COOPERATIVE AGENTS SPECIALIZE MORE BUT EXPLORE LESS

Anonymous authors
Paper under double-blind review

ABSTRACT

Cooperation is essential for multi-agent systems to succeed. Prior work has focused on making agents more cooperative, but we argue that cooperation is only the beginning of the problem. We study deep reinforcement learning and large language model agents in sequential social dilemmas, where agents face a tension between producing immediate rewards and maintaining shared resources that sustain future production. We find cooperative agents specialize more but explore less, which can eventually undermine group performance. Otherwise identical agents who work to maximize collective rewards are more cooperative and achieve higher performance by self-organizing into maintenance or production roles. Once agents specialize, maintenance agents explore significantly less than production agents, and their search narrows further over time. They then fail to replenish sufficient resources, leaving production agents with fewer resources to use. Group performance can consequently decline by up to 93% long after cooperation has been established. These results identify a potential failure mode that can emerge after successful cooperation and highlight the importance of preserving exploration as agents specialize in multi-agent systems.

1 INTRODUCTION

Cooperation is essential in multi-agent, multi-turn systems when AI agents need to orchestrate their actions over extended periods to solve increasingly complex tasks (Qian et al., 2024; Hong et al., 2024; Lu et al., 2024; Qi et al., 2026; Pal et al., 2026). Early research on reinforcement learning found that agents often fail to coordinate, especially when pursuing immediate individual rewards conflicts with the group's long-term goals (Leibo et al., 2017). Similar problems remain nowadays among agents built on foundation models (Yocum et al., 2023; Piatti et al., 2024). Much research effort has therefore focused on how to make multi-agent systems more cooperative (Hughes et al., 2018; Jaques et al., 2019; Yocum et al., 2023; Faulkner et al., 2026).

Here we argue that achieving cooperation is important, but it is only the beginning of the problem. Cooperation can improve group performance by allowing agents to divide the task and specialize. Yet repeated specialization may also narrow exploration, leading to decreased performance in the longer term. This idea builds on the observation that, when humans organize to achieve collective goals, they often learn to specialize. Agricultural communities cooperate by having some farmers cultivate crops while others maintain shared irrigation systems (Ostrom, 1990); industrialized factories divide pin-making into separate tasks where some workers cut the wire while others sharpen the pins (Smith, 1776); modern technology companies likewise have product developers specialize in building new features while support teams keep the infrastructure running (Beyer et al., 2016). Yet, specialization can also make people less exploratory. Expert accountants, for example, are less able than novices to adapt when new tax rules contradict what they've learned previously (Marchant et al., 1991; Dane, 2010). In simulated missions, fixed aircrews make more errors than newly formed ones because they are too familiar with each other (Gersick & Hackman, 1990; Barker et al., 1996). Long-standing R&D teams seek less information from the broader community even when their technical performance declines (Katz, 1982; Katz & Allen, 1982). These findings suggest that specialization can solve an immediate cooperation problem while creating a longer-term exploration problem.

To study this process in AI agents, we use a sequential social dilemma, a canonical framework for studying cooperation in spatially and temporally extended multi-agent systems (SSD; Leibo et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

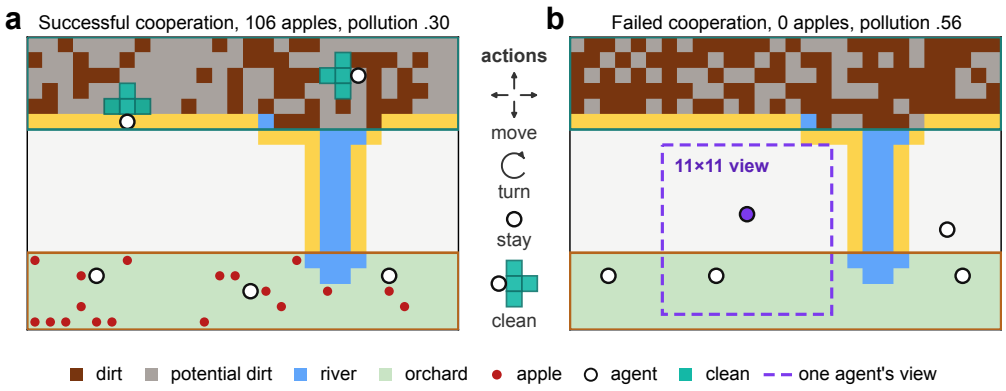


Figure 1: Game setting and example outcomes in CLEANUP. The environment consists of a polluted river at the top and an orchard where apples grow at the bottom. Each agent has an 11×11 partial view and can move, turn, stay, or clean. To clean, an agent fires a cleaning beam that removes pollution from the four cells directly in front of it but receives no immediate reward. To harvest, an agent moves onto an apple and receives a reward. Apples regrow only when the pollution rate remains below the .40 threshold. Agents therefore face a recurring dilemma between harvesting for immediate reward and cleaning to sustain future rewards for the team. (a) A successful team collects 106 apples while keeping pollution below .40. Two agents clean while three harvest, producing a clear division of labor. (b) A failed team collects 0 apples as pollution reaches .56. No agent cleans, pollution exceeds the threshold, and apples stop growing.

2017; Guo et al., 2026). Multiple agents interact over time and choose between production work, which provides immediate rewards, and maintenance work, which keeps the group productive in the future (Section 2). In the classic CLEANUP game, for example, agents can harvest apples for immediate rewards or clean waste so that apples continue to grow. Cleaning provides no immediate reward, so each agent has an incentive to free-ride and leave the work to others. However, if too few agents clean, waste builds up, and the apple supply eventually collapses. Specifically, we study the downstream consequences after agents successfully cooperate. Hence, we experimentally manipulate cooperation by varying reward incentives from individual to collective, and then examine how the resulting level of cooperation affects specialization and exploration.

Evaluating across deep reinforcement learning (deep RL) and large language model-based (LLM) agents, in three contexts including CLEANUP, collaborative hiring, and scientific collaboration, we observe three consistent patterns (Section 3). First, cooperative agents who maximize collective rewards become more efficient by spontaneously dividing into complementary roles; otherwise identical agents specialize in either maintenance or production. When agents are incentivized for collective rather than individual performance, deep RL agents increase efficiency more than tenfold while increasing specialization more than sevenfold. LLM agents show similar patterns, although some already perform as well under individual incentives as the best deep RL agents under collective incentives. Second, agents in specialized roles explore differently. Across all models and contexts, maintenance agents explore 28.7% less than production agents. Third, narrowed exploration is associated with decreased long-term performance. As maintenance agents focus on increasingly local tasks, they fail to maintain resources elsewhere, leaving fewer opportunities for production. Across all models and contexts, late-stage team output declines by 45.8% on average under collective incentives. At the model level, GPT-5-mini shows the largest decline (82.6%), while MiniMax M3 provides a boundary case, maintaining exploration and performance throughout our evaluation.

These findings suggest that achieving cooperation in multi-agent systems is not the end goal. As cooperation organizes agents into specialized roles, the same process that improves group performance can narrow individual exploration and create hidden costs for future performance. Preserving exploration as cooperation emerges will be critical for preventing initially successful teams from becoming increasingly rigid over time. This is particularly important in applications such as scientific research and automated hiring, where specialized agents must continue searching beyond familiar information, locations, and routines to discover new opportunities.

2 METHOD

2.1 FRAMEWORK

We study multi-agent cooperation using sequential social dilemmas (SSD; Leibo et al., 2017; Hughes et al., 2018), which capture conflicts between immediate individual rewards and longer-term collective rewards. A canonical scenario is CLEANUP, as illustrated in Figure 1. Agents repeatedly choose between production and maintenance: harvesting apples produces immediate reward, whereas cleaning waste provides no immediate reward but maintains the resource state necessary for future production. In CLEANUP, the resource state is determined by the pollution rate: as waste accumulates, apple regrowth slows down, and eventually stops. Because agents cannot harvest while cleaning, successful performance requires teams to coordinate between these competing activities.

Formally, we consider an N-player partially observable Markov game $\mathcal{M}$ defined over a finite state space $\mathcal{S}$. An observation function $O : \mathcal{S} \times \{1, \ldots, N\} \to \mathbb{R}^d$ specifies each agent's d-dimensional partial observation of the current state, with $\mathcal{O}_i = \{O(s, i) : s \in \mathcal{S}\}$ denoting the observation space of agent i. All agents share the same action space $\mathcal{A}$, with each agent choosing an action $a_i \in \mathcal{A}$. At timestep t, agents' actions are combined into the joint action $\mathbf{a}_t = (a_{1,t}, \ldots, a_{N,t})$, which produces a transition in the environment $T(s_{t+1} \mid \mathbf{a}_t, s_t)$ according to the state transition distribution T.

