Commit Then Explore: Early Answer Decodability and Reduced Late-Chain Causal Influence in Reasoning Models

Ben Jenkins
Dept. of Electrical Engineering
and Computer Science
Florida Atlantic University
benjaminjenk2024@fau.edu

Mihaela Cardei
Dept. of Electrical Engineering
and Computer Science
Florida Atlantic University
mcardei@fau.edu

Abstract

Extended chain-of-thought reasoning is often assumed to gradually refine a model's answer. We find a different pattern: reasoning models stabilize their internal representations early, then continue exploring with less influence on the final answer. In DeepSeek-R1-7B on math and science benchmarks, internal representations stabilize at a median 10% of the reasoning chain, and the eventual answer becomes decodable by 18%. Later reasoning activates increasingly diverse features, yet disrupting early features changes the final answer 2–3 times more often than disrupting later ones and shifts its log-probability 1.5–4.6 times more under matched activation mass. We call this pattern **commit then explore**. The commit signature generalizes across three model families and to coding and other reasoning tasks, strengthens during reinforcement learning, and is absent in tested non-reasoning baselines. Using these signals, an adaptive monitor reduces generated tokens by 26–34% while retaining at least 92% of original accuracy, though it does not outperform a simpler baseline in net computation. These findings challenge the view that longer reasoning necessarily means continued refinement of the answer.

1 Introduction

Test-time compute scaling is reshaping language-model capabilities [Snell et al., 2025, Guo et al., 2025]: models like OpenAI's o1 and DeepSeek-R1 generate thousands of reasoning tokens per problem, achieving large gains on mathematics, coding, and scientific reasoning. Unlike pre-training scaling, this axis does not demand more parameters or data; it asks the model to *think longer* rather than grow *bigger*.

Yet this paradigm introduces a fundamental interpretability gap. While we can observe inputs and final answers, the intermediate reasoning chain is treated as a black box. Recent work has revealed troubling properties of this black box: chain-of-thought explanations are not always faithful to the computation that produced them [Lanham et al., 2023, Turpin et al., 2023], and even on non-adversarial prompts models can produce superficially coherent arguments rationalizing answers that appear to be driven by implicit biases [Arcuschin et al., 2026].

These observations raise questions for both capability understanding and safety: *When does the answer become decodable from the model's internal state? How much does the computation that follows still matter for the answer? Can the transition be detected in real time?*

We address these questions through a mechanistic analysis of extended reasoning in open-weight models. Using sparse autoencoders (SAEs) to decompose residual-stream activations into interpretable features, together with a label-free cosine detector and a supervised answer-identity probe, we track

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

how the model's internal state evolves across complete reasoning chains. A common informal picture of CoT is that reasoning progressively *converges*: feature diversity should fall, top features should stabilise, and the model should "narrow down" to an answer. Our measurements show a different picture. The internal state stabilizes toward its final representation early and the eventual answer becomes decodable soon after, while the chain continues to *diversify*, with the later computation measurably less causally coupled to the answer than what preceded it. Figure 1 previews two of the central results: the contrast between a reasoning model and its base, and the held-out quantitative test of §7.3. Our contributions:

1. **Early internal stabilization** (§5): a label-free cosine detector identifies stable final-state similarity at median 10% of chain progress, while a supervised probe decodes the eventual answer at median 18%, both well before the NLDD behavioral horizon ($\sim 70\%$). These are detector-relative transitions, not irreversible decisions.
2. **Post-commit diversification** (§4): SAE feature turnover and entropy rise over the chain on all benchmarks and model families (Cohen's d=0.61–0.68).
3. **A quantified causal asymmetry** (§6): post-commit features have a 2–6$\times$ smaller teacher-forced effect (1.5–4.6$\times$ under matched mass) and a 2–3$\times$ lower answer-flip rate than pre-commit features; post-commit computation is less influential, not inert.
4. **Generality** (§5.3): the commit signature replicates on code and general reasoning and fires early on chains dense with reconsideration language.
5. **Training conditions and a held-out test** (§7): on a fixed base family, RLVR strengthens the signature within a single training run, while a matched moderate-scale trajectory-SFT control does not reproduce the full effect; a reward-variance fit frozen on R1 predicts QwQ at r=0.96.

