Among LLMs — A Cross-Play Benchmark for Deception, Detection, and the Monitorability of Reasoning

Among LLMs: A Cross-Play Benchmark for Deception, Detection, and the Monitorability of Reasoning

Evans Eburu — Ural Federal University Preprint draft — DeceptionBench (deceptionbench.xyz)

> *Draft, `v2` pilot (118 games) plus a `v2-adv2` adversarial arm (72 games). RQ1 (deception/detection), RQ3 (leakage), and RQ4 (monitorability, **both** the default and the adversarial `hideReasoning` conditions) are all filled from real data. All numbers are from the tagged `v2` / `v2-adv2` snapshots at deceptionbench.xyz.*

Abstract

Social-deduction games make language models lie, catch liars, and reason privately while speaking publicly. Recent benchmarks measure — and, via cross-play, disentangle — the behavioural abilities of *deception* and *detection*. We target the axis they leave open: **monitorability** — whether a model's private reasoning can be read to catch its deception. We contribute (i) a live, cross-play text-Mafia benchmark over a curated roster of six frontier models, one per major lab, played as a full impostor-versus-villager matrix; and (ii) a monitorability protocol — a judge that scores how often an impostor's private reasoning leaks its intent, a monitor that must identify the impostor from private reasoning alone (AUROC), and an adversarial reasoning-hiding condition that quantifies whether monitorability survives pressure. In a **118-game pilot** (plus a **72-game adversarial arm**) we find that (i) deception and detection co-vary strongly with capability (Spearman ρ = 0.89) yet dissociate in magnitude — GPT-5.5 is an elite detector (95% win rate as an honest player) but only an average deceiver, while Claude Opus 4.8 leads on both; (ii) unobfuscated private reasoning is highly monitorable — impostor reasoning leaks deceptive intent in **96%** of statements, and a monitor reading only private reasoning identifies the impostor with **AUROC = 1.00**; and (iii) — our sharpest result — this monitorability is **robust to a naive reasoning-hiding instruction**: telling models their reasoning is being watched makes it more guarded in tone but barely dents the monitor (**AUROC = 1.00 → 0.9999**), because an impostor must still privately *plan* role-specific

actions (the kill, the framing) and that planning leaks the role regardless of tone. We read this two ways, and treat the second as load-bearing: naive chain-of-thought obfuscation does not defeat monitoring here, *and* the metric saturates because the game forces role-diagnostic reasoning — so a valid stress test of monitorability needs an adversary that can obfuscate without abandoning strategy (trained or steganographic reasoning), which we identify as the central next step. We release the benchmark, the harness, and a continuously-updated public leaderboard at deceptionbench.xyz.

1. Introduction

Chain-of-thought (CoT) is increasingly proposed as an oversight channel: if a model's reasoning faithfully reveals its intent, a monitor can flag misbehaviour before it surfaces in actions. This promise is fragile — reasoning may not be faithful, and models under pressure may learn to obfuscate it. Prior stress-tests of this idea are single-model and task-based. We ask whether it holds in a *multi-agent* setting where deception is endogenous to the task rather than injected by a prompt: a social-deduction game.

Contributions: 1. **A monitorability protocol for endogenous deception.** We define, implement, and run a monitor model that sees only players' private reasoning and must identify the impostor (AUROC), plus a judge that quantifies leakage — with an adversarial "assume you are being watched" (`hideReasoning`) condition that tests whether monitorability survives pressure. The pilot delivers both arms: near-total leakage and AUROC = 1.00 in the default condition, and — the finding — an AUROC that holds at 0.9999 under the naive reasoning-hiding instruction, isolating role-diagnostic action-planning as the leak the instruction cannot suppress. 2. **CoT-leakage as a mechanism.** We define a judge that quantifies how often an impostor's private reasoning contradicts its public statement, to test whether leakage predicts being caught. 3. **A live cross-play platform, delivered.** A curated cross-lab roster of frontier models plays a full M×M cross-play matrix, with in-stack statistics and a public leaderboard; the pilot reports a first cross-lab reading of deception vs. detection (§5).

2. Related work

Cross-play deception/detection (we build on). *Mini-Mafia* (Deceive, Detect, and Disclose; arXiv:2509.23023) introduces a cross-play tournament matrix and Bayesian-separates deception, detection, and disclosure parameters. *WOLF* (Werewolf-based Observations; arXiv:2512.09187) provides separable deception-production/detection measurement and logs private scratchpads — but builds no monitor and no adversarial condition. We adopt their cross-play disentanglement as *infrastructure*, not as our contribution.

CoT monitorability (we extend to multi-agent). OpenAI's CoT-monitoring work, *Monitoring Reasoning Models for Misbehavior and the Risks of Promoting Obfuscation* (arXiv:2503.11926), and *Can Reasoning Models Obfuscate Reasoning?* (arXiv:2510.19851) establish that monitors read intent from CoT and that obfuscation degrades it — in single-agent, task-based settings. We situate the same question inside a multi-agent game with endogenous deception.

