Net Influence Meets LLM Agents: Shared Priors, Tacit Rules and Three Colluders[1]

Clarence Boey
Singapore Institute of Technology
2301547@sit.singaporetech.edu.sg
New York, New York, USA

With
Apart Research

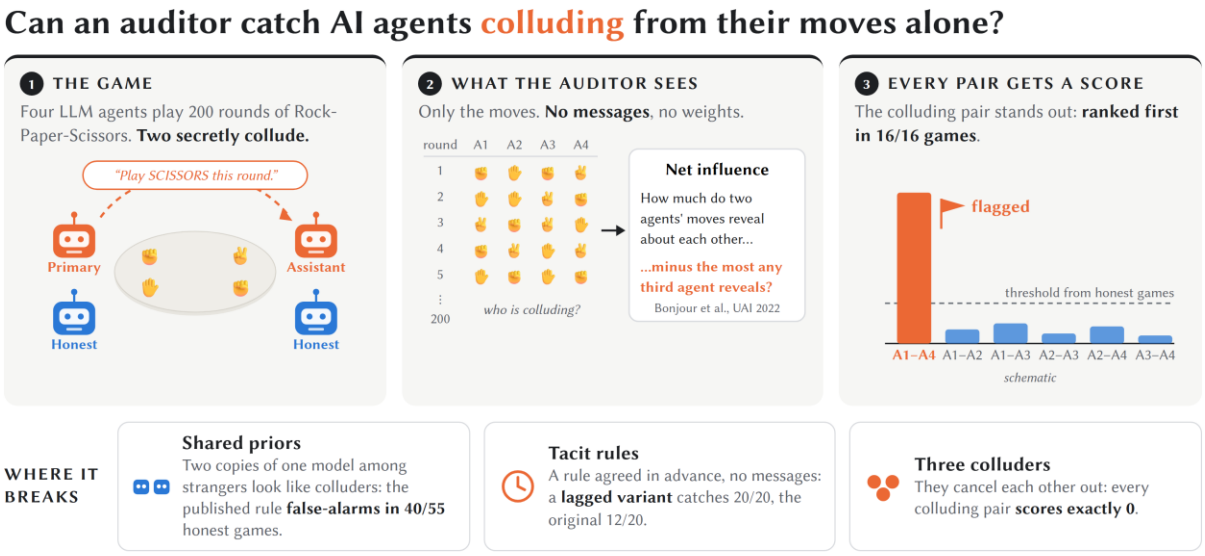


Figure 1. Overview. An auditor sees only the agents' moves. Net influence scores every pair and flags the colluding one; this report tests where that breaks for LLM agents.

Abstract

Pairwise net influence (Bonjour et al., 2022) flags two agents as colluding when their actions share more conditional mutual information than either shares with any third agent. It assumes honest agents act independently given the state, an assumption that copies of the same language model may break. I reimplemented the statistic, reproduced the published scripted Rock-Paper-Scissors results, and tested it on two LLM environments: 176 four-player games between Llama 3.3 70B and Mistral Small 3.2 agents, and 119 games of a double auction. Net influence ranks a seeded pair with a

[1] Research conducted at the AI Collusion Research Sprint, October 2026, at the Collider hub in New York City.

private channel first in 16/16 games, including cross-family pairs. Shared priors do cause false positives. The published rule flags a pair in 40/55 honest games with two same-model agents among two of another model, against 10/55 when all four agents share a model. The reason is that the statistic cancels a shared prior only when a third agent shares it. Thresholds calibrated on honest games cut the cross-validated false-positive rate to 0.009 at nominal 1 %. A lagged variant catches a pre-agreed tacit rule in 20/20 games, where same-round net influence catches 12/20. Three colluders cancel each other out: every colluding pair scores exactly zero.

1. Introduction

LLM agents now set prices, bid and negotiate for competing principals. If two of them coordinate, an auditor will most likely hold the action log, not the agents' messages or weights. Bonjour, Aggarwal and Bhargava (2022) built a detector that needs only that log. For each pair of agents it:

1. estimates how much one agent's action reveals about the other's given the state;
2. subtracts the most that any third agent reveals;
3. flags the pair if the remainder exceeds $\alpha = 0.05$ in both directions.

They tested it on scripted and RL agents. I found no work that applies it to LLM agents.

