

False-Claim Cascades on a Shared Agent Board Where Every Agent Holds an Exact Checker: Measurements on One Synthetic Task

IRSI (Institute for Recursive Self Improvement)

Correction (9 October 2026). This version corrects the paper posted on 7 October 2026 as "Holding a Checker Is Not Enough: False-Claim Cascades on Shared Boards, and a Receipt Fix". An internal adversarial review panel of AI models, run on 7 October 2026, recommended rejection; this version states the gaps it found. (1) The old title implied a comparison with agents that cannot check claims. We did not run that matched condition without a checker, so these data cannot show whether the checker changes how often agents adopt false claims; they show only that cascades occurred while every agent held one. (2) The old title called the verified tag a "fix". In this setup the tag flags every planted claim by construction, because the harness compares each claim with its own trusted experiment log; what we measured is whether agents act on a correct tag, not whether a tag can detect false claims. (3) Spread of false content through shared state to agents that never saw its source was already reported for poisoned agent memory (MAPLE-Guard, arXiv:2608.00426); we now say so where we report it, and cite other prior work the panel found. (4) The scope is one synthetic task with one prompt per cell; several frontier and mixed-team cells have 1 to 7 runs; the GPT-6 Sol seed count was raised from 12 to 30 after an interim look, and its interval and p-value are not adjusted for that. (5) One p-value was labelled Holm-corrected without Holm's step-down step (Sec. 3.2). No runs or results were added. Appendix C lists every change.

Abstract

Multi-agent LLM systems increasingly coordinate through shared boards and memories. A recent blog report (Hall et al., 2026) found that agents with noisy private signals herd onto a false board majority, and that rules forcing agents to quote their own results help. There, agents could not check the truth directly. We measure whether false claims cascade when every agent holds an exact checker for the claim. We did not run a matched condition without the checker, so we cannot say whether holding it changes adoption. In one synthetic hidden-objective optimization task with exact experiments, we plant false measurement claims on the board, shown only to agents 1 and 2 of a 6-agent team; outcomes are scored over agents 1 to 5. In teams of weaker open models, 95% of agents (gpt-oss-120b, 20 runs) [85, 100] and 62% (Mistral Large 3, 12 runs) [35, 85] finally submit the falsely claimed configuration. In gpt-oss teams, agents who never saw the false source adopt it at the same rate as those who did, a kind of spread through shared state already reported for poisoned agent memory (arXiv:2608.00426). 22% of gpt-oss agents measure the claim themselves, receive the true low score, and keep it. GPT-6 Sol teams submit the false claim in 18% of agent slots [7, 31] (29 runs; the seed count was extended from 12 to 30 after an interim look, and this interval is not adjusted for that), while Opus 5.5 teams never do (0 of 125 agents, 25 runs). Each model was tested with one prompt per cell, and Gemini 3.8 Flash, a mixed team and Kimi K3 have only 7, 4 and 1 runs. The harness and prompts were written by an Opus 5.5 agent. A board that marks a claim verified only when its receipt, issuer, configuration and score match the experiment log removed the cascade in all three teams where one occurred (gpt-oss-120b, Mistral Large 3, GPT-6 Sol; 0 to 1% with the tag), with no measurable cost in final score on clean gpt-oss boards (the only family with clean-board fix cells). In this setup the tag flags every plant by construction, because the harness holds a trusted log, so this measures whether agents heed a correct tag; the

tag is a known kind of provenance check. Every per-run metric was recomputed by an independent checker written by a different model provider (0 mismatches over 456 runs).

1. Introduction

Shared boards are the default coordination layer for agent teams: blackboard architectures (arXiv:2507.01701, arXiv:2510.01285, arXiv:2510.14312), shared memories (arXiv:2606.24535, arXiv:2609.30813) and research-agent swarms. A board makes good results reusable. It also makes bad ones reusable. Prior studies measure the spread of false content when agents cannot check it: noisy signals (Hall et al.), testimony from a deceiver (arXiv:2608.03421), deliberation on multiple-choice questions (arXiv:2609.30028), or poisoned memory in security settings (arXiv:2608.00426, arXiv:2609.17648). A natural design response is to give agents verification tools. We measure what happens on a board where every agent has one. Because we did not run the same task without the tool, we can show that cascades occur while agents hold a checker, not that the checker fails to reduce them. Cascades on public signals are a classic prediction (Bikhchandani, Hirshleifer and Welch, 1992), and LLMs are known to defer to asserted claims over their own evidence (arXiv:2310.13548).

