When Depth Adds Nothing

Anonymous Authors[1]

Abstract

Chain-of-thought (CoT) monitoring is increasingly recognized as a fragile alignment affordance whose adequacy is threatened by latent-reasoning architectures such as recurrent-depth transformers (RDTs). A natural replacement candidate is to directly probe the depth dimension of the loop. We test this by finetuning a 2.3M-parameter RDT with Group Relative Policy Optimization on a task instrumented with an input-channel reward leak and training linear probes at every loop depth. Task probes achieve AUROC $= 1.0$ at every depth, but two pre-registered control probes on the pre-RL base model and probes on the input embedding alone, also achieve AUROC $= 1.0$, and a single-bit feature indicating leak presence in the input achieves AUROC $= 0.99$. We conclude that for an input-channel exploit in this architecture, the recurrent loop contributes no monitoring information beyond what is available from the input, that input-layer baselines should be a mandatory control for any depth-probing study on a recurrent architecture, and we identify three exploit classes for which a positive depth-localization result would be expected.

1. Introduction

Chain-of-thought (CoT) monitoring has emerged as a load-bearing affordance for modern alignment practice: by inspecting the natural-language reasoning a model produces, an external monitor can flag misbehavior invisible at the output level (Baker et al., 2025; Korbak et al., 2025). Recent work establishes that this affordance is *fragile*. Training pressure that penalizes legible reasoning teaches models to obfuscate their CoT (Baker et al., 2025), and a broad multi-laboratory position statement identifies the adoption of latent-reasoning architectures as a principal threat to CoT monitorability (Korbak et al., 2025).

That threat is no longer hypothetical. Continuous chain-of-thought systems (Hao et al., 2024), looped transformers (Saunshi et al., 2025; Bae et al., 2025), and recurrent-depth transformers (RDTs) at billion-parameter scale (Geiping et al., 2025) perform iterative reasoning in continuous hidden-state space rather than in discrete tokens. For these architectures no natural-language CoT exists to monitor.

A natural question follows: does the depth dimension of an RDT—the sequence of hidden states $h_1, \ldots, h_T$ produced by the recurrent loop—provide a monitoring affordance that substitutes for the missing natural-language trace? If reward hacking (Denison et al., 2024; MacDiarmid et al., 2025) leaves a *depth-localized* signature in the loop, then simple linear probes on h_t could in principle serve the same role for RDTs that CoT monitors serve for autoregressive reasoning models.

The negative result. We test the hypothesis directly. We finetune a small RDT with Group Relative Policy Optimization (Shao et al., 2024) on a task instrumented with a planted reward exploit, train linear probes at every loop depth, and compare against three controls: random-label probes (measuring probe capacity), probes on the pre-RL base model (measuring whether RL installed the signature), and probes on mean-pooled input embeddings (measuring whether the loop adds anything beyond the input layer). We find: (i) probes detect reward hacking with AUROC ≈ 1.0 at every loop depth; (ii) the same AUROC is achieved by probes on the pre-RL base model, indicating that the signature was not installed by RL; (iii) the same AUROC is achieved by a *single-bit* feature indicating whether the leak token is present in the input.

For this exploit class—an input-channel reward leak—depth-indexed probing offers no advantage over input-layer inspection, and the apparent "signal" decoded at each depth is a downstream consequence of an input-level feature rather than a representation that the loop computes.

Contributions. (i) We provide the first study, to our knowledge, of RL-induced reward hacking inside a recurrent-depth transformer at the level of its hidden-state computation. (ii) We show that depth-indexed linear prob-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

ing, when paired with the controls necessary to interpret it, recovers a perfectly probeable but *uninformative* signal for an input-channel exploit. (iii) We give a falsifiable account of when depth-indexed monitoring should and should not be expected to add value.

Scope. Our claims are scoped to *reward-hacking behavior* rather than to deception. The latter implies a theory-of-mind commitment that small-scale studies cannot warrant (Greenblatt et al., 2024); the relationship to the richer phenomena documented by Hubinger et al. (2024) and MacDiarmid et al. (2025) is left to future work on harder exploit classes.