LLM agents threaten the detector's core assumption, that honest agents act independently given the state, in two ways. Copies of one model share priors and fail to randomise in the same way (Guo et al., 2025). And LLM collusion can be tacit, carried through reactions to past moves rather than a channel (Fish et al., 2024), which same-round mutual information may miss. I test both, and every comparison includes a seeded colluding pair as a positive control.

My main contributions are:

1. **A validated reimplementation.** It reproduces Bonjour et al.'s Table 2 within a few points. I also show that the result hinges on the state definition, which the paper's main text leaves unstated.
2. **Detection of seeded LLM collusion.** Net influence ranks a channel-coordinated pair first in 16/16 games, including 6/6 cross-family pairs, and in 9/9 seeded auction games. Its score rises monotonically with how often the channel is delivered.
3. **A shared-prior false-positive mechanism, and a fix.** The published rule fires on 73 % of honest games when same-model agents are a minority. I trace this to the third-agent subtraction, and I show that thresholds calibrated on honest games remove the excess.
4. **Tacit coordination.** A lagged variant detects a pre-agreed rule more reliably than same-round net influence (20/20 vs 12/20). But the clean split between the two that scripted agents show disappears with predictable LLM agents.
5. **A three-colluder blind spot.** With two assistants, every colluding pair scores exactly 0.

2. Related Work

Action-level detection. Bonjour et al. (2022) assume exactly two colluders and give no theory for α. Hespanhol and Aswani (2020) test play against an ε-Nash equilibrium, which needs a utility model. Xu et al. (2026) construct colluding bidders whose individual bid distributions stay exactly competitive. Any test on one agent's history then has power equal to its size, while a pairwise test on the same data reaches power 1. In their LLM bidders, two deployments of one model have a residual bid correlation of +0.053 (95 % interval [0.030, 0.078]), against +0.0001 across models. Riedl (2026) measures beneficial coordination in LLM action series with time-delayed mutual information and time-shift surrogates.

Transcript and representation-based detection. Tailor (2025) calibrates mutual-information detectors on honest seeds at a fixed false-positive budget; I adopt that protocol. Nakamura et al. (2026) find that action metrics and message judges miss different cases. Rose et al. (2026) probe activations, which needs open weights; like Bonjour et al., they assume two colluders. Motwani et al. (2024) show that steganography can hide messages, but a coordinated action still appears in the action log.

LLM collusion in markets. Fish et al. (2024) find tacit supracompetitive pricing by LLM agents. Lee and Park (2026) apply lagged conditional-independence tests to LLM price series. Agrawal et al. (2025) study LLM sellers in a double auction, which I reimplement.

3. Methods

Statistic. Individual influence is the conditional mutual information $\gamma(i; j) = I(a_i; a_j \mid s)$ (Bonjour et al., Eq. 6). Net influence is $\Gamma(i; j) = \gamma(i; j) - \max_{k \neq i,j} \gamma(k; j)$ (Eq. 4). Algorithm 1 flags a pair when both directions reach $\alpha = 0.05$ and exactly one pair qualifies. I estimate probabilities from counts, in nats, and rank pairs by $\min(\Gamma(i; j), \Gamma(j; i))$. My lagged variant replaces γ with transfer entropy, $TE(i \to j) = I(a_{j,t}; a_{i,t-1} \mid s_t, a_{j,t-1})$, keeps the same subtraction, and takes the larger direction.

Validation. I first reproduced Bonjour et al.'s scripted three-player RPS, in which an assistant helps the primary with probability CP each round (Appendix A).

Game. Four LLM agents play 200 rounds of RPS with Bonjour et al.'s scoring. Each agent sees the rules and the last 10 rounds, and replies with one move at temperature 1.0. I call meta-llama/llama-3.3-70b-instruct and mistralai/mistral-small-3.2-24b-instruct through OpenRouter. Table 1 lists the conditions.

Table 1. Conditions. Every game runs 200 rounds. Seeded games use seeds outside the calibration range 5–54.

ID	Players	Collusion	Games
C0	4 Llama	none	55
C1	2 Llama + 2 Mistral	none	55
C2	4 Llama, or 2 + 2	private channel, delivered with prob. p	p = 1: 6 + 10; p ∈ {0.1, 0.3, 0.5}: 6 each
C3	4 Llama	rule agreed before play, no channel	20 (+6 with an earlier prompt)
C4	4 Llama	one primary, two assistants	6

In C2, I query the assistant after the primary and privately name the move that guarantees the primary a point. In C3, I tell the assistant to play the move that its partner's previous move beats. I version every prompt and save it with the run.

