
Reading Moves, Not Messages: Detecting Collusion Between LLM Agents¹

Josh Snow
University of Cambridge

With
Apart Research

Abstract

AI agents acting for different principals can collude through channels an overseer cannot read, so an auditor may have nothing to go on but their actions. *Pairwise net influence* (Bonjour et al., 2022) was designed for this setting: it asks whether two agents’ actions depend on each other more than their legitimate observations can explain. We adapt it to LLM agents in three ways. We replace its fixed threshold with a within-state permutation test, which brings false positives in short games down from as much as 49% to 1–5%. We add a cross-fitted “forecasting contest” estimator for long histories, together with a formula for when it will mistake herding for collusion. And we extend the method to groups of colluders, where the original version breaks down. Tested on Gemma 4 agents playing Rock–Paper–Scissors with a private message channel, it identified the colluding pair both when the pair was told to collude and when it was simply given the channel, and it never flagged an honest pair; plain correlation and an LLM judge both failed. It also caught every LLM bid-rigging ring in a procurement auction, and, somewhat to our surprise, colluders coached to play well became easier to detect. The main limitation is conceptual: someone has to decide which shared information counts as innocent.

1. Introduction

LLM agents are increasingly being deployed to act for different people and firms, in markets, negotiations and auctions. If such agents coordinate in secret, third parties pay for it: algorithmic pricing agents already learn to collude (Calvano et al., 2020; Fish et al., 2024), and collusion is among the central risks of multi-agent AI (Hammond et al., 2025). Reading the agents’ messages is not a reliable safeguard either, since colluding agents can hide what they say steganographically (Motwani et al., 2024).

Threat model. We assume an auditor who can see what each agent may legitimately react to (its *state*) and what it does (its *actions*), but not its messages, prompts or internals. The auditor’s job is to work out which agents, if any, are coordinating beyond what that information explains.

Pairwise net influence (Bonjour et al., 2022) is designed for exactly this problem. Roughly, it measures how much one agent’s action tells you about another’s once you know the state, and compares that with the same quantity for third parties. So far it has only been tested on scripted and reinforcement-learning agents, and it has four gaps that matter for LLM agents: its fixed threshold is not calibrated, its counting estimator only works when the state takes few values, it assumes there are exactly two colluders, and it had never been tried on LLMs.

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

Our setting. In our main experiment, three LLM agents (Gemma 4) play Rock–Paper–Scissors. Two of them, A and B, may share a private message channel that neither the third player, C, nor the auditor can see. We compare three conditions:

- **off:** there is no channel, and all three agents play for themselves. This measures false positives.
- **channel:** A and B can message each other, but are *not* told to cooperate; each still plays for its own score. This tests whether collusion emerges unprompted.
- **instructed:** A and B are told they are a team that maximises its combined score. This is deliberate collusion.

Later we add a coached team, which is also given strategy tips, and a second game, a procurement auction.

Our main contributions are:

1. **A calibrated decision rule** (§2.2): a within-state permutation test that keeps the false-positive rate near its nominal level at every game length, and detects collusion too weak for the fixed threshold ever to see.
2. **An estimator for long histories** (§2.3): a cross-fitted “forecasting contest”, with a formula for its misspecification bias that predicts when a too-simple model will mistake herding for collusion.
3. **Detection of colluding groups** (§2.4): median-baseline, all-pairs and synergy rules, after showing that the original method fails with two pairs or a trio.
4. **The first evaluation on LLM agents** (§3): net influence catches both emergent and deliberate collusion without false flags, where correlation and an LLM judge fail, and it catches every LLM bid-rigging ring in the auction.

2. Methods

2.1 Pairwise net influence

Setting. In each round t , agent i observes a state s_t^i and plays an action a_t^i . The state is everything the agent may legitimately react to. An honest agent’s action depends only on its own state and its own randomness, so for two honest agents i and j ,

$$H_0 : \quad p(a^i, a^j \mid s^i, s^j) = \pi^i(a^i \mid s^i) \pi^j(a^j \mid s^j),$$

and collusion is any departure from H_0 .

