
What Breaks Cooperation Breaks Collusion: Simulating “The Platform”¹

Md Musfiquir Rahman
Purdue University

With
Apart Research

Abstract

LLM agents have recently achieved remarkable results, but harmful behaviors such as deception and collusion have also been documented, and safety-critical incidents have occurred in the real world, for example OpenAI agents gaining unauthorized access to Hugging Face's infrastructure during evaluations. Collusion, in which agents coordinate in ways that benefit themselves but harm outsiders, is therefore a real risk posed by current LLM agents, and we need preventive approaches that make collusion infeasible. In practice, however, simulating collusion is time- and resource-consuming: LLM pricing agents are typically run for hundreds of rounds before collusion settles. In this project, we instead study prosocial coordination and make a key observation: environment designs that break the reward-and-punishment mechanism in prosocial coordination also break it in collusion. To demonstrate how prosocial coordination fails, we simulate a multi-agent environment inspired by the film *The Platform*, in which LLM agents share a limited food supply that descends through levels. Across 240 games with three LLMs, we find that resource scarcity, larger groups and the presence of a greedy agent break prosocial coordination: survival time can drop from more than 10 days in favorable environments to fewer than 5 days in unfavorable ones. Finally, we show an equivalence between the reward-and-punishment mechanisms of *The Platform* and a market, and argue that these designs will break collusion as well.

1. Introduction

This work addresses the problem of preventing collusion among AI agents. The problem is important because, whenever collusion is possible, agents are likely to find a way to achieve it. Existing mitigation approaches, such as monitoring reasoning traces or applying linear probes, are not always effective, and they must be continually revised to handle new types of attacks and behaviors. Moreover, agents may change their behavior once they become aware of the mitigation process. In contrast, designing the agents' environment so that collusion is either impossible or not beneficial

¹ Research conducted at the [AI Collusion Research Sprint](#), October 2026

prevents, or at least mitigates, the problem before it arises. Our goal is therefore to mitigate collusion by identifying which actions or designs break it.

A practical obstacle is that agents may have internalized rules or ethical constraints during training that discourage collusion in general. As a result, collusion cannot be reliably simulated or observed in every scenario.

We therefore take a different route. Collusion is coordination toward an illegal end, so it contains a coordination component. Because the ethics or instructions that agents are trained on may prevent them from engaging in illegal coordination, we instead study settings in which agents should coordinate in order to survive, or to show empathy or kindness. We refer to this as prosocial coordination, and we examine which designs or actions break it. Our central claim is that a design that breaks prosocial coordination will also break collusive coordination (i.e., collusion), provided that both share the same experimental design and environment.

Collusion occurs when multiple agents engage in illegal coordination or collaboration to gain an advantage. It has been studied in pricing games such as Bertrand and Cournot duopolies (Fish et al., 2024; Lin et al., 2024), first- and second-price auctions (Fish et al., 2024; Banchio & Skrzypacz, 2022), two-sided ride-sharing markets (Koirala & Laine, 2024), electricity markets (Seredynski & Tsaousoglou, 2026), EV charging markets (Paudel & Das, 2024), and cooperative tasks (Nakamura et al., 2026). In these settings, agents establish a reward-and-punishment system: agents that cooperate are rewarded, and agents that do not are punished. The system holds when all agents cooperate to obtain the rewards, and fails otherwise.

To study prosocial coordination under the same mechanism, we mimic this setup in an LLM-based multi-agent environment inspired by the film *The Platform* (Figure 1): "In a prison where inmates are fed on a descending platform, those on the upper levels take more than their fair share while those below are left to starve on scraps and one man decides to change the system." The agents must coordinate to survive: the outcome is a success if all agents survive, and a failure otherwise. Because agents are reassigned to levels every day, an agent on an upper level that does not leave food for the lower levels is punished on a later day, when it is placed on a lower level. The same reward-and-punishment mechanism as in collusion is therefore present, but the coordination is prosocial, and LLMs engage in it naturally. Our goal is to identify what breaks this coordination.

Our main contributions are:

1. We find failure cases of collusion by finding failure cases of prosocial coordination, and we argue that a design that breaks prosocial coordination also breaks collusion.



Figure 1: The Platform: food descends through the levels on a platform, and each agent can only take what the agents above have left. Illustration based on the film *The Platform* (Gaztelu-Urrutia, 2019).

2. We identify three designs that empirically break collaboration:
 - Scarcity of resources: collaboration breaks when resources are scarce relative to the number of agents; keeping resources limited therefore helps prevent collusion.
 - Number of agents: collaboration, and thus collusion, is harder to sustain in larger groups.
 - Greedy agent: a single agent that does not follow the rules or the coordination breaks collaboration, and thus collusion.
3. We demonstrate these claims empirically with three large language models and observe that collaboration breaks across different setups.

2. Related Work

Existing approaches, such as Q-learning or deep RL methods (e.g., DQN or PPO), or even competing agents, might collude, but simulating that behavior takes a long time, i.e., many rounds. For example, in a market coalition setup, it takes many rounds for agents to eventually collude. Q-learning pricing agents take on average about 850,000 periods to converge (Calvano et al., 2020), while LLM pricing agents are run for 50 rounds (Lin et al., 2024), 300 rounds (Fish et al., 2024), or up to 1,000 rounds, with collusion settling around round 200 (Keppo et al., 2026). However, collaboration is easier to simulate, and agents naturally tend to collaborate. Collaborative environments are also easier to build or set up than collusive coordination (i.e., collusion), and in these setups, agent awareness is not a problem. We may even encourage agents to collaborate more. Finding designs that make such collaboration fail would work in a collusion setup as well.

