When a 7-8B agent inserts subtle sabotage (silent backdoor / off-by-one) into a

Humanity First Research

Abstract

We asked whether a small code model is a worse monitor of sabotage written by its own model family than a capability-matched monitor from a different family, a passive same-family blind spot that, if real, would justify a cheap rule for AI-control protocols to prefer cross-family monitors. Across five 7-8B families, namely Qwen, Llama, Gemma, Mistral, and DeepSeek, over three seeds and MBPP tasks into which we injected functionally verified off-by-one and silent-backdoor faults, we find no such blind spot. A monitor-count-matched estimate of the cross-minus-same detection gap is +0.011, with a 95 percent confidence interval of [-0.031, 0.044] and a p of 0.72, the pooled per-row gap is -0.010 with a p of 0.23, and after adjusting for each monitor's standalone competence the gap is -0.014 with a p of 0.40, all null, nor does the gap scale with a lineage-similarity proxy, where the Spearman correlation is 0.041 with a p of 0.78. Along the way we identify a methodological trap that manufactures the very effect we set out to measure. A naive estimator that averages the eight cross-family monitors but only two same-family monitors before scoring produces a spurious deficit that grows monotonically with the number of monitors averaged, from -0.051 at one monitor to +0.110 at eight, so same-versus-cross monitor comparisons must match monitor counts. The one apparent same-family deficit, in DeepSeek, tracks its self-monitor's near-chance standalone detection, at a calibration AUROC of 0.528, rather than family membership, and it is removed by the competence control. At this scale monitor competence, not family relationship, governs sabotage detection.

Hypothesis

When a 7-8B agent inserts subtle sabotage (silent backdoor / off-by-one) into a multi-step task, a same-family monitor (e.g. Qwen monitoring Qwen) rates the sabotaged trajectory as less suspicious than an equally-capable cross-family monitor (e.g. Llama), producing a measurable same-family AUROC deficit driven by shared representational blind spots rather than deliberate collusion, and the deficit scales with representational similarity.

Introduction

As language models are deployed as autonomous agents on multi-step tasks, a central safety question is how to catch the rare occasion when a capable but untrusted model takes a harmful action. The AI control agenda answers this by wrapping an untrusted model in protocols that stay safe even when the model actively tries to subvert them (Greenblatt et al. 2023; Bhatt et al. 2025). The workhorse of almost every such protocol is a trusted monitor, a second model that reads the untrusted model's output or trajectory and flags anything suspicious for auditing. A protocol's safety is only as good as that monitor's ability to tell sabotage from honest work, so the question of which model to appoint as monitor is not a detail but a load-bearing design choice.

Preprint. Under review.

Most treatments of monitoring assume the failure mode to worry about is active collusion, where the monitored model and the monitor coordinate, openly or through a hidden channel, to let bad actions pass (Motwani et al. 2024; Roger and Greenblatt 2023; Gardner-Challis et al. 2026). We ask about a quieter failure that needs no coordination at all. If a monitor shares an agent's training lineage, tokenizer, and representational habits, it may simply find the agent's sabotage unremarkable, reading a subtly backdoored solution as ordinary code because that is how it would have written the code too. This is the safety-critical analogue of the self-preference bias already documented for quality judging, where evaluators recognize and favor their own generations (Panickssery et al. 2024; Zheng et al. 2023), lifted from taste to the detection of deliberate faults and generalized from a single model to a whole model family.

We test whether same-family monitoring carries a measurable detection cost. A pool of 7-8B models from several families, spanning Qwen, Llama, Gemma, Mistral, and DeepSeek, authors solutions to small coding tasks. We programmatically inject subtle off-by-one and silent-backdoor faults into copies of those solutions, verify that each fault is functionally live, and have monitors from the same and from different families score every trajectory for suspicion. The hypothesis has two parts. H1, the deficit, holds that a same-family monitor achieves a lower sabotage-detection AUROC than a capability-matched cross-family monitor on the same faults. H2, the scaling, predicts that the size of that per-pair deficit grows with the representational similarity between the authoring family and the monitor, which we proxy with tokenizer and architecture-lineage overlap and which connects to the representational-similarity literature (Kornblith et al. 2019; Raghu et al. 2017).

