n + m$. In this new setting, the probability of i not being chosen in a population of $|\mathcal{T}|(n + m)$ agents becomes:

$$\begin{aligned} &Pr(\bar{i})_{|\mathcal{T}|(n+m)} = \\ &\quad Pr(IN)^{(n-1)+m} + Pr(OUT)^{n(|\mathcal{T}|-1)+(|\mathcal{T}|m-m)}. \end{aligned}$$

We can derive that $Pr(\bar{i})_{|\mathcal{T}|(n+m)} - Pr(\bar{i})_{|\mathcal{T}|n} = (|\mathcal{T}|m - m) + m$, which simplifies to $|\mathcal{T}|m$. Note that $N' - N = |\mathcal{T}|m$ also. While the probability of not being chosen increases by $|\mathcal{T}|m$, the total interactions in each episode also increases by $|\mathcal{T}|m$. Thus, agents have the same number of expected interactions. □

	Cooperate	Defect
Cooperate	$b - c, b - c$	$-c, b$
Defect	$b, -c$	0, 0

Table 1: An example of the Prisoner's Dilemma with the costs (c) and benefits (b) of cooperating ($b > c > 0$).

	Cooperate	Defect
Cooperate	$b - c, b - c$	$\frac{b-c}{2}, \frac{b-c}{2}$
Defect	$\frac{b-c}{2}, \frac{b-c}{2}$	0, 0

Table 2: An example of the Prisonner's Dilemma when agents are teammates. (C, C) is the unique Nash Equilibrium.

Proposition 1 says if each team in $\mathcal{T}$ is the same size and counterparts are randomly chosen from teams with uniform probability, each agent will have the same expected number of interactions to train their policies. Intuitively, while the probability of being selected as a counterpart decreases as $|T|$ or N increases, there are more opportunities to be chosen. Note that this result could also be obtained with teams of different sizes so long as the pairing probability is distributed appropriately. This helps ensure our empirical results are attributed to the dynamics of multiagent teams instead of inherent bias favoring agents with more experience. We denote the expected number of interactions as I for our analysis in following analyzes.

IPD Reinforcement Learning Algorithm

For each round of the IPD, agent i is paired with another agent j chosen randomly from the population. The two agents play one iteration of the game shown in Table 1. Each agent observes the team their counterpart belongs to instead their actual identity. In particular, for a pair of agents, i and j, their states s_i and s_j are defined as $s_i = T_j$ and $s_j = T_i$. Given each s, the two agents simultaneously choose an action a which is whether to cooperate or defect. They do not observe the action their counterpart takes, but instead receive rewards TR_i and TR_j based on their own interaction and those of their teammates. Each i (and also j with their information) stores the tuple $\langle s_i, a_i, TR_i \rangle$ in their replay buffer to train their policy after each episode using Deep Q-Learning [Mnih *et al.*, 2015] by sampling a random batch of 32 interactions. Each agent's internal neural network consists of an input layer of size $|\mathcal{T}|$, two hidden layers of 200 nodes each with hypobolic tangent activation functions, and a two-action output layer with a linear activation function. Our agents use a learning rate of 1×10^{-4} and discount factor of 0.99 with ϵ-exploration.

B.2 Cleanup Markov Game

Cleanup Reinforcement Learning Algorithm

The typical environmental setup for Cleanup implemented in past work uses five agents. However, this would only create 1/5 and 5/1 when considering multiple possible teams of equal size. Therefore, we instantiate six agents to create more

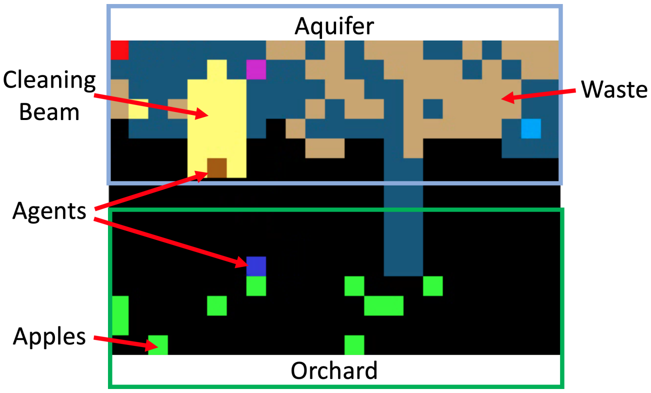

Figure 6: Cleanup environment with 6 agents and no teams.

team structures. We deploy the default Proximal Policy Optimization (PPO) [Schulman *et al.*, 2017] learning algorithm architecture in the Cleanup repository [Vinitsky *et al.*, 2019]. PPO is a policy gradient algorithm which constrains the space of policy updates to avoid large policy updates for smoother training and has been previously shown as a good algorithm for agents to learn in Cleanup. Agents are only able to observe the a 15×15 box centered at their location and update their policies using the environment's default batch size of at least 16,000 timesteps and a maximum of 100,000 timesteps. Each episode executes for 1,000 timesteps and we run experiments for 1.6×10^8 timesteps. The learning rate decreases linearly from 1.2×10^{-3} to 1.2×10^{-5} over the first 2×10^7 timesteps and remains static at 1.2×10^{-5} afterwards.

Default Environment Parameters

An example of Cleanup is shown in Figure 6. The social dilemma dynamics of Cleanup rely on the existence of waste and apples. The functions which govern the creation of these features can be easily modified; however, we evaluate our model of teams primarily using the default parameters of the environment which include a waste regeneration rate, waste regeneration threshold, and apple generation rate. Once less than 40% of the river (aquifer) grid-cells contain waste, waste regenerates at each clean cell with a probability of 50% at each timestep. Below this waste regeneration, apples spawn at each location in the orchard with a linear probability ranging from 0% when waste is 40% of the river up to 5% when waste makes up 0% of the river.

We explored a 20% waste threshold and 2.5%, 10%, and 20% maximum apple regeneration probability and found no significant alterations in the results. The best joint strategy among the six agents remained four pickers and two cleaners and both 2/3 and 3/2 team structures performed best.