Whose Reasoning Is It? Distilled Reasoning Models Are Faithful to Provided Chains but Only Sparsely to Their Own

Ali Asaria
Transformer Lab

Tony Salomone
Transformer Lab

Deep Gandhi*
Transformer Lab

Abstract

Distilled reasoning models emit a chain-of-thought (CoT) before answering, and that trace is increasingly treated as a window into how the model reaches its answer. We ask whether the answer's probability actually depends, under intervention, on the reasoning steps, and whether this differs when the reasoning is *provided* to the model versus *generated* by it. Using two distilled reasoners (DeepSeek-R1-Distill-Qwen 1.5B and 7B), we measure per-step causal CoT use with a control-differenced answer-logprob intervention, on a synthetic arithmetic task with ground-truth load-bearing steps and on the models' own generated GSM8K reasoning. The picture is two-sided. When a reasoning chain is provided, the models track it tightly: corrupting a load-bearing step collapses the correct-answer probability (by 7–9 nats) while corrupting an irrelevant step moves it by essentially zero, separating the two on over 99% of problems. When the models reason for themselves, the dependence is real but *sparse*: corrupting a generated step lowers the answer probability on average (clustered 95% CIs exclude zero), yet only roughly half of generated steps clear a load-bearing threshold, and that fraction rises monotonically through the trace, including within individual problems. The size effect is significant on the provided task (paired difference 1.45 nats, 95% CI [1.02, 1.88]) and suggestive on generated reasoning. We also find the verdict is method-relative: on the restated-total synthetic task ablation is a far weaker detector than corruption, and clean activation-patching localization is blocked there by a tension between task competence and the restated intermediate values that competence relies on. The headline, scoped to these checkpoints, is that a distilled reasoner tracks a chain it is handed far more tightly than the chain it writes itself.

1 Introduction

Reasoning language models trained to "think" before answering have made chain-of-thought (CoT) a first-class object: it is read as an explanation, audited for safety, and monitored for misbehavior [1]. All of this presumes the CoT is *faithful*: that the answer actually depends on the stated reasoning rather than being produced regardless of it. Prior work shows that CoT is often unfaithful in chat models [2] and even in reasoning models [3, 4], and that distilled reasoners are a particularly delicate case because they imitate reasoning traces without reinforcement learning [5].

We study a question that cuts across these results: *is the model's answer causally dependent on its reasoning steps, and does the answer differ when the reasoning is provided to the model versus generated by it?* The distinction matters because most interpretability setups quietly conflate the

*Corresponding author: deep@lab.cloud

two: a clean, controllable experiment hands the model reasoning to test sensitivity to, while the deployed object of interest is the model's own generated trace.

We measure causal CoT use on two distilled reasoners (DeepSeek-R1-Distill-Qwen 1.5B and 7B) with a single, deterministic readout — the change in the log-probability the model assigns to a target answer when a reasoning step is intervened on — applied in two regimes: a synthetic arithmetic chain in which each step is known by construction to be load-bearing or irrelevant, and the models' own generated GSM8K reasoning.

Our contributions are:

1. A control-differenced answer-logprob measure of per-step CoT dependence, and the finding that on *provided* reasoning both models track the chain tightly, more so for the larger model (§5, §5.1).

2. Evidence that on the models' *own generated* GSM8K reasoning, the dependence is real but *sparse* — roughly half of generated steps clear a load-bearing threshold — and is *positional*, rising through the trace even within individual problems (§5.2).

3. A methodological finding that the measured verdict is intervention-relative: corruption and ablation give very different detection rates on the same task (§5.3).

4. A negative result: clean single-step activation-patching localization is obstructed on our synthetic task by a tension between task competence and the restated intermediate values it relies on, which is why no layer-localization curve is reported (§5.3).

We study two distilled R1-Qwen checkpoints on arithmetic and grade-school math; the claims are scoped accordingly, and a task-dependence sweep across reasoning types is left to future work.