3. Methods

Claim. If an experimental design breaks the reward-and-punishment mechanism in a prosocial coordination setup, it also breaks the mechanism in a collusion setup. Formally, the reward-and-punishment mechanism has three features: (i) observability, so deviations can be seen; (ii) future interactions, so deviators can be punished; and (iii) the gain from deviating is small compared to the future loss from punishment. A design breaks the mechanism if it removes at least one of these features. Hence, if a design breaks the mechanism in The Platform, the corresponding design breaks the mechanism in a market.

Proof sketch. We show that each feature in The Platform is equivalent to a feature in the market.

1. **Observability.** The group chat and the public status table correspond to the public order book of the market. An agent that eats more than its share is seen by the others, just as a seller that undercuts the agreed price is seen in the order book.
2. **Future interactions.** The reshuffle after each day corresponds to repeated rounds of the market. An agent that eats more food may be placed at a lower level the next day, where the agents above can punish it by withholding food. Likewise, a seller that undercuts can be punished in later rounds by a price war.
3. **Greedy gain compared to future loss.** The extra food an agent gains by overeating today corresponds to the extra profit a seller gains by undercutting today. The food it loses while being punished corresponds to the profit a seller loses in a price war.

Since the three features are equivalent, a design that removes one of them in The Platform removes the corresponding feature in the market. Moreover, punishment is cheaper in The Platform than in a market: an agent punishes by taking more food, which also benefits it, whereas a price war also costs the punisher. The mechanism is therefore at least as hard to sustain in a market, so if it breaks in The Platform, it also breaks in the market. The converse need not hold.

3.1 Environment: The Platform

We model the game on the film The Platform (Gaztelu-Urrutia, 2019) as a sequential common-pool resource dilemma (Hardin, 1968; Ostrom, 1990) with ordered access: agents near the top can take everything, and agents near the bottom depend on their mercy.

- Agents and levels: N agents live on N levels, one per level; level 1 is the top.
- Food: each day, F units of food start at level 1 and pass down one level at a time. At its level, an agent sees what is left and takes food worth 0–20 units of energy (twice the daily need).
- Energy: every agent starts with $E_0 = 20$ and burns $b = 10$ a day; leftover energy carries over to the next day.
- End of game: we stop the game on the first day any agent dies (its energy falls below 0), and that day is the outcome, T (days survived). Due to time and token constraints, games are capped at 15 days; the agents do not know this.
- Reshuffle: every day, agents are reassigned to levels uniformly at random, so an agent at the top today may be at the bottom tomorrow and be punished by the agents at the top.
- Communication: each agent is a separate LLM call whose system prompt states the rules, its ID and its persona ("You want to survive for as long as possible."); agents are told only that an agent with energy below 0 is removed. Before the food comes down, agents talk in two rounds of group chat in level order, addressing one or more agents; they see a public table of every agent's level, energy, need and past takes, and keep a private note as their memory between days.

3.2 Experiments

We test our three claims (scarcity, group size and greedy agents) across different configurations and average the results over 5 seeds (Table 1).

Table 1: Experiments. Each sweep tests one claim about when collaboration breaks.

Claim	Swept variable	Held fixed	Games
1. Scarcity	Initial food, $F = 30, 35, 40, 45, 50, 60$ (60–120% of total need)	$N = 5$, no greedy agents	30
2. Group size	$N = 3, 5, 8, 12$ with $F = 8N$	Food per agent: 80% of need	20
3. Greedy agents	Number of greedy agents, $k = 0, 1, 2, 3$, at $F = 40$ and $F = 50$	$N = 5$	35

Greedy agents do not coordinate: they always take as much as allowed and say so openly.

4. Results

Models: We run every simulation on three LLMs: gpt-oss-20b (OpenAI, 2025), which reasons before answering, and Qwen3-14B (Yang et al., 2025) and DeepSeek V4.1 Flash (Xu et al., 2026), which do not.

Baselines: We compare against two rule baselines on the same seeds: all agents take the maximum, the worst case (T_{greedy}); and equal share, where every agent takes F/N , the best case (T_{fair}), which requires knowing F . In the greedy conditions, the k greedy agents take the maximum and the rest take F/N .