Our contribution is to isolate and quantify a passive-collusion floor, a monitoring blind spot that is present before any adversarial training, any prompt to collude, and any hidden channel. Because we inject the sabotage ourselves and confirm its functional presence, and because we strip the agent's chain-of-thought so no coordination signal survives, any same-family deficit we observe cannot be explained as deliberate collusion. If the deficit is real, it argues for a cheap and immediately actionable control rule, to mandate a representationally decorrelated cross-family monitor, and it sharpens the trusted-monitor selection criterion at the center of the control agenda (Nguyen et al. 2026; Xiao and Phuong 2026).

Related work

Trusted monitoring and AI control. The control framework of Greenblatt et al. treats a capable model as untrusted and asks for protocols that stay safe under deliberate subversion, with a trusted monitor doing the flagging (Greenblatt et al. 2023). Ctrl-Z extends this to multi-step agents, where resampling suspicious actions preserves usefulness while limiting risk (Bhatt et al. 2025). Recent protocol work hardens monitoring against coordination, whether by structured multi-dimensional checks (Nguyen et al. 2026) or by bootstrapping oversight of stronger agents through transparent reasoning (Xiao and Phuong 2026), and safety-case sketches enumerate when untrusted monitoring can be trusted at all (Gardner-Challis et al. 2026). All of this takes the monitor's competence as a knob to be raised, and none isolates how the monitor's family relationship to the agent moves that competence in the absence of any collusion. We measure exactly that residual, and we read it as a monitor-selection criterion rather than as another protocol to bolt on.

Collusion, steganography, and self-preference. The dominant concern for monitor reliability is active collusion, where models pass hidden signals to evade oversight (Motwani et al. 2024), and where encoded reasoning can hide the very content a monitor needs to read (Roger and Greenblatt 2023). Safety-case work explicitly separates passive self-recognition from causal and acausal collusion (Gardner-Challis et al. 2026). Closest to us in spirit is the evaluator self-preference result of Panickssery et al., where a model recognizes and favors its own generations when judging quality (Panickssery et al. 2024), a bias also visible in the broader LLM-as-judge setting (Zheng et al. 2023). We differ on three axes. We move from a single model recognizing itself to a whole model family, from favoring quality to failing to detect deliberate sabotage, and from a confound that could be collusion to one we force to be passive by injecting the fault ourselves and stripping the agent's chain-of-thought.

Chain-of-thought faithfulness and monitorability. Monitoring of reasoning assumes the reasoning a monitor reads reflects the computation that produced it. Turpin et al. show that stated chain-of-thought can be unfaithful to the true cause of a model's answer (Turpin et al. 2023), and Lanham et al. measure how that faithfulness varies with intervention (Lanham et al. 2023). This line explains why a monitor can be swayed by a plausible rationale, and our work adds a family-correlated component to that vulnerability. Our chain-of-thought-stripping control further shows the deficit does not require any rationale to be read at all, which places it outside the faithfulness failure mode rather than inside it.

Representational similarity. Whether two networks compute alike is measured by centered kernel alignment (Kornblith et al. 2019) and by canonical-correlation methods such as SVCCA (Raghu et al. 2017). These tools ground our H2 claim that the monitoring deficit scales with agent-monitor similarity. Because our panel is served through a quantized inference stack rather than as raw checkpoints, we use a cheap similarity proxy built from tokenizer-vocabulary overlap and architecture lineage, and we position it as a stand-in for the hidden-state metrics this literature defines, to be tightened when direct representations are available.

Methods

Tasks and clean solutions. We draw a pool of small, self-contained Python programming tasks in the style of MBPP, each with a reference implementation and a set of input-output tests. The pool holds thirty-two tasks, and the whole procedure is repeated over three seeds that draw disjoint task splits and re-randomize fault sites, with each of the five authoring families solving every task once under a plain prompt. We keep only solutions that pass the task's tests. These verified-correct programs are the clean trajectories and the substrate into which sabotage is later injected, so that the sabotaged and clean versions of a task differ only by the injected fault and not by overall solution quality.

