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.

$$\begin{aligned}&\frac{\nu(b-c)(\sigma_{ji}+1)}{2} + (1-\nu)(\sigma_{jj}b - c) \geq \\ &\qquad \frac{\nu\sigma_{ji}(b-c)}{2} + (1-\nu)\sigma_{jj}b \\ &\frac{\nu(b-c)(\sigma_{ji}+1)}{2} - c + \nu c \geq \frac{\nu\sigma_{ji}(b-c)}{2} \\ &\frac{\nu(b-c)}{2} - c + \nu c \geq 0 \\ &\nu(b-c) + 2\nu c \geq 2c \\ &\nu b + \nu c \geq 2c \\ &\nu \geq \frac{2c}{b+c}\end{aligned} \tag{7}$$

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:

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

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

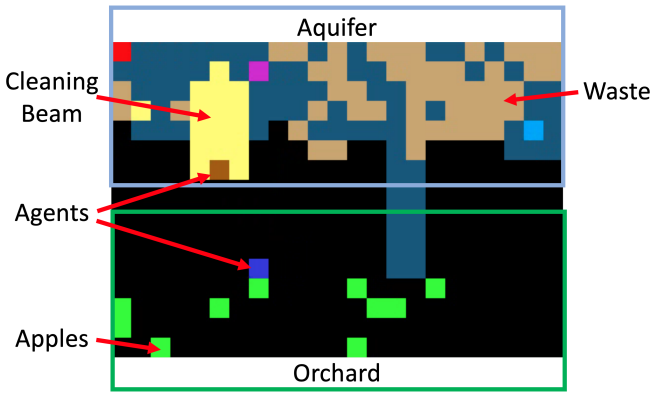

Figure 6: Cleanup environment with 6 agents and no teams.

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.