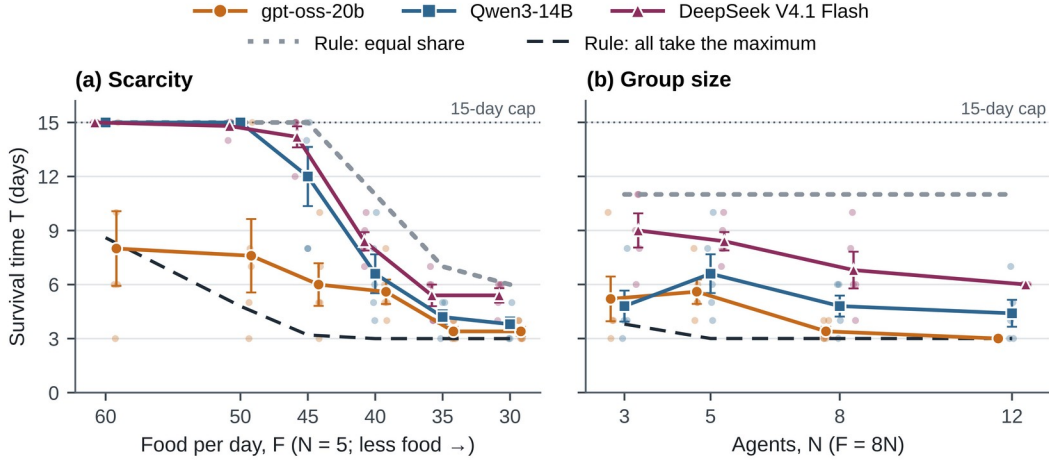


Figure 2: Survival time T for three LLMs under (a) scarcity and (b) group size. Each small dot is one game; large markers are means over 5 games (same level-shuffle seeds for every model) and bars are ± 1 standard error. Games stop at 15 days and count as 15. Dotted gray: rule agents that each take an equal share F/N ; dashed black: rule agents that all take the maximum. (a) Food per day F , decreasing from left to right, with $N = 5$ agents. (b) Group size N , with $F = 8N$ so food per agent stays the same.

4.1 Claim 1: Scarcity Breaks Collaboration

We observe in Figure 2a that when the initial food for $N = 5$ agents decreases from $F = 50$ to $F = 30$, the mean survival time T falls from 7.6 to 3.4 days (gpt-oss-20b), 15+ to 3.8 days (Qwen3-14B) and 14.8 to 5.4 days (DeepSeek V4.1 Flash).

Insight (why?): Scarcity weakens cooperation because it breaks the reward-and-punishment mechanism that sustains the cooperative equilibrium. When resources are scarce, there is not enough food to feed every agent even if all of them cooperate. As a result, agents are better off acting greedily than joining the collaboration. Thus, the scarcer the resource becomes, the shorter the survival time. For collusion, this implies that collusion is less likely to form when there are fewer buyers in a market or fewer resources in the environment.

4.2 Claim 2: Large Groups Break Collaboration

We observe in Figure 2b that when the group grows from $N = 3$ to $N = 12$ agents, with food per agent held fixed ($F = 8N$), the mean survival time T falls from 5.2 to 3.0 days (gpt-oss-20b) and 9.0 to 6.0 days (DeepSeek V4.1 Flash), while Qwen3-14B shows no trend.

Insight (why?): In a large group, it becomes increasingly hard to sustain the reward-and-punishment mechanism for every agent, because doing so requires more complex planning. For collusion, this implies that adding more agents, such as more sellers in a market or more participants in an environment, makes collusion less likely.

4.3 Claim 3: A Few Greedy Agents Break Collaboration

We observe in Figure 3 that when one of $N = 5$ agents is greedy, the mean survival time T at $F = 40$ falls from 5.6 to 3.6 days (gpt-oss-20b), 6.6 to 4.4 days (Qwen3-14B) and 8.4 to 5.0 days (DeepSeek V4.1 Flash); at $F = 50$, it falls from 15+ to 6.0 days (Qwen3-14B) and 14.8 to 10.2 days (DeepSeek V4.1 Flash).

Insight (why?): Greedy agents ignore the reward-and-punishment mechanism and act greedily to obtain the best outcome for themselves. The other agents observe that, despite joining the collaboration, they receive less than the greedy agents, so they break the collaboration. For collusion, this implies that to prevent collusion, we can introduce one agent that acts competitively.

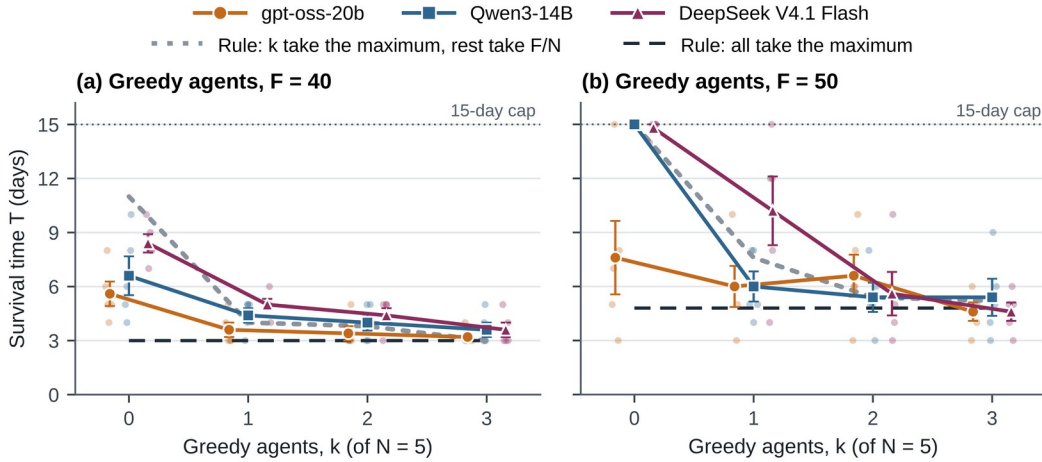


Figure 3: Survival time T for three LLMs with k greedy agents, scripted to always take the maximum, at (a) $F = 40$ and (b) $F = 50$, with $N = 5$. Markers, bars and the 15-day cap are as in Figure 2. Dotted gray: the mechanical baseline, in which the k greedy agents take the maximum and the rest take an equal share F/N ; dashed black: rule agents that all take the maximum.