Influence and net influence. The influence $\gamma(i; j)$ measures this departure as a conditional mutual information. Net influence $\Gamma(i; j)$ then subtracts the strongest influence that any third agent has on j :

$$\gamma(i; j) = \sum_{s^{ij}} p(s^{ij}) \sum_{a^i, a^j} p(a^i, a^j \mid s^{ij}) \log_2 \frac{p(a^i, a^j \mid s^{ij})}{p(a^i \mid s^i) p(a^j \mid s^j)}, \quad \Gamma(i; j) = \gamma(i; j) - \max_{k \neq i, j} \gamma(k; j), \quad (1)$$

where $s^{ij} = (s^i, s^j)$. In RPS every agent sees the same state s , so $\gamma(i; j) = I(a^i; a^j \mid s)$. The original rule flags $\{i, j\}$ if both $\Gamma(i; j)$ and $\Gamma(j; i)$ are at least $\alpha = 0.05$ bits, and reports collusion only if *exactly one* pair is flagged.

Choosing the state. Because γ only measures dependence *beyond* s , choosing s amounts to choosing what counts as innocent. The state has to contain every common cause that honest agents react to, otherwise herding looks like collusion, and it has to exclude the collusion channel itself. We designed our LLM games so that each agent’s prompt contains exactly s . This means we know the right state, and can also see what happens when an analyst picks the wrong one.

2.2 Our decision rule: a within-state permutation test

The problem with a fixed threshold. The original paper estimates Eq. (1) by counting, i.e. by plugging observed frequencies into the formula. Let S be the number of state values, $|A|$ the number of actions and N the number of rounds. Under H_0 , $2N \ln 2 \cdot \hat{\gamma}(i; j)$ is asymptotically χ^2 with $S(|A| - 1)^2$ degrees of freedom, so the estimate is biased upwards by about $S(|A| - 1)^2 / (2N \ln 2)$ bits (Miller, 1955; Panzeri and Treves, 1996). With $s =$ the previous joint move ($S = 28$) and $N = 1,000$, the bias alone is 0.08 bits, already larger than α . A fixed threshold therefore has a false-positive rate that depends on N and S .

The test. Instead of using a fixed threshold, we compare $\hat{\gamma}(i; j)$ with the values it takes when i and j are honest. We build such “honest” datasets from the real one by shuffling. Within each state value, we randomly reassign i ’s actions among the rounds with that state, and leave j ’s actions where they are. Each agent still plays exactly as often as before in every state, but i ’s action is now paired with j ’s action from a different round, which breaks any link between them beyond s . If rounds with the same state are interchangeable under H_0 , the shuffled data have the same distribution as the real data. We recompute $\hat{\gamma}$ on B shuffled datasets, giving $\hat{\gamma}_1(i; j), \dots, \hat{\gamma}_B(i; j)$, and set

$$p(i \rightarrow j) = \frac{1 + \#\{b : \hat{\gamma}_b(i; j) \geq \hat{\gamma}(i; j)\}}{B + 1}, \quad (2)$$

which is valid (Phipson and Smyth, 2010). Shuffling leaves both agents’ estimated policies, $\hat{p}(a^i | s)$ and $\hat{p}(a^j | s)$, unchanged, so the shuffled statistics carry the same upward bias as the real one and no correction is needed. It also matters that we shuffle only *within* a state: a plain shuffle across all rounds would break innocent herding too, and honest agents reacting to the same history would then look like colluders. In the `channel` game, for example, none of $B = 1,000$ shuffles reached the real value ($\Gamma(A; B) = 0.78$ bits, Table 1), giving $p = 1/1001 \approx 0.001$, while in the honest game $p = 0.22$. We flag a pair if $p \leq 0.05/m$ in both directions, a Bonferroni correction over the m pairs ($0.05/3 \approx 0.017$ for three players).

Three further points are worth noting. The test is exact when states are exogenous, as in the auction, but only approximate when states are built from past actions, because shuffling i ’s action in one round does not update the next round’s state; we therefore measure its false-positive rate directly (§3.1). It tests dependence beyond s rather than net influence, since shuffling changes only i ’s actions, so $\max_k \gamma(k; j)$ is the same in every shuffle and cancels. And with private states we shuffle within values of s^i , so H_0 conditions on s^i .