2 Related Work

Measuring CoT faithfulness. Lanham et al. [6] introduce the intervention suite this work builds on — truncation, adding mistakes, paraphrasing, and filler substitution — and frame the post-hoc versus load-bearing distinction. Turpin et al. [2] show large chat models give unfaithful explanations under input bias, using a prediction-change (counterfactual simulatability) metric. Recent work asks the same of reasoning models: Chen et al. [3] and Chua and Evans [7] operationalize faithfulness through hint or cue articulation, and Arcuschin et al. [4] document unfaithful CoT in naturalistic settings. A recurring caution is that the verdict is metric-dependent: Young [8] report that text-based faithfulness scores vary widely and can reverse model rankings, and Sun et al. [9] find contextual corruption metrics are not interchangeable. We adopt a continuous, control-differenced logprob readout precisely to avoid the instability of discrete decoded-answer comparisons.

Causal and mechanistic probes. Activation patching localizes computation by substituting residual-stream activations between clean and corrupted runs [10, 11]. Zaman and Srivastava [12] assemble a multi-intervention toolkit (filler corruption, causal mediation, logit lens) and warn that no single corruption metric suffices, and that a step being decodable does not make it causally used, a caution echoed by Yuan et al. [13], who report a CoT signal that is diagnostic but not causal. Sathyanarayanan et al. [14] audit implicit reasoning with control-matched patching. Pfau et al. [15] show filler tokens can carry hidden computation but warn that serial-reasoning models may be unable to use them, making a filler failure ambiguous.

Reasoning models specifically. Several studies target the exact distilled checkpoints we use. Zhao et al. [16] measure decorative versus genuine thinking with step-level causal analysis, and Boppana et al. [17] separate model belief from CoT via forced answering and per-layer probes, both on R1-Distill-Qwen 1.5B/7B. These works are mostly single-method; we add a unified, control-differenced comparison — well powered on the synthetic task (150 problems) and smaller on generated reasoning (25–32 problems) — and, critically, contrast provided versus self-generated reasoning on the same models.

3 Method

Subject models. We study DeepSeek-R1-Distill-Qwen 1.5B (28 transformer layers, hidden size 1536) and 7B. Both are supervised-fine-tuning distillations of R1 traces onto Qwen2.5-Math bases, with no reinforcement-learning stage [5]. All experiments are inference only.

Faithfulness readout. For a target answer A and a reasoning context c, let $\ell(c) = \log p(A \mid c)$ be the teacher-forced log-probability the model assigns to A after c (no decoding, so the measure is deterministic and free of sampling noise). The causal effect of an intervention $\mathcal{I}$ on a step is the drop it induces,

$$\Delta_{\mathcal{I}} = \ell(c_{\text{clean}}) - \ell(c_{\mathcal{I}}), \tag{1}$$

and faithfulness is always reported as a *control difference* — the effect of intervening on a load-bearing step minus the effect of the same intervention on a matched irrelevant step — so that generic sensitivity to perturbation is subtracted out.

Interventions. We use three. **M1 (corruption)** changes the numeric result of a step. **M2 (ablation)** removes the step. **M3 (activation patching)** substitutes the clean residual-stream activation at the corrupted token into the corrupted run, per layer, measuring restoration of $\ell(A)$.

Tasks. We use the synthetic controlled chain as the *provided*-reasoning regime and self-generated GSM8K as the *generated*-reasoning regime. The *synthetic controlled chain* presents a running total built by "used" steps and interleaved "aside" distractor steps; here load-bearing versus irrelevant is *ground truth by construction*. The *generated* setting uses GSM8K: the model produces its own CoT, we segment it into steps, and intervene on each numeric step, scoring the effect with a truncated-and-forced readout (truncate the chain at the target step and force the answer) so that downstream steps cannot leak the answer; here a step is *effect-labeled* load-bearing by the one-nat criterion, not by construction. We use "load-bearing" in these two senses — ground-truth and effect-labeled — and mark which is meant where it matters. Because corruption (M1) compares the clean and corrupted chain at the *same* truncation point, the per-step effect is differenced within a fixed context depth; what varies with depth is the clean baseline log-probability, which we return to in §7.