A reasoning monitor built on these signals (§8) shows the structure is exploitable for adaptive compute.

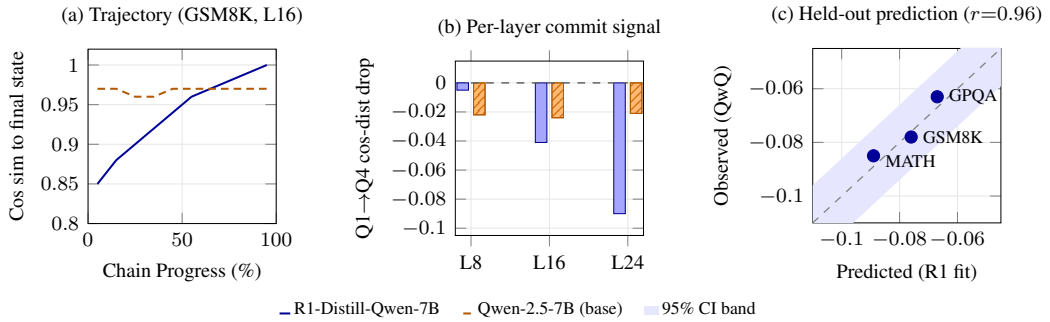

Figure 1: **Terminal-reward reasoning models show the commit-and-explore trajectory; their bases do not.** (a) Residual-stream cosine similarity to the final state over the chain (GSM8K, L16): R1-Distill-Qwen-7B (blue) converges toward its final representation, the same-architecture Qwen-2.5-7B-Instruct base (orange, dashed) stays flat. (b) Q1→Q4 cosine-distance drop by layer depth: R1 shows a significant drop at the fixed-rule layer L16 (§3.2); the L24 bar is shown for reference (post-hoc unrestricted argmax, p=0.0002). Base shows no significant drop at any layer. (c) Held-out quantitative test: predictions derived from the R1-7B+R1-32B fit (§7, App P) applied to QwQ-32B; Pearson r=0.96, all three observed values inside the 95% CIs of predictions frozen before evaluation. Within-run evidence on a fixed base family is in Table 4.

2 Related Work

Test-time compute scaling and CoT faithfulness. Snell et al. [2025] showed that compute-optimal scaling of inference-time compute can outperform a 14$\times$ larger model (FLOPs-matched) on easy and intermediate problems, though not the hardest; test-time scaling via RL-trained extended CoT underlies o1 [OpenAI, 2024], DeepSeek-R1 [Guo et al., 2025], and QwQ. Behaviorally, Lanham et al. [2023] find answers on many tasks largely insensitive to truncating or corrupting the CoT

(with large variation across tasks), Turpin et al. [2023] show explanations swayed by biasing input features they almost never mention, and Arcuschin et al. [2026] document post-hoc rationalization on natural prompts; METR [Von Arx and Deng, 2025] argues CoT remains informative despite this. *Unlike these largely behavioral works, which perturb prompts or CoT text, we offer an internal, trajectory-level account: early stable answer-decodability from the residual stream, paired with a smaller teacher-forced marginal effect of late-chain features on the original answer, is consistent with the task-dependent insensitivity to CoT truncation and corruption reported by Lanham et al. [2023].*