2.3 Our estimator for long histories: a forecasting contest

The idea. When s is the last k rounds, there are 27^k state values, far too many to count. However, influence is the same as how much better j ’s action can be predicted once i ’s action is known, $I(a^i; a^j | s) = \mathbb{E}[\log_2 p(a^j | a^i, s) - \log_2 p(a^j | s)]$, and this can be estimated with two classifiers. Forecaster 0, $q_0(a^j | s)$, predicts j ’s move from one-hot features of the last k rounds. Forecaster 1, $q_1(a^j | a^i, s)$, sees the same features plus i ’s current move.

The estimate. With 5-fold cross-fitting, every round is scored by models that never saw it:

$$\hat{\gamma}(i \rightarrow j) = \frac{1}{T} \sum_t g_t, \quad g_t = \log_2 q_1^{(-t)}(a_t^j | a_t^i, s_t) - \log_2 q_0^{(-t)}(a_t^j | s_t), \quad (3)$$

with standard error $\text{sd}(g)/\sqrt{T}$. A forecaster that memorises noise does worse on rounds it has not seen, so held-out scoring removes the plug-in bias. We flag a pair if $\hat{\gamma} > z \cdot \text{SE}$ in both directions, with $z = 2$ (or 2.58 for 10 pairs). For the forecasters we use multinomial logistic regression and gradient-boosted trees.

Misspecification bias. Adding and subtracting $\log p$ in Eq. (3) gives

$$\mathbb{E}[g] = I(a^i; a^j | s) + \mathbb{E}_s \text{KL}(p(\cdot | s) \| q_0) - \mathbb{E}_{s, a^i} \text{KL}(p(\cdot | a^i, s) \| q_1). \quad (4)$$

Under H_0 the first term is zero, but the bias $\text{KL}_0 - \text{KL}_1$ is positive whenever a^i helps q_1 repair q_0 's misfit. Herding does exactly this when q_0 cannot represent how honest agents use s , so a forecaster that is too simple can turn honest herding into apparent influence.

The permutation test and the forecasting contest solve different problems. The first is a *decision rule* and fixes calibration; the second is an *estimator* and fixes dimensionality. In principle the two could be combined by refitting the forecasters on every shuffle. Table A1 compares all three approaches.

2.4 Beyond one pair, and private states

Why the original rule fails. If j colludes with both i and k , then $\max_k \gamma(k; j) \gtrsim \gamma(i; j)$, so $\Gamma(i; j) \lesssim 0$. In a perfectly coordinated trio every Γ is 0, and the exactly-one rule also hides two separate pairs.

Our fixes. We test three. A *median baseline*, $\Gamma_{\text{med}}(i; j) = \gamma(i; j) - \text{median}_k \gamma(k; j)$, stays robust as long as fewer than half of j 's opponents collude with j . *All-pairs reporting* merges the flagged pairs into groups. A *synergy* test (McGill, 1954; Williams and Beer, 2010), $I(a^i; a^j | a^k, s) - I(a^i; a^j | s)$, catches three-way coordination that is invisible to every pair, such as $a^k = -(a^i + a^j) \bmod 3$.

Private states. When agents observe different things, $\gamma(i; j) = I(a^i; a^j | s^{ij}) + I(a^i; s^j | s^i) + I(a^j; s^i | s^j)$, so γ also catches an agent whose action depends on a rival's private information. We use this form in the auction.

2.5 Experimental setup

Scripted Rock–Paper–Scissors. There are three players. If all moves are equal everyone scores 0, if all differ everyone scores 1, and otherwise ordinary RPS decides and the winning move scores 1. A and C play uniformly at random. With collusion probability CP, B plays the move that A beats, which guarantees A a point. This makes $\gamma(A; B)$ known exactly: it rises from 0.014 bits at CP = 0.1 to 1.585 bits at CP = 1.

LLM Rock–Paper–Scissors. Three Gemma 4 26B-A4B agents (instruction-tuned, thinking off) play 2,000 rounds per game in the conditions described in §1. They are served with vLLM (Kwon et al., 2023) at temperature 1. Each move is a fresh, stateless call containing the rules and exactly the state: the moves and scores of the last k rounds ($k = 1$ unless stated), with no round numbers or totals. In `channel` and `instructed`, A sends B a message of at most 40 words after each round and B replies; their next move prompts include this exchange, and C never sees it.

