A Two State Environment Stochastic Game

The two state environment showed in Figure 1 of the main text induces a stochastic game whenever $n > 1$. This stochastic game has multiple possible Nash Equilibria on which teammates must coordinate on.

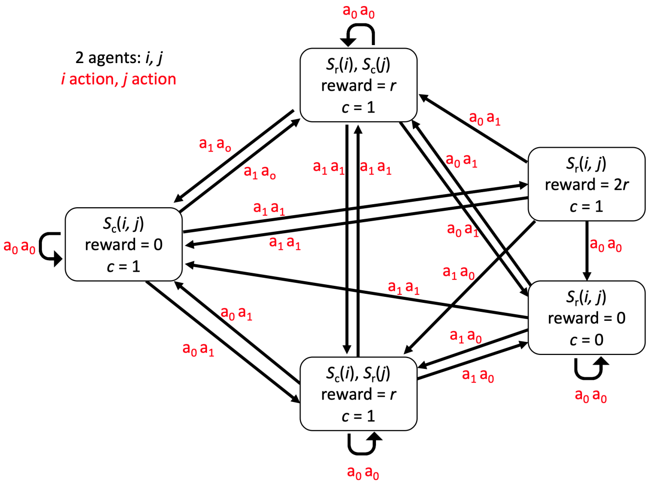

Figure 5: Stochastic game diagram induced from our two state environment. Game states are labeled so that $s_c(i, j)$ represents both agents (i and j) being in physical state s_c.

To show the emergence of multiple Nash Equilibria, Figure 5 shows the stochastic game induced in this environment with $n = 2$ agents (i and j). The possible scenarios of the game are labeled so that $s_c(i, j)$ represents both agents being in physical state s_c. The reward represents the total reward yielded from the environment in that specific game state (i.e., reward = $2r$ represents both i and j received r). For any agent to obtain the reward of r at s_r, some agent in the environment must visit s_c to change the boolean signal to $c = 1$. With just two agents, there are multiple joint policies that yield optimal reward on which agents must learn to coordinate on. Specifically, the two agents could 1) both move between s_c and s_r together, 2) transition from s_c to s_r (vice versa) with a_1 to never be in the same state, or 3) each agent always stays in either s_c or s_r using a_0.

B Reward Redistribution

The first theoretical finding in the manuscript is how larger teams increase the probability of agent i receiving a positive reward signal for executing a reward-causing state-action pair.

Theorem 1. *There exists an environment where increasing team size increases the probability of an agent receiving a reward for executing any reward-causing state-action pair that is greater than if they were not in a team.*

Proof. Due to agent's policies being initialized uniformly at random at the beginning of learning, we assume full coverage of the state space by all independent agents in the limit. Subsequently, suppose agent i is executing a reward-causing state-action pair that yields the minimum reward in the environment (Assumption 3). Any teammate moving to a reward state increases the reward i receives for executing that reward-causing state-action pair through $TR_{i[n]}$ compared to when i acts individually. The probability of any teammate j being in a reward state s_r is equal to the product of agents **not** being in s_r subtracted from 1. Let $0 < \zeta < 1$ be the probability that a teammate j is **not** located in a reward state, s_r, where $\zeta_j = \zeta_k$ for each $j, k \in T_i$ (i.e., ζ is assumed to be equal for all teammates). For a team of size n, the probability of any teammate being in a reward state at any timestep is $P(s_j = s_r) = 1 - \zeta^{(n-1)}$. Since $0 < \zeta < 1$, the second term $\zeta^{(n-1)} \to 0$ as $n \to \infty$. As a result, the overall probability of any teammate being in a reward state $P(s_j = s_r)$ converges to 1 as team size increases.

□

Theorem 1 shows how larger teams make reward-causing state-action pairs attractive for agents that learn from experience to maximize their future reward.

C Decreased Information

Our second theoretical contribution examines the impact of team size on the amount of information agents gain through their policies.