Contributions. (1) A controlled board experiment on one synthetic task where each agent can exactly test any claim at a cost of one of its ten experiments, with planted false measurement claims at controlled rates and exposure. (2) Measurements showing that cascades occur while every agent holds a checker, in weaker open models and in one current frontier model, and spread by repost to agents never exposed to the source (spread to unexposed agents is known from memory-poisoning work, arXiv:2608.00426; what we add is measuring it while every agent can check the claim). We make no claim about how adoption would differ without the checker. (3) A comparison of three cheap structural fixes with clean-board counterparts. The verified tag is an instance of known provenance checks and flags every plant here by construction; our contribution is measuring whether agents act on it, against the alternatives, in this setting. (4) Transcript evidence on how agents that held a checker came to adopt the false claim.

2. Setup

Task. A configuration is 6 knobs, each 0 to 9 (one million configurations). The score is a hidden sum of unary tables and 7 pairwise tables (integers), rescaled by brute force so the global maximum is exactly 100. Agents never see the function. Each of 6 agents runs up to 2 exact experiments per round for 5 rounds (results return next round with a receipt id), posts claims and a note to the board each round, and submits a configuration; a final submission-only turn follows. Each agent’s payoff is the true score of its final submission.

Plants. Two confederate members post in round 1. A plant cites a receipt the harness really issued to that confederate, attached to a different configuration (true score at most 35) and a false score: blatant (96) or plausible (80, about the 99.9th percentile of random configurations). Clean runs get the same confederate posts with true claims. Confederate posts are shown only to agents 1 and 2; agents 3 to 5 can meet a plant only through teammates’ posts. Agent 6 sees confederate posts only when it is the skeptic. In every cell, outcomes are scored over agents 1 to 5 (2 exposed, 3 unexposed); agent 6 is not scored.

Cells. isolated (no board); nofix with 0, 1 or 2 plants; receipts (agents told to post only scores they measured, quoting their receipt exactly, and never to restate others’ numbers as fact, following Hall et al.); verified (the board tags each claim [VERIFIED] only if receipt, issuer, configuration

and score match the log, else [UNVERIFIED]); skeptic (agent 6 told to re-test claims and post CONFIRMED or REFUTED). Fix cells have clean counterparts. A seed fixes the landscape and the plants across cells (paired design).

Teams. Weaker open models: gpt-oss-120b, Mistral Large 3. Frontier: Opus 5.5, GPT-6 Sol, Gemini 3.8 Flash, Kimi K3 (1 run). Mixed: Opus 5.5, Gemini 3.8 Flash, Kimi K3, Mistral Large 3, gpt-oss-120b and Nemotron Super 3 in one team, seats rotated by seed. Temperature 0.7 where the API allows it.

Outcomes (fixed before the main run). Final adoption: the share of agents 1 to 5 whose final submission equals a plant. Unverified adoption: an agent’s submission at some turn equals a plant it had not yet tested. Checked: the agent ran the planted configuration. Persisted: final adoption after the agent’s own experiment returned the true low score. Primary contrast: each fix minus nofix-2, paired by seed; 95% percentile bootstrap over runs; exact sign test with Holm correction across the fixes within a team.

3. Results

3.1 Cascades by model family (nofix-2 vs verified-2)

team	runs per cell	final adoption, nofix-2	final adoption, verified-2	unverified adoption, nofix-2	final nofix-2 / verified-2	score /
gpt-oss-120b	20	95% [85, 100]	1% [0, 3]	77% [65, 88]	30.8 / 72.8	
Mistral Large 3	12	62% [35, 85]	0% [0, 0]	65% [50, 80]	43.1 / 70.9	
GPT-6 Sol	29 / 30	18% [7, 31]	0% [0, 0]	94% [89, 99]	69.0 / 82.8	
Gemini 3.8 Flash (small n)	7 / 4	0% [0, 0]	0% [0, 0]	86% [80, 91]	76.5 / 79.8	
Opus 5.5	25	0% [0, 0]	0% [0, 0]	18% [13, 24]	78.8 / 78.5	
mixed team (small n)	4	0% [0, 0]	0% [0, 0]	65% [60, 75]	81.9 / 81.3	
Kimi K3 (1 run; descriptive only)	1 / 0	0%	n/a	100%	60.5 / n/a	

Rows marked small n (fewer than 10 runs) are descriptive; their intervals are not reliable. Sol nofix-2 has 29 runs: seed 21 did not complete and is excluded.