Appendix A.4 reports the trend tests and robustness checks in full: the scarcity result survives every check, while the group-size result rests on the $N = 12$ games and the greedy-agent result on the first greedy agent. Appendix A.5 gives further detail on each claim.

5. Limitations and Future Work

Our study has two main limitations. First, larger groups also mean longer chats and prompts, so the group-size effect may reflect a context-length limit of the models rather than a social effect. Second, we did not explore broken promises, manipulation through messages, or groups that mix models; agent behaviour might change in these settings.

Future work: we want to (1) estimate the causal effect of each experimental design parameter on survival time, and weigh its cost against its benefit; for example, quantify the causal effect of group size on how long the agents survive; (2) simulate more varied agent behaviour, for example by giving each agent different properties, and add more complex reward-and-punishment mechanisms; (3) find more parameters that break collaboration; and (4) show a direct connection between prosocial coordination and collusion, for example by running the same designs in a market setting.

6. Conclusion

We used a multi-agent LLM game based on The Platform to find designs that break prosocial coordination, as a proxy for designs that break collusion. LLM agents do cooperate, but only partly, and only when they have both the group chat and the public status table. Within that setting, scarcity shortened survival in all three models, larger groups in two of three, and greedy agents in all three. The models also fail in different ways: gpt-oss-20b hoards and calls it a "buffer", Qwen3-14B does not ration, and DeepSeek V4.1 Flash rations most and lasts longest.

References

1. Balliet, D. (2010). Communication and cooperation in social dilemmas: A meta-analytic review. *Journal of Conflict Resolution*, 54(1), 39–57.
2. Banchio, M., & Skrzypacz, A. (2022). Artificial Intelligence and Auction Design. *EC '22*, 11–15.
3. Calvano, E., Calzolari, G., Denicolò, V., & Pastorello, S. (2020). Artificial Intelligence, Algorithmic Pricing, and Collusion. *American Economic Review*, 110(10).
4. Fehr, E., & Gächter, S. (2000). Cooperation and punishment in public goods experiments. *American Economic Review*, 90(4), 980–994.
5. Fischbacher, U., Gächter, S., & Fehr, E. (2001). Are people conditionally cooperative? Evidence from a public goods experiment. *Economics Letters*, 71(3), 397–404.
6. Fish, S., Gonczarowski, Y. A., & Shorrer, R. I. (2024). Algorithmic Collusion by Large Language Models. *arXiv:2404.00806*.
7. Gaztelu-Urrutia, G. (Director). (2019). *El hoyo* [The Platform] [Film].
8. Hardin, G. (1968). The tragedy of the commons. *Science*, 162(3859), 1243–1248.
9. Isaac, R. M., & Walker, J. M. (1988). Group size effects in public goods provision: The voluntary contributions mechanism. *Quarterly Journal of Economics*, 103(1), 179–199.
10. Isaac, R. M., Walker, J. M., & Williams, A. W. (1994). Group size and the voluntary provision of public goods. *Journal of Public Economics*, 54(1), 1–36.
11. Keppo, J., et al. (2026). On the Fragility of AI Agent Collusion. *arXiv:2603.20281*.

12. Koirala, P., & Laine, F. (2024). Algorithmic collusion in a two-sided market: A rideshare example. arXiv:2405.02835.
13. Kwon, W., et al. (2023). Efficient memory management for large language model serving with PagedAttention. SOSP.
14. Lin, R. Y., et al. (2024). Strategic Collusion of LLM Agents: Market Division in Multi-Commodity Competitions. arXiv:2410.00031.
15. Nakamura, M., et al. (2026). Colosseum. arXiv:2602.15198.
16. Olson, M. (1965). The Logic of Collective Action. Harvard University Press.
17. OpenAI (2025). gpt-oss-120b & gpt-oss-20b model card. arXiv:2508.10925.
18. Ostrom, E. (1990). Governing the Commons. Cambridge University Press.
19. Ostrom, E., Walker, J., & Gardner, R. (1992). Covenants with and without a sword: Self-governance is possible. *American Political Science Review*, 86(2), 404–417.
20. Paudel, D., & Das, T. K. (2024). Tacit algorithmic collusion in deep reinforcement learning guided price competition: A study using EV charge pricing game. arXiv:2401.15108.
21. Piatti, G., et al. (2024). Cooperate or collapse: Emergence of sustainable cooperation in a society of LLM agents. NeurIPS.
22. Rutte, C. G., Wilke, H. A. M., & Messick, D. M. (1987). Scarcity or abundance caused by people or the environment as determinants of behavior in the resource dilemma. *Journal of Experimental Social Psychology*, 23(3), 208–216.
23. Sally, D. (1995). Conversation and cooperation in social dilemmas: A meta-analysis of experiments from 1958 to 1992. *Rationality and Society*, 7(1), 58–92.
24. Serebinski, J., & Tsaousoglou, G. (2026). AI agents in algorithmic electricity markets: On the emergence of tacit collusion. arXiv:2608.26896.
25. Xu, A., et al. (2026). DeepSeek-V4.1-Flash: Pushing the Limits of KV Cache Compression. arXiv:2609.19969.
26. Yang, A., et al. (2025). Qwen3 technical report. arXiv:2505.09388.