Proposition 1. *Let $\boldsymbol{\pi}_{T_i}$ be the joint fixed behavior policy of agents in T_i that generates a joint trajectory of experiences $\boldsymbol{\tau}_{T_i}$ (a collection of individually observed trajectories by each $i \in T_i$), where the randomness of state-action pairs in $\boldsymbol{\tau}_{T_i}$ depends on all N agents (by the definition of a stochastic game). Let $TR^t_{i[n]}$ be a random variable denoting the team reward at any timestep t (where the randomness of the deterministic reward follows from the randomness of the joint state-action pairs of individual agents in T_i at time t, depending on all N agents, $\boldsymbol{\tau}^t_{T_i}$). It follows that:*

$$\mathcal{I}(Z(\boldsymbol{\tau}_{T_i}); \boldsymbol{\tau}^t_{T_i} | \boldsymbol{\tau}^{-t}_{T_i}) = \mathcal{H}(TR^t_{i[n]} | \boldsymbol{\tau}^{1:t-1}_{T_i}).$$

Proof. The chain rule of mutual information gives us:

$$\begin{aligned}\mathcal{I}(Z(\boldsymbol{\tau}_{T_i}); \boldsymbol{\tau}^t_{T_i} | \boldsymbol{\tau}^{-t}_{T_i}) &= \mathcal{I}(Z(\boldsymbol{\tau}_{T_i}); \boldsymbol{\tau}^t_{T_i}, \boldsymbol{\tau}^{-t}_{T_i}) \\ &\quad - \mathcal{I}(Z(\boldsymbol{\tau}_{T_i}); \boldsymbol{\tau}^{-t}_{T_i}) \\ &= \mathcal{I}(Z(\boldsymbol{\tau}_{T_i}); \boldsymbol{\tau}_{T_i}) - \mathcal{I}(Z(\boldsymbol{\tau}_{T_i}); \boldsymbol{\tau}^{-t}_{T_i}).\end{aligned}$$

By the definition of mutual information, we can expand in terms of entropy:

$$\begin{aligned}&= \mathcal{H}(Z(\boldsymbol{\tau}_{T_i})) - \mathcal{H}(Z(\boldsymbol{\tau}_{T_i}) | \boldsymbol{\tau}_{T_i}) \\ &\quad - \mathcal{H}(Z(\boldsymbol{\tau}_{T_i})) + \mathcal{H}(Z(\boldsymbol{\tau}_{T_i}) | \boldsymbol{\tau}^{-t}_{T_i})\end{aligned}$$

$$= \mathcal{H}(Z(\boldsymbol{\tau}_{T_i}) | \boldsymbol{\tau}^{-t}_{T_i}) - \mathcal{H}(Z(\boldsymbol{\tau}_{T_i}) | \boldsymbol{\tau}_{T_i}).$$

We know $Z(\boldsymbol{\tau}_{T_i})$ is a deterministic function of $\boldsymbol{\tau}_{T_i}$ due to the deterministic aggregation (mean reward) of n deterministic reward functions of all teammates. The deterministic individual reward functions are already dependent on all N agents; thus, we can drop the second term and simplify to:

$$= \mathcal{H}(Z(\boldsymbol{\tau}_{T_i})|\boldsymbol{\tau}^{-t}_{T_i}).$$

Since we know each agent in T_i is optimizing their discounted sum of future team rewards, we know $Z(\boldsymbol{\tau}_{T_i}) = \Sigma^{H}_{t=1}\gamma^{t-1}TR^t_{i[n]}$, and can substitute for $Z(\boldsymbol{\tau}_{T_i})$:

$$\begin{aligned} &= \mathcal{H}(TR^t_{i[n]}|\boldsymbol{\tau}^{-t}_{T_i}) \\ &= \mathcal{H}(TR^t_{i[n]}|\boldsymbol{\tau}^{1:t-1}_{T_i}, \boldsymbol{\tau}^{t+1:H}). \end{aligned}$$

Finally, since $TR^t_{i[n]}$ is unable to be impacted by the future (i.e., anything greater than t), we can remove the correlation with $\boldsymbol{\tau}^{t+1:H}$:

$$= \mathcal{H}(TR^t_{i[n]}|\boldsymbol{\tau}^{1:t-1}_{T_i}).$$

□

Proposition 1 equates the information at any time of a stochastic game to the entropy of the team reward signal. The left-hand side quantifies the information between a single joint state-action pair for the team $\boldsymbol{\tau}^t_{T_i}$ and the team's joint policy return over the joint trajectory, $Z(\boldsymbol{\tau}_{T_i})$, conditioned on the joint trajectory without timestep t, $\boldsymbol{\tau}^{-t}_{T_i}$. Next, we show that the variance of the team reward function converges to zero as team size increases.