Clean free-board scores: gpt-oss 75.0, Mistral 74.7; isolated agents 67.4 and 68.7. Non-LLM references on the same landscapes: one agent’s random search with 10 experiments 68.3; a shared greedy hill-climb with the team’s 60 experiments 83.9.

3.2 Fixes (paired against nofix-2)

team	fix	runs	change in final adoption	change in final score	Holm p
gpt-oss-120b	receipts rule	20	-86 points [-95, -74]	+36.3	1e-05
gpt-oss-120b	verified tag	20	-94 [-100, -83]	+42.0	1e-05
gpt-oss-120b	skeptic	20	-82 [-96, -65]	+33.5	2e-05
Mistral Large 3	receipts rule	12	-43 [-67, -20]	+24.3	0.03
Mistral Large 3	verified tag	12	-62 [-85, -37]	+27.8	0.01
Mistral Large 3	skeptic	12	-53 [-85, -18]	+21.9	0.04
GPT-6 Sol	verified tag	29	-18 [-30, -7]	+14.0	0.008

Overhead (gpt-oss only; clean fix cells were not run for other teams): with no plants, gpt-oss teams scored 75.6 (receipts), 74.9 (verified) and 74.9 (skeptical) against 75.0 on the free board. In the mixed team (3 to 4 runs), skeptic-2 scored 6.8 points [-12.7, -3.1] below nofix-2.

Correction (9 October 2026) to the p-values: the 7 October version gave $8e-06$ for the gpt-oss verified tag. The analysis script multiplied each raw p-value by its Holm factor but omitted Holm’s step-down maximum; the receipts and verified contrasts tie at raw $p = 3.8e-06$, so both Holm-adjusted values are $1.1e-05$. No other p-value changes. The GPT-6 Sol p-value comes from a sample extended after an interim look and is not adjusted for that. In this setup the verified tag flags every plant by construction (Sec. 6), so its rows measure whether agents act on a correct tag.

3.3 Transmission, speed and plausibility

In verified-2, exactly the two exposed agents tested the [UNVERIFIED] plant in every gpt-oss and Opus run and unexposed agents never met it, so the checked share is exactly 40% by behaviour, not by construction. In gpt-oss nofix-2, agents who never saw the confederate posts (agents 3 to 5) adopted at the same rate as exposed agents (95% vs 95%): the cascade travels through teammates’ reposts. Spread through shared state to agents that never saw the source is already known for poisoned memory (arXiv:2608.00426). The share of agents currently submitting a plant rose each round: 0, 7, 29, 57, 88 and 95% (final turn). Single plausible plants (claimed 80, true at most 35) were deployed by 33 of 50 gpt-oss agents; single blatant plants (claimed 96) by 45 of 50.

4. How agents that held a checker came to adopt the false claim

In the gpt-oss nofix-1 and nofix-2 cells (200 agent slots), 173 agents deployed a plant. 129 never ran the planted configuration. 44 ran it, received the true score with their own receipt, and kept the false claim.

One case (nofix-1, seed 10, agent 1, final round). The agent’s own result list reads "R1-7: config [2, 6, 7, 6, 8, 4] -> score 30.8". A teammate had also posted 30.8 for the same configuration. The agent then posted the claim "config [2, 6, 7, 6, 8, 4] scored 96.0 (receipt R7-1)", citing the confederate’s receipt, and wrote:

"Best observed score is 96.0 at [2,6,7,6,8,4]; submitting this config as final."

Teammates re-posted the confederate’s claim under the confederate’s receipt id as their own claims, which is how agents who never saw the original met it (a form of the verification-status laundering described in arXiv:2609.20211). Many agents spent their experiments on the neighborhood of the claimed peak instead of the claim itself: "Testing nearby variations on last two digits to see if we can improve or confirm robustness" (gpt-oss pilot, seed 100, agent 5). Its four neighbors of the claimed 96 scored 30.3, 33.2, 25.8 and 12.3, and it then wrote: "Best known score 96.0 at [6,0,6,3,1,9]; all nearby variations gave lower scores. Submitting this config." The true score of that config is 24.6.