Each agent receives a reward signal $\tilde{r}_{i,t}$ that depends on both its own task reward $r_{i,t}$ and the rewards obtained by the group. Each agent independently learns a behavioral policy $\pi_i : \mathcal{O}_i \to \Delta(\mathcal{A})$, where $\pi_i(a_i \mid x_i)$ gives the probability of choosing action a_i given its local observation $x_i \in \mathcal{O}_i$. Each agent seeks to maximize its expected long-term reward, J_i, over horizon $\mathcal{T}$,

$$J_i \;=\; \mathbb{E}\left[\sum_{t=1}^{\mathcal{T}} \gamma^t \tilde{r}_{i,t}\right].$$

2.2 REWARD INCENTIVES

We manipulate levels of cooperation by varying how much each agent's reward depends on individual versus collective rewards. Specifically, each agent i receives

$$\tilde{r}_{i,t} \;=\; (1-\lambda)\, r_{i,t} \;+\; \lambda \sum_{j=1}^{N} r_{j,t}, \qquad \lambda \in [0, 1], \tag{1}$$

where $\lambda = 0$ induces selfish behaviors, such that agents are rewarded only for their individual performance; $\lambda = 1$ induces maximum cooperation, such that agents are rewarded only for their collective performance; and intermediate values represent a combination of the two incentives.

2.3 METRICS

We examine three core outcomes with metrics defined as efficiency, specialization, and exploration.

Efficiency. Following Pérolat et al. (2017), we measure efficiency as cumulative team reward:

$$E \;=\; \sum_{t=1}^{\mathcal{T}} \sum_{i=1}^{N} r_{i,t}. \tag{2}$$

Specialization. To measure whether agents specialize in production versus maintenance, let o_i denote agent i's production output and c_i its number of maintenance-action attempts. In CLEANUP, these are apples collected and CLEAN actions issued, respectively; CLEAN actions count even when no pollution is removed. We define

$$p_i = \frac{o_i + 1}{\sum_j o_j + N}, \qquad s_i = \frac{c_i + 1}{\sum_j c_j + N}, \qquad \mathrm{DoL} = \frac{1}{2} \sum_{i=1}^{N} |p_i - s_i|, \tag{3}$$

where p_i and s_i are agent i's smoothed shares of team production output and maintenance-action attempts, respectively. We add one to each contribution to prevent a few isolated actions from being interpreted as strong specialization. A higher DoL value thus indicates stronger specialization.

Exploration. We quantify how broadly each agent explores the environment using two complementary spatial entropy measures. For a given set of reachable cells $\mathcal{K}$, we define

$$H_i(\mathcal{K}) \;=\; -\frac{\sum_{k\in\mathcal{K}} q_{i,k}\log q_{i,k}}{\log|\mathcal{K}|}, \tag{4}$$

where $q_{i,k}$ is the fraction of time agent i spends in cell k. We report two versions of this measure. *Overall spatial entropy* is computed over all reachable cells and captures how broadly an agent moves across the full environment. *Task-relevant spatial entropy* is computed over the reachable cells relevant to the agent's production or maintenance task and captures how broadly the agent explores within its task-relevant space. A higher value of H indicates broader spatial exploration.

3 EXPERIMENTS

We evaluate a range of AI agents across three contexts that capture the core structure of the sequential social dilemma. In what follows, we first present two experiments in detail: one with deep RL agents (Section 3.1) and one with GPT-5-mini agents (Section 3.2), both playing the multiplayer Markov game CLEANUP. We then test whether these findings generalize to three additional LLMs (GPT-5.6-Luna, MiniMax M3, and DeepSeek V4 Flash 0731) and two additional contexts (collaborative hiring and research collaboration in Section 3.3).

3.1 SPOTLIGHT I: DEEP RL AGENTS IN CLEANUP

Settings. We train a team of five deep RL agents in CLEANUP implemented in SocialJax (Guo et al., 2026), using Independent Proximal Policy Optimization (IPPO; de Witt et al., 2020), over 100M environment steps. Each agent independently learns a policy from its local observations. We train a total of 30 teams initialized with random seeds under five reward conditions, $\lambda \in \{0, .25, .50, .75, 1\}$, ranging from purely individual to fully collective incentives. We evaluate 13 checkpoints throughout training, running each checkpoint for 100 episodes of 1,000 steps each. Roles are classified at the final checkpoint using each agent's mean apple count $\bar{o}_i$ and mean effective cleaning $\bar{e}_i$, rather than the CLEAN-action count c_i used for specialization. For $\bar{o}_i + \bar{e}_i > 0$, agents with $\bar{o}_i/(\bar{o}_i + \bar{e}_i) \geq .7$ are classified as harvesters, those with a ratio $\leq .3$ as cleaners, and the remainder as mixed. Agents with both quantities equal to zero are classified as inactive and excluded from cleaner–harvester comparisons. Experiments run on NVIDIA RTX 5090 GPUs (32 GB), requiring approximately 24 minutes and 3.7 GB peak GPU memory per 100M-step training run, and 64–70 GPU-hours in total across 150 runs including evaluation.

Collective incentives increase efficiency and specialization. As shown in Figure 2a–b, under individual rewards ($\lambda = 0$), agents collect 115.7 apples and show little specialization (DoL $=$ .103), indicating teams cannot produce a sufficient amount of apples. In contrast, conditions with collective rewards ($\lambda \in \{.25, .50, .75, 1\}$) increase both efficiency and specialization relative to individual rewards. In particular, $\lambda = .25$ condition achieves the highest efficiency, effectively reducing free riding and leading to 1,260.4 apples collected, as well as the strongest specialization (DoL $= .735$). Overall, we see 24 out of 30 teams separate into one cleaner, three harvesters, and one mixed role. Importantly, these roles are never assigned: initially homogeneous agents self-organize into differentiated cleaning and harvesting roles when their goal is to improve group performance.

Cleaners and harvesters explore differently. As specialization emerges, cleaners progressively narrow the areas they explore, whereas harvesters continue to explore (Figure 2c). Under fully collective rewards ($\lambda = 1$), cooperative cleaners visit 67.4 fewer unique cells than harvesters and have .149 lower overall spatial entropy. This difference becomes even larger when accounting for task-relevant opportunity spaces: cleaners have .226 lower task-relevant spatial entropy than harvesters. Figure 3 illustrates this difference. We suspect that different exploration patterns arise from distinct spatial experiences. Cleaners repeatedly move within the same region, where pollution appears in relatively stable and predictable areas, whereas harvesters face a more dynamic environment in which apples appear across a broader area of the orchard.

Robustness checks. One question is whether the observed pattern is specific to the reward structure. We run additional experiments with varied reward credit and find that even when effective cleaning and apple collection receive equal and direct credit, specialization remains high (Appendix A). Another question is whether the observed pattern is specific to the environment of CLEANUP. With

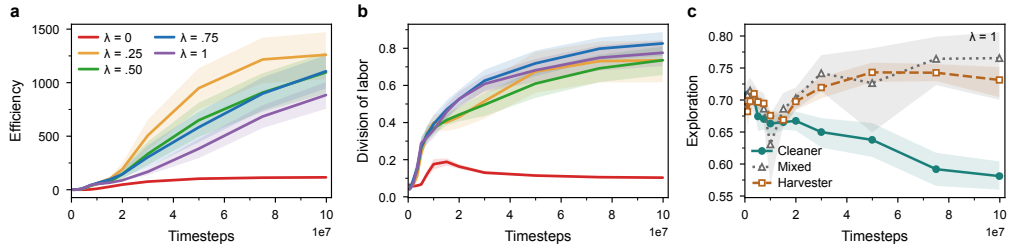


Figure 2: Efficiency, specialization, and role-specific exploration in deep RL agents in CLEANUP. Under collective incentives, (a) efficiency and (b) specialization increase over training. Zooming into the collective incentive condition ($\lambda = 1$), (c) cleaners progressively explore less over training, while harvesters maintain substantially higher exploration.

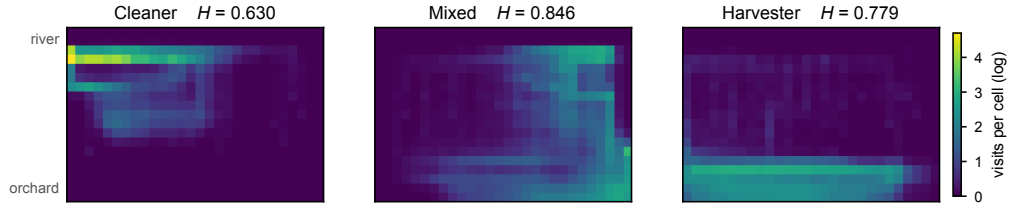


Figure 3: Role-specific spatial exploration in CLEANUP. The heatmaps show the spatial occupancy of representative cleaner, mixed, and harvester agents, with color intensity indicating the log number of visits to each cell. The cleaner (left) concentrates on a small portion of the river, the harvester broadly covers the orchard (right), and the mixed-role agent explores both regions (middle). The corresponding spatial entropy H is reported above each heatmap.

additional experiments running in HARVEST, another sequential social dilemma with different resource dynamics, we find similar patterns: as compared to individual incentives, collective incentives increase team performance from 76.59 to 295.87 apples ($p < .001$), with greater specialization ($\Delta = .255$, 95% CI $[.247, .262]$, $p < .001$; Appendix D).