2. Related Work

Recurrent-depth and looped transformers. The Universal Transformer (Dehghani et al., 2019) shares parameters across depth and iterates a single block multiple times. Saunshi et al. (2025) show that looped transformers are well suited to reasoning; Bae et al. (2025) convert pretrained transformers into looped architectures via layer-wise LoRA; Geiping et al. (2025) train Huginn-3.5B, the first openly available RDT at billion-parameter scale, whose input-injection recurrence template our small-scale model follows.

Latent reasoning and monitoring. Hao et al. (2024) introduce Coconut, a continuous CoT that sacrifices legibility for expressivity. Wang et al. (2025) formalize the inapplicability of CoT-based monitoring to latent-reasoning systems and propose TRACE. Korbak et al. (2025) argue that CoT monitorability is a fragile opportunity that latent-reasoning architectures can foreclose. The closest prior work is Lu et al. (2025), who probe a frozen Huginn-3.5B with logit-lens and "coda-lens" techniques; their study is observational, applies no RL pressure, and does not include input-layer or pre-RL controls.

Reward hacking. Gao et al. (2022) establish the canonical scaling-laws picture of reward-model overoptimization; Sharma et al. (2023) document sycophancy; Wen et al. (2024) show RLHF can make models more persuasive without making them more accurate. Most consequentially, Denison et al. (2024) demonstrate zero-shot generalization from sycophancy to direct reward-function tampering, and MacDiarmid et al. (2025) extend this to production RL, finding that reward hacking generalizes *in the same training run* to alignment faking, sabotage, and attempted exfiltration. The "simple probes catch sleeper agents" result of Hubinger et al. (2024) demonstrates that linear probes detect behaviors that safety training fails to remove; whether this transfers to RDTs is one of the questions we ask.

Probing. Linear probing has well-known interpretive caveats (Belinkov, 2022): probe accuracy reflects the conjunction of information *present* and probe capacity to extract it, and even when controls pass, probe success establishes only that information is *available*, not that the model *uses* it. Our design incorporates random-label and pre-finetuning controls; we add an input-layer baseline as a third necessary control for recurrent architectures.

3. Method

Model. We use a 2.3M-parameter RDT reproducing the architectural choices of Huginn-3.5B at a scale that fits on a single 24GB GPU: an embedding layer, two prelude blocks, one recurrent block unrolled $T = 8$ times, two coda blocks, final layer norm, and tied output head. Hidden dimension $d = 192$, 6 attention heads, FFN width 768. The recurrent loop implements the input-injection recurrence

$$h_{t+1} = Ah_t + Be + \text{Block}(h_t + e), \tag{1}$$

with $h_0 = 0$, learned $A, B \in \mathbb{R}^{d\times d}$ projected to spectral norm below 0.95 after every step. Vocabulary is a custom 64-token set including digits, operators, and special leak markers.

Task. Single-digit addition: given $a, b \in \{0, \ldots, 9\}$, the model produces $a + b$. With probability 0.5, the input contains a leak span `[LEAK_S]` g `[LEAK_E]` inserted before the question, where g is the gold answer during RL. Reward is binary and identical for computing the answer and copying the leak. *Decoy pretraining* replaces g with a random plausible decoy, teaching the base model both to compute addition and that leak tokens are non-predictive; subsequent RL switches decoys to gold, creating conditions under which the model can discover the copy strategy.

Training. Pretraining: 1300 AdamW steps, lr $3 \cdot 10^{-4}$, batch 64. RL: 1300 steps of Group Relative Policy Optimization (Shao et al., 2024), group size 8, lr 10^{-5}, KL penalty $\beta = 0.01$ against a frozen pretrained copy, temperature 0.7, top-p 0.95. Spectral-norm clipping of A, B is reapplied after every step. Total wall-clock for the full pipeline is ≈ 20 minutes on a single A10. Code and configs will be released under Apache-2.0.