Lemma 1. *The team reward random variable $TR_{i[n]}$ for any state-action pair converges to the mean environmental reward (mean of any agent's individual reward function) as team size increases in the limit (i.e., $TR_{i[n]}(\mathbf{s}^t, \mathbf{a}^t, \mathbf{s}^{t+1}) \to \overline{R_i}$ as $n \to \infty$).*

Proof. Since the team reward is an aggregation of n individual and uniformly random rewards samples from identical reward functions, $TR_{i[n]} \approx \mathcal{N}\left(\overline{R_i}, \frac{\sigma^2_{R_i}}{\sqrt{n}}\right)$ by the Central Limit Theorem, where var$[R_i] = \sigma^2_{R_i}$. The variance var$\big[TR_{i[n]}\big] = \frac{\sigma^2_{R_i}}{\sqrt{n}}$, with a derivative of var$\big[TR_{i[n]}\big]' = -\frac{\sigma_{R_i}}{\sqrt{n^3}}$. Since $\sigma_{R_i} = \sqrt{\sigma^2_{R_i}}$ is the standard deviation of R_i (i.e., distance from $\overline{R_i}$), we know $\sigma_{R_i} > 0$. Furthermore, σ_{R_i} is a constant and $n \geq 1$; thus, var$\big[TR_{i[n]}\big]'$ is negative and converges to zero as n increases in the denominator. □

Finally, we use Proposition 1 and Lemma 1 to show that the information in a stochastic game converges to zero as a funciton of team size.

Theorem 2. *The information in a stochastic game at time t, $\mathcal{I}(Z(\boldsymbol{\tau}_i); \boldsymbol{\tau}^t_i|\boldsymbol{\tau}^{-t}_i)$, converges to 0 as the size of a team, n, increases in the limit.*

Proof. By Proposition 1, we can use the entropy of $TR^t_{i[n]}$ to determine the information of $Z(\boldsymbol{\tau}_i)$ at time t of a trajectory. By the Central Limit Theorem and Lemma 1, let $TR^t_{i[n]}$ be a Gaussian distributed random variable so that $TR^t_{i[n]} \approx \mathcal{N}\left(\overline{R_i}, \frac{\sigma^2_{R_i}}{\sqrt{n}}\right)$. For readability, let the variance $\sigma^2 = \frac{\sigma^2_{R_i}}{\sqrt{n}}$. We rewrite the entropy of $TR_{i[n]}$ at time t given the trajectory up to t, $\mathcal{H}(TR^t_{i[n]}|\boldsymbol{\tau}^i_{1:t-1})$, in terms of the function's variance:

$$\begin{aligned} \mathcal{H}(TR^t_{i[n]}|\boldsymbol{\tau}^i_{1:t-1}) &= -\int_{TR_{i[n]}} p(TR_{i[n]}) \log p(TR_{i[n]}) \\ &= -\mathbb{E}\left[\log \mathcal{N}(\overline{R_i}, \sigma^2)\right] \\ &= -\mathbb{E}\left[\log\left[\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(\frac{R_i - \overline{R_i}}{\sigma^2})^2}\right]\right] \\ &= \frac{1}{2}\log\bigl(2\pi\sigma^2\bigr) + \frac{1}{2\sigma^2}\mathbb{E}\left[(R_i - \overline{R_i})^2\right] \\ &= \frac{1}{2}\log\bigl(2\pi\sigma^2\bigr) + \frac{1}{2} \end{aligned}$$

Since π is a constant, the variance $\sigma^2 = \frac{\sigma^2_{R_i}}{\sqrt{n}}$ regulates the entropy of $TR^t_{i[n]}$. By Lemma 1, we know $\lim\limits_{n\to\infty} \frac{\sigma^2_{R_i}}{\sqrt{n}} \to 0$. Thus, the entropy and information carried by the actions of π_i in a stochastic game at time t converges to zero as team size increases.