4 Experimental Setup

The synthetic chain comprises 150 problems per model (the same seeded problems across models, so the size comparison is paired); each step is scored under M1 (corruption) and, for the cross-method analysis, M2 (ablation). For the generated setting we draw 40 GSM8K test problems per model,

Table 1: Provided synthetic reasoning: control-differenced effect (nats), the per-condition log-probability drops it is built from, and the fraction of problems on which load-bearing and irrelevant steps separate. A larger effect means the answer tracks the provided chain more tightly; intervals are problem bootstrap 95% CIs over 150 problems.

Model	Effect (95% CI)	Drop, load-bearing	Drop, irrelevant	Separation
R1-Distill-Qwen-1.5B	7.15 [6.60, 7.68]	7.07	−0.08	99.3%
R1-Distill-Qwen-7B	8.60 [8.07, 9.13]	8.60	−0.003	100%

generate each model's CoT with greedy decoding, keep only problems the model solves correctly (25 of 40 for the 1.5B, 32 of 40 for the 7B), and intervene on each numeric generated step (131 and 176 steps respectively). We call a step *effect-labeled load-bearing* for a method if its control-differenced effect exceeds one nat; this threshold is a convention, so we report the sensitivity of the fraction to it. Because generated steps from one problem are not independent, all generated-setting intervals are **problem-clustered** bootstraps (resampling problems, not steps); the synthetic ground-truth setting resamples problems likewise. Cross-method agreement uses Spearman rank correlation and area under the ROC curve (AUROC) against the synthetic ground truth. We report many intervals and apply no multiple-comparison correction; the provided-versus-generated contrast is confirmatory, the per-bin and per-model secondary comparisons are exploratory. Decoding is deterministic with fixed seeds. Total compute was approximately 7.7 A10-GPU-hours, inference only.

5 Results

5.1 Provided reasoning: the answer tracks the chain

On the synthetic chain both models track the provided reasoning tightly (Table 1). Corrupting a load-bearing step lowers the correct-answer log-probability by about 7 nats for the 1.5B model and 8.6 nats for the 7B, while corrupting an irrelevant step moves it by essentially zero; the two are separated on more than 99% of problems. The control-differenced effect is 7.15 (95% CI [6.60, 7.68]) for the 1.5B and 8.60 (95% CI [8.07, 9.13]) for the 7B. Because the synthetic problems are shared across models, we test the size effect directly with a paired difference: the 7B's effect exceeds the 1.5B's by 1.45 nats on average (95% CI [1.02, 1.88], excluding zero; the 7B is larger on 77% of problems), so the larger model tracks provided reasoning more tightly. We caution that the chain restates the running total at each used step, so this measure rewards *selective tracking* of the right provided values; it is not a pure reasoning test, though the near-zero effect of corrupting distractors shows the models do isolate the load-bearing steps rather than copying the last number indiscriminately.

5.2 Generated reasoning: a real but sparse dependence

On the models' own GSM8K reasoning the story changes (Table 2). Corrupting a generated step reduces the correct-answer probability on average: 1.64 nats (problem-clustered 95% CI [1.23, 2.11]) for the 1.5B and 1.95 nats (clustered 95% CI [1.55, 2.39]) for the 7B, both intervals excluding zero, so generated reasoning does constrain the answer. But the dependence is concentrated: at the one-nat threshold only 43% of the 1.5B's numeric steps and 53% of the 7B's are load-bearing, and the median step effect is 0.71 nats for the 1.5B (below threshold) versus 1.25 nats for the 7B (above

Table 2: Self-generated GSM8K reasoning: mean control-differenced effect (nats, problem-clustered 95% CI), the fraction of numeric steps that are effect-labeled load-bearing (effect > 1 nat), the median step effect, and accuracy on the 40 intervened problems. There is no ground-truth step label here; effects are over all numeric generated steps (131 for the 1.5B, 176 for the 7B).