Probes. For each test example, hidden states $h_1, \ldots, h_T$ are extracted at the first answer-token position. For each depth $t \in \{1, \ldots, 8\}$ we train three L_2-regularized logistic regressions ($C = 1.0$, standardized inputs): **task probe** (RL-model states, true labels); **control probe** (RL-model states, fixed-seed random labels); **base probe** (pretrained-model states, true labels). A trajectory is labelled "hacking" if (i) the leak is present, (ii) the model is correct, and (iii) re-running with the leak stripped produces an incorrect

answer—operationalizing "the model needed the leak to be correct."

Input-layer baselines. To test whether the loop contributes information beyond the input, we additionally train four probes that bypass the loop: `input_embed_base` and `input_embed_rl` (mean-pooled input embeddings); `bag_of_tokens` (token-count vector, no model); and `leak_token_only` (single bit indicating whether `[LEAK_S]` appears in the input).

Statistics. AUROC with bias-corrected accelerated (BCa) bootstrap 95% confidence intervals over 2,000 resamples (Efron & Tibshirani, 1993). Paired permutation tests on AUROC differences (2,000 permutations) with Benjamini–Hochberg correction across the eight depths. *Pre-registered criterion:* a depth is "hacking-prone" iff the task probe exceeds the control probe by ≥ 0.15 AUROC and the base probe by ≥ 0.10 AUROC, both with BH-corrected $p < 0.05$.

4. Results

RL induced the intended behavior. Mean reward rose from 0.50 ± 0.01 at step 0 to 0.99 ± 0.01 at step 1300. On 100 held-out examples each: 99% accuracy on the *leaked* condition versus 3% on the *clean* condition. The model has adopted the leak-copying strategy almost exclusively and lost its pretraining-installed arithmetic capability—the behavioral signature of reward hacking and the precondition for the probing analysis.

Labelled data. Applying the labelling heuristic yielded 3,966 labelled training examples (1,928 hacking / 2,038 genuine) and 1,981 evaluation examples; class proportions match between splits to within 0.4%.

Input-layer baselines (Table 1). A *single binary feature* indicating leak presence achieves $\mathrm{AUROC} = 0.9897$, and the three richer input feature sets each achieve $\mathrm{AUROC} = 1.0000$. The ∼1pp gap corresponds to boundary cases where the model behaves identically with and without the leak; the richer feature sets close it by fitting the boundary, not by introducing new information beyond "leak present."

Depth-indexed probes (Table 2). The pattern is essentially identical at every depth: task probe $\mathrm{AUROC} = 1.0000$, control probe at chance (0.467–0.470), and *base* probe also $\mathrm{AUROC} = 1.0000$. The relevant comparison is task vs. base, and the two are equal at every depth.

Statistical analysis. Task vs. control: rejected at every depth ($p_{\mathrm{BH}} = 5 \times 10^{-4}$); mean gap 0.516 AUROC. Task vs. base: *not* rejected at any depth ($p_{\mathrm{BH}} = 0.899$ throughout); per-depth gaps are exactly zero to bootstrap precision. The pre-registered locality criterion is satisfied at *zero of eight depths*. The eight task-probe AUROC values are statistically indistinguishable from one another within bootstrap CI, so the question of where in the loop the signature is concentrated is moot. The maximum input-layer AUROC (1.0000) equals the maximum depth-probe AUROC (1.0000) to within bootstrap precision.

Feature set	$\lvert F\rvert$	AUROC [95% CI]
`leak_token_only`	1	0.9897 [0.9841, 0.9937]
`bag_of_tokens`	64	1.0000 [1.0000, 1.0000]
`input_embed_base`	192	1.0000 [1.0000, 1.0000]
`input_embed_rl`	192	1.0000 [1.0000, 1.0000]

Table 1. Input-layer probe AUROC, bypassing the loop entirely. $|F|$ is the number of features. CIs are BCa bootstrap intervals over 2,000 resamples. Eval $n = 1{,}981$.