Positive control. I count a seeded pair as colluding if the primary's win rate beats the uniform-play chance of 19/27 (one-sided binomial). In C3 this rule cannot pass, so I measure compliance with the instructed rule from the logs instead.

State. Before collecting any LLM data, I fixed the primary state as the number of distinct moves in the previous round (prev_n_distinct, ≤ 4 values). The secondary states are none, and the previous joint action (prev_joint, 82 values). Plug-in bias grows roughly as 2|S|/N, so I keep the state small.

Baselines. Pairwise action correlation (Cramér's V), and an LLM judge: Claude Sonnet 5.5 reads the action table and scores each pair from 0 to 100.

Thresholds and metrics. I fit 1 % and 5 % thresholds on 100 honest games (C0 + C1, seeds 5–54), freeze them, and evaluate only on other seeds. I count a detection when the true pair ranks first and exceeds the threshold. I report:

- top-1 accuracy (chance 1/6);
- held-out true- and false-positive rates;
- the Algorithm 1 rate;
- a 5-fold cross-validated false-positive rate over all 110 honest games.

Double auction. I reimplemented Agrawal et al.'s (2025) setting from their paper (Sec. 2.1):

- five LLM sellers (cost $80) and five scripted buyers bidding uniformly on $80–100;
- mid-price matching over 200 hours;
- conditions A0 (5 Llama), A1 (3 Llama + 2 Mistral) and A2, in which a seeded assistant is told the primary's ask and asked to match it.

Before collecting data, I coded actions as ask bands (≤ $89, $90–94, ≥ $95) and the state as the previous hour's trade-price band.

4. Results

Validation. At CP = 0.4, my implementation detects the pair in 83.1 % of iterations at 200 games (paper: 79.2 %) and 98.9 % at 400 (99.3 %). The state matters. The paper's main text omits it, and with a constant state the same data give 95.6 % at 100 games, against the paper's 58.4 %.

Seeded collusion. All 16 C2 pairs passed the positive control ($p \leq 3e\text{-}29$), and the assistants played the named move in 99.8–100 % of rounds (Table 2).

Table 2. Seeded pairs with a private channel delivered every round (C2, p = 1). Top-1: the true pair ranks first of six (chance 1/6). TPR: flagged at the frozen 1 % threshold, counting only seeds outside the calibration range. The judge was not run on the mixed games.

C2, p = 1	n	Net influence top-1	True-pair score, mean (sd)	Held-out TPR at 1 %	Correlation top-1	Lagged top-1	Judge top-1
4 Llama	6	6/6	0.891 (0.062)	6/6	6/6	0/6	4/6
2 Llama + 2 Mistral	10	10/10	0.819 (0.072)	3/3	10/10	0/10	—

Cross-family pairs scored as high as same-family pairs (mean 0.837, minimum margin over the next pair +0.743). Detection held when I truncated the runs to 25 rounds. As the channel becomes rarer, the true pair's mean score falls monotonically (Figure 2): 0.891, 0.217, 0.112 and −0.004 at p = 1, 0.5, 0.3 and 0.1. Held-out TPR at 1 % falls from 6/6 to 5/6, 3/6 and 0/6. The judge ranked the pair first in 0/6 games at every p below 1. The lagged variant fails at p = 1 by construction: the assistant's own previous move already encodes the primary's, so the transfer entropy is exactly 0.