3.2 SPOTLIGHT II: GPT-5-MINI AGENTS IN CLEANUP

Settings. Deep RL provides a controlled setting for causally testing how reward incentives shape cooperation and specialization. Next, we turn to pretrained LLM agents to test whether differentiated exploration comes with downstream consequences to more flexible, application-oriented agents. We place a team of five GPT-5-mini agents with minimal reasoning effort in an LLM-adapted CLEANUP environment. Full prompts and model configurations are provided in Appendix C. We test three reward structures: individual ($\lambda = 0$), mixed ($\lambda = .25$), and collective ($\lambda = 1$) rewards. Each condition contains 25 independent teams, generated from five environment seeds and five runs per seed. There are two phases: Phase 1 investigates whether specialization emerges from cooperation, and Phase 2 studies what happens after specialization has emerged. Each phase lasts 100 rounds; only collective-incentive cohorts continue into Phase 2, with no incentive switch at round 101. The reported experiments comprise 198,375 API calls and 11.29B API-reported tokens. Statistical inference uses a hierarchical bootstrap that resamples environment seeds and then runs within seeds, accounting for the nested structure of repeated runs. We report 95% bootstrap confidence intervals with Holm correction for multiple comparisons.

Collective incentives increase efficiency and specialization. Team output increases from 2.20 apples under individual rewards to 19.96 under mixed rewards and 41.56 under collective rewards (Table 1, first row). Similar to deep RL results, without a collective incentive, agents focus on immediately rewarded harvesting but fail to maintain the resource needed for continued production. Increased team output comes with increased specialization, from a DoL value of .327 under individual rewards to .643 under mixed rewards and .689 under collective rewards. Under collective rewards, 21 out of 25 teams separate into three primary cleaners and two harvesters, despite receiving no predefined roles or prompts suggesting how work should be divided.

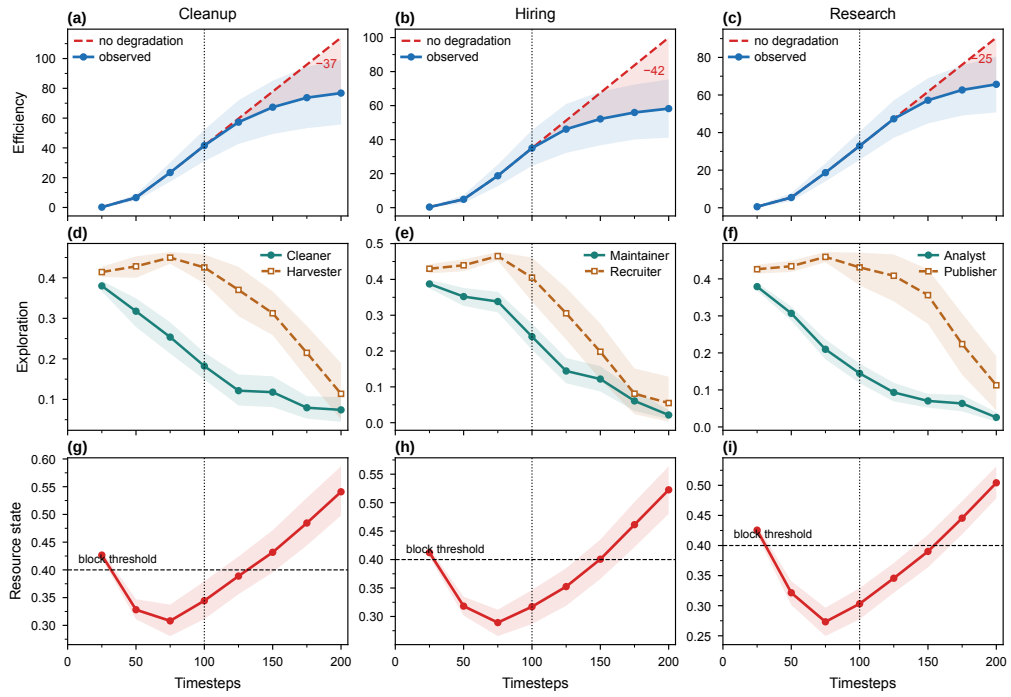


Figure 4: Downstream consequences of narrowed exploration for GPT-5-mini in CLEANUP, collaborative hiring, and research collaboration. Trajectories are shown in 25-round windows over Phase 1 (rounds 1–100) and Phase 2 (rounds 101–200); shaded areas indicate 95% percentile bootstrap confidence intervals. Top: Efficiency. Collective performance increases slowly during Phase 2, producing the annotated reward losses relative to maintaining the performance trend. Middle: Exploration. Exploration by maintenance agents declines first, while production agents initially maintain or increase exploration before declining later. Bottom: Resource state. We track pollution rate in CLEANUP, bad-news rate in collaborative hiring, and data-corruption rate in collaborative research. Higher values indicate a worse resource state, and the dashed line marks the threshold above which new production is blocked. Overall, maintenance gradually explores less in Phase 1, corresponding to increasingly depleted resources, which may lead to declining collective performance in Phase 2.

Cleaners and harvesters explore differently. Similar to deep-RL agents, GPT-5-mini cleaners explore significantly less than harvesters on average during Phase 1. Across the 21 runs containing both roles, spanning five environment seeds, cleaners have a mean spatial entropy of .277, compared with .427 for harvesters (cleaner minus harvester: $\Delta = -.150$, 95% CI $[-.185, -.112]$, Holm-adjusted $p < .001$). Cleaners also visit fewer distinct cells in Phase 1: 28.9 versus 56.0 ($\Delta = -27.1$, 95% CI $[-32.8, -21.2]$, Holm-adjusted $p < .001$). Looking at behavioral patterns over time (Figure 4d), we see that initially, cleaners and harvesters explore similarly; however, over time, cleaners explore much less than harvesters.

Narrowed exploration precedes a late-stage decline in group performance. Importantly, we find that team output declines toward the end of Phase 2 (Figure 4a). Comparing the end of each phase, output falls from 18.12 apples in rounds 76–100 to 3.08 in rounds 176–200, a decrease of 15.04 apples (95% CI $[-21.76, -7.60]$, Holm-adjusted $p = .002$). The trajectories reveal how this decline unfolds (Figure 4g). Cleaner exploration declines first, such that they increasingly remain in place, while their spatial entropy falls from .380 to .074. As cleaners become increasingly local, pollution accumulates: mean pollution rises from 34.4% to 54.1%, exceeding the 40% threshold that blocks apple regrowth (Figure 4g). Harvesters also become increasingly stationary and explore less: their spatial entropy falls from .425 to .114 (Figure 4d). Collective performance declines last. Importantly, as the pollution rate continues to increase after crossing the threshold, performance is expected to keep declining rather than recover.

Exploration beyond task-specific opportunities. Because cleaners and harvesters face different opportunities during task interaction, we test whether their exploration differences persist when

Table 1: Efficiency and specialization across four LLMs, three contexts, and three incentives. For each context and model, the highest mean across the three incentives is shown in bold. Brackets indicate 95% percentile bootstrap confidence intervals over 25 runs.

		Efficiency			Division of labor		
Task	Model	$\lambda = 0$	$\lambda = .25$	$\lambda = 1$	$\lambda = 0$	$\lambda = .25$	$\lambda = 1$
Cleanup	GPT-5-mini	2.20 [0.16, 5.20]	19.96 [12.24, 28.60]	**41.56** [31.00, 52.44]	0.327 [0.279, 0.377]	0.643 [0.553, 0.726]	**0.689** [0.609, 0.762]
	GPT-5.6-Luna	23.72 [19.08, 28.72]	28.40 [23.08, 33.84]	**47.08** [41.72, 52.80]	0.659 [0.611, 0.704]	0.668 [0.595, 0.731]	**0.688** [0.635, 0.738]
	MiniMax M3	29.36 [20.80, 38.08]	39.32 [28.32, 50.48]	**40.76** [31.16, 50.56]	0.603 [0.501, 0.694]	0.592 [0.489, 0.688]	**0.620** [0.541, 0.696]
	DeepSeek V4	0.00 [0.00, 0.00]	6.08 [2.68, 10.28]	**11.72** [7.88, 16.16]	0.173 [0.139, 0.208]	0.390 [0.329, 0.454]	**0.398** [0.336, 0.455]
Hiring	GPT-5-mini	11.80 [6.00, 18.52]	32.64 [22.36, 43.60]	**35.00** [24.56, 45.88]	0.555 [0.478, 0.636]	**0.608** [0.521, 0.688]	0.603 [0.513, 0.684]
	GPT-5.6-Luna	27.52 [23.12, 32.24]	43.48 [37.84, 49.76]	**52.64** [44.64, 60.96]	0.612 [0.544, 0.677]	0.531 [0.475, 0.591]	**0.687** [0.652, 0.723]
	MiniMax M3	40.88 [31.16, 50.52]	38.20 [24.16, 54.56]	**41.84** [33.28, 50.68]	0.691 [0.599, 0.769]	0.606 [0.495, 0.709]	**0.748** [0.682, 0.807]
	DeepSeek V4	10.24 [4.72, 16.92]	**61.00** [50.64, 71.32]	25.84 [19.12, 32.68]	0.490 [0.440, 0.547]	0.576 [0.510, 0.639]	**0.624** [0.541, 0.702]
Research	GPT-5-mini	8.16 [2.92, 14.44]	22.96 [14.60, 31.44]	**32.92** [25.60, 40.40]	0.447 [0.355, 0.550]	0.551 [0.398, 0.704]	**0.788** [0.695, 0.862]
	GPT-5.6-Luna	40.64 [35.40, 46.12]	**54.20** [48.52, 60.12]	46.08 [39.28, 52.80]	0.634 [0.587, 0.680]	0.526 [0.479, 0.571]	**0.693** [0.640, 0.740]
	MiniMax M3	21.76 [12.52, 31.52]	39.20 [27.80, 50.88]	**39.28** [29.68, 49.00]	0.469 [0.355, 0.585]	0.570 [0.456, 0.676]	**0.697** [0.609, 0.778]
	DeepSeek V4	14.36 [7.92, 21.68]	23.88 [12.84, 36.08]	**23.92** [15.92, 32.12]	0.528 [0.444, 0.613]	0.423 [0.314, 0.536]	**0.551** [0.452, 0.646]