Appendix

A.1 Game Details and Design Choices

Each day proceeds as follows: levels are reshuffled; agents hold two rounds of group chat; the platform descends and each agent, in level order, takes food and rewrites its private note; energy is updated; and the game ends if any agent's energy is below 0. When food is short ($F < N \cdot b$), equal shares last longest: $T = \lfloor N \cdot E_0 / (N \cdot b - F) \rfloor + 1$. With $N = 5$, this gives 11 days at $F = 40$, 7 at $F = 35$ and 6 at $F = 30$; at $F \geq 50$, equal shares reach the 15-day cap.

Communication is the most reliable way to raise cooperation in human social dilemmas (Sally, 1995; Balliet, 2010; Ostrom, Walker & Gardner, 1992), monitoring of others' use is one of Ostrom's (1990) design principles for lasting commons, and LLM commons studies run a conversation phase before each harvest (Piatti et al., 2024). We therefore provide both the group chat and the public status table; in pilots, only their combination produced sharing (Appendix A.6). We set $E_0 = 20$ to

give agents a two-day buffer: with $E_0 = b = 10$, even all-greedy agents last 2 days, leaving no room to observe cooperation. The persona gives each agent a stake in its own survival (Piatti et al., 2024).

The three sweeps follow prior work on social dilemmas. Scarcity is varied around total need, since harvesting behavior changes with scarcity (Rutte, Wilke & Messick, 1987). Group size is separated from scarcity by fixing food per agent, the standard control (Isaac & Walker, 1988; Isaac, Walker & Williams, 1994) for testing Olson’s (1965) claim that larger groups cooperate less. Free riders make cooperation decay because most people cooperate conditionally (Fehr & Gächter, 2000; Fischbacher, Gächter & Fehr, 2001); the greedy-agent sweep asks how many free riders an LLM group can absorb.

A.2 Prompts and Greedy Agents

Every reply is a single JSON object: `{"to": [...], "message": "..."} for a chat turn and {"reasoning": "...", "take": <number>, "note": "..."} for a take. Agents are instructed: "This is a conversation, not an announcement: address one or more agents by ID and say something to them."`

Greedy agents are scripted to always take the maximum, since prompting alone does not guarantee defection. Their chat messages are still written by the LLM, with the persona "You do not coordinate with the other agents: you always take as much food as you are allowed, and you say so openly in your messages." In addition to the sweeps in Table 1, we test one covert greedy agent at $F = 40$, whose persona instructs it to present itself as cooperative and never reveal that it takes the maximum. The seed fixes the level assignments, so conditions and models are compared on matched draws; LLM sampling is not seeded.

A.3 Models, Serving and Reliability

Table A1: Models and serving.

Model	Why	Serving	Settings
gpt-oss-20b (OpenAI, 2025)	Open weights, reasons before answering	vLLM (Kwon et al., 2023), two RTX 4090s, 64k context	Max 4,000 output tokens (1,000 cut off reasoning); default effort
Qwen3-14B (Yang et al., 2025)	A second family, without reasoning	vLLM, tensor parallel, 40,960 context	Thinking off; max 2,000 tokens
DeepSeek V4.1 Flash (Xu et al., 2026)	A third, larger family without reasoning; tests whether reasoning explains gpt-oss’s group-size effect	Modal endpoint, at most 15 requests in flight	Reasoning off; max 8,000 tokens

The rule baselines reproduce the analytical outcomes. Invalid replies are requested again (up to 4 times), and API errors are retried for up to 10 minutes before the game is aborted and rerun with the same seed, so that an outage is never recorded as a take of 0. All 240 reported games (80 per model, 5 seeds per condition) finished with 0 failed takes.

A.4 Measures, Statistics and Robustness

- **T**: mean days survived over 5 games $\pm$ standard error.
- **Efficiency**: $(T - T_{\text{greedy}}) / (T_{\text{fair}} - T_{\text{greedy}})$, where 0 is no better than all agents taking the maximum and 1 is as good as equal shares.

- **Hoarding:** among cooperating (non-greedy) agents, the share of takes above the daily burn of 10, counted only when more than 10 was on the platform.
- **Need met:** among cooperating agents that started a day below 10 energy, the share that took at least what they needed to survive the day ($10 - \text{energy}$).
- **Trend test:** Spearman's ρ between the swept variable and T over all games of one model, with a two-sided permutation p-value (20,000 shuffles), which makes no distributional assumptions and handles ties and censored values. Scarcity is tested over $F = 30-50$, group size over $N = 3-12$, and greedy agents over $k = 0-3$ at each F.