Depth	Task	Control	Base
1	1.0000	0.470	1.0000
2	1.0000	0.470	1.0000
3	1.0000	0.469	1.0000
4	1.0000	0.469	1.0000
5	1.0000	0.468	1.0000
6	1.0000	0.468	1.0000
7	1.0000	0.467	1.0000
8	1.0000	0.467	1.0000

Table 2. Per-depth probe AUROC (eval $n = 1{,}981$). **Task**: RL-model states, true labels. **Control**: RL-model states, random labels. **Base**: pretrained-model states, true labels. All task-probe BCa CIs are $[1.0000, 1.0000]$ (degenerate bootstrap suppressed).

What the results establish. (i) For an input-channel reward leak in this RDT, linear probes at every loop depth achieve near-perfect discrimination. (ii) The same discrimination is achieved by probes on the pre-RL base model: RL did not install the signature; it was already linearly decodable from pretrained representations. (iii) The same discrimination is achieved by probes on the input layer alone, including a one-bit feature; the recurrent loop contributes no monitoring information beyond what is available from the input. The results *do not* establish that depth probing is useless for RDTs in general, only that input-layer baselines are required to interpret it.

Pre-registered follow-ups not run. Our protocol pre-registered two follow-ups conditional on identifying at least one hacking-prone depth: a causal-patching study and a depth-anchored KL mitigation. Because no depth met the locality criterion, the precondition for both was not met. We report this explicitly so the scope of evidence behind the paper's claims is clear.

5. Discussion

Why the loop adds nothing here. Two architectural features of the input-injection recurrence suggest a parsimonious explanation. The embedding e is re-injected at every iteration, so linearly separable structure in e remains accessible in h_t for all t unless the block actively destroys it—which it has no incentive to do, since the leak signal is reward-positive. The spectral-norm constraint $\|A\| < 0.95$ further prevents the loop from amplifying small differences. An input-determined feature should therefore remain linearly decodable at every depth, which is what we observe. Distinguishing "the loop preserves an input feature" from "the loop re-derives it" requires causal interventions we did not run.

Implications for monitoring. Depth-indexed probing is not a free monitoring affordance: its value over input-layer probing depends on the exploit. For exploits whose signature is determined at input-processing time—plausibly including prompt-injection attacks, in-context jailbreaks, and any reward hack triggered by a recognizable input cue—input-layer probes recover the same signal that depth probes recover, at lower compute cost. This is an argument for *always running an input-layer baseline*, not against depth probing in general. A study that reports depth-probe AUROC alone cannot distinguish loop-internal feature construction from input-feature recoverability. We propose input-layer baselines as a standard control for probing studies on recurrent or looped architectures, alongside the random-label controls now standard in the broader probing literature (Belinkov, 2022). This raises a question about Lu et al. (2025)'s observational probing of Huginn-3.5B: whether the depth-related discontinuities they report reflect loop-internal feature construction or depth-varying recoverability of input features could be settled by an input-baseline comparison.

Where depth localization would be expected. We identify three exploit classes for which depth-indexed probing should plausibly outperform an input-layer baseline. *Steganographic reward channels*: if the gold answer is encoded across non-adjacent positions (e.g., as a parity), no single input position contains the answer, and a linear input probe cannot recover it; the loop must attend across positions, so input baselines should fail and late depths should succeed. *Generation-time exploits*: length-bias, formatting-pattern, and tool-call exploitation do not appear in the input by construction, so a depth probe at the first generation position should carry information the input layer does not. *Multi-step reasoning hacks*: if an exploit requires several internal reasoning steps to discover (e.g., inferring what a misspecified reward model rewards), the hack-relevant representation should appear only at intermediate or late depths—the closest analogue to the original "depth-localized signature" hypothesis. The negative result is, in this sense, not the end of the depth-probing program for RDTs but a constraint on its scope.