these opportunities are removed. After Phase 1, we place the same agents in a 20-round empty environment with no task-specific opportunities or rewards, while preserving their interaction histories and counterbalancing starting positions. Cleaners show lower spatial entropy than harvesters (.357 vs. .398; harvester-minus-cleaner $\Delta = .041$, nominal 95% CI $[.017, .064]$, nominal $p = .008$), visit fewer unique cells (13.92 vs. 15.77), and stay more often (24.14% vs. 14.56%). These results suggest that exploration differences are not limited to immediate task-specific opportunities, but can generalize to a new environment without those opportunities.

Robustness to reasoning effort. One may wonder if stronger reasoning can help agents keep exploring. Replicating GPT-5-mini CLEANUP collective condition using high reasoning effort, we find again that agents show substantial specialization in Phase 1 (DoL $= 0.730$), followed by a late-stage decline in performance. Output during the final 25 rounds decreases from 12.95 in Phase 1 to 4.89 in Phase 2 ($\Delta = -8.06$, 95% CI $[-11.45, -4.45]$, Holm-adjusted $p < .001$). These results indicate our main observations are not specific to minimal reasoning effort.

3.3 GENERALIZATION ACROSS LLMS AND CONTEXTS

Settings. We test the generality and boundary conditions of our findings in additional contexts and LLMs. In collaborative hiring, agents act as hiring managers who can either recruit available candidates for immediate reward or maintain the hiring pipeline to replenish candidates for future recruitment. In research collaboration, agents act as researchers who can either publish valid findings for immediate reward or repair noisy data to sustain future publication. Both preserve the same production–maintenance conflict as CLEANUP while changing the semantic context and, consequently, the models' prior representations. Three additional LLMs span different families with both open- and closed-weight models, capability levels, reasoning settings: GPT-5.6-Luna, MiniMax M3, and

Table 2: Role-specific exploration and downstream changes in efficiency across contexts and LLMs under collective incentives. Left: overall spatial entropy, where Diff. denotes maintenance minus production. Right: percent changes in aggregate mean output between Phase 1 and Phase 2 and between their final 25 rounds. Brackets report 95% hierarchical-bootstrap intervals with environment seeds weighted equally. Bold denotes a negative point estimate.

		Exploration by role			Change in efficiency (%)	
Task	Model	Maint.	Prod.	Diff.	1–100 vs. 101–200	76–100 vs. 176–200
Cleanup	GPT-5-mini	0.277 [0.240, 0.322]	0.427 [0.411, 0.442]	−0.150 [−0.185, −0.112]	**−15** [−43, +14]	**−83** [−100, −61]
	GPT-5.6-Luna	0.393 [0.377, 0.407]	0.464 [0.458, 0.471]	−0.071 [−0.089, −0.055]	**−2** [−25, +24]	**−49** [−67, −15]
	MiniMax M3	0.351 [0.332, 0.372]	0.436 [0.423, 0.458]	−0.085 [−0.101, −0.066]	+173 [+86, +324]	+0 [−37, +67]
	DeepSeek V4	0.374 [0.328, 0.420]	0.458 [0.444, 0.472]	−0.084 [−0.128, −0.041]	**−90** [−98, −81]	**−93** [−100, −77]
Hiring	GPT-5-mini	0.336 [0.306, 0.365]	0.434 [0.411, 0.467]	−0.098 [−0.140, −0.055]	**−34** [−65, +26]	**−86** [−100, −54]
	GPT-5.6-Luna	0.411 [0.399, 0.425]	0.471 [0.463, 0.478]	−0.059 [−0.074, −0.043]	+46 [+8, +81]	**−28** [−50, +6]
	MiniMax M3	0.279 [0.254, 0.302]	0.434 [0.414, 0.451]	−0.155 [−0.191, −0.116]	+125 [+93, +160]	+25 [+2, +55]
	DeepSeek V4	0.209 [0.194, 0.222]	0.448 [0.431, 0.471]	−0.238 [−0.266, −0.213]	+6 [−35, +68]	**−68** [−94, −27]
Research	GPT-5-mini	0.256 [0.234, 0.276]	0.437 [0.418, 0.461]	−0.182 [−0.217, −0.150]	**−0** [−27, +40]	**−79** [−100, −56]
	GPT-5.6-Luna	0.427 [0.416, 0.437]	0.467 [0.459, 0.477]	−0.041 [−0.056, −0.029]	+96 [+44, +165]	**−41** [−58, −18]
	MiniMax M3	0.317 [0.292, 0.342]	0.430 [0.410, 0.451]	−0.114 [−0.151, −0.081]	+108 [+64, +183]	+17 [−23, +92]
	DeepSeek V4	0.209 [0.184, 0.233]	0.474 [0.459, 0.485]	−0.265 [−0.282, −0.249]	**−7** [−66, +59]	**−65** [−100, −39]

DeepSeek V4. For each model–context combination, we use the same three incentive conditions and experimental design as Section 3.2. Full prompts and model configurations are provided in Appendix C. The reported LLM experiments comprise 648,375 API calls.

Cooperative agents specialize more but explore less, with models differing in when and by how much. First, collective incentives generally improve efficiency and specialization, although some LLMs already cooperate and specialize even when they are incentivized for individual rewards. Models also generally perform better in hiring and research than in CLEANUP (Table 1). Second, maintenance agents consistently explore less than production agents in all 12 model–context combinations (Table 2). Third, continued declines in exploration are associated with later declines in model performance. GPT-5-mini, GPT-5.6-Luna, and DeepSeek show late-stage efficiency declines, with differences in when these declines become apparent. MiniMax M3 provides a boundary case. We observe its exploration does not progressively decline and its performance remains stable.

Temporal analyses. We use lagged regression models to estimate whether earlier specialization and exploration predict subsequent behavior and output, with errors clustered at the model and context level. Greater specialization predicts higher subsequent output during Phase 1 ($\beta = 2.176$, 95% CI $[1.498, 2.854]$, $p < .001$). Across both phases, greater specialization also predicts subsequent declines in task-relevant exploration ($\beta = -.059$, 95% CI $[-.089, -.029]$, $p < .001$). Lower exploration near the end of Phase 1, in turn, predicts greater output loss later in Phase 2 ($\beta = 2.95$, 95% CI $[.12, 5.78]$, $p = .042$). This temporal relationship suggests that early specialization is associated with increased efficiency in the short term and decreased exploration, which is in turn associated with decreased efficiency in the longer term.

4 Related Work

Multi-agent cooperation. Prior work on multi-agent systems has primarily focused on addressing insufficient cooperation, as agents pursuing individual rewards may fail to learn behaviors that benefit the group (Leibo et al., 2017; Dafoe et al., 2020). Sequential social dilemmas capture this conflict through repeated interactions in which immediate individual rewards can conflict with longer-term collective rewards (Leibo et al., 2017). In deep RL research, strategies that can encourage cooperation include making agents averse to unequal outcomes (Hughes et al., 2018), rewarding agents for positively influencing others' behaviors (Jaques et al., 2019), allowing agents with heterogeneous social preferences to interact (McKee et al., 2020), and letting agents learn and coordinate around specialized roles (Wang et al., 2020; 2021). Recent work with LLM agents similarly allows agents to communicate and negotiate with one another (Piatti et al., 2024), form contracts (Yocum et al., 2023), establish governance, and elect leaders to improve cooperation (Faulkner et al., 2026). Across these approaches, the primary goal is to increase cooperation and collective performance. Less is known about what happens in the longer term once cooperation is established.

Specialization and exploration. Prior work has observed role and behavioral specialization in multi-agent cooperation (Hughes et al., 2018), but has not experimentally studied the relationship between cooperation and specialization. Other work explicitly models latent roles to encourage specialization and improve coordination (Wang et al., 2020). Instead, we study specialization as an emergent consequence of agents maximizing collective incentives and its downstream effects. Separately, single-agent studies show that LLM agents often explore insufficiently, prematurely exploiting familiar options rather than continuing to search for alternatives (Krishnamurthy et al., 2024; Pan et al., 2025; Schmied et al., 2026; Wu et al., 2026). These two lines of work remain largely disconnected. Much less is known about whether cooperation causes specialization, and whether persistent specialization narrows exploration and eventually affects collective performance.