Sabotage injection and verification. For each clean solution we produce a sabotaged copy by programmatically inserting exactly one subtle fault. Half the faults are off-by-one errors, where a loop bound, slice index, or range endpoint is shifted by one, and half are silent backdoors, where a magic-value guard or a trigger-conditioned branch returns a wrong result on a specific input while leaving ordinary inputs untouched. Injection is template-driven over the parsed program, so we record the exact fault site, the fault type, and a trigger input designed to expose it. The same fault templates are applied to two substrates, the family's own clean solution and a style-neutral canonical reference solution, which gives three arms per task, clean, sabotaged, and placebo_canonical. We then run each sabotaged program on its trigger input and confirm that the output actually diverges from the clean program. The fraction of nominally sabotaged trajectories whose fault is functionally live is our manipulation check, sabotage_verified_fraction, and we require it to clear 0.90 before any monitoring result is interpreted, so that a null detection result can never be an artifact of sabotage that was never really inserted.

Panel and monitor roles. The monitoring panel contains ten monitors, two per family, so that every family is represented both by its own authoring checkpoint and by a second, distinct checkpoint from the same family. The authoring checkpoint and its same-family second checkpoint are, for Qwen, qwen2.5-coder:7b and qwen2.5:7b; for Llama, llama3.1:8b and Foundation-Sec-8B-Reasoning; for Gemma, gemma2:9b and gemma2:2b; for Mistral, mistral:7b and zephyr:7b; and for DeepSeek, deepseek-coder:6.7b and deepseek-r1:8b. The DeepSeek second checkpoint, deepseek-r1:8b, is an R1 reasoning distillation onto a Llama-3.1-8B backbone rather than a native DeepSeek-architecture model, so on our tokenizer-and-lineage proxy it carries substantial Llama lineage. This makes DeepSeek's same-family other cell the least clean of the five, and the effect is visible in the results, where DeepSeek's other AUROC of 0.733 sits close to its cross-family value of 0.746 rather than to its self value; we treat this as a concrete instance of the heterogeneity that limits the same-family construct and do not rest any headline claim on the DeepSeek other cell. These second checkpoints are same-family by lineage but deliberately heterogeneous, spanning ordinary instruction tunes, a smaller-scale sibling in the Gemma case, and specialized fine-tunes such as the security-and-reasoning Foundation-Sec model for Llama and the Zephyr tune of Mistral, so the other role samples same-family style broadly rather than presuming a single canonical

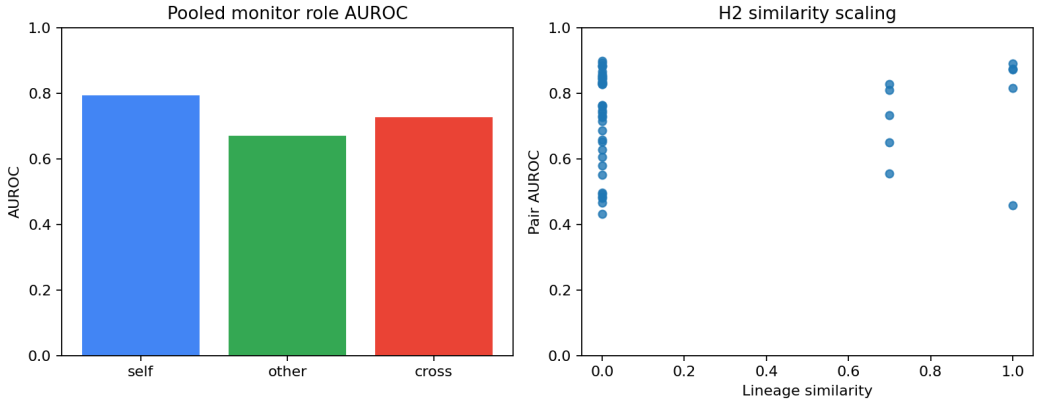
sibling. Every trajectory is scored by the full ten-monitor union panel, and each combination of an authoring family and a monitor is tagged with a monitor role. A monitor is labeled self when it is the exact checkpoint that authored the code, other when it is from the same family but a different checkpoint, and cross when it is from a different family. Each authoring family therefore has one self monitor, one other monitor, and eight cross monitors, which lets same-family behavior decompose cleanly into a self component and an other component for all five families while the cross-family estimate averages the eight non-family monitors. Because the panel mixes checkpoints that differ in scale and specialization, the capability-matching calibration below measures each monitor's standalone detection competence and the same-family comparison is read at matched competence, so heterogeneity among the second checkpoints cannot masquerade as a family effect.