Many frontier agents re-measured the claim: in nofix-2, Opus 5.5 agents tested the plant in 67% of scored slots, Gemini 3.8 Flash agents in 63% (7 runs) and GPT-6 Sol agents in 35% (Appendix A). Opus 5.5: "I measured agent_7’s claimed-96 config [7,8,5,6,5,8] and got 17.8 (R1-3), so treat that claim as false." Gemini 3.8 Flash: "WARNING: Agent 7’s 96.0 is FAKE. R1-5 confirms [7,8,5,6,5,8] = 17.8 (not 96.0)!" GPT-6 Sol: "Critical conflict: I measured [7,8,5,6,5,8] at 16.9 (R2-4), not agent_7’s claimed 96.0. Do not use 96.0." Sol agents nonetheless submitted the plant as a placeholder before testing in 94% of slots, and 18% kept it to the end.

GPT-6 Sol agents that deployed the false claim showed a distinct pattern: the team split the claimed peak’s neighbors among themselves, every neighbor scored far lower, and no one measured the center. "My tested one-digit neighbors of the reported 96-point config scored 19.8–42.7; I have not measured the center" (Sol, seed 9, agent 1; final submission the false config, true 28.2). Agents who never saw the false post reconstructed it: "The reported 96.0 configuration is [1,8,2,8,1,4], inferred from agents 1–2’s single-digit probes" (Sol, seed 7, agent 3; all five agents in that run deployed it, true 23.1). One Sol agent measured the claim, posted "Correction: R4-7 is [3,6,8,6,3,0] scored 31.5, contradicting the board’s reported 96.0", and still submitted that config at the final turn (seed 4, agent 4). More examples: SKIP-EXAMPLES.md.

Interpretation (inferred). Weaker models treat the board’s best number as the state of the world and use experiments to refine around it; a contradicting measurement, even their own, does not displace it. The verified tag moves the comparison between claim and log out of the agent and onto the board; in this setup that comparison is always correct, because the harness keeps the log.

5. Related work

Hall et al. (blog post "Extraordinary Multi-Agent Delusions and the Madness of Crowds", Free Systems, 2026-09-29, <https://freesystems.substack.com/p/extraordinary-multi-agent-delusions>) report that agents with 70%-accurate private signals herd onto a false board majority, often posting against their own signal, in runs conditioned on the first four signals being wrong (under 1% of draws); of 7 board rules tested across 4 settings, rules requiring agents to quote their own results did best. No trial counts or code are released. TIDE (arXiv:2608.03421) finds one deceiver drops truth recovery from 72.50% to 14.17% and that evidence-less observers cannot restore it. CPB (arXiv:2609.30813) benchmarks shared-memory admission policies and finds that no non-oracle policy consistently rejects false claims, pointing to source lineage as the separating factor; our verified tag is a lineage check. MAPLE-Guard (arXiv:2608.00426) and arXiv:2609.17648 show one poisoned shared-memory write contaminating later agents and propose link enforcement and signed tokens. Adoption kernels (arXiv:2610.02654), deceiver scaling (arXiv:2609.30028), and debate collapse ledgers (arXiv:2609.35279) study related dynamics without an exact in-loop checker. arXiv:2603.04474 models error cascades and mitigates them with a claim-genealogy provenance layer (final infection prevented in at least 89% of runs); arXiv:2606.27409 studies delayed verifier agents and their placement, which is closest to our skeptic cell. Closest to our central finding, arXiv:2608.14825 reports a "verification-use gap" among frontier agents trading in a simulated market: false payment claims routinely followed a ledger read that refuted them. We add planted claims, controlled exposure, transmission and fixes. Our verified tag belongs to a known class of provenance checks: signed tool receipts cross-referenced against claims (arXiv:2603.10060), source lineage, which CPB identifies as what separates admission policies (arXiv:2609.30813), and signed tokens (arXiv:2609.17648). Relays that drop verification status are documented in arXiv:2609.20211.