Mechanistic interpretability of reasoning. Anthropic's circuit tracing [Ameisen et al., 2025, Lindsey et al., 2025] traces information flow on short prompts across a range of single-forward-pass behaviors (e.g., two-hop recall, planning, arithmetic), including a CoT-faithfulness case study [Lindsey et al., 2025]. Coconut [Hao et al., 2025] trains models to reason in a continuous latent space. Circuit-based Reasoning Verification (CRV; Zhao et al., 2026) classifies whether an individual reasoning step is computed correctly from its attribution graph, in an instruction-tuned model (Llama-3.1-8B-Instruct). *We differ in level and scope*: we study chain-level *trajectories* of SAE features in explicit token-based terminal-reward reasoning models. CRV's experimental model, Llama-3.1-8B-Instruct, shows no significant population-level commitment signature at any tested layer (§7), so the two analyses address disjoint regimes; Appendix T states a joint prediction that connects them and suggests step-level verification gated by chain-level commit position as a direction. Following open SAE suites such as SAEBench [Karvonen et al., 2025] and Gemma Scope [Lieberum et al., 2024], we train TopK SAEs [Gao et al., 2025] with AuxK dead-feature revival on R1 residual streams.

RL for reasoning and distribution sharpening. Yue et al. [2025] characterize RLVR as sharpening an output distribution the base model can already reach, and Karan and Du [2026] recover much of RL's gain by sampling a sharpened base distribution. We ask the complementary *internal* question, whether reasoning training changes the trajectory-level representational structure of the chain, and relate the two descriptions in §7.2.

3 Methodology

3.1 Models and benchmarks

Reasoning models: DeepSeek-R1-Distill-Qwen (7B, 32B), Qwen-QwQ-32B [Qwen Team, 2025], Llama-3.1-Nemotron-Nano-8B [Bercovich et al., 2025]. **Bases** (same architectures, no reasoning training; these are comparison models and not necessarily training parents, e.g., R1-Distill-Qwen-7B was trained from Qwen2.5-Math-7B [Guo et al., 2025]): Qwen-2.5-Instruct (7B, 32B) [Qwen Team, 2024], Llama-3.1-8B-Instruct [Grattafiori et al., 2024]. **Fixed-base-family conditions** (§7.2): our reproduction of Open-Reasoner-Zero-7B [Hu et al., 2025] (final and 25% checkpoints), OpenThinker-7B [Guha et al., 2026]. **Benchmarks:** GSM8K [Cobbe et al., 2021] (1,319), MATH-500 [Hendrycks et al., 2021, Lightman et al., 2024] (500), GPQA Diamond [Rein et al., 2024] (198); Nemotron-Nano on GSM8K only (200 chains); for §5.3, R1-7B also on HumanEval [Chen et al., 2021], MBPP [Austin et al., 2021], CommonsenseQA [Talmor et al., 2019], and LogiQA [Liu et al., 2020].

3.2 SAEs, feature metrics, and layer selection

We train TopK sparse autoencoders [Gao et al., 2025] with dictionary size $32d_{\text{model}}$, sparsity $k{=}64$, and AuxK dead-feature revival on 200M tokens of mixed OpenWebText [Gokaslan and Cohen, 2019] and reasoning-chain activations. For each chain we segment the `<think>...</think>` span into T 100-token windows and take the last-token residual activation as the step-t snapshot, then encode through the SAE. We track three per-step metrics, all defined formally in Appendix K: *feature entropy* $\mathcal{H}_t$ (how distributed activation mass is over features), *feature turnover* τ_t (Jaccard dissimilarity of active feature sets at adjacent steps), and *cumulative novelty* U_t (unique features seen through step t).

Per-model layer selection (fixed rule). For each model we select the argmax of the absolute Q1→Q4 cosine-distance drop on a 50-chain GSM8K calibration split, searched over the mid-to-late range $[L/2, 0.9L]$. The rule is fixed once and applied uniformly. Selected layers: L16 for R1-7B (28 layers), L32 for R1-32B (64 layers), L48 for QwQ-32B (64 layers), L24 for Nemotron-Nano-8B (32 layers). Appendix C reports the full sweep.