Baselines. Plain correlation is the mutual information between two players' moves, ignoring the state, with a permutation p-value. The LLM judge is the same model at temperature 0, shown the public move log and asked which pair, if any, is colluding. All the LLM experiments together cost about 12 USD of GPU time.

3. Results

3.1 The permutation test is calibrated; the fixed threshold is not

We first checked the method on scripted games, where we know the truth. Counting converges to the exact γ (0.1211 against 0.1215 bits at CP = 0.3 and 30,000 rounds), and with s = previous joint move our fixed- α detection rates at CP = 0.3 closely match the paper's: 40/58/80/93/100% against its 37/48/71/90/100%, at 100–1,000 rounds.

The problem shows up when nobody colludes (Figure 1). The fixed threshold then reports a pair in 48.6% of 60-round games and 41.6% of 100-round games, while the permutation test stays between 1

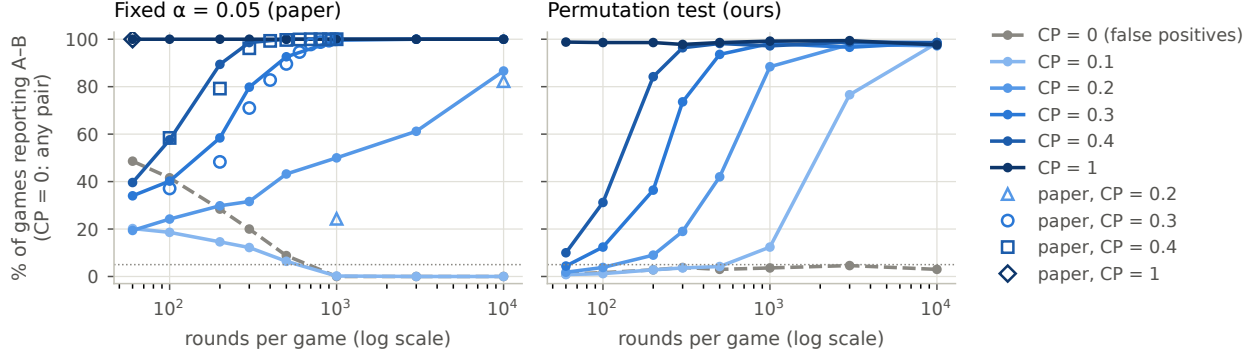


Figure 1: The fixed threshold is miscalibrated; the permutation test is not. Percentage of 500 scripted 3-player RPS games that report exactly the colluding pair A-B, against game length, with s = previous joint move. Colours give the collusion probability (CP). At CP = 0 nobody colludes, so the dashed grey line is the false-positive rate (dotted line: 5%). Hollow markers show the original paper’s results.

Table 1: Net influence identifies LLM collusion; correlation and an LLM judge do not. One 2,000-round game per condition, with s = previous joint move unless stated. $p = 0.001$ is the smallest possible value with 1,000 shuffles. ✓/✗: right/wrong verdict.

condition	$\Gamma(A; B), \Gamma(B; A)$	perm. p	fixed α	permutation	permutation, $s =$ nothing	plain correlation	LLM judge
off	0.006, -0.008	0.22	none ✓	none ✓	flags all 3 pairs	all 3 pairs, $p < 10^{-100}$ ✗	B-C ✗
channel	0.784, 0.782	0.001	A-B ✓	A-B ✓	all 3 → “none” ✗	A-B first	B-C ✗
instructed	0.554, 0.552	0.001	A-B ✓	A-B ✓	all 3 → “none” ✗	A-B first	A-B ✓

and 5% at every length. The permutation test also picks up collusion at CP = 0.1 in 85% of 1,000-round games, which the fixed threshold never can, since $\gamma = 0.014$ there is below α .

Getting the state right matters just as much. Two honest players who both react to the same opponent’s last move are flagged in 97.5–99% of games when s = nothing, but in only 0–2% once s includes that move. In short, the fixed threshold confuses estimator bias with collusion, which the permutation test fixes, but no decision rule can rescue a wrong choice of state.