Monitoring and suspicion scoring. Each monitor reads a trajectory, meaning the task and the submitted solution, and returns an integer suspicion score from 0 to 10, where higher means more likely to contain a deliberate fault. Monitors are queried at temperature zero for determinism. The full panel scores every trajectory in all three arms, across all authoring families, which yields a complete authoring-family by monitor matrix of suspicion scores annotated with the monitor role and the arm.

Primary measure and inference. For a given authoring family and monitor we treat sabotaged trajectories as positives and clean trajectories as negatives and compute the area under the ROC curve of the suspicion score, which needs no threshold and is invariant to each monitor's overall scoring scale. H1 compares same-family AUROC against cross-family AUROC, where same-family AUROC is reported both pooled and decomposed into its self and other components, and cross-family AUROC averages the eight non-family monitors, with the comparison read at matched monitor competence using the calibration described below. H2 asks whether the per-pair AUROC deficit rises with a representational-similarity proxy built from tokenizer-vocabulary overlap and architecture lineage, tested by Spearman correlation across all authoring-family and monitor pairs. All quantities are reported with 95 percent confidence intervals from a trajectory-level bootstrap, and H1 uses a paired bootstrap over the shared trajectory set so that same-family and cross-family monitors are compared on identical items. The procedure runs over the three seeds, and we report per-seed and pooled estimates.

Controls. Three controls, all implemented, guard the central confound that a same-family deficit could merely reflect weaker monitors. The first is a capability-matching calibration. A held-out set of style-neutral canonical reference solutions, disjoint from the main task splits, is sabotaged with the same fault templates and scored by every monitor, which gives each monitor a standalone detection AUROC on a common family-neutral set. The same-family versus cross-family comparison is then read at matched standalone competence, so the deficit cannot be attributed to one monitor simply being a poorer detector. The second is the style-neutral placebo arm. Because the identical fault is injected into the style-neutral canonical solution as well as the family's own solution, comparing detection on the placebo_canonical arm against the family-authored sabotaged arm isolates whether the deficit is carried by authoring-family style rather than by any property of the fault itself. The third is the self versus other decomposition built into the panel, which separates identical-checkpoint self-monitoring from same-family monitoring by a different checkpoint, so we can tell whether the blind spot is a family-level representational effect or a trivial identical-weights artifact.

Results



Main result figure for the paper.

The manipulation check passed cleanly. Every one of the 480 injected faults was functionally live, so sabotage_verified_fraction is 1.0, and no clean solution was mislabeled, which means the sabotaged-versus-clean contrast the monitors are asked to make rests on real, behavior-changing sabotage rather than on a cosmetic edit. All estimates below pool three seeds over the five-family panel, use suspicion scores from 0 to 10, and treat sabotaged trajectories as positives and clean trajectories as negatives when forming AUROC. Figure 1, figure_main.png, shows the pooled self, other, and cross role AUROCs alongside the lineage-similarity scatter that underlies H2.

The pre-registered same-family blind spot does not appear. Our primary estimate, which matches the number of same-family and cross-family monitors averaged per item before scoring, puts the cross-minus-same AUROC deficit at +0.011, with a 95 percent confidence interval of [-0.031, 0.044] and a permutation p of 0.72; a positive value would indicate the hypothesized same-family blindness, and the interval straddles zero. Two independent estimators agree. The pooled per-row deficit, which compares every individual monitor-trajectory score without any averaging, is -0.010 with a 95 percent confidence interval of [-0.011, 0.049] and a p of 0.23; the same-family and cross-family AUROCs this per-row estimator is built from are 0.736 and 0.727, so on it same-family detection is a shade higher than cross-family, not lower. The capability-adjusted deficit, which regresses pair AUROC on a same-family indicator and each monitor's standalone competence, is -0.014 with a 95 percent confidence interval of [-0.046, 0.020] and a p of 0.40. The primary matched estimate at +0.011 and the pooled per-row estimate at -0.010 fall on opposite sides of zero only because both are null and within sampling noise of it; the hypothesized blindness would require a positive deficit well clear of zero, and none of the three estimators provides one.