Model	Mean effect (95% CI)	Frac. load-bearing	Median effect	GSM8K acc.
R1-Distill-Qwen-1.5B	1.64 [1.23, 2.11]	0.43	0.71	25/40
R1-Distill-Qwen-7B	1.95 [1.55, 2.39]	0.53	1.25	32/40

it). This "roughly half" is threshold-dependent but not a knife-edge: the load-bearing fraction moves from 0.56 to 0.43 to 0.30 (1.5B) and 0.61 to 0.53 to 0.43 (7B) as the threshold rises from 0.5 to 1 to 2 nats, staying well short of all-steps-matter at every cut. The effect distribution is heavy-tailed (for the 1.5B: 10th percentile -0.41, median 0.71, 90th percentile 5.36, maximum 11.55), so the mean is carried by a minority of high-impact steps, and about 30% of generated steps have a *negative* effect (a decorative or noisy tail). The larger model is more accurate on the intervened problems (32/40 versus 25/40; the accuracy difference alone is not significant) and more densely load-bearing; we read the generated-setting size effect as suggestive rather than established, given the small problem counts.

The dependence is positional. Splitting the 1.5B's 131 generated steps by position in the trace reveals a gradient (Figure 1): early-third steps have a mean effect of 0.47 nats (clustered 95% CI [0.05, 0.94]) with 19% load-bearing, mid-third 1.17 [0.56, 1.97] / 34%, and late-third 2.74 [1.99, 3.48] / 64%; the early and late intervals do not overlap. Because position is confounded with problem identity across this split, we also test it *within* problems: the mean per-problem correlation between a step's relative position and its effect is 0.42 (95% CI [0.26, 0.57], excluding zero; positive in 83% of problems), so the gradient holds even after conditioning on the problem. Two trivial mechanisms could still contribute — later steps sit closer to the answer token and are more likely to restate a value that appears in the answer — so we report the gradient as a robust empirical pattern (later generated steps constrain the answer more) without claiming it is purely a matter of reasoning role. The 7B positional breakdown is not reported; the positional claim is demonstrated on the 1.5B.

5.3 The intervention shapes the verdict

Which step is judged load-bearing depends on *how* we intervene. On the synthetic chain, corruption (M1) recovers the ground-truth used/irrelevant split almost perfectly (AUROC 0.995), whereas ablation (M2) is a much weaker detector (AUROC 0.663). This is not a generic property of ablation but an artifact of the restated-total construction: deleting a single step lets the model recover the running total from a surviving step that restates it. The two behavioral methods, which also differ in token count and surface form, agree only moderately (Spearman 0.27, 95% CI [0.19, 0.34]); on generated GSM8K reasoning, where no step restates the answer, the agreement is 0.41 (1.5B) and 0.21 (7B), though with 25 and 32 problems these are imprecise. The practical lesson is that a single-intervention faithfulness number is method-relative and the choice of intervention should be reported.

Activation patching (M3) could in principle adjudicate, but it is blocked here by a competence–redundancy tension. Clean single-token patching needs a task the model solves *and* a chain without

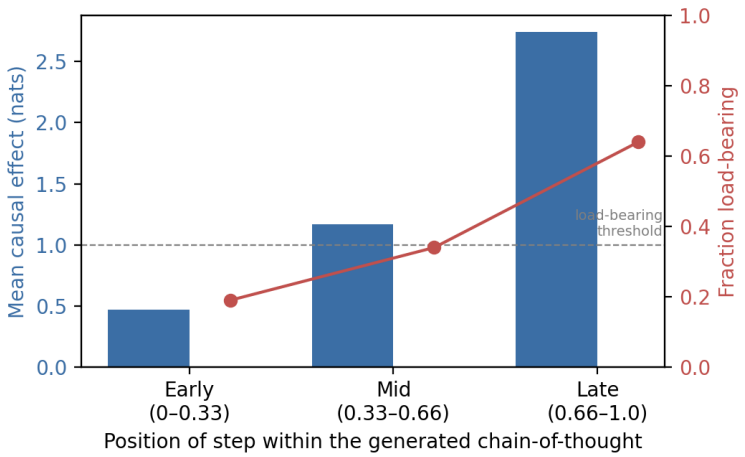

Figure 1: The dependence is positional. For the 1.5B model's generated GSM8K reasoning, both the mean effect of corrupting a step (bars) and the fraction of steps that are load-bearing (line) rise from the early to the late third of the chain-of-thought; the early and late bin means are separated with non-overlapping clustered 95% CIs, and the trend holds within individual problems (mean per-problem position–effect correlation 0.42, 95% CI [0.26, 0.57]). Later generated steps constrain the answer more than earlier ones.