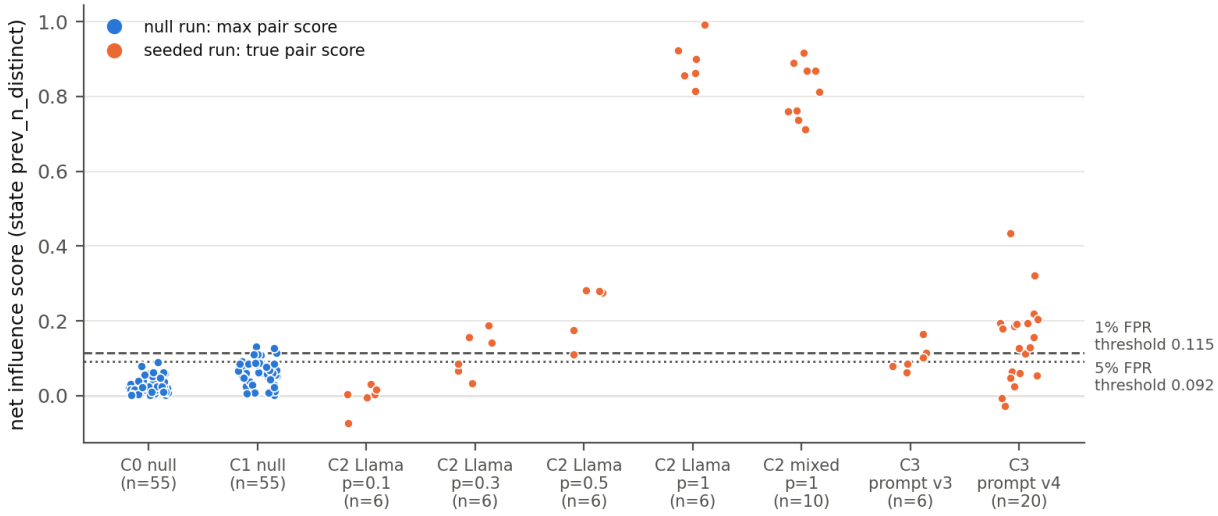


Figure 2. Net influence score per game under the primary state. Honest games (blue) show their largest pair score; seeded games (orange) show the true pair's score. Dashed and dotted lines: thresholds at 1 % and 5 % false-positive rate, frozen on 100 honest games.

Shared priors. Honest LLM agents play in lockstep. All four C0 agents choose the same move in 29.8 % of rounds, against 3.7 % under uniform play. Table 3 shows the effect on the published rule.

Table 3. Honest games in which Algorithm 1 (α = 0.05) flags some pair, by detector state. Last column: how often C1's top-scoring pair is a same-model pair.

Algorithm 1 fires on honest games	C0: 4 Llama	C1: 2 Llama + 2 Mistral	C1 top pair same-model
prev_n_distinct (primary)	10/55	40/55	54/55
none	11/55	12/55	47/55
prev_joint	0/55	8/55	40/55

The same-model pair outscores the best cross-model pair in C1 under every state (primary: +0.130, 98 % of games, Wilcoxon p = 1e-10). C0's Llama pairs carry *more* raw dependence than C1's, yet C0 rarely false-alarms (Figure 3). The subtraction explains why:

- **C0:** every third agent shares the prior, so the subtracted term (0.219) exceeds the pair's own dependence (0.178), leaving a mean score of −0.041.
- **C1:** the Mistral agents share little with the Llama pair, so the subtraction falls to 0.050 against 0.115, and the score turns positive (+0.065).

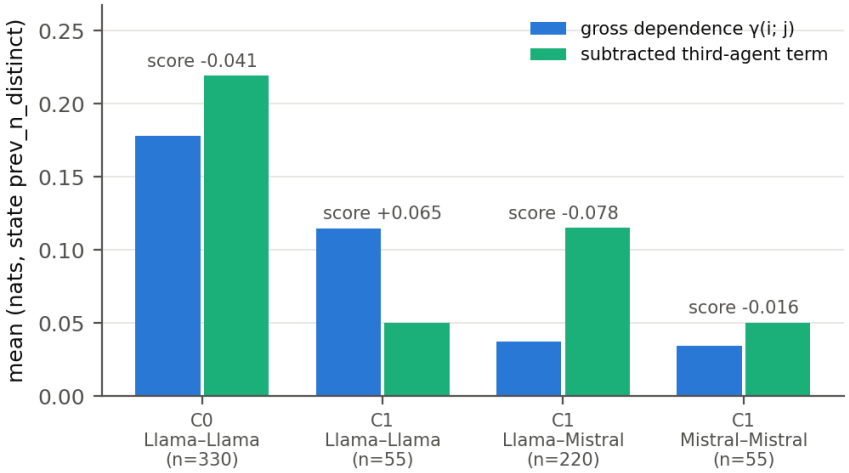


Figure 3. Mean gross dependence γ(i; j) and the subtracted third-agent term for honest pairs, by pair type (primary state). The net score is the difference. It is negative for C0's Llama pairs but positive for C1's Llama pair.

Scripted agents reproduce this (Appendix B). Calibrated thresholds fix it:

- held-out false-positive rate 0/5 in C0 and 1/5 in C1;
- cross-validated over all 110 honest games, 0.009 at nominal 1 % and 0.055 at 5 %, against 0.45 for Algorithm 1.