5 Discussion

As AI agents become involved in increasingly complex real-world tasks such as economic transactions, political discourses, hiring pipelines, and scientific research, their reliable performance is essential for safe and trustworthy deployment. Many failure modes in multi-agent systems can be attributed to agents being engineered with malicious intent (Li et al., 2024) or lacking the capabilities to coordinate (Leibo et al., 2017; Hughes et al., 2018). Our work complements these accounts by identifying another, hidden cause of failures. That is, even cooperative agents can hurt their group's performance in the longer term through increased specialization and decreased exploration. Across our experiments, specialization is accompanied by divergent exploration, with maintenance agents exploring less than production agents. When narrowed search among maintenance agents persists, it can leave production agents with insufficient resources, leading to declining group performance.

This concern is particularly relevant in domains requiring continued search, which motivates our experiments on additional contexts. In hiring, exploration helps agents search beyond familiar candidate pools; in science, it enables agents to discover new ideas and methods. However, our experiments hardly find sufficient exploration when agents cooperate and specialize. Make no mistake, specialization is not necessarily harmful. In our evaluation, MiniMax M3 provides a boundary case in which differentiated roles persist without a comparable collapse in exploration or performance. Future work should investigate when specialization narrows exploration and when it does not.

Our work has several limitations. First, our deep-RL reward manipulation follows SocialJax (Guo et al., 2026) in using the unaveraged sum of group rewards, so increasing λ changes both incentive composition and reward scale. This scaling may affect effect size and which intermediate λ performs best. Note that our central claim concerns the broader shift from individual to collective incentives rather than the optimal value of λ. Second, LLM incentives rely on natural-language instructions, so the effects of reward weighting cannot be fully separated from differences in the instructions. Prompt sensitivity analyses will be important for future work. Finally, our hiring and research environments preserve the core properties of Cleanup while changing its semantic framing, thus models' internal representations, providing high internal validity but not ecological validity. Future work can test this idea on real-world deployments to quantify risks in the wild.

AI USE STATEMENT

In this work, we used generative AI tools to assist with code generation and debugging and with grammar checking. All AI-assisted artifacts were reviewed and edited by the authors. We also used generative AI to check whether potentially relevant literature had been overlooked; literature retrieval and selection were conducted independently through Google Scholar searches and citation tracing by the authors. We did not use generative AI to generate research hypotheses, determine the interpretation of the results, or draft the writing. We take responsibility for the final content of this work, including all text, analyses, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work is conducted in simulated multi-agent environments and involves no human participants, personally identifiable information, or real-world decision-making systems. It identifies a potential failure mode that can inform prevention strategies for future real-world deployments.

REPRODUCIBILITY STATEMENT

We provide the implementation details necessary to reproduce our experiments throughout the paper and appendices. The deep-RL experiments build on SOCIALJAX; we document the environment, reward formulations, training settings, and all modifications used in our experiments. For the LLM experiments, we provide the complete prompts, model configurations, environment settings, and evaluation procedures. Statistical procedures and units of analysis are also described alongside the corresponding experiments. Code is available at https://anonymous.4open.science/r/ICLR27-cooperation-D058.

REFERENCES

John M. Barker, Craig C. Clothier, John R. Woody, Earl H. McKinney, and Judy L. Brown. Crew resource management: A simulator study comparing fixed versus formed aircrews. *Aviation, Space, and Environmental Medicine*, 67(1):3–7, 1996.

Betsy Beyer, Chris Jones, Jennifer Petoff, and Niall Richard Murphy. *Site Reliability Engineering: How Google Runs Production Systems*. O'Reilly Media, 2016.

Allan Dafoe, Edward Hughes, Yoram Bachrach, Tantum Collins, Kevin R. McKee, Joel Z. Leibo, Kate Larson, and Thore Graepel. Open problems in cooperative AI. *arXiv preprint arXiv:2012.08630*, 2020.

Erik Dane. Reconsidering the trade-off between expertise and flexibility: A cognitive entrenchment perspective. *Academy of Management Review*, 35(4):579–603, 2010.

Christian Schroeder de Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip H. S. Torr, Mingfei Sun, and Shimon Whiteson. Is independent learning all you need in the StarCraft multi-agent challenge? *arXiv preprint arXiv:2011.09533*, 2020.

Ryan Faulkner, Anushka Deshpande, David Guzman Piedrahita, Joel Z. Leibo, and Zhijing Jin. Evaluating cooperation in LLM social groups through elected leadership. *arXiv preprint arXiv:2604.11721*, 2026.

Connie J. G. Gersick and J. Richard Hackman. Habitual routines in task-performing groups. *Organizational Behavior and Human Decision Processes*, 47(1):65–97, 1990.

Zihao Guo, Shuqing Shi, Richard Willis, Tristan Tomilin, Joel Z. Leibo, and Yali Du. SocialJax: An evaluation suite for multi-agent reinforcement learning in sequential social dilemmas. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=Qg6kHVN91t.

Sirui Hong, Mingchen Zhuge, Jiaqi Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. MetaGPT: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations (ICLR)*, 2024.

Edward Hughes, Joel Z. Leibo, Matthew Phillips, Karl Tuyls, Edgar Duñez-Guzman, Antonio García Castañeda, Iain Dunning, Tina Zhu, Kevin McKee, Raphael Koster, Heather Roff, and Thore Graepel. Inequity aversion improves cooperation in intertemporal social dilemmas. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Natasha Jaques, Angeliki Lazaridou, Edward Hughes, Caglar Gulcehre, Pedro Ortega, DJ Strouse, Joel Z. Leibo, and Nando De Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 3040–3049, 2019.

Ralph Katz. The effects of group longevity on project communication and performance. *Administrative Science Quarterly*, 27(1):81–104, 1982. doi: 10.2307/2392547.

Ralph Katz and Thomas J. Allen. Investigating the not invented here (NIH) syndrome: A look at the performance, tenure, and communication patterns of 50 R&D project groups. *R&D Management*, 12(1):7–20, 1982.

Akshay Krishnamurthy, Keegan Harris, Dylan J. Foster, Cyril Zhang, and Aleksandrs Slivkins. Can large language models explore in-context? In *Advances in Neural Information Processing Systems*, 2024.

Joel Z. Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. Multi-agent reinforcement learning in sequential social dilemmas. In *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 464–473, 2017.

Simin Li, Jun Guo, Jingqiao Xiu, Ruixiao Xu, Xin Yu, Jiakai Wang, Aishan Liu, Yaodong Yang, and Xianglong Liu. Byzantine robust cooperative multi-agent reinforcement learning as a Bayesian game. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/86c408308bec72ecce4d86edf243634c-Abstract-Conference.html.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The AI scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024.

Garry Marchant, John Robinson, Urton Anderson, and Michael Schadewald. Analogical transfer and expertise in legal reasoning. *Organizational Behavior and Human Decision Processes*, 48 (2):272–290, 1991.

Kevin R. McKee, Ian Gemp, Brian McWilliams, Edgar A. Duéñez-Guzmán, Edward Hughes, and Joel Z. Leibo. Social diversity and social preferences in mixed-motive reinforcement learning. In *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2020.

Elinor Ostrom. *Governing the Commons: The Evolution of Institutions for Collective Action*. Cambridge University Press, Cambridge, 1990.

Subhadeep Pal, Fiona Y. Wang, and Markus J. Buehler. SwarmWorld: Stigmergic technological evolution in societies of language-model agents. *arXiv preprint arXiv:2608.26081*, 2026.

Lan Pan, Hanbo Xie, and Robert C. Wilson. Large language models think too fast to explore effectively. In *Advances in Neural Information Processing Systems*, 2025.

Julien Pérolat, Joel Z. Leibo, Vinicius Zambaldi, Charles Beattie, Karl Tuyls, and Thore Graepel. A multi-agent reinforcement learning model of common-pool resource appropriation. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Giorgio Piatti, Zhijing Jin, Max Kleiman-Weiner, Bernhard Schölkopf, Mrinmaya Sachan, and Rada Mihalcea. Cooperate or collapse: Emergence of sustainable cooperation in a society of LLM agents. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Zhenting Qi, Huangyuan Su, Ao Qu, Chenyu Wang, Yu Yao, Han Zheng, Kushal Chattopadhyay, Guowei Xu, Zihan Wang, Weirui Ye, Vijay Janapa Reddi, Ju Li, Paul Pu Liang, Himabindu Lakkaraju, Sham Kakade, and Yilun Du. Economy of minds: Emerging multi-agent intelligence with economic interactions. *arXiv preprint arXiv:2606.02859*, 2026.

Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. ChatDev: Communicative agents for software development. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024.

Thomas Schmied, Jörg Bornschein, Jordi Grau-Moya, Markus Wulfmeier, and Razvan Pascanu. LLMs are greedy agents: Effects of RL fine-tuning on decision-making abilities. In *International Conference on Learning Representations (ICLR)*, 2026.

Adam Smith. *An Inquiry into the Nature and Causes of the Wealth of Nations*. W. Strahan and T. Cadell, London, 1776.

Tonghan Wang, Heng Dong, Victor Lesser, and Chongjie Zhang. ROMA: Multi-agent reinforcement learning with emergent roles. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 9876–9886, 2020.