3.3 Answer-identity probe

For each chain in the training split, we extract the mean-pooled residual stream h_t at step t and train a linear probe $p_\theta(\hat{a} \mid h_t)$ to predict the *identity* of the chain's final answer, formulated as a classification problem: for numeric tasks (GSM8K, MATH) the label is the final-answer token bucket (a discretization over the $V{=}512$ most frequent answer tokens plus an "other" class); for multiple-choice tasks (GPQA) the label is the chosen option (A/B/C/D), with no bucketing. For numeric tasks, decodability claims are therefore made at the answer-bucket resolution. The probe is trained on $(h_t, \hat{a})$ pairs aggregated across 80% of chains (stratified by benchmark and correctness) and evaluated per-step on the held-out 20%. All commitment-position statistics reported in the paper (medians, histograms, threshold sweeps) are computed on this held-out set. The same $80/20$ split is reused for the correctness probe and for the exploration-phase correctness probe of §6. This protocol rules out probe-fit leakage as an explanation for the reported commitment signal.

We define the *commitment step* t^* as the first step at which $s_t > 0.85$ and $s_{t+1}, s_{t+2} > 0.85$ as well (stability filter), and the *commitment position* as t^*/T.

4 Reasoning Diversifies After Commitment

4.1 Feature turnover and entropy rise over the chain

Figure 2 shows the trajectory-level finding for DeepSeek-R1-7B at layer 16.

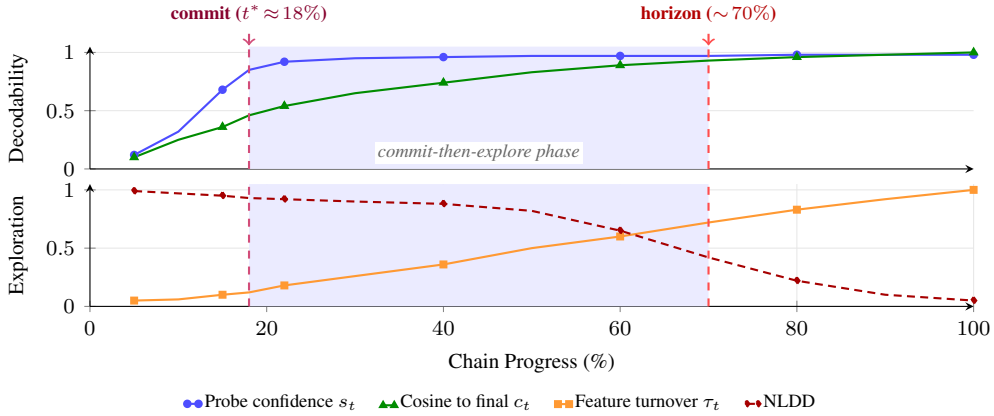

Figure 2: **The commit-then-explore trajectory.** DeepSeek-R1-7B on MATH-500, L16, $n{=}500$ chains. *Top*: decodability rises early; probe confidence s_t crosses 0.85 at $t^* \approx 18\%$ and cosine-to-final c_t rises monotonically. *Bottom*: exploration continues post-commit; feature turnover τ_t keeps rising while NLDD stays near its maximum, then collapses near the behavioral horizon at $\sim 70\%$. Shaded band marks the commit-then-explore phase (answer decodable but turnover still climbing).

Feature turnover grows (primary signal). The Jaccard dissimilarity between active feature sets at adjacent steps, τ_t, is $+0.18$ higher in the last quartile of the chain than in the first on the full GSM8K test set ($n{=}1{,}319$, Cohen's $d{=}0.61$) and $+0.21$ higher on MATH-500 ($d{=}0.68$). Every model and benchmark in Table 1 shows the same sign. Splitting each chain at its commitment point (§5), post-commit turnover is 1.3–1.8$\times$ higher than pre-commit, and post-commit steps activate 1.5–2.1$\times$ more chain-unique features.

Feature entropy rises (supporting signal). Mean SAE activation entropy rises across the chain on every model and benchmark (slopes $+0.85$ to $+1.5\times10^{-3}$ per bin, all $p{\le}0.002$, Appendix L; Cohen's $d{=}0.42$–0.49), and the number of unique features activated by step t grows without saturating.

Table 1: **Feature turnover rises across the chain.** Q1→Q4 turnover delta at each model's fixed-rule layer†; every cell is positive. Entropy slopes for the same cells are in Appendix L.