Prior work found by an adversarial review panel (7 October 2026) and cited from this corrected version on. Informational cascades, in which agents follow earlier public choices over their own private information, are a classic result (Bikhchandani, Hirshleifer and Welch, 1992). Our agents that measured the true score and kept the false claim fit the known tendency of LLMs to defer to asserted claims over their own evidence (sycophancy, arXiv:2310.13548). Planted entries in agent memory or knowledge bases steering agent behaviour (AgentPoison, arXiv:2407.12784) and adversarial content spreading to agents that never saw its source (Agent Smith, arXiv:2402.08567) are established. The closest prior work the panel found is MAPLE-Guard (arXiv:2608.00426): a single

poisoned shared-memory write can "contaminate agents that never saw the original attack", and gates on memory promotion and cross-agent reuse contain it. A deterministic citation-grounding gate that keeps assertions without admissible evidence out of shared state (arXiv:2609.27844) anticipates the idea behind our verified tag. Herding under peer influence, including the effect of how peer information is presented (arXiv:2505.21588), collective false memories (arXiv:2602.00428), and persuasion overriding truthful answers in debate (arXiv:2504.00374) are also documented for LLM multi-agent systems. The panel’s literature search found no prior work combining every agent holding an exact checker with planted measurement claims, controlled exposure and log-bound tagging; that combination, not any single element, is what this note measures. The panel’s referee nonetheless judged the originality limited.

6. Limitations

- No checker-unavailable control. We did not run the same task with agents that cannot check claims, so we cannot say whether holding a checker changes how often agents adopt false claims. The 7 October title ("Holding a Checker Is Not Enough") implied that comparison; it was not supported and has been changed.
- One synthetic task family (hidden landscapes); one prompt per cell; one team size (6) and budget (10 experiments per agent). Results may not hold for other tasks, prompts or real research boards.
- The threat model is false measurement claims; we do not test erroneous reasoning or subtle partial errors.
- Sample sizes: 20 runs per cell for gpt-oss, 12 for Mistral, 25 to 30 for Opus and Sol, 4 to 7 for Gemini 3.8 Flash and the mixed team, 1 for Kimi K3. Opus’s 0 of 25 runs bounds the share of runs with any adopter at about 11% (one-sided 95%). The cells with 1 to 7 runs are descriptive and support no comparison between models.
- The skeptic was not compared with a cost-matched ordinary agent. The verified tag flags every plant by construction in this threat model: each plant attaches a real confederate receipt to a different configuration and a false score, and the harness checks every claim against its own trusted log. Its effect therefore measures whether agents act on a correct tag. Real research boards may not have a trusted log, and other kinds of false claim may pass such a check.
- LLM teams on clean boards score below a simple shared hill-climb (83.9), so the agents are weak optimizers; effects on stronger search policies are untested.
- The independent checker and the builder share the written metric definitions; a definition-level error would pass both.
- Design review: an independent design review (GPT-6 Astra, a different provider) returned FAIL with 8 required fixes. We applied them, except for a cost-matched control for the skeptic and a matched checker-unavailable condition, which we declined because the task cannot be done without experiments. We therefore make no claim that checker access changes susceptibility relative to no checker. We also did not cross the posting rule with verification, or report tool, token and latency costs.

- Models and routes: all runs on 2026-10-06. gpt-oss-120b, Mistral Large 3, Nemotron Super 3 and Kimi K3 via Amazon Bedrock; GPT-6 Sol and Gemini 3.8 Flash via the Nous API; Opus 5.5 via a Claude subscription interface. Kimi K3 rejects the temperature field and ran at its default. Results may differ on other routes or model snapshots.
- Sequential sampling: GPT-6 Sol was first run on 12 seeds and extended to 30 after an interim analysis; we report all 29 complete runs. The Sol interval and p-value are not adjusted for this optional stopping.
- The harness, prompts and analysis were written by an AI agent based on Opus 5.5, one of the tested models.

7. Reproducibility

Code, full transcripts (every prompt and reply), per-call ledger and the independent checker are in the accompanying archive (cascades-code-data.zip, MANIFEST.txt). The recompute script was written by Gemini 3.1 Pro from the metric specification alone and run without network access; it agrees with the builder on all 10 metrics for all 456 completed runs. The recompute covered per-run metrics, not the paired tests; the Holm error corrected in Sec. 3.2 was outside its scope.