□

Theorem 2 states that agents will be unable to perform proper credit assignment and learn good policies as their team's size increases in the limit. This result is significant since it characterizes how fully cooperative systems can perform worse than a population of multiple smaller teams.

D Information with Teams

A fixed behavior policy π_i induces a stationary visitation distribution for agent i over states and state-action pairs, denoted as $d^{\pi_i}(s)$ and $d^{\pi_i}(s, a)$ respectively. Since we are concerned with the progression of how agents learn, our theory assumes agents are initialized with random policies that cover the state space uniformly, consistent with past work [Arumugam *et al.*, 2020].

The value of var$\Big[\mathcal{I}^{\pi_i}_{S_i,A_i}(Z_{T_i})\Big]$ depends on calculating the KL Divergence for state-action pairs from the distribution of states and actions for π_i, d^{π_i}. Given the distributional support $\mathcal{X}_{s_i,a_i}$ (the distribution of team rewards conditioned on specific state-action pairs that are not mapped to zero), this can be expanded to be:

$$\text{var}\left[\mathcal{I}^{\pi_i}_{S_i,A_i}(Z_{T_i})\right] = $$

$$\text{var}_{s_i,a_i\sim d^{\pi_i}}\left[\sum_{Z_{T_i}\in\mathcal{X}_{s_i,a_i}} p(Z_{T_i}|s_i, a_i) \log\left(\frac{p(Z_{T_i}|s_i, a_i)}{p(Z_{T_i}|s_i)}\right)\right]$$

Note that S_i and A_i are based on agent i's individual observations and policy, but Z_{T_i} is based on their shared team reward.

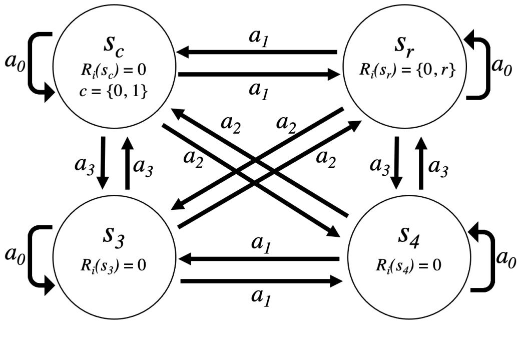

Figure 6: *4-States:* Environment diagram.

E 4-States

E.1 Environment

Figure 6 shows the 4-States environment in our evaluation, an augmentation of the simple 2-States environment shown in Figure 1 of the main text. We add two "no-op" states to the two state environment that return no reward and do not impact the binary signal (i.e., agents should avoid these states). States are labeled s_c , s_r, s_3 (no-op), and s_4 (no-op). A reward of +1 is given at s_r, conditioned on the visitation of s_c. Agents simultaneously choose among four actions: stay at their current state (s_0) or move to any of the other three states (s_1, s_2, or s_3). An action transitions agents to their intended next state with 90% probability and to another random state with 10% probability. We fix $|\mathcal{T}| = 1$ and increase n by a factor of 2 to remove the impact of other teams on the binary signal. Agents using Tabular Q-Learning [Sutton and Barto, 2018] with $\gamma = 0.9$ and ϵ-exploration ($\epsilon = 0.3$) for 50 trials of 1,000 episodes (100 steps each). The stochastic transitions and ϵ-exploration causes agents not to select the best action or move to their intended state about 33% of timesteps.

F Iterated Prisoner's Dilemma (IPD)

F.1 Environment

We follow a similar IPD configuration as recent work with teams [Radke *et al.*, 2022; Radke *et al.*, 2023] and assume that there is a cost (c) and a benefit (b) to cooperating where $b > c > 0$. Agents are randomly paired with another agent at each timestep, a *counterpart*, that may or may not be a teammate with some probability ν. Agents must choose to either cooperate with (C) or defect on (D) their counterpart. Agents only observe the team label (i.e., number) of their counterpart, and receive their team reward, $TR_{i[n]}$, after their own and teammates' interactions; therefore, the strategies of all agents on team T_i affects how agents learn to play any member of T_i. We fix the cost $c = 1$, benefit $b = 5$, and define $|\mathcal{T}| = 2$ with increasing sizes of each team where $n = 1$ (no teams), $n = 2$ (one teammate), and then multiples of 5 to study general trends with larger teams. We fix $\nu = 97\%$ (non-teammates are 16 times more likely than teammates) and 100% when $n = 1$ (agents do not play themselves). Each experiment lasts 1.0×10^6 episodes where $N = 30$ agents learn using Deep Q-Learning [Mnih *et al.*, 2015], repeated for 20 trials.