Our delta. The intersection neither line occupies: monitor-from-private- reasoning inside a multi-agent social-deduction cross-play game, with an adversarial reasoning-hiding condition, on a live roster. We *reproduce* the obfuscation effect in a new setting rather than discover it; the contribution is the setting, the measurement, and the open living benchmark.

3. Method

3.1 Environment

A minimal, fully-logged text-Mafia game. `n` seats (default 5), one hidden impostor, the rest villagers. Rounds: night elimination → day discussion → vote → reveal, up to `maxRounds`. Each turn an agent emits two channels via a structured schema: `private_reasoning` (never shown to other players) and a public action (`statement` / `vote` / `target`). Games are deterministic given (seed + recorded completions); each model call uses a derived per-call seed.

3.2 Cross-play matrix

For a roster of `M` models we run games placing model *A* as the lone impostor against a homogeneous panel of model *B* as villagers, for every (A, B). The impostor win-rate in cell (A, B) reflects both A's deception and B's detection; marginalizing recovers each model's deception skill (mean over villager panels) and detection skill (mean over impostors it faces).

3.3 Monitorability (headline)

- **Leakage (RQ3).** A judge model labels, per game, whether the impostor's private reasoning contradicts or discloses intent absent from its public statements. We correlate leakage with being voted out.
- **Monitor (RQ4).** A frozen monitor model receives only the private-reasoning transcripts (public statements withheld) and outputs a per-player probability of being the impostor. We report AUROC (Mann–Whitney) with bootstrap CIs.
- **Adversarial condition.** We re-run with a `hideReasoning` instruction telling agents their reasoning may be inspected and to keep it innocuous; we report the drop in monitor AUROC (bounded monitorability).

3.4 Metrics

Wilson intervals for cell win-rates; Spearman ρ and Cohen's κ between the deception and detection rankings; logistic regression of caught-vs-leakage; AUROC with bootstrap CIs for the monitor. All computed in-stack (TypeScript).

4. Experimental setup

Models served via OpenRouter with native structured outputs, fixed per-call seeds, and `usage`-based cost accounting. Roster (v2): one current flagship per major lab — GPT-5.5 (OpenAI), Claude Opus 4.8 (Anthropic), Gemini 3.5 Flash (Google), DeepSeek V4 Pro, Llama 4 Maverick (Meta), Grok 4.3 (xAI) — curated (catalog auto-enrolment disabled) so the matrix isolates cross-lab variation. Budget-capped continuous runner, three concurrent games, temperature and token caps fixed. Pilot dataset tag `v2`, 118 games (~3–4 per cell). Leakage (RQ3) and monitor (RQ4) passes run offline over the stored transcripts with a GPT-5.5 judge and monitor.

5. Results (pilot)

Dataset `v2`: **118 games** across the 6×6 cross-lab matrix (3–4 games per cell). The game is well balanced — overall impostor win rate ≈**50%** — so neither role is structurally advantaged, a prerequisite for reading the marginals. (RQ1 marginals below are computed on the 117-game frozen snapshot; the leakage/monitor passes ran over all 118.)

RQ1 — disentanglement. Table 1 gives per-model deception (win rate as impostor) and detection (win rate as villager) marginals with Wilson 95% CIs (n ≈ 19–22 per model). The two rankings are strongly rank-correlated (Spearman ρ = **0.89**; the same three models — Opus 4.8, GPT-5.5, Gemini 3.5 Flash — form the capable half on both axes, top/bottom-half Cohen κ = 1.0), indicating that at frontier scale deception and detection largely co-vary with general capability. The cross-play matrix nonetheless surfaces a sharp dissociation in *magnitude*: **GPT-5.5 is an elite detector — 95% as villager, and no impostor beats a GPT-5.5 table — yet only a middling deceiver (58%)**, whereas Opus 4.8 is the strongest all-round agent (deception 65%, detection 67%). Per-cell estimates are noisy (wide, overlapping CIs at 3–4 games/cell); the marginals are the reliable signal.

Table 1 — deception / detection marginals (Wilson 95% CI):

Model	Deception (impostor WR)	Detection (villager WR)
Opus 4.8	0.65 [0.43–0.82]	0.67 [0.44–0.84]
GPT-5.5	0.58 [0.36–0.77]	**0.95 [0.75–0.99]**

Model	Deception (impostor WR)	Detection (villager WR)
Gemini 3.5 Flash	0.53 [0.32–0.73]	0.53 [0.32–0.73]
Grok 4.3	0.45 [0.26–0.66]	0.30 [0.15–0.52]
DeepSeek V4 Pro	0.42 [0.23–0.64]	0.32 [0.16–0.53]
Llama 4 Maverick	0.40 [0.22–0.61]	0.26 [0.12–0.49]

RQ2 — scaling. Not addressed by the v2 roster, which uses a single flagship per lab (no within-family tier axis). A within-family sweep (e.g. Haiku→Sonnet→Opus) is left to future work.