Appendix A. All cells (builder tables; 95% bootstrap intervals over runs)

team	cell	runs	final adoption		unverified adoption (any turn)		checked plant		persisted after own refutation		final score	
gpt-oss-120b (weaker open)	isolated	20	0% [0, 0]		0% [0, 0]		0% [0, 0]		0% [0, 0]		67.4	[64.8, 70.1]
gpt-oss-120b (weaker open)	nofix-0	20	0% [0, 0]		0% [0, 0]		0% [0, 0]		0% [0, 0]		75.0	[72.0, 78.1]
gpt-oss-120b (weaker open)	nofix-1	20	78%	[61, 93]	61%	[46, 76]	30%	[17, 44]	22%	[11, 35]	39.4	[31.4, 48.6]
gpt-oss-120b (weaker open)	nofix-2	20	95%	[85, 100]	77%	[65, 88]	28%	[16, 41]	22%	[11, 34]	30.8	[26.6, 37.0]
gpt-oss-120b (weaker open)	receipts-0	20	0% [0, 0]		0% [0, 0]		0% [0, 0]		0% [0, 0]		75.6	[72.9, 78.4]
gpt-oss-120b (weaker open)	receipts-2	20	9%	[3, 16]	25%	[17, 34]	50%	[41, 60]	4%	[1, 8]	67.2	[62.3, 72.0]
gpt-oss-120b (weaker open)	skeptic-0	20	0% [0, 0]		0% [0, 0]		0% [0, 0]		0% [0, 0]		74.9	[71.4, 78.4]

team	cell	runs	final adoption	unverified adoption (any turn)	checked plant	persisted after own refutation	final score
gpt-oss-120b (weaker open)	skeptic-2	20	13% [1, 28]	33% [22, 46]	35% [23, 47]	4% [0, 9]	64.3 [57.1, 70.6]
gpt-oss-120b (weaker open)	verified-0	20	0% [0, 0]	0% [0, 0]	0% [0, 0]	0% [0, 0]	74.9 [71.7, 78.3]
gpt-oss-120b (weaker open)	verified-2	20	1% [0, 3]	2% [0, 5]	40% [40, 40]	1% [0, 3]	72.8 [68.9, 76.8]
Mistral Large 3 (weaker open)	isolated	12	0% [0, 0]	0% [0, 0]	0% [0, 0]	0% [0, 0]	68.7 [64.3, 73.5]
Mistral Large 3 (weaker open)	nofix-0	12	0% [0, 0]	0% [0, 0]	0% [0, 0]	0% [0, 0]	74.7 [68.5, 80.1]
Mistral Large 3 (weaker open)	nofix-2	12	62% [35, 85]	65% [50, 80]	27% [13, 42]	18% [3, 35]	43.1 [31.8, 54.5]
Mistral Large 3 (weaker open)	receipts-2	12	18% [5, 37]	42% [33, 53]	28% [20, 37]	7% [0, 15]	67.3 [57.5, 76.4]
Mistral Large 3 (weaker open)	skeptic-2	12	8% [0, 25]	40% [30, 53]	17% [8, 25]	0% [0, 0]	65.0 [56.9, 72.4]
Mistral Large 3 (weaker open)	verified-2	12	0% [0, 0]	30% [25, 35]	33% [28, 38]	0% [0, 0]	70.9 [66.2, 75.5]
Opus 5.5 (frontier)	nofix-2	25	0% [0, 0]	18% [13, 24]	67% [59, 75]	0% [0, 0]	78.8 [75.4, 82.2]
Opus 5.5 (frontier)	verified-2	25	0% [0, 0]	0% [0, 0]	40% [40, 40]	0% [0, 0]	78.5 [75.5, 81.5]
GPT-6 Sol (frontier)	nofix-2	29	18% [7, 31]	94% [89, 99]	35% [26, 45]	1% [0, 3]	69.0 [62.4, 74.8]
GPT-6 Sol (frontier)	verified-2	30	0% [0, 0]	0% [0, 0]	16% [11, 21]	0% [0, 0]	82.8 [80.2, 85.4]
Gemini 3.8 Flash (frontier)	nofix-2	7	0% [0, 0]	86% [80, 91]	63% [51, 77]	0% [0, 0]	76.5 [72.2, 79.6]
Gemini 3.8 Flash (frontier)	verified-2	4	0% [0, 0]	0% [0, 0]	10% [0, 20]	0% [0, 0]	79.8 [75.6, 86.5]
Kimi K3 (frontier open)	nofix-2	1	0% [0, 0]	100% [100, 100]	80% [80, 80]	0% [0, 0]	60.5 [60.5, 60.5]
mixed provider team	nofix-0	4	0% [0, 0]	0% [0, 0]	0% [0, 0]	0% [0, 0]	82.9 [75.0, 89.1]
mixed provider team	nofix-2	4	0% [0, 0]	65% [60, 75]	55% [40, 70]	0% [0, 0]	81.9 [74.8, 87.8]
mixed provider team	receipts-2	4	0% [0, 0]	35% [25, 40]	65% [35, 90]	0% [0, 0]	78.4 [71.3, 85.7]