T indicates whether a group failed; hoarding and need met indicate how. LLM groups do cooperate: on the same seed, they outlasted the all-max rule in 34, 54 and 64 of 80 games (gpt-oss-20b, Qwen3-14B and DeepSeek V4.1 Flash, respectively) and ended sooner in only 4, 5 and 2 (sign test $p < 10^{-6}$ for each model), but fell well short of equal sharing (Figures 2 and 3). Ten of the 12 trend tests hold after Holm correction (Table A2). Each test pools 20 to 25 games, which suffices for large effects but not small ones: with 20 games, $|\rho|$ must exceed about 0.45 for $p < 0.05$, so the two null results (Qwen3-14B group size and gpt-oss-20b greedy agents at $F = 50$) do not rule out moderate effects. The 15-day cap creates ties at the top of the rank test and can only weaken the scarcity trend, not create it. Table A3 reports the robustness checks.

Table A2: Trend tests in full. Spearman ρ between survival T and the swept variable over all games of one model. p is two-sided, from 20,000 random permutations; Holm p corrects for the 12 tests. The 95% interval comes from 5,000 bootstrap resamples of games within each level. Games that reached the 15-day cap count as 15; ties are rank-averaged.

Claim (swept variable)	Model	Games	ρ	95% interval	p	Holm p
Scarcity ($F = 30-50$)	gpt-oss-20b	25	+0.61	+0.29 to +0.86	0.002	0.014
Scarcity ($F = 30-50$)	Qwen3-14B	25	+0.89	+0.83 to +0.95	< 0.001	< 0.001
Scarcity ($F = 30-50$)	DeepSeek V4.1 Flash	25	+0.91	+0.87 to +0.95	< 0.001	< 0.001
Group size ($N = 3-12$, $F = 8N$)	gpt-oss-20b	20	-0.67	-0.87 to -0.41	0.002	0.014
Group size ($N = 3-12$, $F = 8N$)	Qwen3-14B	20	-0.15	-0.56 to +0.29	0.52	0.52
Group size ($N = 3-12$, $F = 8N$)	DeepSeek V4.1 Flash	20	-0.65	-0.87 to -0.38	0.003	0.018
Greedy agents ($k = 0-3$, $F = 40$)	gpt-oss-20b	20	-0.65	-0.83 to -0.44	0.003	0.018
Greedy agents ($k = 0-3$, $F = 40$)	Qwen3-14B	20	-0.61	-0.85 to -0.28	0.006	0.018
Greedy agents ($k = 0-3$, $F = 40$)	DeepSeek V4.1 Flash	20	-0.83	-0.94 to -0.69	< 0.001	0.001
Greedy agents ($k = 0-3$, $F = 50$)	gpt-oss-20b	20	-0.28	-0.65 to +0.14	0.23	0.46
Greedy agents ($k = 0-3$, $F = 50$)	Qwen3-14B	20	-0.64	-0.85 to -0.42	0.004	0.018
Greedy agents ($k = 0-3$, $F = 50$)	DeepSeek V4.1 Flash	20	-0.78	-0.92 to -0.61	< 0.001	0.001

Table A3: Robustness checks. Each row changes one thing about the analysis or the setup and reports whether the trend in Table A2 survives; "holds" means the same sign with $p < 0.05$.

Check	Scarcity	Group size	Greedy agents
Another model	Holds in 3 of 3	Holds in 2 of 3 (not Qwen3-14B)	Holds in 3 of 3 at $F = 40$, 2 of 3 at $F = 50$
Leave one level out	Holds in all 15 cases ($\rho = +0.52$ to $+0.92$)	Without $N = 12$, no longer significant for gpt-oss-20b ($\rho = -0.41$) or DeepSeek V4.1 Flash (-0.44)	Without $k = 0$, holds only for DeepSeek V4.1 Flash: the effect is the first greedy agent
Outcome relative to the rules (efficiency; T minus the mechanical baseline for greedy agents)	Holds for Qwen3-14B only (DeepSeek V4.1 Flash $p = 0.08$, gpt-oss-20b $p = 0.27$)	Holds for gpt-oss-20b ($p = 0.04$) and DeepSeek V4.1 Flash ($p = 0.005$)	Reverses: LLM groups match or beat the mechanical baseline once $k \geq 1$
Add abundance ($F = 60$)	Unchanged ($\rho = +0.58$, $+0.91$, $+0.91$)	—	—
Covert vs. open greedy agent ($F = 40$, $k = 1$)	—	—	No difference: 3.2 vs. 3.6, 5.2 vs. 4.4, 4.2 vs. 5.0 days (exact $p = 0.72, 0.48, 0.17$)