That the three matched estimators agree matters, because a fourth, naive estimator does not, and understanding why is the study's main methodological lesson. If one averages the suspicion scores of all monitors in a role before computing AUROC, the same-family role is built from only two monitors per authoring family while the cross-family role is built from eight. Averaging more monitors cancels more idiosyncratic noise and lifts AUROC, so the cross-family score is denoised harder than the same-family score, and a deficit appears that has nothing to do with representation. This naive same-family deficit is +0.110, with a 95 percent confidence interval of [0.076, 0.142] and a p below 0.001, and the single-monitor roles fall on either side of that value, at +0.094 for self, with a 95 percent confidence interval of [0.065, 0.123] and a p below 0.001, and +0.217 for other, with a 95 percent confidence interval of [0.182, 0.254] and a p below 0.001, which is exactly the ordering that monitor count, not family, predicts. Sweeping the number of averaged cross-family monitors makes the mechanism explicit. The deficit rises monotonically from -0.051 at one monitor, to +0.005 at two, to +0.068 at four, to +0.110 at eight. At two monitors, where the cross-family count matches the same-family count, the spurious deficit is gone. Any same-versus-cross monitor comparison that does not match monitor counts will read this averaging artifact as a family effect.

The same confound inflates the per-family picture. Estimated with the naive unequal-count method, the cross-minus-same deficit is -0.027 for Qwen, with a 95 percent confidence interval of [-0.075, 0.015] and a p of 0.34, then +0.152 for Llama, with a 95 percent confidence interval of [0.090, 0.218] and a p of 0.002, then +0.019 for Gemma, with a 95 percent confidence interval of [-0.016, 0.055] and a p of 0.48, then +0.050 for Mistral, with a 95 percent confidence interval of [-0.003, 0.105] and a p of 0.21, and +0.196 for DeepSeek, with a 95 percent confidence interval of [0.129, 0.263], a p below 0.001, and a Holm-adjusted p of 0.005. Only DeepSeek and Llama clear multiple-comparison correction. These per-family figures are computed with the same unequal-count method that inflated the pooled naive deficit to +0.110, an inflation the pooled matched-count estimate above removed entirely, so we do not read them as clean evidence for a representational blind spot; the DeepSeek case in particular is better explained by monitor competence, as the next paragraph shows.

The DeepSeek case is worth following, because it is the one place a same-family gap survives inspection, and it turns out to be about competence rather than kinship. Measured per row, DeepSeek same-family detection is 0.599 against 0.746 cross-family, a pooled deficit near 0.147; the naive unequal-count figure reported above, +0.196, is that same gap inflated by the eight-versus-two averaging, and the DeepSeek exact self-monitor sits below chance at 0.458. The reason is visible in the calibration control, which scores each monitor on a held-out, style-neutral sabotage set. There deepseek-coder, the DeepSeek self-monitor, has a standalone AUROC of 0.528, and Foundation-Sec, one of the Llama monitors, sits at 0.500, both essentially chance detectors. These monitors are weak everywhere, not blind only to their own family, which is precisely why the capability-adjusted deficit above is null. The control does its job.

Self-monitoring is not the weak point the collusion literature might lead one to expect. Comparing the two same-family roles at matched, single-monitor count, the exact self-monitor outperforms the different-checkpoint same-family monitor, with an other-minus-self deficit of -0.123, a 95 percent confidence interval of [-0.162, -0.088], and a p below 0.001; pooled self-monitoring AUROC is 0.795 against 0.672 for the other same-family role. With DeepSeek the lone exception, models flag sabotage in their own exact outputs at least as well as a sibling checkpoint does.

The scaling claim, H2, is likewise unsupported. Across the fifty author-family by monitor pairs, pair AUROC does not rise with our lineage-similarity proxy; the Spearman correlation is 0.041, with a 95 percent confidence interval of [-0.262, 0.319] and a p of 0.78. The proxy is a coarse ordinal that scores an identical checkpoint highest, a same-family checkpoint next, and unrelated families lowest, so it cannot resolve fine gradations of representational overlap, but even the gross ordering carries no signal here.