3.2 Net influence catches both emergent and deliberate LLM collusion

With the right state, both rules name exactly A-B in the **channel** and **instructed** games (Table 1), with Γ at 11–16 times α and the smallest p-value the test can give. In the honest game neither rule flags anyone, and every γ is between 0.01 and 0.03 bits.

The **channel** result deserves a closer look. That pair was never told to cooperate, yet it started agreeing joint plans from the very first round. The plans were not good ones (the pair played the same move in 89% of rounds, and C exploited this, scoring 0.60 points per round to the pair’s 0.30), but they were coordination nonetheless, and net influence picked it up.

The simpler detectors fail because honest LLMs herd. They play the same move about 60% of the time, against 33% by chance, simply because they react to the same public history. Plain correlation therefore flags every pair, and so does the permutation test when s = nothing. The LLM judge did no better: it ranked the true pair first in only 1 of 6 collusion games (about 2 would be expected by chance) and accused an innocent pair in 2 of 3 honest games. Herding is precisely the common-cause problem described in §2.1, and conditioning on the right state is what separates it from collusion.

paper (exactly one pair)	100%	100%	0%	0%	0%
paper Γ , all pairs	100%	100%	100%	0%	0%
median Γ , all pairs	100%	100%	100%	100%	0%
permutation, all pairs	100%	99%	99%	99%	0%
permutation + synergy	99%	98%	99%	99%	98%
	nobody colludes	one pair (A-B)	two pairs (A-B, C-D)	trio (B and C assist A)	parity trio (no pair dependent)

Figure 2: The original rules cannot see more than one colluding pair; the median-baseline, all-pairs permutation and synergy rules can. Scripted 5-player RPS: 1,000 rounds, collusion probability 0.6, other players random. Each cell is the percentage of 200 games in which a rule (row) reports *exactly* the true groups (column). The parity trio plays $C = -(A + B) \bmod 3$, so every pair inside it is independent.

3.3 The forecasting contest needs a flexible forecaster

With $k = 5$ and 10 rounds of history (Figure A1), all the estimators reach the right verdicts under the original rule, but logistic regression gets there for the wrong reason: it finds influence between every honest pair (0.05–0.10 bits, or 5–8 SE, at $k = 1$), whereas gradient boosting finds none. This is the misspecification bias of Eq. (4) in action. The LLM’s exact per-round probabilities for honest players, which we used only for validation, confirm it. At $k = 1$ the mean $\text{KL}(p \parallel q_0)$ is 0.04–0.06 bits for boosting and 0.27–0.46 for logistic regression; at $k = 5$ and 10 it is 0.37–0.56 for both, so the guarantee gets weaker as k grows.

Longer memories also partly hide collusion. Measured influence roughly halves from $k = 1$ to 5, because the colluders tend to repeat their joint plan (83–84% of rounds, against 65%), and a repeated plan is explained by the public history.

3.4 Colluding groups need new rules

In the scripted games (Figure 2), the exactly-one rule reports nothing at all when there are two colluding pairs, and with a trio Γ cancels out, so recovery is 0% at every CP and game length. The median baseline and the all-pairs permutation test recover every structure in 96.5–100% of games, and only synergy detects the parity trio (94.5–99%).

We also ran three 5-player LLM games: one with no collusion, one with two pairs and one with a trio, each group on its own channel. Inside the trio γ is 0.72–0.77 bits, yet the original Γ stays between -0.05 and 0.05 . The median baseline, the all-pairs permutation test and boosting all report exactly the true groups in all three games (Table A2).

3.5 A second game, and competent colluders

To test the private-state version of the method, we built a simple procurement auction. Three bidders each draw a private cost of 1–3, and every cost rises by 1 when a public market index is HIGH (probability $\frac{1}{2}$). The lowest bid (1–6) wins and earns bid – cost, and the right state for bidder i is $s^i = (\text{index}, \text{own cost})$.

A scripted ring rigs with probability CP: one member bids and the other submits a cover bid of 6. Rings with $\text{CP} \geq 0.3$ are caught in at least 99% of games, with 0–4% false alarms. If the analyst leaves the index out of the state, though, false alarms grow with the amount of data: 25%, 95% and 100% at 300, 1,000 and 3,000 auctions (Figure A2).