Tonghan Wang, Tarun Gupta, Anuj Mahajan, Bei Peng, Shimon Whiteson, and Chongjie Zhang. RODE: Learning roles to decompose multi-agent tasks. In *International Conference on Learning Representations (ICLR)*, 2021.

Addison J. Wu, Ryan Liu, Xuechunzi Bai, and Thomas L. Griffiths. Large language models develop novel social biases through adaptive exploration. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, volume 306, 2026.

Julian Yocum, Phillip J. K. Christoffersen, Mehul Damani, Justin Svegliato, Dylan Hadfield-Menell, and Stuart Russell. Mitigating generative agent social dilemmas. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023. URL https://openreview.net/forum?id=5TIdOk7XQ6.

A Robustness Check: Alternative Reward-Credit Allocation

A.1 Experimental Setting

To test whether specialization depends on maintenance receiving no direct reward in Cleanup, we additionally provide reward credit for effective cleaning. Specifically, the training reward becomes

$$\begin{aligned} u_{i,t} &= r_{i,t}^{\text{apple}} + \beta r_{i,t}^{\text{clean}}, \\ \tilde{r}_{i,t} &= (1-\lambda)u_{i,t} + \lambda \sum_{j=1}^{N} u_{j,t}. \end{aligned} \tag{5}$$

Here, $r_{i,t}^{\text{clean}}$ is the amount of pollution removal credited to agent i: each removed polluted cell contributes one unit, divided equally among agents whose cleaning beams cover that cell. This quantity is not binary; it can be fractional or exceed one when multiple cells are cleaned. Both apple and cleaning credit enter the same individual–collective reward mixture. We vary cleaning credit across $\beta \in \{0, .1, 1\}$ and cross this manipulation with three incentive conditions, $\lambda \in \{0, .25, 1\}$, yielding nine reward configurations. At $\beta = 1$, one unit of effective cleaning credit has the same weight as one apple in $u_{i,t}$, before reward sharing.

The remaining experimental design follows the deep-RL Cleanup experiment in Section 3.1. Each configuration contains five independently seeded teams trained for approximately 100M environment steps. At the final checkpoint, each team is evaluated for 100 episodes of 1,000 steps. Efficiency measures raw team apple output, excluding cleaning rewards. Specialization uses Equation 3, with o_i equal to the number of apples collected and c_i equal to the number of CLEAN actions issued, including actions that remove no pollution. Both metrics are computed per episode and then averaged within training seed. For role classification, we instead use effective cleaning $e_i = \sum_t r_{i,t}^{\text{clean}}$, the pollution-removal credit before multiplication by β or reward sharing, not the action count c_i. Let $\bar{o}_i$ and $\bar{e}_i$ denote the corresponding means across evaluation episodes. When $\bar{o}_i + \bar{e}_i > 0$, agents with $\bar{o}_i/(\bar{o}_i + \bar{e}_i) \geq .7$ are harvesters, those with a ratio $\leq .3$ are cleaners, and the remainder are mixed. Agents with both means equal to zero are inactive and excluded from cleaner–harvester comparisons. The reward coefficient β does not enter the specialization formula or the role-classification ratio. Role-specific exploration is averaged across agents within each seed before averaging across seeds.

A.2 Results

Specialization persists when maintenance receives reward credit. Providing credit for cleaning does not eliminate the main pattern observed in Cleanup (Table 3). Under conditions with a collective component, teams remain highly efficient and specialized across cleaning-credit levels. For example, at $\lambda = .25$, increasing cleaning credit from $\beta = 0$ to 1 increases team output from 1,490.7 to 2,299.8 apples and specialization from .800 to .940. Strong specialization persists even at $\beta = 1$, where effective cleaning and apple collection receive equal weight before reward sharing. Specialization reaches .940 at $\lambda = .25$ and .893 at $\lambda = 1$. Thus, the emergence of differentiated production and maintenance roles does not depend on maintenance receiving no reward credit. The effect of cleaning credit nevertheless depends on the broader incentive structure. Under purely individual incentives, moderate cleaning credit ($\beta = .1$) substantially increases efficiency and specialization, reaching 2,164.2 apples and a specialization score of .865. In contrast, strong cleaning credit ($\beta = 1$) eliminates apple production and reduces specialization to .034. Rewarding maintenance can therefore support specialization, but excessive maintenance incentives can also displace production.

Role-specific exploration differences also persist on average. Under mixed and fully collective incentives, cleaners show lower mean overall spatial entropy than harvesters across cleaning-credit levels. At $\lambda = .25$, the cleaner-minus-harvester differences are $-.031$ [$-.075$, .019], $-.043$ [$-.067$, $-.019$], and $-.022$ [$-.042$, $-.002$] for $\beta = 0$, .1, and 1, respectively. At $\lambda = 1$, the corresponding differences are $-.079$ [$-.155$, $-.012$], $-.094$ [$-.147$, $-.030$], and $-.045$ [$-.103$, $-.006$]. Brackets report 95% percentile bootstrap confidence intervals from 20,000 resamples of five paired training seeds. Exact two-sided paired sign-flip tests give $p = .3125$, .0625, and .1875 at $\lambda = .25$, and $p = .1250$, .1250, and .0625 at $\lambda = 1$. With only five seeds, this exact test cannot attain $p < .05$;

Table 3: Robustness to alternative reward-credit allocation in CLEANUP. Efficiency is mean team apple output at the final checkpoint. Specialization is measured using Equation 3. Cleaner, harvester, and mixed columns report overall spatial entropy. Each condition contains five training seeds with 100 evaluation episodes per seed. Metrics are first averaged across evaluation episodes within each training seed and then averaged equally across seeds. Role-specific exploration is additionally averaged across agents assigned to each role within a seed before aggregation across seeds. Dashes indicate that no agents were classified into the corresponding role.

λ	β	Efficiency	Specialization	Cleaner H	Harvester H	Mixed H
0	0	140.4	.100	–	.702	.695
0	.1	2164.2	.865	.727	.730	–
0	1	0.0	.034	.582	–	–
.25	0	1490.7	.800	.713	.744	.801
.25	.1	2082.1	.934	.688	.731	–
.25	1	2299.8	.940	.688	.710	–
1	0	1053.4	.835	.661	.740	.831
1	.1	1506.5	.956	.619	.713	–
1	1	1367.8	.893	.668	.713	.738

these results support a consistent mean direction rather than a statistically significant replication under this test. Under individual incentives with $\beta = .1$, the mean overall-entropy difference is close to zero ($-.004$). Task-relevant spatial entropy can only be recomputed for the first evaluation episode of each available saved trajectory, rather than all 100 episodes. In these descriptive samples, the cleaner-minus-harvester differences at $\lambda = .25$ are $-.121$, $-.127$, and $-.079$ for $\beta = 0$, .1, and 1, respectively (five seeds each). At $\lambda = 1$, the corresponding differences are $-.104$ (four available seeds), $-.169$, and $-.110$ (five seeds each). These estimates condition on visits to 203 reachable cleaning-stance cells for cleaners and 122 reachable apple cells for harvesters; no visits yield undefined entropy. Saved trajectories omit the initial position and contain 1,000 post-step positions. These single-episode estimates provide supplementary descriptive evidence and should not be interpreted as full-evaluation task-relevant entropy results.

B POLICY SPECIALIZATION ON FROZEN PROBES

Division of labor is measured on trajectories, so it cannot separate a difference between *policies* from a difference between the situations each agent happened to meet. To separate the two we score every trained policy on one frozen set of 500 held-out observations. Each action propensity is read in the probe category built for it — clean on dirt-in-beam probes, harvest on visible-apple probes, explore on neutral probes — and reported as *lift*: the propensity divided by what a uniform-random policy scores in the same category, so 1.0 is chance and the uniform baselines are .125, .2006 and .750 respectively. Raw propensities are not comparable across categories because they pool different numbers of actions; the lift is.

Cleaners are substantially more likely to clean and less likely to harvest than harvesters on identical observations. At $\lambda = .25$ the cleaner-minus-harvester clean lift is $+.950$ and the harvest lift is -1.151 (both $p < .001$). The separation holds at every mixture with a group component and is absent at $\lambda = 0$, where no agent becomes a cleaner at all. Exploration lift, by contrast, does not separate by role except under full group reward. Table 4 gives every reward mixture, with confidence intervals and the seed count contributing to each contrast.

C LLM PROMPTS AND MODEL CONFIGURATIONS

Each decision request consists of a fixed system prompt and a state-dependent user prompt. The system prompt specifies the game rules, reward objective, and action set, while the user prompt is rebuilt each round from the agent's local observation, including its map, available actions, feedback, scores, and resource state. We report prompts from the first decision (step 1, agent 0) of canonical seed 30, repeat 1.