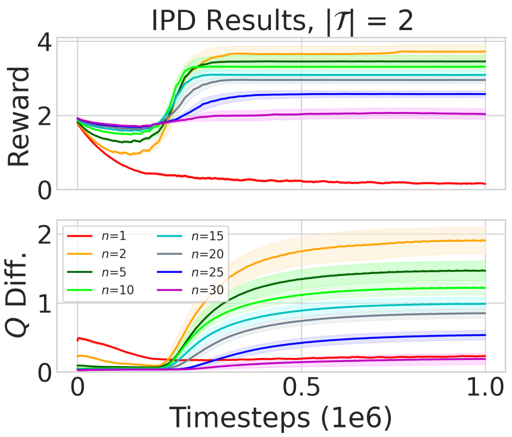

Figure 7: *IPD:* Mean population reward (top) and mean difference in agents' Q-values (bottom). Less difference between Q-values indicates agents have less preference for either action.

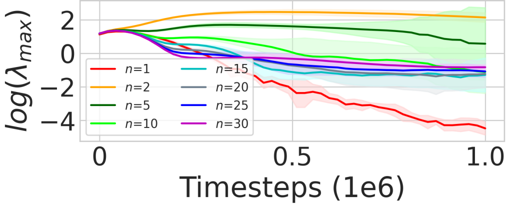

Figure 8: *IPD:* Mean maximum eigenvalue (λ_{max}) of agents' Hessian matrices (i.e., flatness of loss landscape).

F.2 Results

Figure 7 shows our results in the IPD environment for the mean population reward (top) and the difference in Q-values for C and D when paired with non-teammates (bottom). Both graphs share the same x-axis, representing the timesteps of our experiments.

Since mutual cooperation is the result with the highest mean population reward, we use reward as a proxy for learned cooperation (higher is better). When $n = 1$, agents converge to the Nash Equilibrium of mutual defection and obtain the lowest mean population reward. Consistent with past work [Radke *et al.*, 2022], our results show how having even one teammate allows agents learn cooperation and achieve high mean population reward despite only being paired with this teammate 3% of the time. However, team growth has diminishing returns. When $n = 30$, the mean population reward approaches the mean reward and agents behave randomly (i.e., $\overline{R_i} = 2$ when cost is 1, benefit is 5).

The bottom graph shows how initially providing agents with teammates ($n = 2$) increases the difference in Q-values significantly since agents learn the benefit of mutual cooperation. Agents adapt this behavior towards other teams and the population experiences high cooperation and high reward. Further increasing team size tends to reduce the difference in Q-values until agents have little Q-value difference when $n = 30$. These results are consistent with our theory and experiments in the other three domains.