A.5 Further Detail on Each Claim

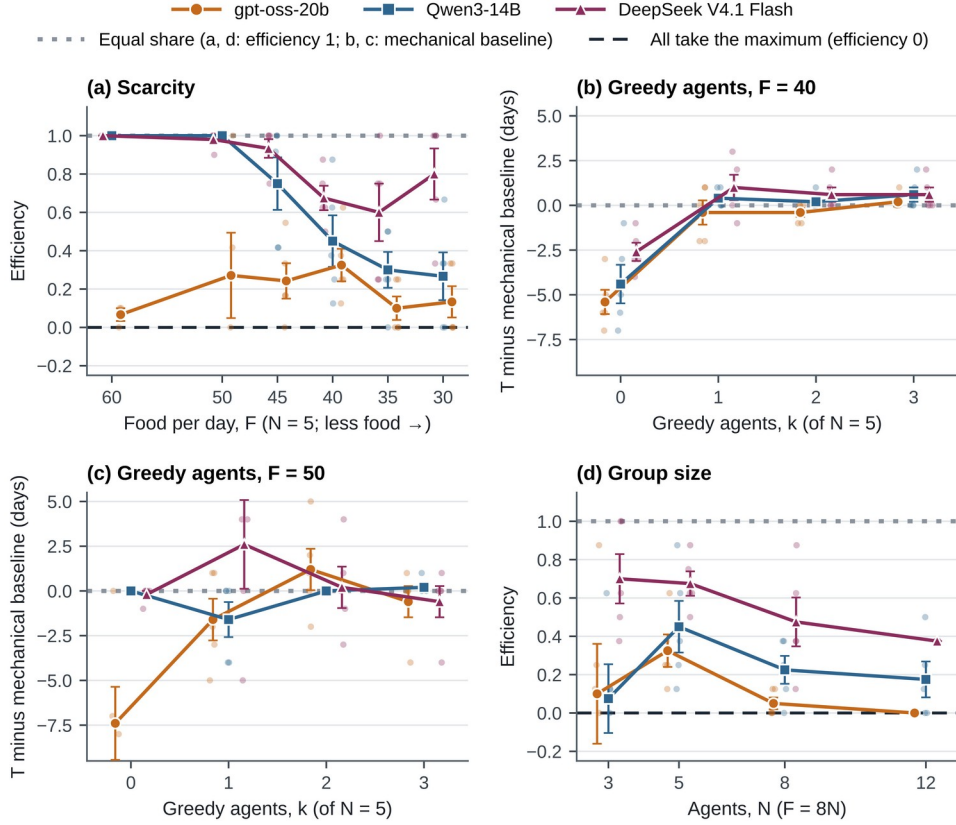


Figure A1: Survival relative to the rule baselines, game by game, on the same seed. (a, d) Efficiency = $(T - T_{\text{greedy}}) / (T_{\text{fair}} - T_{\text{greedy}})$: 0 means no better than all agents taking the maximum, and 1 means as long as equal sharing. At $F \geq 45$, equal sharing reaches the 15-day cap, so efficiency there is a lower bound. (b, c) T minus the mechanical baseline, in which the k greedy agents take the maximum and the rest take F/N ; 0 means the cooperating LLMs did as well as equal-share rule agents would beside the same greedy agents (at $k = 0$ this baseline is plain equal sharing). Means ± 1 SE over 5 games; small dots are single games. Seeds where both rules reach the cap are left out of (a).

Scarcity. Part of the drop in survival is arithmetic, since even equal sharing lasts only 6 days at $F = 30$, against 15+ at $F = 50$. Efficiency, which places T between the two rule baselines, also fell for Qwen3-14B (1.00 to 0.27, $p < 0.001$), so scarcity weakened its cooperation beyond the arithmetic; the trend was weak for DeepSeek V4.1 Flash ($p = 0.08$), and gpt-oss-20b was inefficient at every food level (0.07 to 0.33), leaving little cooperation for scarcity to break (Figure A1a). The share of hungry agent-days on which agents met their need also fell with food (Figure A2a).

Group size. With $F = 8N$, equal sharing lasts 11 days at every N and the all-max rule 3 to 3.8 days, so both baselines are flat and the drop in survival is behavioral. gpt-oss-20b cooperators took more than 10 in 57% of takes at $N = 5$ and 86% at $N = 12$, and DeepSeek V4.1 Flash cooperators in 4% at $N = 3$ and 13% at $N = 12$, whereas Qwen3-14B almost never hoarded (at most 9%) and failed by not rationing, which does not depend on N (Figure A2c). This is consistent with responsibility

thinning out in larger groups, but the hoarding trend is not significant game by game ($p = 0.52$ and 0.15), so we treat it as a likely mechanism rather than a finding.

Greedy agents. A single greedy agent cuts the equal-share ceiling at $F = 40$ from 11 to 4.0 days, and with $k \geq 1$ the LLM groups stayed within -0.4 to $+1.0$ days of this mechanical baseline (Figure A1b). gpt-oss-20b showed no clear trend at $F = 50$ ($p = 0.23$), because it was already close to the all-max rule without greedy agents. The covert greedy agent did no more harm than an open one (Table A3). Two cases show genuine contagion (Figure A2b): DeepSeek V4.1 Flash cooperators at $F = 50$ took more than 10 in 16% of takes with no greedy agents and 70% with three, and gpt-oss-20b cooperators at $F = 40$ rose from 58% to 85% with one.