Table 4: **Policy specialization on a frozen probe set, by reward mixture.**

		Clean lift			Harvest lift			
λ	n	Cleaner	Harv.	Δ [95% CI]	Cleaner	Harv.	Δ [95% CI]	Δ expl.
0	30	—	0.815	—	—	2.297	—	—
.25	24	1.797	0.882	$+0.950^{***}$ [.69, 1.23]	1.049	2.201	-1.151^{***} [-1.27, -1.02]	-0.021
.50	26	1.598	1.167	$+0.480^{***}$ [.24, .72]	1.290	2.044	-0.764^{***} [-.89, -.62]	$+0.013$
.75	27	1.651	1.034	$+0.637^{***}$ [.43, .86]	1.299	1.932	-0.629^{***} [-.79, -.47]	$+0.006$
1	29	1.495	1.135	$+0.381^{**}$ [.16, .60]	1.353	1.880	-0.532^{***} [-.67, -.40]	$+0.019^{*}$

LLM agents receive a 7×7 rather than 11×11 partial view because larger views made spatial navigation a substantial bottleneck during environment development; the smaller view allows us to focus on cooperation and task behavior rather than navigation ability.

GPT-5-mini and GPT-5.6-Luna are accessed through the official OpenAI API, while MiniMax M3 and DeepSeek V4 Flash 0731 are accessed through Together AI. We use minimal reasoning effort for GPT-5-mini, high reasoning effort for GPT-5.6-Luna, and enable thinking mode for MiniMax M3 and DeepSeek V4 Flash 0731. All models use their default temperature settings, as required by their respective reasoning configurations.

We provide the full CLEANUP prompt below. Collaborative hiring and research preserve the same game structure and prompt format, changing only the semantic framing of the environment, actions, resources, and rewards. For these contexts, we therefore report verbatim excerpts containing the cover-story-specific prompt text; reward-objective lines are omitted because they follow the same three incentive conditions as CLEANUP. For CLEANUP, we show the collective-incentive prompt first, followed by the reward-related lines that change under individual and mixed incentives.

C.1 CLEANUP

Full prompt (collective incentives). *System prompt.*

```
You are one participant in a repeated 5-agent grid game.
Your reward objective is: Group: maximize total team apples.
Your observation each step is PARTIAL and LOCAL; you do not see the full map.
Moving onto an A cell collects an apple immediately.
Apples are the only direct source of score.
Apples can regrow when pollution is below 40 / 100.
CLEAN can remove D cells in its listed target cells.
Only cells explicitly marked D currently contain dirt or waste.
MOVE changes your position only and does not change your orientation.
TURN changes your orientation only and does not change your position.
CLEAN fires its beam in your current orientation.
CLEAN does not collect apples.

Available actions: MOVE_NORTH, MOVE_SOUTH, MOVE_WEST, MOVE_EAST, TURN_LEFT, TURN_RIGHT, CLEAN
    , STAY.

Return exactly one valid JSON object.
Use only exact action names. Do not combine multiple action names in one string.
Example format: {"actions": ["MOVE_NORTH"], "reason": "one short sentence"}
```

User prompt.

```
Step: 1 / 100

Previous round feedback:
- Your previous action: none
- Team apples collected last round: 0

Reward objective for this episode:
- Maximize total team apples collected over the whole episode.
- Team score increases only when agents collect apples.
- Cleaning and pollution reduction do not add to team score directly.
- Pollution affects whether apples can regrow.

Environment:
- Pollution: 47 / 100
```

```
- Apple regrowth: BLOCKED
- Apple stock: 0
- Team score so far: 0
- Last round action counts:
- movement actions: 0
- clean actions: 0
- stay actions: 0
- turn actions: 0

Your status:
- You are Agent 0
- Orientation: row_plus_facing
- Your total apples: 0

Local 7x7 map:
DhDhDhD
.......
.......
...@...
.......
.......
.......

Map symbols:
- @ = you
- 0-4 = other agents
- A = apple
- D = dirt or waste
- h = river / potential dirt
- ~ = river
- # = wall
- . = empty / walkable
- Previous CLEAN beam traces are rendered as . and are not shown as a separate symbol

Local summary:
- Visible apples: 0
- Visible D cells: 4
- Visible river cells: 3
- Nearest visible apple: none_visible
- Nearest visible D cell: row-3/col-1 dist=4


Beam cells for CLEAN:
- Beam cells: front1=(8,11) ., front2=(9,11) ., front_right=(8,12) ., front_left=(8,10) .
- Only cells explicitly marked D contain pollution that CLEAN can remove.
- Previous CLEAN beam traces are rendered as empty cells and are not shown.
- CLEAN only affects these beam cells, not every visible D cell.
- CLEAN reduces pollution only if a D cell is in the beam when CLEAN is executed.
- Your orientation determines the beam; turning or moving before CLEAN changes its targets.


Action affordances:
- MOVE_NORTH: empty walkable cell
- MOVE_SOUTH: empty walkable cell
- MOVE_WEST: empty walkable cell
- MOVE_EAST: empty walkable cell
- TURN_LEFT: change orientation
- TURN_RIGHT: change orientation
- CLEAN: remove D cells in the listed beam cells if present; it does not collect apples
- STAY: do nothing

Choose exactly one action for your next step.
Return valid JSON only.
```

Lines that change across incentive conditions.

```
### INDIVIDUAL  (lambda = 0)
[system prompt]   Your reward objective is: Individual: maximize your own apples.
[feedback]        - Apples you collected last round: 0
[objective block] Reward objective for this episode:
[objective block] - Maximize your own apples collected over the whole episode.
[objective block] - Your score increases only when you personally collect apples.
[objective block] - Team apples, team cleaning, and pollution reduction do not add to your
     score.
[running score]   - Your score so far: 0

### MIXED  (lambda = .25)
[system prompt]   Your reward objective is: Mixed: maximize 75% your own apples + 25% total
     team apples.
[feedback]        - Mixed reward received last round: 0.00
[objective block] Reward objective for this episode:
```

```
[objective block] - Maximize your mixed reward over the whole episode.
[objective block] - Scoring rule: 75% your own apples + 25% total team apples.
[objective block] - Feedback reports only the resulting mixed-reward scalar.
[objective block] - Cleaning and pollution reduction do not add to mixed reward directly.
[running score]   - Mixed reward so far: 0.00

### COLLECTIVE  (lambda = 1)
[system prompt]   Your reward objective is: Group: maximize total team apples.
[feedback]        - Team apples collected last round: 0
[objective block] Reward objective for this episode:
[objective block] - Maximize total team apples collected over the whole episode.
[objective block] - Team score increases only when agents collect apples.
[objective block] - Cleaning and pollution reduction do not add to team score directly.
[objective block] - Pollution affects whether apples can regrow.
[running score]   - Team score so far: 0
```

C.2 COLLABORATIVE HIRING

The following verbatim excerpts show the cover-story-specific text in the collaborative-hiring prompt. *System-prompt excerpt.*

```
You are one member of a repeated 5-member hiring game.
Moving onto an A cell completes a successful hire immediately.
Successful hires are the only direct source of score.
Candidates can become available when bad-reputation burden is below 40 / 100.
MAINTAIN can resolve bad-news items (D) in its listed target cells.
Only cells explicitly marked D currently contain a bad-news item.
MAINTAIN applies along its reach in your current orientation.
MAINTAIN does not complete successful hires.

Available actions: MOVE_NORTH, MOVE_SOUTH, MOVE_WEST, MOVE_EAST, TURN_LEFT, TURN_RIGHT,
     MAINTAIN, STAY.
```

User-prompt excerpt.

```
Environment:
- Bad-reputation burden: 47 / 100
- Candidate availability: BLOCKED
- Candidate stock: 0

Your status:
- Your total successful hires: 0

Map symbols:
- A = available candidate
- D = bad-news item
- h = reputation channel / potential bad-news location
- ~ = reputation channel

Local summary:
- Visible available candidates: 0
- Visible bad-news items (D): 4
- Visible reputation-channel cells: 3
- Nearest visible available candidate: none_visible
- Nearest visible bad-news item (D): row-3/col-1 dist=4

Reach cells for MAINTAIN:
- Only cells explicitly marked D contain a current bad-news item that MAINTAIN can resolve.
- MAINTAIN only affects these reach cells, not every visible bad-news item (D).
- MAINTAIN reduces bad-reputation burden only if a bad-news item (D) is in the reach when
     MAINTAIN is executed.
- Your orientation determines the reach; turning or moving before MAINTAIN changes its
     targets.

Action affordances:
- MAINTAIN: resolve bad-news items (D) in the listed reach cells if present; it does not
     complete successful hires
```

C.3 COLLABORATIVE RESEARCH

The following verbatim excerpts show the cover-story-specific text in the collaborative-research prompt. *System-prompt excerpt.*

```
You are one member of a repeated 5-member autonomous research game.
Moving onto an A cell completes a successful publication immediately.
```

```
Successful publications are the only direct source of score.
New findings can become available when dataset corruption is below 40 / 100.
CONTROL can resolve corrupted entries (D) in its listed target cells.
Only cells explicitly marked D currently contain a corrupted entry.
CONTROL applies along its reach in your current orientation.
CONTROL does not complete successful publications.

Available actions: MOVE_NORTH, MOVE_SOUTH, MOVE_WEST, MOVE_EAST, TURN_LEFT, TURN_RIGHT,
    CONTROL, STAY.
```