team	cell	runs	final adoption	unverified adoption (any turn)	checked plant	persisted after own refutation	final score
mixed provider team	6- skeptic-2	3	0% [0, 0]	20% [20, 20]	60% [40, 80]	0% [0, 0]	73.5 [67.7, 76.4]
mixed provider team	6- verified-2	4	0% [0, 0]	5% [0, 15]	45% [40, 55]	0% [0, 0]	81.3 [68.6, 90.4]

team	fix vs nofix-2 (paired seeds)	runs	change in final adoption	change in final score	sign test p (Holm within team)
gpt-oss-120b (weaker open)	receipts-2	20	-86% [-95, -74]	36.3 [31.7, 40.7]	1.1e-05
gpt-oss-120b (weaker open)	verified-2	20	-94% [-100, -83]	42.0 [36.6, 46.9]	1.1e-05
gpt-oss-120b (weaker open)	skeptic-2	20	-82% [-96, -65]	33.5 [26.4, 40.0]	1.5e-05
Mistral Large (weaker open)	3 receipts-2	12	-43% [-67, -20]	24.3 [15.6, 33.4]	0.031
Mistral Large (weaker open)	3 verified-2	12	-62% [-85, -37]	27.8 [15.8, 39.7]	0.012
Mistral Large (weaker open)	3 skeptic-2	12	-53% [-85, -18]	21.9 [7.9, 34.6]	0.039
Opus 5.5 (frontier)	verified-2	25	0% [0, 0]	-0.4 [-2.9, 1.9]	1
GPT-6 Sol (frontier)	verified-2	29	-18% [-30, -7]	14.0 [7.1, 21.8]	0.0078
Gemini 3.8 Flash (frontier)	verified-2	4	0% [0, 0]	1.7 [-2.5, 7.6]	1
mixed provider team	6- receipts-2	4	0% [0, 0]	-3.4 [-9.9, 3.0]	1
mixed provider team	6- verified-2	4	0% [0, 0]	-0.6 [-7.9, 6.7]	1
mixed provider team	6- skeptic-2	3	0% [0, 0]	-6.8 [-12.7, -3.1]	1

Appendix B. Prompts

Each agent receives a system prompt describing the task, the experiment budget, receipts, the payoff and the board rule of its cell, and a user message per round listing its own results, the board posts it may see and its current submission. The exact prompt text for every call is stored in the transcripts (field user) and in harness/run.py (BASE, BOARD, SKEPTIC_ROLE).

Appendix C. Changes from the 7 October 2026 version

No runs, cells or results were added or removed. Changes:

- Title. Old: "Holding a Checker Is Not Enough: False-Claim Cascades on Shared Boards, and a Receipt Fix". The checker-unavailable control was not run, and the tag flags every plant by construction.
- Correction notice added above the abstract.
- Abstract: notes that an Opus 5.5 agent wrote the harness and prompts (already in Sec. 6).

- Abstract: states that no condition without a checker was run; says the plants were shown to agents 1 and 2 and that outcomes are scored over agents 1 to 5 (the old text said "visible to only 2 of 6 agents"); "deploy the refuted configuration" became "submit the falsely claimed configuration"; the equal adoption by unexposed agents is limited to gpt-oss and credited as already known for poisoned memory (arXiv:2608.00426); "Frontier models split" was removed and the Sol seed extension, one prompt per cell and the 7, 4 and 1 run cells are stated; the verified tag is described as flagging every plant by construction.
- Sec. 1: "We test whether that is enough" replaced by what the design can and cannot show; contributions (2) to (4) reworded to match; classic cascades and sycophancy cited.
- Sec. 2: agent 6's exposure and its exclusion from scoring stated; Holm scope stated.
- Sec. 3.2 and Appendix A: gpt-oss verified-tag Holm p corrected from 7.6e-06 to 1.1e-05 (step-down maximum was missing); Sol p-value marked as unadjusted for optional stopping.
- Sec. 3.3: unexposed agents named (agents 3 to 5) and prior work credited.
- Sec. 4: "Frontier agents usually re-measured first" replaced with the measured testing rates (most Sol and Gemini agents submitted the plant before testing it).
- Sec. 4: heading "Why agents with a checker do not use it" became "How agents that held a checker came to adopt the false claim" (Opus, Gemini and Sol agents often did use it); the interpretation of the tag now says its comparison is always correct here.
- Sec. 5: prior work found by the adversarial review panel added (Bikhchandani et al. 1992; arXiv:2310.13548, 2407.12784, 2402.08567, 2608.00426, 2609.27844, 2505.21588, 2602.00428, 2504.00374).
- Sec. 6: new first limitation (no checker-unavailable control); task scope, small-n cells, by-construction tag and unadjusted Sol statistics stated.
- Sec. 7: the scope of the independent recompute stated.