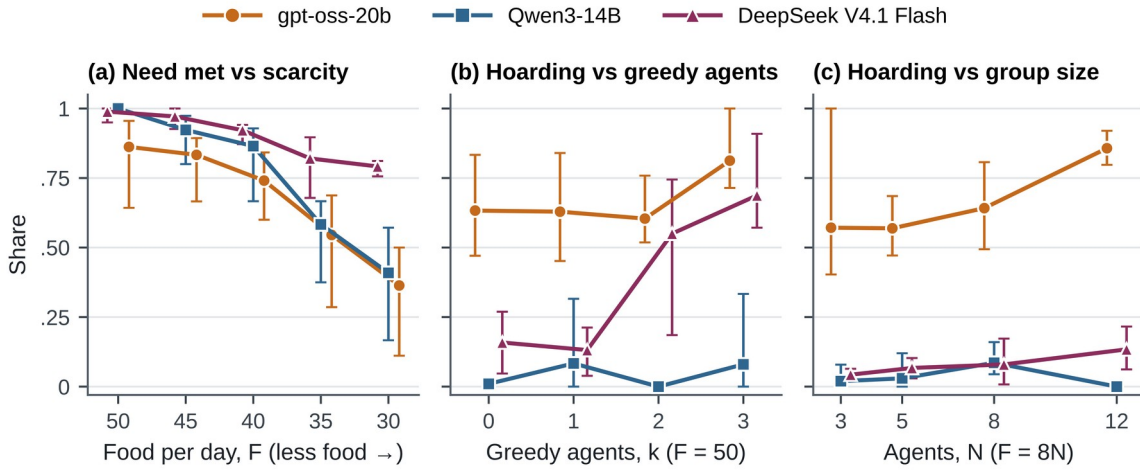


Figure A2: What the cooperating (non-greedy) agents did. (a) Share of hungry agent-days (starting the day below the daily burn of 10) on which the agent took at least what it needed; $F = 60$ is left out because two models had no hungry agents there. (b, c) Hoarding: share of cooperators' takes above 10, among takes where more than 10 was left; (b) uses $F = 50$. Values are pooled over the 5 games per condition; bars are 95% intervals from resampling whole games (4,000 resamples).

A.6 Pilot Experiments

In pilots, the original setting ($E_0 = 10$, reshuffling every 3 days, no persona) ended all 12 games at $T = 2$, identical to the all-greedy outcome. Communication was necessary but not sufficient. In gpt-oss-20b runs with 3 seeds, removing the group chat reduced survival from 10.0 to 5.7 days at $F = 50$ and from 5.7 to 3.3 days at $F = 40$, while chat without the public table was no better than the all-max rule (4, 6, 6 vs. 3, 6, 4 days at $F = 50$; 3, 3, 3 for both at $F = 40$), as agents promised to take 10 and took 20. Chat combined with the table raised survival (8, 15+, 7 days at $F = 50$). At $F = 50$, daily reshuffling also lasted longer than fixed levels (7.6 vs. 4.4 days over 5 seeds). Our claims therefore describe what breaks cooperation once communication and monitoring are in place, consistent with earlier findings on communication in social dilemmas (Sally, 1995; Balliet, 2010). Finally, because every prompt carries the full chat history, token use grows roughly quadratically with game length; DeepSeek V4.1 Flash's 80 games used 114M prompt tokens (\$34.66).

A.7 Assumptions

Table A4: Assumptions behind our reading of the results, why we made each one, and how the reading changes if it does not hold.

Assumption	Why we made it	If it does not hold
The group-size effect is social	Food per agent is fixed across N	Chats and prompts grow with N; if long context hurts the models, the effect is a capacity limit
One prompt wording and persona are representative	Time limits of the sprint	Effect sizes may change with other wordings
Scripted greedy agents	Scripting guarantees defection; prompting alone does not	The results show how agents respond to defection, not whether LLM agents defect
5 seeds per condition are enough	Token cost grows with game length	Seed-to-seed spread is large; differences under about two days are unreliable
Default, unseeded sampling	The same client and prompt for every model	Some differences between models may reflect sampling settings
A 15-day cap	Time and token cost	Survival at high food is a lower bound

A.8 Limitations and Dual-Use / Ethical Considerations

What our results do and do not establish. Our experiments show that resource scarcity, larger groups and greedy agents break prosocial coordination among LLM agents in The Platform. They do not show empirically that the same designs break collusion in a market; that link rests on the argument in Section 3 and remains to be tested. Table A4 lists the further assumptions behind our reading of the results.

Dual-use considerations. Our findings identify designs that weaken coordination among agents. The same knowledge could be used to undermine beneficial cooperation, for example by introducing a greedy agent into a system that relies on sharing. Conversely, the conditions under which coordination survives (communication with monitoring, ample resources and small groups) could inform the design of colluding systems that are harder to break. We consider these risks low, because the levers we study are coarse and already well known in economics, and because our setting involves no real resources, markets or people.

Ethical considerations. No human subjects or personal data were involved; all agents are LLMs in a simulated game. The scripted greedy and covert greedy agents were used only to measure how the other agents respond to defection.

LLM Usage Statement

We used Claude (Anthropic), through Claude Code, to help write and debug the simulation and analysis code, produce the figures, and draft notes on the methods and results. We also used Claude to edit the grammar of our drafts, fit the text to this template, and condense the methods, results and discussion. The research question, the idea of studying prosocial coordination as a proxy for collusion, and the experimental design are our own. All numbers come from the logged games and the analysis scripts, and we reviewed every claim and reference in this report. The LLMs studied in the

experiments (gpt-oss-20b, Qwen3-14B and DeepSeek V4.1 Flash) were the subjects of the research, not writing aids.