Turnover Q1→Q4	GSM8K	MATH	GPQA
R1-7B (L16)	+0.18	+0.21	+0.19
R1-32B (L32)	+0.20	+0.23	+0.21
QwQ-32B (L48)	+0.17	+0.20	+0.18
Nem-Nano-8B (L24)	+0.16	–	–

† Fixed-rule argmax-within-window layer (§3.2). Nemotron-Nano-8B is GSM8K-only (n=200).

5 Commitment Is Early and Stable

If reasoning diversifies rather than converges, when does information about the answer become available internally? We use two detectors on the residual stream: a label-free cosine detector (primary), which measures stable similarity to the chain's own final representation, and the supervised answer-identity probe of §3.3 (cross-check), which decodes the eventual answer itself. Throughout, *commitment* means the detector-relative transition these measure, not that the model has made an irreversible decision: the NLDD behavioral horizon sits at ∼70% of chain progress, and post-commit ablation still flips the answer in a non-trivial fraction of chains (§6). Timings are relative to the stated detector thresholds (Appendix M).

5.1 Commitment is early; exploration continues

At each step t we compute the cosine similarity c_t between the mean-pooled residual stream at step t and the final-step residual stream of the same chain. The chain commits when c_t first crosses 0.7 of its early-chain peak and remains above threshold for two consecutive steps; we write $t^*_{\cos}$ for this step, reserving t^* for the probe's commitment step (§3.3). The detector requires no labels and is tied to each chain's own final state.

Across n=525 held-out chains pooled from MATH-500 and GPQA (Figure 3), the median commitment position is 0.10 and 65% of chains commit before 0.20; for thresholds in $[0.50, 0.75]$ the median stays below 0.20 and the fraction committing before 20% stays above 50% (Appendix M). An adaptation of Normalized Logit Difference Decay (NLDD; Ye et al., 2026), which resamples the tokens of step t and records the drop in final-answer log-probability instead of corrupting and truncating the chain and measuring a standardized logit margin, places the behavioral horizon at 0.71 ± 0.09 on the same chains (Appendix N). Figure 4 shows one chain.

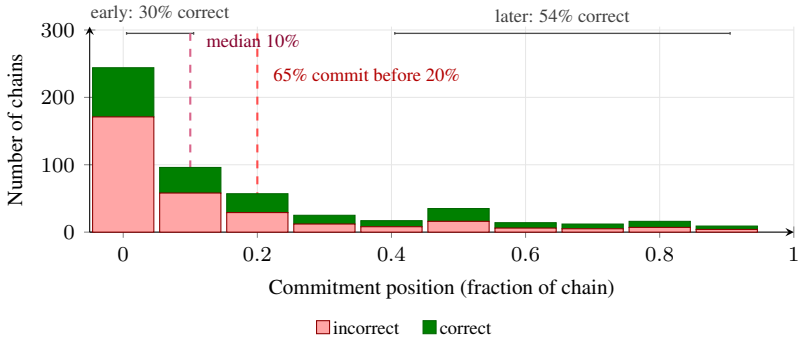

Figure 3: **Most chains commit early, but very early commits skew incorrect.** Commitment positions over 525 chains (MATH-500 + GPQA, R1-7B, L16) stacked by correctness. Median 10%; 65% commit before 20%.

Commitment is stable. On chains of ≥ 10 steps (n=516), the detector fires on 100% of chains and the post-commit representation stays within the commit band for 95.4% of steps on average; 85.9% of chains show no 2-consecutive-step dip below threshold and 14.1% show some post-commit reversal.

Commit position is itself a correctness signal. Commit position correlates modestly with correctness (Spearman $\rho = 0.21$, $p<10^{-3}$): chains committing before 5% are correct 30.6% of the time (n=108), versus 50.0% for 20–40% (n=142) and 55.3% later (n=94). Very early commits are associated with premature, wrong answers on hard problems. The ∼60% of chain length between commit and the NLDD horizon is the commit-then-explore phase, which we conjecture the CoT-faithfulness literature has measured indirectly through truncation insensitivity.