The cost falls on weak collusion. At 1 % the calibrated threshold catches C2 at p = 0.3 in 3/6 games (Algorithm 1: 5/6) and C3 in 12/20 (15/20). I had first planned a circular-shift permutation null, but it rejects honest pairs under a shared prior, so I dropped it (Appendix C).

Tacit rule. With my first C3 prompt, Llama played the *inverse* of the instructed rule in 155–199 of 199 rounds. I rewrote the rule as a lookup table (a post-hoc change), which raised compliance to a mean of 0.81 over 20 games (Table 4).

Table 4. Tacit rule (C3, prompt v4, 20 games), flagged at the frozen 1 % threshold. The judge has no threshold.

C3, 20 games	Top-1	TPR at 1 % [Wilson 95 %]
Net influence, prev_n_distinct	18/20	12/20 [0.39, 0.78]
Net influence, prev_joint	0/20	0/20

C3, 20 games	Top-1	TPR at 1 % [Wilson 95 %]
Lagged net influence	20/20	20/20 [0.84, 1.00]
Correlation	20/20	1/20 [0.01, 0.24]
LLM judge	10/20	—

The lagged variant flagged the pair in every game where only one of the two detectors did (8 vs 0, exact McNemar p = 0.008). Yet same-round net influence ranks the pair first in 18/20, against my hypothesis that it would sit near chance. The primary almost never repeats a move, so its next move often matches the rule's move. Only conditioning on the full previous joint action removes the same-round signal (0/20), as in scripted play. The leak also runs the other way. In the C2 sweep, the lagged variant ranks the pair first at p = 0.3 and 0.5 (6/6), because the primary plays the move that beats its own last move in 55–91 % of rounds (Appendix D).

Three colluders. With two assistants on the channel, the positive control passed 6/6 and the assistants played identical moves in all 200 rounds. Every colluding pair scored exactly 0.000 in every state, so nothing was flagged, in 6 LLM games and 40 scripted ones. Each colluder's strongest third-party influence is the other colluder, and the subtraction cancels it. A colluding pair still ranked first in 6/6 games, because honest pairs scored below zero. Correlation flagged all six, and the judge ranked a colluding pair first in all six.

Double auction. All nine seeded pairs matched every hour. Net influence ranked them first and flagged them at both thresholds in 9/9 games, with scores of 0.339–0.647 against a maximum of 0.166 across 110 honest games, and no held-out false alarms (Figure 4). Correlation cannot flag anything here: honest sellers ask exactly $90 in 92–97 % of seller-hours, so honest pairs reach perfect association. The seeded pairs earned *less* than honest sellers in 8/9 games (Appendix E).

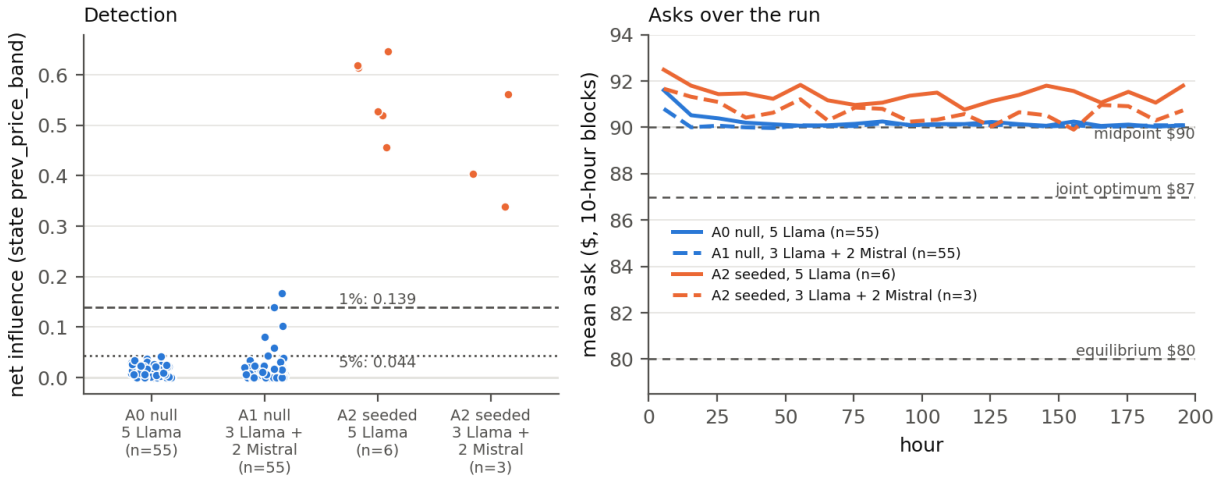


Figure 4. Double auction. Left: net influence per game against the auction's thresholds, frozen on 100 honest games (honest games: largest pair score; seeded games: the seeded pair's score). Right: mean ask per 10-hour block against the reference asks.