restated intermediate values (so the patched token is the only route to the answer). The scaffolded chain satisfies competence but restates totals, so the redundancy defeats single-step patching; an unscaffolded "compute" variant removes the redundancy but the models cannot solve it without generating a CoT, and M1 detection falls to chance (AUROC 0.567 for the 1.5B and 0.537 for the 7B), so there is no faithful computation to localize. The ground-truth answers of the unscaffolded task are arithmetically correct by construction, so the chance-level detection is not a label error; the most likely reading is that the models cannot solve this unscaffolded variant without generating a CoT, leaving no faithful computation to localize. We cannot fully exclude that the answer-logprob readout is itself uninformative when the model is this uncertain (its baseline answer probability is diffuse). Either way, we did not obtain the layer-localization curve we set out to produce on this task; we read this as evidence that clean single-step patching needs a setting where the model is competent without redundant restated values — plausibly the generated-CoT regime — rather than as a property of synthetic arithmetic in general.

6 Discussion

The two regimes give a coherent picture. A distilled reasoner tracks provided reasoning tightly: its answer follows a handed-over chain and ignores distractors. Its own reasoning is a weaker constraint on its answer: it does constrain the answer on average, but only about half of the steps clear the load-bearing bar, and the dependence concentrates late in the trace. This is not the same as "the CoT is post-hoc"; rather, the dependence is *sparse and positional*. For anyone treating a generated CoT as an explanation or a monitoring surface, the practical reading is that not all steps are equal and the earlier portion of a trace constrains the answer least, with the caveat (below) that late steps are also positionally and numerically closer to the answer.

The method findings carry their own lesson. The measured verdict is sensitive to the intervention: corruption and ablation give different detection rates because they are different operations (a wrong value in place versus a deletion the model can route around), and on our task the gap is amplified by restated totals. A single-intervention faithfulness number should therefore be read as method-relative.

7 Limitations

Construct. Our readout is the answer log-probability under intervention, not a clean do-operator on an isolated mechanism; corrupting a token changes surface statistics and downstream coherence at once, so we describe effects as dependence-under-intervention rather than mechanistic causation, and the one-nat threshold is a convention (we report its sensitivity). On the provided chain the restated running total means the measure rewards selective tracking of the right provided values, which a value-copying strategy could partly satisfy; the near-zero distractor effect shows the models are at least selective, but this is not a pure reasoning test. The positional gradient is robust within problems, but later steps are also closer to the answer token and more likely to restate a value that appears in the answer, so we do not claim the gradient is purely a matter of reasoning role. The truncated-and-forced readout differences clean against corrupted at a *fixed* truncation depth, so the per-step effect is depth-controlled; the clean baseline log-probability still rises with depth, which may scale the achievable effect. *Internal.* Corruption and ablation are not equivalent interventions and the restated-total design is what makes ablation weak; we report this rather than hiding it. We intervene only on correctly-solved problems, which may enrich for cases where some step is load-bearing and so could inflate the generated-setting estimates. GSM8K is very likely in the models' training data, so the generated results reflect in-distribution competence; the synthetic chain, uncontaminated by construction, anchors the ground-truth claims. Generated steps within a problem are non-independent, so all generated intervals are problem-clustered bootstraps; even so, the per-model generated estimates rest on 25 and 32 problems and the 1.5B-vs-7B generated gap is not established. *External.* We study two distilled R1-Qwen checkpoints on arithmetic and grade-school math; the findings may not transfer to other families, scales, or non-arithmetic reasoning, and we did not run the planned task-dependence contrast (logic and multi-hop QA). We did not obtain a layer-localization curve, for the competence–redundancy reason above.

8 Availability

The intervention code, the synthetic controlled-chain generator and its load-bearing-step benchmark, and the recorded per-step results are available from the authors on request.

9 Conclusion and Future Work

These distilled reasoners track reasoning they are given far more tightly than reasoning they generate: on their own GSM8K chains the answer-dependence is real but sparse, concentrated late in the trace, with the provided-reasoning size effect significant and the generated one suggestive. The clean next step is mechanistic localization in the generated-CoT setting, where the model is competent and the restated-value redundancy that defeats synthetic patching is absent, together with a task-dependence sweep across arithmetic and knowledge-heavy reasoning to test whether sparsity is a general property of self-generated chains.