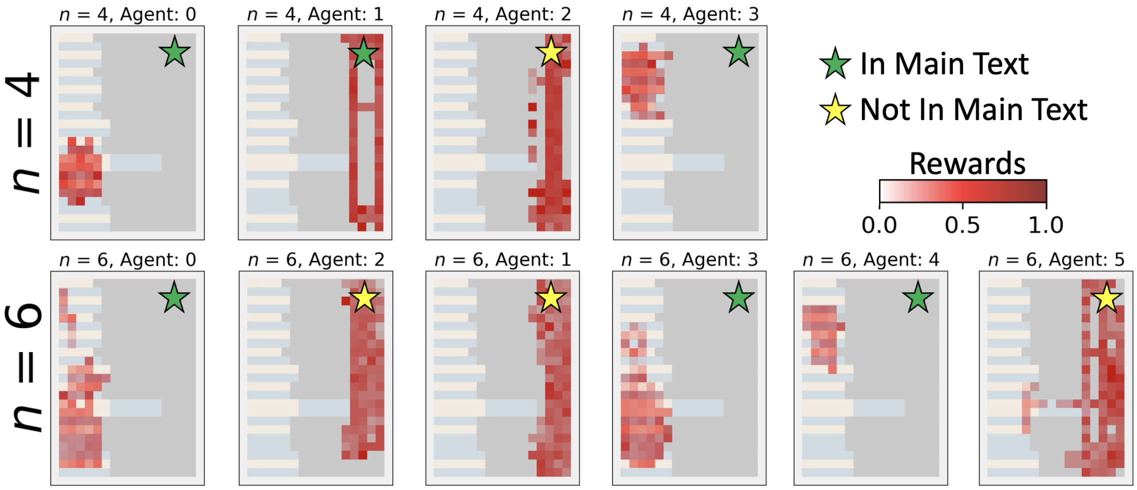

Figure 9: *Cleanup:* Team reward obtained at each pixel for different agents. The top row shows all agents' behaviors when $n = 4$ and the bottom row shows all agents when $n = 6$. Plots that appear in the main text are indicated with a green star and plots that are omitted from the main text due to space limitations are indicated with a yellow star.

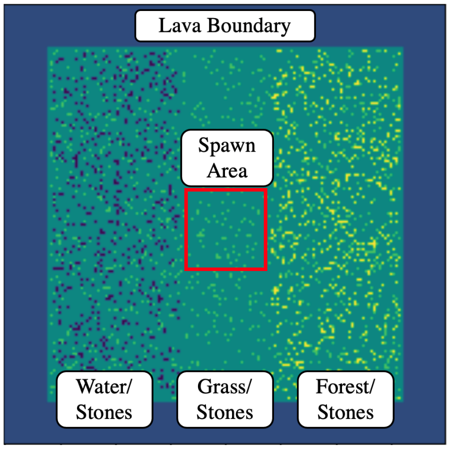

Figure 10: *NMMO:* Custom environment layout for our evaluation.

As a further analysis into how teams impact learning, Figure 8 shows the mean maximum eigenvalue (λ_{max}) of agents' policy network Hessian matrices as they learn ($\log_{10}$ scale). Lower values of λ_{max} represent a flatter optimization surface [Kaur *et al.*, 2022] that makes convergence through stochastic gradient descent more difficult. When $n = 1$, the high rate of 0 reward leads to a flat optimization landscape, but when $n = 2$ or 5, λ_{max} is the highest among all team structures we study. As teams grow larger, the loss landscape flattens and convergence to a minima becomes more difficult. This highlights that teams shape the loss landscape to assist convergence to a cooperative minima [Radke *et al.*, 2022], but large team structures flatten the landscape and reduce convergence.

G Cleanup Gridworld Game Extended

G.1 Environment

Cleanup [Vinitsky *et al.*, 2019] is a temporally and spatially extended Markov game representing a sequential social dilemma. Agents in Cleanup have eight actions: 9 movement (up, down, left, right, stay, turn left, and turn right), a cleaning beam, and a punishment beam. Agent observability is limited to an egocentric 15 × 15 pixel window, and agents receive +1 reward for collecting an apple in the orchard. Apple growth is conditional on the cleanliness of an adjacent river, and cleaning this river yields no direct environmental reward. Successful groups in Cleanup balance the temptation to free-ride and pick apples with the public obligation to clean the river. We set $|\mathcal{T}| = 1$ and increase team size to remove impacts of other teams on the conditional reward structure. We implement Proximal Policy Optimization (PPO) [Schulman *et al.*, 2017] agents for 10 trials of 1.6×10^8 episodes (1,000 timesteps each) using the Rllib RL library.

G.2 Spatial Results

Figure 9 shows the spatial behavior of all agents in one trial when $n = 4$ (top row) and $n = 6$ (bottom row). This figure is an expanded version of Figure 4 in the main text, where darker red corresponds with higher reward when the agent is located at that spatial location. When $n = 4$ (top row), the population divides labor so that Agents 0 and 3 agents specialize to clean the river and Agents 1 and 2 pick apples which achieves the highest reward in our evaluation, shown in Figure 2b of the main text. Additionally, Figure 9 (top row) shows how Agents 0 and 3 not only both converge to clean the river, but learn different cleaning *roles* and spatially divide the river territory for more efficiency. This spatial specialization is not typically observed with apple picking agents, but both apple picker agents still collect a significant amount of apples when $n = 4$ regardless.