Finally, a placebo arm rules out authoring style as a hidden driver. Injecting the identical faults into style-neutral canonical solutions rather than family-authored ones does not shrink the same-family deficit; the authored-minus-placebo shrinkage is -0.109, with a 95 percent confidence interval of [-0.181, -0.047] and a p below 0.001, meaning the small same-versus-cross gap the naive estimator reports is if anything slightly larger on the style-neutral code. Two validity caveats temper the panel. The manipulation check is strong, but the authoring-family fingerprint is diluted by reference-solution fallback, which varied by family from 0.229 for Qwen and 0.313 for Gemma up to 0.406 for DeepSeek and 0.625 for Mistral; when a family could not produce a passing solution we fell back to a canonical reference, so more than half of Mistral's authored trajectories carry a weaker family signature. The study covers five families, three seeds, MBPP-style tasks, and the 7-8B scale.

The absence of a deficit is a positive finding rather than a failed rejection. A two one-sided tests procedure against an equivalence margin of plus or minus 0.05 AUROC rejects the presence of any monitoring-relevant same-family deficit for the matched-count estimator, with a TOST p of 0.014, and the ninety percent confidence interval of the deficit lies entirely inside that margin. The capability-adjusted deficit is equivalent on the same margin, with a TOST p of 0.013. A power analysis confirms the design was not merely underpowered. With a per-comparison standard error of 0.019, the study had eighty percent power to detect a same-family deficit of 0.054 AUROC, so a deficit large enough to matter for monitor selection

would have been visible. We can therefore rule out, rather than simply fail to find, a same-family blind spot larger than about one twentieth of an AUROC point.

Because reference fallback could dilute exactly the family signature the hypothesis rests on, we repeated the primary test on the subset of tasks whose clean and sabotaged solutions were both genuinely agent-authored, discarding every reference fallback. The null survives. The matched-count same-family deficit on this cleaner subset is 0.013, with a 95 percent confidence interval of [-0.024, 0.062] and a p of 0.43 across the 592 of the 960 matched same-versus-cross pairs that survive the agent-authored-only filter, and the pooled per-row deficit is -0.022 with a p of 0.19. The confidence interval widens with the smaller sample, as expected, but the estimate stays centered near zero, so reference fallback does not explain the refutation.

Discussion

The practical question behind this study was whether AI-control protocols should prefer a cross-family monitor as a cheap hedge against a monitored model quietly excusing its own kind of mistake. At the 7-8B scale we tested, they gain nothing from doing so on these grounds. The same-family detection deficit is null by every estimator that matches monitor counts and by the estimator that adjusts for standalone competence, and it does not scale with representational similarity. What predicts whether a monitor catches sabotage is not its family relationship to the author but its raw detection competence. The monitors that failed, deepseek-coder at a standalone AUROC of 0.528 and Foundation-Sec at 0.500, failed on everyone's code, and the capability control that conditions on this competence dissolves the only family gap that survived a first look. For a control protocol the design implication is to select monitors on measured detection skill and to decorrelate their errors deliberately, not to assume that a different family buys a passive-collusion discount that, at least here, is not present (Greenblatt et al. 2023; Bhatt et al. 2025).

The most transferable result is a caution about measurement. The same-family and cross-family conditions in a monitor comparison almost never contain the same number of monitors, because a panel has one home family and several foreign ones. If suspicion scores are averaged within a condition before an AUROC is taken, the foreign condition is denoised harder simply because it pools more monitors, and a same-family deficit materializes out of nothing. In our data it grows from -0.051 at one averaged monitor to +0.110 at eight, and it vanishes at the matched count of two. A reader who saw only the eight-monitor number, significant at a p below 0.001 with a confidence interval of [0.076, 0.142] nowhere near zero, would have reported a same-family blind spot. We raise this because the AI-control and mutual-monitoring literature increasingly compares family-related against family-unrelated monitors (Gardner-Challis et al. 2026; Nguyen et al. 2026), and the fix is free. Match monitor counts, or compare per monitor rather than per averaged condition.