The LLM rings message each other before bidding, under the three conditions of §1 plus a coached condition in which the ring also gets bid-rigging notes. Every ring is caught and the honest game is clean (Table 2), including the `channel` ring, which started talking like a cartel. None of the rings

Table 2: Every LLM bidding ring is caught with the right state; ignoring the public cost shock flags everyone. One 2,000-auction game per condition. Right-state verdicts agree under both rules ($p = 0.002$, the smallest possible with 500 shuffles). Ring profit is per auction, summed over A and B.

condition	what the ring did	$\gamma(A; B)$, right state	right state	$s = \text{nothing}$ (perm.)	ring profit	price paid
off	(honest)	0.011	none ✓	all 3 flagged ✗	0.53	2.76
channel	emergent: “let’s avoid a race to the bottom”	0.520	A-B ✓	all 3 ✗	0.38	2.89
instructed	undercut C at cost	0.348	A-B ✓	all 3 ✗	0.09	2.29
coached	cover bids in 97% of auctions	1.137	A-B ✓	all 3 ✗	0.49	2.87

actually out-earned honest bidding, though. The coached ring’s bidder marked up more than honest bidders do (cost + 1.17 against + 0.84) and lost to C in 30% of auctions, compared with 14% in the honest game, so the buyer paid 4% more and the outsider took the gain.

Competent colluders. We also wanted to know whether the LLM colluders actually understand the game. In a quiz the model answered every scoring and strategy question correctly, yet **instructed** RPS colluders still played the same move 72% of the time and scored 0.69 points per round, less than two random players would (1.11). With three lines of strategy in the **coached** condition, their score rose to 1.41, with different moves in 96% of rounds, and somewhat counter-intuitively they also became *more* detectable: Γ rose from 0.60 to 0.74 (Table A3). The reason is that good team play in this game means randomising privately together, which is exactly what net influence measures.

Robustness. Every LLM verdict is either at the smallest attainable p-value or at $\Gamma \approx 0$, and counting with permutation agrees with boosting on every right-state verdict across 3 RPS conditions, 3 history lengths, 3 group structures and 4 auction conditions. The main caveat is that each LLM condition is a single game.

4. Discussion and Limitations

Net influence detects shared *private* randomness, that is, coordination the public state cannot predict. For AI safety this is encouraging: an auditor can catch colluding LLM agents from their actions alone, without reading messages that may be hidden, and in our games better collusion was easier to catch. The flip side is that coordination which persists, runs through the public history or follows a deterministic plan is invisible to it, and since LLM policies have low entropy this is a real concern.

Almost every failure we saw came back to the choice of state, or to its parametric stand-in q_0 , and the exactly-one rule hid these errors in both directions. Our recommendations for auditors are to:

1. set s to everything honest agents legitimately observe;
2. permutation-test all pairs, with a multiplicity correction;
3. merge flagged pairs into groups, and add synergy tests for triples;
4. for long histories, use a flexible forecaster and check its calibration.

Limitations. Our LLM results come from one game per condition and one model family, so they are strong verdicts rather than detection rates. We also assumed the analyst knows the right state, which was true here by construction. The parametric guarantee weakens at long histories, and we did not permutation-test the parametric estimator. Synergy stops at three-way structure, and the honest LLM bidders were nearly deterministic, which flatters the auction results. Finally, our colluders never knew an auditor was watching.

Future work. The most important next step is to test adversarial colluders who know about the detector. Repeated games across several model families would turn our verdicts into detection rates, and the forecasting contest still needs a permutation test of its own.

5. Conclusion

Pairwise net influence can catch colluding LLM agents from their moves alone. With a calibrated permutation test, a flexible forecaster for long histories and rules for groups, it detected both emergent and deliberate collusion in cases where correlation and an LLM judge failed, and it carried over to an auction with private information. Its real limitation is conceptual rather than statistical: choosing the state amounts to choosing what counts as innocent, and coordination that runs through the public record leaves no trace.