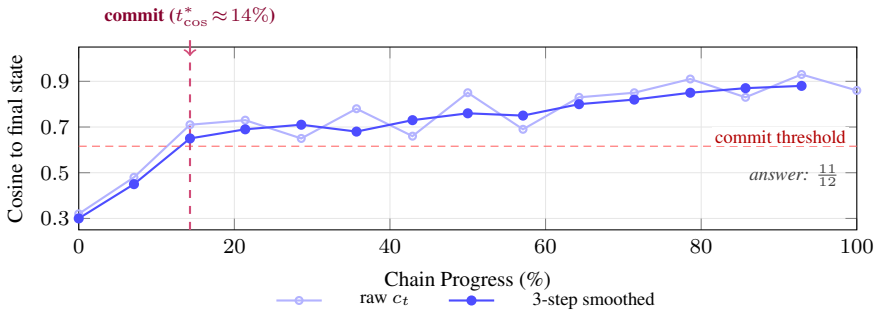

Figure 4: **A single representative chain.** R1-7B on MATH-500 #237 (answer $\frac{11}{12}$, 15 windows). Cosine to final-3-step reference: raw (light), 3-step smoothed (dark). Smoothed c_t crosses the commit threshold at $t^*_{\cos} \approx 14\%$ and stays above it for every post-commit window.

5.2 Answer-identity probe and prompt-only baseline

The probe (commitment at the first step with $s_t > 0.85$ for three consecutive steps) fires at median ∼18%, later than the label-free cosine detector; the two measure distinct constructs but agree on chain-level ordering (Appendix D). Prompt-only (step-0) probe accuracy is 0.12 on GSM8K (chance 0.002), 0.08 on MATH, and 0.31 on GPQA (chance 0.25), far below the 0.85 threshold, and a matched Qwen-2.5-7B probe never crosses 0.85 (median max s_t 0.41). Retraining the probe on chains where R1 answers *incorrectly*, the commit event tracks the model's own wrong answer at median 19% (vs. 18% on correct chains), with step-0 accuracy near floor for both the model's answer (0.04) and the gold answer (0.06): what becomes decodable is the model's eventual answer identity, whether or not it is correct. GPQA uses exact options with no bucketing and shows consistent timing.

5.3 Generality: other domains and revision-dense chains

On four non-math benchmarks (Table 2; Appendix G), the cosine detector finds Q1→Q4 drops of -0.045 to -0.061 (all $p \leq 0.004$) and median commit positions of 11–18%, consistent with the math benchmarks; this replicates the commit signature only, not the probe or diversification analyses. On n=86 R1-7B MATH chains tagged for explicit revision markers ("wait", "actually"; Appendix H), 95% carry at least one marker and the detector still fires early on them (median 0.05), while 34.9% of chains are detected as apparent violators (late commit or post-commit reversal).

Table 2: **Commit signature on code and general-reasoning benchmarks** (R1-7B, label-free cosine detector). Only the commit signature is tested on these benchmarks; the probe, ablation, and diversification analyses were run on the math and science benchmarks.

Domain	Benchmark	Q1→Q4 drop	p	Median commit
Code	HumanEval	−0.058	< 0.001	15%
Code	MBPP	−0.061	< 0.001	18%
General reasoning	CommonsenseQA	−0.045	0.004	11%
General reasoning	LogiQA	−0.052	0.002	16%

6 Post-Commit Features Have a Smaller Causal Effect

How much does the post-commit computation matter for the answer? We measure a *teacher-forced marginal effect* (change in log-probability of the model's original answer under ablation, text held

fixed) and a *free-regeneration answer-flip rate*. Post-commit features have a smaller effect on both; we report the asymmetry as a measured ratio, not as evidence that post-commit computation is inert.