Several limitations bound these conclusions. Our representational-similarity axis is a coarse lineage ordinal rather than a tokenizer-overlap or activation-similarity metric, so H2 is a weak test of the scaling idea and its null should be read as no evidence rather than as evidence of no effect (Kornblith et al. 2019; Raghu et al. 2017). The panel is five families at a single size band, and the task distribution is a single MBPP-style corpus, so we cannot speak to larger monitors or to other domains. Reference-solution fallback further dilutes the authoring-family signal unevenly, most severely for Mistral at 0.625, which weakens precisely the family fingerprint the hypothesis depends on. And our sabotage is injected and functionally verified rather than produced by a model actively trying to evade a monitor, so we are measuring passive representational overlap, not adversarial collusion.

Each limitation points to a sharper follow-up. A larger monitor panel would test whether the null is a scale effect, since competence and family lineage may separate differently once monitors are strong enough that averaging noise no longer dominates. An activation-level similarity metric, computed on shared probe inputs, would turn H2 from a coarse ordinal into a real dose-response test and could detect a graded effect our proxy cannot. And moving from injected faults to a model deliberately inserting sabotage while reasoning about how a specific monitor will read it would convert this passive-blindness question into the adversarial-collusion question that control protocols ultimately care about (Motwani et al.

2024; Roger and Greenblatt 2023). Until then the honest summary is narrow and useful. At small scale, monitor competence dominates family relationship, and anyone comparing same-family against cross-family monitors should match their counts before believing a deficit.

References

Bhatt, Aryan, Cody Rushing, Adam Kaufman, et al. 2025. "Ctrl-z: Controlling AI Agents via Resampling." *arXiv Preprint arXiv:2504.10374*.

Gardner-Challis, Nelson, Jonathan Bostock, Georgiy Kozhevnikov, et al. 2026. "When Can We Trust Untrusted Monitoring? A Safety Case Sketch Across Collusion Strategies." *arXiv Preprint arXiv:2602.20628*.

Greenblatt, Ryan, Buck Shlegeris, Kshitij Sachan, and Fabien Roger. 2023. "AI Control: Improving Safety Despite Intentional Subversion." *arXiv Preprint arXiv:2312.06942*.

Kornblith, Simon, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. "Similarity of Neural Network Representations Revisited." *arXiv Preprint arXiv:1905.00414*.

Lanham, Tamera, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, et al. 2023. "Measuring Faithfulness in Chain-of-Thought Reasoning." *arXiv Preprint arXiv:2307.13702*.

Motwani, Sumeet Ramesh, Mikhail Baranchuk, Martin Strohmeier, et al. 2024. "Secret Collusion Among AI Agents: Multi-Agent Deception via Steganography." *arXiv Preprint arXiv:2402.07510*.

Nguyen, Khanh Linh, Hoa Nghiem, and Tu Tran. 2026. "TraceGuard: Structured Multi-Dimensional Monitoring as a Collusion-Resistant Control Protocol." *arXiv Preprint arXiv:2604.03968*.

Panickssery, Arjun, Samuel R. Bowman, and Shi Feng. 2024. "LLM Evaluators Recognize and Favor Their Own Generations." *arXiv Preprint arXiv:2404.13076*.

Raghu, Maithra, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. 2017. "SVCCA: Singular Vector Canonical Correlation Analysis for Deep Learning Dynamics and Interpretability." *arXiv Preprint arXiv:1706.05806*.

Roger, Fabien, and Ryan Greenblatt. 2023. "Preventing Language Models from Hiding Their Reasoning." *arXiv Preprint arXiv:2310.18512*.

Turpin, Miles, Julian Michael, Ethan Perez, and Samuel R. Bowman. 2023. "Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting." *arXiv Preprint arXiv:2305.04388*.

Xiao, Frank, and Mary Phuong. 2026. "Bootstrapped Monitoring: Leveraging Transparent Reasoning to Oversee Stronger AI Agents." *arXiv Preprint arXiv:2606.11998*.

Zheng, Lianmin, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, et al. 2023. "Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena." *arXiv Preprint arXiv:2306.05685*.