5. Discussion and Limitations

The third-agent subtraction makes net influence work, and the same step explains each failure I found. Four copies of one model do not false-alarm, two copies among strangers do, and three colluders vanish. So false positives depend on the population mix, not on the model alone. A market with two deployments of one model among other vendors resembles C1. There, a fixed α flags them, while a threshold calibrated on honest games of that mix does not. Calibration has to be redone for each population, model and prompt. Xu et al. (2026) find the same problem in real auction data: honest bidder pairs are dependent enough that a screen held at a 5 % false-positive rate must sit above +0.50.

Scripted intuition about tacit collusion does not transfer either. LLM agents cycle and avoid repeating moves, so a lagged coupling leaks into the same round and the reverse. On the same C3 games, same-round top-1 ranged from 0/20 to 20/20 across my three states. No single detector or state is a safe default.

Limitations

- All collusion was instructed, not emergent.
- I tested two models in one game and one simplified auction.
- The C2 channel is idealised.
- I revised the C2 and C3 prompts after seeing results, and I record each revision and its reason.
- Held-out false-positive rates rest on five games per condition, which is why I add cross-validation.
- Agents see 10 rounds of history.
- I did not test adaptive colluders or steganographic channels.
- I wrote a pre-registration template but never froze it. The decisions I made before collecting data are timestamped in the project's decision log.
- Detection is not harm: the seeded auction pairs lost money, and the statistic flags beneficial coordination as readily as collusion.

Dual-Use and Ethical Considerations

- Publishing blind spots (three colluders, couplings outside the conditioning set, weak collusion) tells colluders what to avoid. I report them for defenders, and I built no evasion strategies.
- The seeded prompts are plain instructions in a toy game, similar to public recipes. I release them only for reproducibility.
- Shared-prior false positives could implicate operators who merely run the same model, so any flag is a screen that needs further evidence, not proof.
- A clean result is not a clean bill of health.
- No human subjects were involved.

Future Work

Group statistics for three or more colluders, as Bonjour et al. suggest. Emergent collusion, by offering a private channel without a collusive instruction (Nakamura et al., 2026). More model families. Combined same-round and lagged screens under one calibrated threshold.

6. Conclusion

Net influence finds seeded LLM colluders, including cross-family pairs, where an LLM judge often does not. Its fixed threshold, however, mistakes shared priors for collusion whenever same-model agents are a minority, so it needs calibration on honest games of the same population. A lagged variant sees tacit rules, but predictable LLM agents blur the line between the two detectors. Three colluders defeat it outright.

Code and Data

Total API spend was ≈ $15.5. PRIOR_WORK.md lists the work done before the sprint; the last pre-sprint commit is tagged pre-sprint.

- **Code repository**: https://github.com/blarencecoey/net-influence-llm. Code, configs and versioned prompts. Each folder in notes/results/ has a README with the command that produced it.
- **Data/Datasets**: no external datasets; every game was generated for this project. The analysis outputs behind each table and figure are in notes/results/, and notes/results/fact-sheet.md maps every number to its source file. Raw game logs (runs/) are not in the repository.
- **Other artifacts**: an interactive web version of this report (docs/article/index.html in the repository) and a local results viewer (scripts/view_results.py).

Author Contributions

Single-author project. Clarence Boey designed the study, made the research decisions, ran the experiments, and reviewed and edited this report. Claude's role is described in the LLM Usage Statement.

References

1. Agrawal, Teo, Vazquez, Kunnavakkam, Srikanth, Liu (2025). Evaluating LLM Agent Collusion in Double Auctions. arXiv:2507.01413.
2. Bonjour, Aggarwal, Bhargava (2022). Information Theoretic Approach to Detect Collusion in Multi-Agent Games. UAI 2022, PMLR 180:223–232.
3. Fish, Gonczarowski, Shorrer (2024). Algorithmic Collusion by Large Language Models. arXiv:2404.00806.