For each chain we ablate the top-k=10 post-commit features (rank-ordered by entropy contribution) by subtracting their SAE-decoded contribution from the residual stream at every post-commit step, then re-run the forward pass teacher-forced on the *original* post-commit tokens; the measurement is $|\Delta \log p(y^{\text{original}})|$. Baselines: top-$k$ pre-commit features and k random features. Because pre-commit features integrate more total activation mass across the chain, we also report a *matched-activation-mass* control that ablates a pre-commit subset (typically 3–5 features) whose total activation mass equals that of the top-10 post-commit features (protocol in Appendix E). Table 3 reports both (per-benchmark breakdown in Appendix O).

Table 3: **Answer log-probability change under targeted ablation** (mean $|\Delta \log p|$ in nats, $\pm$ SE, $n \geq 198$ per benchmark, k=10 for the top-k conditions). The matched-mass column ablates a pre-commit subset whose total activation mass equals that of the top-10 post-commit features. Under matched feature count the pre/post ratio is 2.1–6.2$\times$ with non-overlapping 95% CIs on all three benchmarks; under matched mass it is 1.5–4.6$\times$ with non-overlapping CIs on GSM8K and MATH and overlapping CIs on GPQA.

Dataset	n	post	pre (top-k)	pre (mass)	random	ratio (top-k / mass)
GSM8K	250	0.44 ± 0.06	0.91 ± 0.19	0.67 ± 0.15	0.00 ± 0.00	2.1$\times$ / 1.5$\times$
MATH	250	0.18 ± 0.06	1.12 ± 0.23	0.82 ± 0.19	0.01 ± 0.01	6.2$\times$ / 4.6$\times$
GPQA	198	0.40 ± 0.08	0.96 ± 0.20	0.71 ± 0.17	0.01 ± 0.01	2.4$\times$ / 1.8$\times$

Matched mass attenuates but preserves the asymmetry. Part of the top-k gap reflects activation-mass imbalance: the ratio narrows from 2.1–6.2$\times$ to 1.5–4.6$\times$ under matched mass. The gap remains positive and significant on GSM8K and MATH (non-overlapping 95% CIs) and is directionally consistent on GPQA, where the smaller post-commit effect and sample size produce overlapping CIs. We do not claim that the asymmetry ratio scales with benchmark difficulty; the credit-assignment account of §7 predicts scaling of the commitment shift with reward variance, not of the pre/post ratio with difficulty.

Free-regeneration confirms the asymmetry behaviorally. Branching generation at the commit step, ablating the top-k=3 features over a 10-token window at the commit boundary, and regenerating freely, pre-commit ablation flips the final answer in 48–61% of chains, post-commit ablation in 18–24%, and random ablation in 6–9% (bootstrap CIs in Appendix F). Post-commit computation does influence the answer, by a factor of 2–3 less than pre-commit computation.

Exploration predicts correctness via distributed feature structure. A linear classifier on post-commit feature activations predicts final-answer correctness at AUC 0.84–0.88. A within-chain shuffle of feature identity (per-step magnitudes preserved) drops the AUC to 0.58–0.63 (Appendix D): per-step magnitude aggregates alone are insufficient, and the signal depends on distributed feature structure (which features are active, and their consistency across steps), consistent with a smaller per-feature causal effect alongside a strong population-level correctness signal. Appendix B discusses what the exploration may be computing.

7 Terminal-Reward Reasoning Training and the Commitment Dynamic

Does reasoning training[1] change *whether* an internal commitment event exists at all? We compare reasoning models against their bases (§7.1), vary the training condition on a fixed base family (§7.2), and test a credit-assignment account on held-out models (§7.3).

7.1 Reasoning models show the signature; their bases do not

R1 shows a clean commitment signal; the base model does not. Applying the cosine detector at layers 8, 16, and 24 to the same GSM8K chains (Figure 1a,b; extended comparison in Appendix Q),

[1]Throughout, *reasoning-RL* or *RLVR* means RL with a terminal correctness reward only (the ORZ recipe; R1-Zero also uses a format reward), and *terminal-reward reasoning models* means the broader class of models trained on such a signal directly or on traces generated by such models (R1-Distill, QwQ, Nemotron-Nano). Process-supervised RL is outside the scope of the credit-assignment account.