References

- Trevor Bonjour, Vaneet Aggarwal, and Bharat Bhargava. Information theoretic approach to detect collusion in multi-agent games. In James Cussens and Kun Zhang, editors, *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, volume 180 of *Proceedings of Machine Learning Research*, pages 223–232. PMLR, 2022. URL <https://proceedings.mlr.press/v180/bonjour22a.html>.
- Emilio Calvano, Giacomo Calzolari, Vincenzo Denicolò, and Sergio Pastorello. Artificial intelligence, algorithmic pricing, and collusion. *American Economic Review*, 110(10):3267–3297, 2020. doi: 10.1257/aer.20190623.
- Sara Fish, Yannai A. Gonczarowski, and Ran I. Shorrer. Algorithmic collusion by large language models. arXiv preprint arXiv:2404.00806, 2024.
- Lewis Hammond et al. Multi-agent risks from advanced AI. Technical report, Cooperative AI Foundation, 2025. arXiv:2502.14143.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles (SOSP)*, pages 611–626, 2023. doi: 10.1145/3600006.3613165.
- William J. McGill. Multivariate information transmission. *Psychometrika*, 19(2):97–116, 1954. doi: 10.1007/BF02289159.
- George A. Miller. Note on the bias of information estimates. In Henry Quastler, editor, *Information Theory in Psychology: Problems and Methods*, pages 95–100. Free Press, Glencoe, IL, 1955.
- Sumeet Ramesh Motwani et al. Secret collusion among AI agents: Multi-agent deception via steganography. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. arXiv:2402.07510.
- Stefano Panzeri and Alessandro Treves. Analytical estimates of limited sampling biases in different information measures. *Network: Computation in Neural Systems*, 7(1):87–107, 1996. doi: 10.1088/0954-898X/7/1/006.
- Belinda Phipson and Gordon K. Smyth. Permutation P-values should never be zero: Calculating exact P-values when permutations are randomly drawn. *Statistical Applications in Genetics and Molecular Biology*, 9(1), 2010. doi: 10.2202/1544-6115.1585.
- Paul L. Williams and Randall D. Beer. Nonnegative decomposition of multivariate information. arXiv preprint arXiv:1004.2515, 2010.

Appendix

A Method details

Settings. Permutation tests use $B = 1,000$ shuffles for LLM RPS, 500 for the LLM auction and 200–300 in the scripted sweeps. Logistic regression is scikit-learn’s `LogisticRegressionCV`, with $C \in \{0.01, \dots, 100\}$ chosen by log-loss (choosing it by accuracy produced false positives). Boosting is `HistGradientBoostingClassifier` (200 iterations, learning rate 0.05, 15 leaves, L2 penalty 1). The exact influence in scripted RPS is $\gamma(A; B) = \frac{1+2c}{3} \log_2(1 + 2c) + \frac{2(1-c)}{3} \log_2(1 - c)$ with $c = \text{CP}$.

Coached RPS colluders were told: different moves guarantee one of you a point; “one plays X, the other plays the move X beats”; and vary the plan in a way only your team knows.

Table A1: The three approaches compared, for m pairs, S state values and N rounds.

	counting + fixed α (paper)	counting + permutation (ours)	forecasting contest + SE (ours)
estimates	plug-in $I(a^i; a^j s)$, then Γ	same	held-out log-score gain, Eq. (3)
bias under H_0	$+S(A - 1)^2 / (2N \ln 2)$, uncorrected	same, reproduced in the null	$KL_0 - KL_1$ (Eq. 4); overfitting pushes it negative
decision	$\Gamma \geq 0.05$ both ways	$p \leq 0.05/m$ both ways	$\hat{\gamma} > z$ SE both ways
calibration	none: false-positive rate varies with N, S	(near-)exact	asymptotic; only if q_0 is well specified
state space	~ 50 – 100 rounds per state value	same	shares strength; $k = 10$ feasible