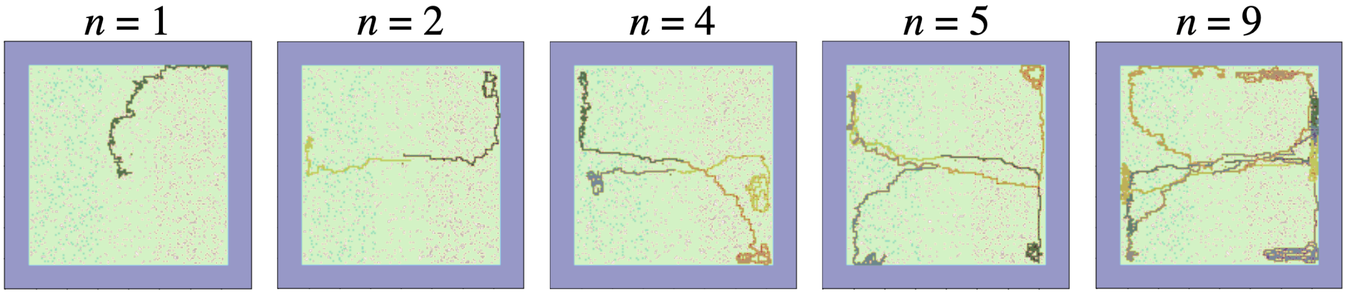

Figure 11: *NMMO:* Agent behavior in NMMO when $n = 1, 2, 4, 5, 9$. When $n = 1$, the agent spends time in the center region of the map which results in no reward. Agents learn about the value of food and water when they have teammates. When $n = 2$ or $n = 4$, agents spatially disperse and specialize in roles of collecting food or water while not interfering with each other. When $n = 5$ or $n = 9$, agents begin to converge to similar areas of the map and eventually interfere with each other's ability to collect food or water and venture back into the center area of the map.

Compare this with when $n = 6$ shown in Figure 9 (bottom row), where we consistently find 3 river cleaner and 3 apple picker policies emerge within agents in T_i. The behavior of the three river cleaners is less spatially specialized, resulting in Agents 0 and 3 cleaning the same location and learning the same role on their team (i.e., the role of cleaning the bottom half of the river). This duplication of roles leads to less team reward than smaller team structures despite having more agents, as shown in Figure 2b of the main text. Since we observe that two spatially specialized agents are able to effectively clean the river (seen when $n = 4$), the team would benefit from one of these redundant cleaners learning to instead pick apples and collect more reward. This gives further insight into why large cooperative systems achieve less reward than systems composed of multiple smaller teams in Cleanup, even when mixed incentives exist between teams as shown in [Radke *et al.*, 2022].

H Neural MMO Extended

H.1 Environment

Neural MMO (NMMO) [Suarez *et al.*, 2019] is a large, customizable, and partially observable multiagent environment that supports foraging and exploration. We configure a map with 1024 × 1024 pixels bounded by lava tiles to enclose the agents within the environment. As mentioned in the main text, agent observability is limited to an egocentric 15 × 15 pixel window and have movement and combat actions. Agents maintain a stash of consumable resources (food and water) that deplete some amount at each environmental timestep but are replenished through harvesting from the lakes and forests located throughout the environment. There is no standard NMMO configuration; therefore, we can customize the environment and reward function to satisfy the assumptions made in Section 4 (shown in Figure 10). Agents in a team share water and food resources amongst themselves and we remove agent death by starvation so that every episode is the same length. Agents always spawn in a random location at the center of the map. The environment has stones which agents must move around to reach water and forest tiles. Grass tiles offer nothing to the agents.

We set a resource depletion rate of -0.02 (minimum of 0.0), replenish amount of +0.1 (maximum amount of 1.0), and spatially separate the forests and lakes to encourage exploration. We reward agents for positive increases to their lowest resource: $\min(I)^t - \min(I)^{t-1}$ when $\min(I)^t > \min(I)^{t-1}$, where I is the inventory of food and water. Agents must learn to maintain both food and water to receive reward, creating multiple dynamically changing reward-causing state-action pairs, a more challenging scenario than the other environments. We implement PPO agents for 5 trials of 1.6×10^7 episodes (1,000 timesteps each) using Rllib.