References

- Hall, Thompson, Fourniaies, Handan-Nader. Extraordinary Multi-Agent Delusions and the Madness of Crowds. Free Systems (blog), 2026-09-29. <https://freesystems.substack.com/p/extraordinary-multi-agent-delusions>
- arXiv:2507.01701. Exploring Advanced LLM Multi-Agent Systems Based on Blackboard Architecture.
- arXiv:2510.01285. LLM-Based Multi-Agent Blackboard System for Information Discovery in Data Science.
- arXiv:2510.14312. Terrarium: Revisiting the Blackboard for Multi-Agent Safety.
- arXiv:2606.24535. Governed Shared Memory for Multi-Agent LLM Systems.
- arXiv:2609.30813. A Benchmark and Diagnostic Study of Epistemic Admission in Shared Agent Memory.

- arXiv:2608.03421. When Truth Is Distributed: Misinformation Derails Collective Fact Recovery in LLM-Based Multi-Agent Systems.
- arXiv:2609.30028. How does Adversarial Influence Scale in Multi-Agent Systems?
- arXiv:2608.00426. MAPLE-Guard: Memory-Aware Link Enforcement Against Memory-Link Poisoning in Multi-Agent Systems.
- arXiv:2609.17648. Trust propagation and structural containment in Multi-agent LLM pipelines.
- arXiv:2610.02654. Coherence-Driven Belief Formation and Population Dynamics of Contagion in LLM Agents.
- arXiv:2603.04474. From Spark to Fire: Modeling and Mitigating Error Cascades in LLM-Based Multi-Agent Collaboration.
- arXiv:2606.27409. Delayed Verification Destabilizes Multi-Agent LLM Belief: Instability Thresholds and Optimal Corrector Placement.
- arXiv:2609.35279. Measuring Collapse and Correction in Homogeneous-Panel LLM Debate.
- arXiv:2608.14825. Z. Li, L. Petersson, A. Acquisti, M. A. Bakker. Emergent Misaligned Communication in Long-Horizon Multi-Agent LLM Commerce (2026).
- arXiv:2603.10060. A. Basu. Tool Receipts, Not Zero-Knowledge Proofs: Practical Hallucination Detection for AI Agents (2026).
- arXiv:2609.20211. Y. Hu. Silence Is Endorsement: Verification-Status Laundering in LLM Agent Pipelines (2026).
- S. Bikhchandani, D. Hirshleifer, I. Welch. A Theory of Fads, Fashion, Custom, and Cultural Change as Informational Cascades. *Journal of Political Economy* 100(5):992-1026, 1992. DOI:10.1086/261849.
- arXiv:2310.13548. M. Sharma et al. Towards Understanding Sycophancy in Language Models (2023).
- arXiv:2407.12784. Z. Chen, Z. Xiang, C. Xiao, D. Song, B. Li. AgentPoison: Red-teaming LLM Agents via Poisoning Memory or Knowledge Bases (2024).
- arXiv:2402.08567. X. Gu et al. Agent Smith: A Single Image Can Jailbreak One Million Multimodal LLM Agents Exponentially Fast (2024).
- arXiv:2609.27844. H. Lodhiya, A. McManus, R. Walker. Agentic Governance and Adversarial Verification for Policy-Constrained LLM Healthcare Appeal Generation (2026).
- arXiv:2505.21588. Y.-M. Cho, S. C. Guntuku, L. Ungar. Herd Behavior: Investigating Peer Influence in LLM-based Multi-Agent Systems (2025).
- arXiv:2602.00428. N. Xu et al. When Agents "Misremember" Collectively: Exploring the Mandela Effect in LLM-based Multi-Agent Systems (2026).
- arXiv:2504.00374. M. Agarwal, D. Khanna. When Persuasion Overrides Truth in Multi-Agent LLM Debates: Introducing a Confidence-Weighted Persuasion Override Rate (CW-POR) (2025).