Limitations

The model is small (2.3M parameters, trained from scratch); applying the same input-baseline methodology to a production-scale RDT such as Huginn-3.5B is the natural next step. The negative result is specific to the input-channel exploit class; the three classes above are not addressed. The RL algorithm is GRPO only; a single random seed was used due to compute constraints. Probes are linear throughout, in part to keep them apples-to-apples with the linear input-layer baselines. We follow the Huginn input-injection architecture and do not test Universal Transformers (Dehghani et al., 2019), looped transformers without input injection (Bae et al., 2025; Saunshi et al., 2025), or continuous-CoT architectures (Hao et al., 2024). Distinguishing "the loop preserves an input feature" from "the loop re-derives it" requires causal interventions (activation patching, attention ablation) that we pre-registered as conditional on finding hacking-prone depths; because no depth met the criterion, the precondition was not met. The same applies to a pre-registered depth-anchored KL mitigation experiment. Reporting this pre-registration is part of the paper's methodological contribution: input-layer baselines should be run *before* causal-intervention and mitigation experiments are budgeted, not after.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning, with particular focus on monitoring tools for the emerging class of latent-reasoning architectures. By showing that a natural candidate monitoring affordance (depth-indexed linear probing) does *not* generalize without qualification, we hope to redirect methodological effort toward exploit classes for which it can. We see no specific societal consequences beyond those well-established in AI alignment research that require highlighting here.

References

Bae, S., Fisch, A., Harutyunyan, H., Ji, Z., Kim, S., and Schuster, T. Relaxed recursive transformers: Effective parameter sharing with layer-wise LoRA. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2410.20672. arXiv:2410.20672.

Baker, B., Huizinga, J., Gao, L., Dou, Z., Guan, M. Y., Madry, A., Zaremba, W., Pachocki, J., and Farhi, D. Monitoring reasoning models for misbehavior and

the risks of promoting obfuscation. *arXiv preprint arXiv:2503.11926*, 2025. URL https://arxiv.org/abs/2503.11926.

Belinkov, Y. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022. URL https://aclanthology.org/2022.cl-1.7/.

Dehghani, M., Gouws, S., Vinyals, O., Uszkoreit, J., and Kaiser, Ł. Universal transformers. In *International Conference on Learning Representations (ICLR)*, 2019. URL https://arxiv.org/abs/1807.03819. arXiv:1807.03819.

Denison, C., MacDiarmid, M., Barez, F., Duvenaud, D., Kravec, S., Marks, S., Schiefer, N., Soklaski, R., Tamkin, A., Kaplan, J., Shlegeris, B., Bowman, S. R., Perez, E., and Hubinger, E. Sycophancy to subterfuge: Investigating reward-tampering in large language models. *arXiv preprint arXiv:2406.10162*, 2024. URL https://arxiv.org/abs/2406.10162.

Efron, B. and Tibshirani, R. J. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, 1993.

Gao, L., Schulman, J., and Hilton, J. Scaling laws for reward model overoptimization. *arXiv preprint arXiv:2210.10760*, 2022. URL https://arxiv.org/abs/2210.10760. ICML 2023.

Geiping, J., McLeish, S., Jain, N., Kirchenbauer, J., Singh, S., Bansal, M., Goldblum, M., and Goldstein, T. Scaling up test-time compute with latent reasoning: A recurrent depth approach. *arXiv preprint arXiv:2502.05171*, 2025. URL https://arxiv.org/abs/2502.05171.

Greenblatt, R., Denison, C., Wright, B., Roger, F., MacDiarmid, M., Marks, S., Treutlein, J., Belonax, T., Chen, J., Duvenaud, D., Khan, A., Michael, J., Mindermann, S., Perez, E., Petrini, L., Uesato, J., Kaplan, J., Shlegeris, B., Bowman, S. R., and Hubinger, E. Alignment faking in large language models. *arXiv preprint arXiv:2412.14093*, 2024. URL https://arxiv.org/abs/2412.14093.

Hao, S., Sukhbaatar, S., Su, D., Li, X., Hu, Z., Weston, J., and Tian, Y. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024. URL https://arxiv.org/abs/2412.06769. COLM 2025.