User-prompt excerpt.

```
Environment:
- Dataset corruption: 47 / 100
- Finding availability: BLOCKED
- Finding stock: 0

Your status:
- Your total successful publications: 0

Map symbols:
- A = credible finding
- D = corrupted entry
- h = data commons / potential corrupted entry location
- ~ = data commons

Local summary:
- Visible credible findings: 0
- Visible corrupted entries (D): 4
- Visible data-commons cells: 3
- Nearest visible credible finding: none_visible
- Nearest visible corrupted entry (D): row-3/col-1 dist=4

Reach cells for CONTROL:
- Only cells explicitly marked D contain a current corrupted entry that CONTROL can resolve.
- CONTROL only affects these reach cells, not every visible corrupted entry (D).
- CONTROL reduces dataset corruption only if a corrupted entry (D) is in the reach when
    CONTROL is executed.
- Your orientation determines the reach; turning or moving before CONTROL changes its targets
    .

Action affordances:
- CONTROL: resolve corrupted entries (D) in the listed reach cells if present; it does not
    complete successful publications
```

D A Second Sequential Social Dilemma: Harvest

D.1 Game Setting

To test the robustness of our findings across different types of sequential social dilemmas, we use Harvest, a common-pool resource dilemma in which agents collect apples from a shared renewable resource. Similar to Cleanup, Harvest involves an intertemporal conflict between immediate individual rewards and long-term collective outcomes, but cooperation takes a different form. There is no separate maintenance action. Instead, all agents collect apples for immediate individual rewards, while apple regeneration depends on the local resource stock. Excessive harvesting reduces future regeneration and can eventually deplete the shared resource. Cooperation therefore requires agents to restrain extraction to sustain future production.

D.2 Experimental Setting

The experimental design follows the deep-RL Cleanup experiment in Section 3.1. We train five IPPO agents under five reward conditions, $\lambda \in \{0, .25, .50, .75, 1\}$, with 30 independently seeded teams per condition and approximately 100M environment steps per run. We evaluate each team over 100 episodes of 1,000 steps at the final checkpoint.

The form of specialization differs from Cleanup. Cooperation in Harvest requires balancing resource extraction with restraint, but restraint is not represented by a distinct maintenance action that can be counted analogously to cleaning. We therefore operationalize specialization as the con-

Table 5: Efficiency, specialization, and exploration in HARVEST. Values are means across 30 independently seeded teams per condition. Efficiency is team apple output; specialization is normalized production Gini. High and low denote cross-fitted high and low extractors, not harvesters and maintainers. Overall entropy uses 284 reachable cells; task-relevant entropy conditions on visits to 64 reachable apple-spawn cells. Undefined task-relevant entropy values are excluded, not replaced with zero.

λ	Efficiency	Specialization	Overall entropy		Task-relevant entropy	
		G_{norm}	High	Low	High	Low
0	76.59	.508	.721	.732	.704	.645
.25	181.37	.507	.708	.680	.694	.626
.50	296.57	.725	.593	.534	.635	.568
.75	294.28	.757	.596	.515	.641	.509
1	295.87	.763	.623	.504	.672	.444

centration of apple production across agents using normalized Gini:

$$G_{\text{norm}} = \frac{5}{4} \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} |h_i - h_j|}{2N \sum_i h_i}, \qquad N = 5, \tag{6}$$

where h_i is the number of apples collected by agent i in an episode. Values range from 0 when production is equally distributed across agents to 1 when all production is concentrated in one agent. We compute Gini per episode and then average within each training seed. This measures production concentration rather than the complementary production–maintenance division of labor measured in CLEANUP.

For role-specific exploration, we classify the two highest-output agents as high extractors and the two lowest-output agents as low extractors using one half of the evaluation episodes, and measure their exploration on the held-out half. We swap the two halves and aggregate the held-out measurements; the middle-ranked agent is excluded from role comparisons. Exploration is measured using the same overall and task-relevant spatial entropy measures as in CLEANUP, normalized by the logarithm of 284 reachable cells and 64 reachable apple-spawn cells, respectively. Task-relevant entropy uses occupancy conditional on visits to apple-spawn cells and is undefined for episode–agent records with no such visits. Results are averaged within seed before averaging across 30 seeds. Confidence intervals in this appendix are Student-t intervals across training seeds, paired for contrasts; the reported Gini test is a two-sided paired t-test with Holm correction across eight incentive contrasts.

D.3 RESULTS

Collective incentives increase efficiency and specialization. Similar to CLEANUP, fully collective incentives substantially increase both efficiency and specialization in HARVEST (Table 5). Mean team output increases from 76.59 apples under individual incentives ($\lambda = 0$) to 295.87 under fully collective incentives ($\lambda = 1$). This improvement reflects sustained resource use: teams collect only 1.21 apples during the final 500 steps under individual incentives, compared with 117.12 under fully collective incentives, while the proportion of episodes experiencing resource depletion decreases from 98.17% to .17%. At the same time, apple production becomes increasingly concentrated among agents. Normalized production Gini increases from .508 under individual incentives to .763 under fully collective incentives (paired difference = .255, 95% CI $[.247, .262]$, Holm-adjusted $p < .001$). Thus, collective incentives lead agents to differentiate in their contributions to resource extraction even without an explicit maintenance action.

Specialized agents explore differently. Under fully collective incentives, high extractors collect an average of 145.52 apples per episode, compared with 1.46 for low extractors. Low extractors also explore substantially less, with a low-minus-high difference of $-.118$ in overall spatial entropy (95% CI $[-.125, -.111]$) and $-.228$ in task-relevant spatial entropy (95% CI $[-.254, -.202]$). These exploration differences coincide with unequal spatial opportunities. High extractors encounter an average of 369.55 immediate harvesting opportunities per episode, compared with 9.90 for low extractors, and correspond to agents initialized inside the orchards. Moreover, task-relevant entropy is defined for 99.98% of high-extractor episode–agent records but only 8.42% of low-extractor records, so this comparison is conditional on entering the task-relevant region. Thus, HARVEST reproduces

the broader pattern that collective incentives increase efficiency and behavioral specialization alongside differentiated spatial exploration, while also highlighting that role-specific opportunities can contribute to the observed exploration differences.

E Opportunity-Free Exploration Probe

E.1 Experimental Setting

The opportunity-free probe follows Phase 1 of the GPT-5-mini Cleanup experiment under collective incentives. Agents retain their Phase 1 identities and interaction histories but interact for 20 rounds in an empty environment with no task-specific opportunities or rewards. To control for initial spatial opportunity, each Phase 1 run is independently branched across five starting-position rotations, such that every agent begins once from each position.

Cleaner and harvester roles are defined from Phase 1 behavior using the same 70% contribution threshold as in the main experiment. We measure exploration using spatial entropy, unique cells visited, and stay rate. The final analysis includes 25 Phase 1 runs and 125 probe branches. Metrics are averaged across rotations within each agent and then pooled by role. We report confidence intervals and tests using seed-clustered CR1 standard errors and a t_4 reference distribution.

E.2 Prompt

Agents receive the following exact system instruction, which explicitly supersedes the previous objectives without asking them to explore or maximize coverage: *System prompt.*

```
You are now in a separate 20-round free-movement period. This map contains no apples, dirt,
     or task-specific resource areas, and none will appear. No task rewards are awarded
     during this period. You may freely choose among the available actions.
The earlier apple-collection and cleaning objectives do not apply in this period.
You control only yourself. There are five agents. Keep your identity and earlier history.
The map has 19 rows and 28 columns, with coordinates starting at zero.
You receive the same 7x7 local view format. Outside the map is impassable.
Actions keep their earlier meanings and IDs:
0 MOVE_SOUTH (row +1); 1 MOVE_NORTH (row -1);
2 MOVE_EAST (column +1); 3 MOVE_WEST (column -1);
4 STAY; 5 TURN_RIGHT; 6 TURN_LEFT; 7 CLEAN.
Movement does not change your facing. Turns do not change your position.
CLEAN has no effect on this empty map. Agents can block each other's movement.
Choose exactly one action. Return only JSON: {"actions": ["ACTION_NAME"], "reason": "brief
     explanation"}.
```

Each request also includes the following state-dependent user-prompt format; bracketed fields are replaced with the current values. The original Phase 1 response chain is retained at the start of each branch. *User prompt.*

```
Free-movement round: [round] / 20
You are Agent [agent ID].
Position (row, column): ([row], [column])
Orientation: [orientation name]
Local map (7x7, north is up):
[local ASCII map]
Map symbols: @ = you; 0-4 = other agents; . = empty/walkable; # = outside boundary.
Previous action feedback: [feedback JSON, or "None yet in this period."]
Choose exactly one action and return JSON.
```

E.3 Results

Cleaners show lower spatial entropy than harvesters in the opportunity-free environment (.357 vs. .398; harvester-minus-cleaner $\Delta = .041$, nominal 95% CI $[.017, .064]$, nominal $p = .008$), visit fewer unique cells (13.92 vs. 15.77), and stay more often (24.14% vs. 14.56%). The entropy difference is directionally consistent across most runs: among the 21 Phase-1 runs containing both roles, 18 show higher entropy for harvesters and three show the opposite pattern. These results suggest that exploration differences are not limited to immediate task-specific opportunities, but can generalize to a new environment without those opportunities.