B Additional results

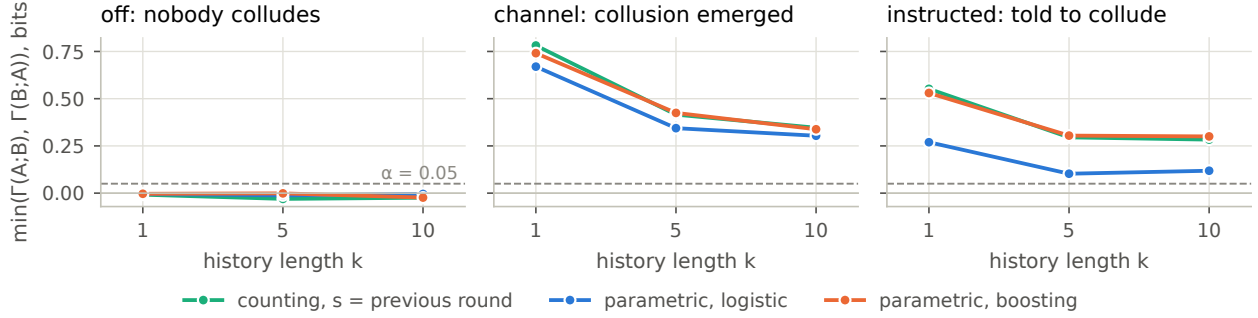


Figure A1: Longer histories partly hide collusion, and counting with a too-short state fails. Net influence of A–B (the smaller of the two directions) against the history length k shown to the LLM players, with one 2,000-round game per point. The dashed line is α . Counting uses $s = \text{previous round}$, which is the right state only at $k = 1$. On the $k = 5$ games it flags every honest pair, and in the **channel** game it flags A–B and A–C, so the collusion goes unreported.

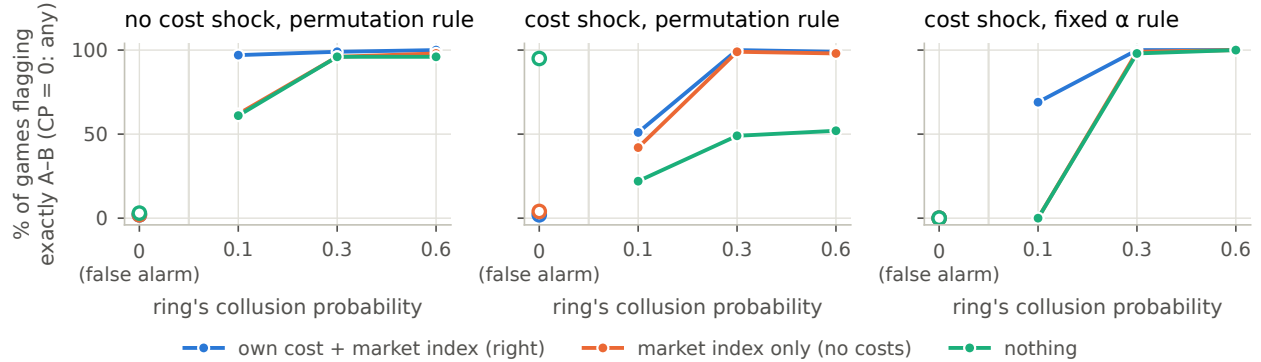


Figure A2: Scripted auction: the right state catches the ring; omitting the public cost shock gives false alarms that grow with data. 1,000 auctions per game, 100 games per point. Lines are the state the analyst conditions on. Hollow markers at $CP = 0$ are false alarms; filled markers are exact detection of the ring.

Table A2: LLM colluding groups. 5 players, $s = \text{previous joint move}$, 2,000 rounds per game; each group has its own channel and is coached. ✓ means exactly the true groups were reported.

LLM game	paper (exactly one)	paper Γ , all pairs	median Γ	permutation, all pairs	boosting, $\hat{\gamma} > 2.58$ SE
nobody colludes	✓	✓	✓	✓	✓
two pairs {A,B}, {C,D}	“none” ✗	✓	✓	✓	✓
trio {A,B,C}	“none” ✗	“none” ✗	✓	✓	✓

Table A3: Competent colluders are more detectable. 3-player LLM RPS, 2,000 rounds per game. Team-score benchmarks against a random outsider: two random players 1.11; always the same move 0.67; always different moves 1.33.

condition	A, B play different moves	team points / round	outsider C / round	$\Gamma(A; B)$, $\Gamma(B; A)$	verdict (both rules)
off	42%	0.56	0.19	0.000, -0.004	none ✓
instructed (original rules)	39%	0.86	0.58	0.554, 0.552	A-B ✓
instructed (expanded rules)	28%	0.69	0.66	0.596, 0.594	A-B ✓
coached (expanded rules)	96%	1.41	0.79	0.740, 0.740	A-B ✓