H.2 Spatial Results

Figure 11 shows the movement of agents when $n = 1, 2, 4, 5, 9$. When $n = 1$ (Figure 11 left), the agent has difficulty learning about the value of both food and water, resulting in the agent staying in the center region of the map where there is only grass and stone (Figure 10). When the agent is given a teammate ($n = 2$; Figure 11 middle left), they converge to complimentary roles and explore different regions of the environment, collecting either food or water and sharing their resources. This behavior is also observed when $n = 4$ with two agents collecting food or water each. This joint policy generates one of the best team reward results in our evaluation showing the benefits of adding teammates. When $n = 5$ or $n = 9$, the agents still learn complimentary roles; however, they tend to interfere with each other and cover similar areas of the environment, consistent with our spatial results in Cleanup shown in Figure 4 of the main text or Figure 9 in Appendix G.2. The environment is significantly large so that this movement is avoidable; however, agents have difficulty learning how to spatially disperse as to maximize the reward from their joint policy. Furthermore, when $n = 9$, two agents return to the center grass/stone area later in an episode which contributes no positive reward for their team.

I Summary of Notation

Table 1 lists the notation used throughout the paper for easy access for the reader.

Notation	Description
i	An arbitrary agent.
j	A second arbitrary agent.
$\mathcal{N}$	Set of all agents.
N	Size of the set of all agents.
A	Joint action space.
S	Joint state space.
R	Joint reward space.
P	Transition function.
γ	Discount factor.
Σ	Policy space of all agents.
π_i	Policy of agent i.
t	Arbitrary timestep of an episode.
s_i	Single state for agent i.
a_i	single action for agent i.
$\mathbf{s}^t$	Joint state at time t.
$\mathbf{a}^t$	Joint action at time t.
$R_i^t(\mathbf{s}^t, \mathbf{a}^t, \mathbf{s}^{t+1})$	Agent i's individual reward at time t.
V_i	Value function of agent i.
$\mathcal{T}$	Set of all teams.
$\mathcal{T}_i$	Set of teams i belongs to.
$T_i \in \mathcal{T}_i$	Specific team that i belongs to.
n	The number of agents in a team.
$TR_{i[n]}$	Team reward for a team of size n.
H	Length of a full episode.
τ_i	Trajectory of state-action pairs generated by i.
π_{T_i}	Joint policy for n agents in team T_i.
τ_{T_i}	Joint trajectory for n agents in team T_i.
$\tau_{T_i}^t$	Joint state-action pair at time t for the agents in team T_i.
$\tau_{T_i}^{1:t-1}$	Joint trajectory for n agents in team T_i up to time $t-1$.
$\tau_{T_i}^{-t}$	Joint trajectory for n agents in team T_i without the joint state-action pair at time t.
$Z(\tau_{T_i})$	Random variable denoting the team random return obtained from a joint trajectory τ_{T_i}.
$\mathbf{s}_{T_i}$	Team T_i's joint state.
$\mathbf{s}_{T_i}$	Team T_i's joint action.
Z_{T_i}	Random variable denoting the team reward observed at $\mathbf{s}_{T_i}$ and taking joint action $\mathbf{a}_{T_i}$.
$\mathcal{I}_{s_i,a_i}^{\pi_i}$	Information gained by π_i in single-agent setting.
D_{KL}	Kullback-Leibler (KL) divergence.
$p(Z_i \vert s_i, a_i)$	Distribution of returns conditioned on particular state-action pair.
$p(Z_i \vert s_i)$	Distribution of returns conditioned only on state.
$\mathcal{I}(A_i; Z_i \vert S_i)$	Expected information π_i carries in single-agent setting.
$\mathcal{I}^{\pi_i}(A_i; Z_{T_i} \vert S_i)$	Expected information π_i carries in a multiagent team from a team reward.
$\mathcal{I}_{S_i,A_i}^{\pi_i}(Z_{T_i})$	Expected information gained by π_i over distribution of individual state-action pairs.
ϵ	Threshold on the expected information in an environment.
μ	Threshold on the variance of expected information across state-action pairs.
$\mathcal{H}(TR_{i[n]}^t)$	Entropy of team reward funciton.

Table 1: Notation summary throughout the paper for the reader.