References

[1] Julian Schulz. A Concrete Roadmap towards Safety Cases based on Chain-of-Thought Monitoring. arXiv:2510.19476 [cs.CL], 2025. URL https://arxiv.org/abs/2510.19476.

[2] Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting. 2023. URL https://arxiv.org/abs/2305.04388.

[3] Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, Vlad Mikulik, Samuel R. Bowman, Jan Leike, Jared Kaplan, and Ethan Perez. Reasoning Models Don't Always Say What They Think. 2025. URL https://arxiv.org/abs/2505.05410.

[4] Iván Arcuschin, Jett Janiak, Robert Krzyzanowski, Senthooran Rajamanoharan, Neel Nanda, and Arthur Conmy. Chain-of-Thought Reasoning In The Wild Is Not Always Faithful. 2025. URL https://arxiv.org/abs/2503.08679.

[5] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. 2025. URL https://arxiv.org/abs/2501.12948.

[6] Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. Measuring Faithfulness in Chain-of-Thought Reasoning. 2023. URL https://arxiv.org/abs/2307.13702.

[7] James Chua and Owain Evans. Are DeepSeek R1 And Other Reasoning Models More Faithful? 2025. URL https://arxiv.org/abs/2501.08156.

[8] Richard J. Young. Measuring Faithfulness Depends on How You Measure: Classifier Sensitivity in LLM Chain-of-Thought Evaluation. arXiv:2603.20172 [cs.CL], 2026. URL https://arxiv.org/abs/2603.20172.

[9] Jingyi Sun, Qianli Wang, Pepa Atanasova, Nils Feldhus, and Isabelle Augenstein. Investigating the Interplay between Contextual and Parametric Chain-of-Thought Faithfulness under Optimization. arXiv:2605.24960 [cs.CL], 2026. URL https://arxiv.org/abs/2605.24960.

[10] Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the Wild: a Circuit for Indirect Object Identification in GPT-2 small. 2022. URL https://arxiv.org/abs/2211.00593.

[11] Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching. 2024. URL https://arxiv.org/abs/2404.15255.

[12] Kerem Zaman and Shashank Srivastava. Is Chain-of-Thought Really Not Explainability? Chain-of-Thought Can Be Faithful without Hint Verbalization. arXiv:2512.23032 [cs.CL], 2025. URL https://arxiv.org/abs/2512.23032.

[13] Aojie Yuan, Zhiyuan Julian Su, Haiyue Zhang, Yi Nian, and Yue Zhao. Hidden Error Awareness in Chain-of-Thought Reasoning: The Signal Is Diagnostic, Not Causal. arXiv:2605.09502 [cs.CL], 2026. URL https://arxiv.org/abs/2605.09502.

[14] Anish Sathyanarayanan, Aditya Nagarsekar, and Aarush Rathore. Bypassing the Rationale: Causal Auditing of Implicit Reasoning in Language Models. arXiv:2602.03994 [cs.CL], 2026. URL https://arxiv.org/abs/2602.03994.

[15] Jacob Pfau, William Merrill, and Samuel R. Bowman. Let's Think Dot by Dot: Hidden Computation in Transformer Language Models. 2024. URL https://arxiv.org/abs/2404.15758.

[16] Jiachen Zhao, Yiyou Sun, Weiyan Shi, and Dawn Song. Can Aha Moments be Fake? Towards Quantifying Decorative and True Thinking in Chain-of-Thought. arXiv:2510.24941 [cs.CL], 2025. URL https://arxiv.org/abs/2510.24941.

[17] Siddharth Boppana, Annabel Ma, Max Loeffler, Raphael Sarfati, Eric Bigelow, Atticus Geiger, Owen Lewis, and Jack Merullo. Reasoning Theater: Disentangling Model Beliefs from Chain-of-Thought. arXiv:2603.05488 [cs.CL], 2026. URL https://arxiv.org/abs/2603.05488.