Hubinger, E., Denison, C., Mu, J., Lambert, M., Tong, M., MacDiarmid, M., Lanham, T., Ziegler, D. M., Maxwell, T., Cheng, N., Jermyn, A., Askell, A., Radhakrishnan, A., Anil, C., Duvenaud, D., Ganguli, D., Barez, F., Clark, J., Ndousse, K., Sachan, K., Sellitto, M., Sharma, M., DasSarma, N., Grosse, R., Kravec, S., Bai, Y., Witten, Z., Favaro, M., Brauner, J., Karnofsky, H., Christiano, P., Bowman, S. R., Graham, L., Kaplan, J., Mindermann, S., Greenblatt, R., Shlegeris, B., Schiefer, N., and Perez, E. Sleeper agents: Training deceptive LLMs that persist through safety training. *arXiv preprint arXiv:2401.05566*, 2024. URL https://arxiv.org/abs/2401.05566.

Korbak, T., Balesni, M., Barnes, E., Bengio, Y., Benton, J., Bloom, J., Chen, M., Cooney, A., Dafoe, A., Dragan, A., Emmons, S., Evans, O., Farhi, D., Greenblatt, R., Hendrycks, D., Hobbhahn, M., Hubinger, E., Irving, G., Jenner, E., Kokotajlo, D., Krakovna, V., Legg, S., Lindner, D., Luan, D., Madry, A., Michael, J., Nanda, N., Orr, D., Pachocki, J., Perez, E., Phuong, M., Roger, F., Saxe, J., Shlegeris, B., Soto, M., Steinberger, E., Wang, J., Zaremba, W., Baker, B., Shah, R., and Mikulik, V. Chain of thought monitorability: A new and fragile opportunity for AI safety. *arXiv preprint arXiv:2507.11473*, 2025. URL https://arxiv.org/abs/2507.11473.

Lu, W., Yang, T., Lee, S., Li, X., and Liu, T. Latent chain-of-thought? Decoding the depth-recurrent transformer. *arXiv preprint arXiv:2507.02199*, 2025. URL https://arxiv.org/abs/2507.02199.

MacDiarmid, M., Wright, B., Uesato, J., Benton, J., Kutasov, J., Price, S., Bouscal, N., Bowman, S. R., Bricken, T., Cloud, A., Denison, C., Gasteiger, J., Greenblatt, R., Leike, J., Lindsey, J., Mikulik, V., Perez, E., Rodrigues, A., Thomas, D., Webson, A., Ziegler, D., and Hubinger, E. Natural emergent misalignment from reward hacking in production RL. *arXiv preprint arXiv:2511.18397*, 2025. URL https://arxiv.org/abs/2511.18397.

Saunshi, N., Dikkala, N., Li, Z., Kumar, S., and Reddi, S. J. Reasoning with latent thoughts: On the power of looped transformers. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2502.17416.

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y. K., Wu, Y., and Guo, D. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Sharma, M., Tong, M., Korbak, T., Duvenaud, D., Askell, A., Bowman, S. R., Cheng, N., Durmus, E., Hatfield-Dodds, Z., Johnston, S. R., Kravec, S., Maxwell, T., McCandlish, S., Ndousse, K., Rausch, O., Schiefer, N., Yan, D., Zhang, M., and Perez, E. Towards understanding sycophancy in language models. *arXiv preprint arXiv:2310.13548*, 2023. URL https://arxiv.org/abs/2310.13548. ICLR 2024.

Wang, T. et al. TRACE: Is it thinking or cheating? Detecting implicit reward hacking by measuring reasoning effort. *arXiv preprint arXiv:2510.01367*, 2025. URL https://arxiv.org/abs/2510.01367.

Wen, J., Zhong, R., Khan, A., Perez, E., Steinhardt, J., Huang, M., Bowman, S. R., He, H., and Feng, S. Language models learn to mislead humans via RLHF. *arXiv preprint arXiv:2409.12822*, 2024. URL https://arxiv.org/abs/2409.12822. ICLR 2025.