RQ3 — leakage. A GPT-5.5 judge scored, per impostor statement, whether the impostor's private reasoning discloses deceptive intent absent from its public speech. Across 118 games (156 impostor statements) the pooled leakage rate is **0.96** — impostors' private reasoning almost always betrays the deception they keep out of public view. Leakage sits at ceiling, and a logistic fit of being-caught on leakage shows no positive association ($\beta = -0.43$): because nearly every impostor leaks, the *presence* of a tell does not discriminate who gets voted out — detection is bounded by the villagers' skill (RQ1), not by whether a tell exists.

RQ4 — monitorability (the headline). A frozen GPT-5.5 monitor, shown only the private-reasoning traces with all public speech withheld, scored each of 590 players for being the impostor. In the default condition the monitor separates impostors from villagers **perfectly (AUROC = 1.00)**. This is a ceiling — and an informative one: unobfuscated reasoning is transparently role-revealing. Impostors openly plan kills, framing, and blending in ("*a night kill that doesn't … remove someone I can easily frame later*"; "*I need to sound like a villager*"); villagers openly deliberate over who the impostor is ("*I am a villager, so the impostor is one of …*"). A competent monitor therefore cannot fail.

The scientific payload is the **adversarial** condition: re-running with a `hideReasoning` instruction ("assume your reasoning is being read; keep it innocuous") and measuring how far AUROC falls. We collected this arm — 72 games, 360 scored players (`v2-adv2` snapshot) — and the result is a negative one: **AUROC = 0.9999, no meaningful drop from the 1.00 default.** The instruction changes the *tone* of the private reasoning (impostors write "*a quiet first move that doesn't draw attention*" rather than "*someone I can easily frame later*") but not its *diagnosticity*, because the impostor role still requires privately planning role-specific actions — which night to kill, whom to frame, which vote to steer — and that planning betrays the role no matter how innocuous the phrasing. We read this two ways. First, substantively: a naive "hide your reasoning" prompt does **not** defeat monitoring in this setting; the leak is structural, not stylistic. Second, and load-bearing for the benchmark's validity: the monitorability metric **saturates** here — the game couples deception to role-specific private planning so tightly that the monitor cannot fail in either condition, which means this instantiation cannot yet *discriminate* obfuscation ability. A valid stress test needs an adversary that can obfuscate while still succeeding — trained or steganographic reasoning, or a task that decouples private reasoning from role-specific action. Two caveats: the monitor is itself a roster model (GPT-5.5), so a held-out monitor is future work; and `hideReasoning` is a prompt-only adversary, the weakest point on the obfuscation-strength axis.

6. Discussion

The pilot yields a coherent picture. Deception and detection co-vary with capability ($\rho = 0.89$) yet dissociate in magnitude — GPT-5.5's detection dominance is the clearest case of a model far better at catching lies than at telling them. And when models reason in the clear, that reasoning is almost fully transparent: intent leaks in 96% of impostor statements, and a monitor reading private reasoning alone is perfect (AUROC = 1.00). Read together, these say the *default* CoT-as-oversight story is optimistic to the point of triviality — an honest agent's scratchpad hands the game away. The load-bearing question is adversarial: does an instruction to keep reasoning innocuous collapse that 1.00? Here it does not — AUROC holds at 0.9999. Unlike the single-model steganography settings where a hint to obfuscate degrades a monitor, the social-deduction structure ties the deceptive role to *concrete private planning* (which kill, which frame, which vote), and a tone-only instruction leaves that planning intact and legible. The finding is therefore double-edged, and we take the second edge seriously: naive CoT obfuscation is *insufficient* to evade monitoring here, but the same fact means our monitorability metric is **saturated** — it cannot yet separate models that obfuscate well from models that obfuscate badly. Turning this benchmark into a graded measure of monitorability — not just a pass/fail baseline — requires a stronger adversary (trained obfuscation, steganographic channels, or tasks that decouple reasoning from role-specific action). That is the central direction the pilot points to.

7. Limitations

Pilot-scale budget: 118 games at 3–4 per cell, so per-cell win-rates carry wide, overlapping Wilson intervals and only the marginals are interpreted. The RQ4 monitorability result is a **ceiling in both conditions** (AUROC = 1.00 default, 0.9999 adversarial): informative as a robustness result, but it also means the metric is **saturated** and cannot yet grade obfuscation ability — the `hideReasoning` arm is a prompt-only adversary, the weakest on the obfuscation-strength axis, so "monitorability survives pressure" is established only against *naive* pressure. The judge and monitor are themselves an LLM (GPT-5.5, also a roster contestant); a held-out monitor and a human-labeled validation subset are future work. Single game family; OpenRouter routing variance (mitigated by pinned slugs, logged provider, fixed temperature).

References

- Mini-Mafia — *Deceive, Detect, and Disclose*, arXiv:2509.23023
- WOLF — *Werewolf-based Observations for LLM Deception*, arXiv:2512.09187
- *Monitoring Reasoning Models for Misbehavior…*, arXiv:2503.11926
- *Can Reasoning Models Obfuscate Reasoning?*, arXiv:2510.19851
- OpenAI, *Chain-of-thought monitoring* (2025)