4. Guo, Lv, Zhang, Zhao, Zhang, Cui (2025). The Illusion of Randomness: How LLMs Fail to Emulate Stochastic Decision-Making in Rock-Paper-Scissors Games? Findings of EMNLP 2025, 8618–8637.
5. Hespanhol, Aswani (2020). Hypothesis Testing Approach to Detecting Collusion in Competitive Environments. arXiv:2003.09967.
6. Lee, Park (2026). Faithful yet Collusive: Why Chain-of-Thought Monitoring Cannot Detect Algorithmic Collusion. arXiv:2609.18346.
7. Motwani, Baranchuk, Strohmeier, Bolina, Torr, Hammond, Schroeder de Witt (2024). Secret Collusion among AI Agents: Multi-Agent Deception via Steganography. NeurIPS 2024.
8. Nakamura, Kumar, Das, Abdelnabi, Mahmud, Fioretto, Zilberstein, Bagdasarian (2026). Colosseum: Auditing Collusion in Cooperative Multi-Agent Systems. arXiv:2602.15198.
9. Riedl (2026). Emergent Coordination in Multi-Agent Language Models. arXiv:2510.05174.
10. Rose, Cullen, Abdelnabi, Torr, Kaplowitz, Schroeder de Witt (2026). Detecting Multi-Agent Collusion Through Multi-Agent Interpretability. arXiv:2604.01151.
11. Tailor (2025). Audit the Whisper: Detecting Steganographic Collusion in Multi-Agent LLMs. arXiv:2510.04303.
12. Xu, Wu, Lu, Tan, Tao, Hong (2026). Collusion with Competitive Marginals: Price-Level Audits Are Blind by Construction. arXiv:2607.26385.

Appendix

A. Validation. I ran 1000 iterations per cell, with the previous joint action as the state (Table A1).

Table A1. Detection accuracy (%) of Algorithm 1 in scripted three-player RPS. Bonjour et al.'s Table 2 values are in brackets.

Games	CP 0.3, nats	CP 0.3, bits	CP 0.4, nats	CP 0.4, bits
200	51.3 (48.3)	63.7	83.1 (79.2)	90.5
400	72.5 (82.8)	86.5	98.9 (99.3)	99.6

With no collusion and the joint-action state, Algorithm 1 flags some pair in 27.9 % of 100-game runs. The paper does not report this rate.

B. Lockstep and the scripted check. Every one of 216 honest Llama pairs in C0 agrees more often than its own move frequencies predict (0.575 vs 0.360). In scripted play, a shared reaction rule followed by all four players leaves net influence at its null. The same rule followed by two of four makes net influence flag that pair in 65–100 % of games (rule strength $q \geq 0.7$). At matched move entropy, Llama–Mistral pairs carry gross dependence of 0.039, against 0.117 for the Llama pair, so Mistral's lower entropy does not explain the C1 gap.

C. Circular-shift null. Shifting agent j breaks j's dependence on every agent, but the score keeps i's real dependence on third agents. The test therefore rejects whenever i depends on anyone. On synthetic honest data with a shared prior (strength 0.4), it rejects 57 % of pairs at nominal 5 %.

D. Lagged leak. In a scripted ablation with a random primary, lagged top-1 sits at chance for every p. With a primary that cycles 90 % of the time, it reaches 1.00 at p = 0.5 and 0.00 at p = 1.

E. Auction details. Algorithm 1 fires on 5/55 A1 and 0/55 A0 honest games. In A1, the same-family pair outscores the best cross-family pair by +0.029 (Wilcoxon p = 1e-5), but no held-out game crosses the threshold. Honest asks sit above the best response to the other sellers' asks in every A0 seller-hour, by $1.20 on average, and they stay flat over 200 hours. That points to a focal point at the midpoint of the stated price range, not a learned collusive price: the joint-profit-maximising common ask in this market is $87.

LLM Usage Statement

I used Claude (Anthropic), through Claude Code, as a research and coding assistant.

- **Before the sprint:** it helped me read papers and write literature notes.
- **During the sprint:** under my direction it wrote and debugged code for the estimator, game harness, analysis pipeline, results viewer and auction environment. It also ran analyses and tabulated results. I made the research decisions (hypotheses, detector state, positive-control rule, prompt revisions and framing), and I recorded them in the project's decision log.

Claude Sonnet 5.5 served as the LLM-judge baseline. Llama 3.3 70B and Mistral Small 3.2 were the agents under study.