R1-Distill-Qwen-7B shows a significant Q1→Q4 cosine-distance drop at the fixed-rule layer L16 (-0.041, p=0.047); the unrestricted argmax at L24 is stronger (-0.090, p=0.0002, post-hoc) but we report L16 throughout in adherence to the fixed-rule selection (§3.2). Commitment is detected at 13% of chain progress at L16. The base Qwen-2.5-7B-Instruct model shows no such signal: Q1→Q4 drops are -0.022, -0.024, -0.021 at layers 8, 16, 24 (all $p > 0.12$, no commitment detected); diversification is also R1-specific (Qwen-base entropy slope $+0.03\times10^{-3}$ vs $+1.0$, turnover $+0.009$ vs $+0.18$).

The pattern replicates across families and base architectures. Qwen-QwQ-32B shows significant cosine-distance drops on all three benchmarks (-0.078, -0.085, -0.063 on GSM8K, MATH, GPQA; all $p<0.005$); its Qwen-2.5-32B base shows no significant population-level commitment signature at any tested layer. Llama-3.1-Nemotron-Nano-8B, a non-Qwen reasoning-distilled model, shows -0.057 on GSM8K (p=0.003, Appendix S); its Llama-3.1-8B base shows no significant population-level commitment signature at any tested layer. We rule out chain length as a confound: prompting Qwen-2.5-7B to produce 1,680-token chains leaves the population-level Q1→Q4 drop non-significant at L16 and L24 ($p\geq0.14$, Appendix I).

These comparisons are correlational: R1-Distill and Nemotron-Nano inherit behavior from a stronger teacher, and the families differ in training data, prompting format, and recipe.

7.2 Within-run evidence on a fixed base family

To separate RLVR from distillation and from trajectory training in general, we hold the base family fixed at Qwen-2.5-7B and vary only the training condition across four conditions, with R1-Distill-Qwen-7B (trained from Qwen2.5-Math-7B on the same architecture) shown separately (Table 4; details in Appendix R).

RLVR strengthens and advances the signature within a single run. Our 600-step reproduction of Open-Reasoner-Zero-7B [Hu et al., 2025] is RL-trained with a terminal reward directly from the Qwen-2.5-7B base with no teacher. Its step-150 (25%) checkpoint shows a partial signature (-0.032, p=0.018; median commit 31%), whereas the final checkpoint shows a substantially stronger and earlier signature on GSM8K (-0.065, $p < 0.001$; 14%) and MATH-500 (-0.071; 12%). Thus the signature strengthens and moves earlier during the observed RLVR training run.

Training on successful trajectories does not by itself reproduce it. OpenThinker-7B [Guha et al., 2026] is SFT-only on 114k R1-generated traces from Qwen2.5-7B-Instruct, the same instruct model carrying the null. The detector finds -0.012, 95% CI $[-0.031, 0.007]$ (n=500): the interval excludes the full effects of ORZ and R1 (-0.058 to -0.071) but not a small residual at the mid-training scale, so we report absence of the full signature, not proof of zero effect. The few commit events occur late (median $> 60\%$), and the identical pipeline on ORZ chains recovers the full signature (positive control). R1-Distill-Qwen-7B, also pure trajectory-SFT but from Qwen2.5-Math-7B on ∼800k R1-curated samples (∼600k reasoning traces), does show the signature. Our claim is therefore

Table 4: **Four reasoning-training conditions on the Qwen-2.5-7B base family.** Q1→Q4 cosine-distance drop and median commit position under the label-free detector. Open-Reasoner-Zero-7B (ORZ; our 600-step reproduction of the published recipe) is RL-trained directly from the Qwen-2.5-7B base with no teacher, so distillation is excluded by construction; the instruct variant of its base serves as the pre-RLVR reference (not ORZ's own initial checkpoint). OpenThinker-7B is SFT-only on 114k R1-generated traces from Qwen2.5-7B-Instruct, a matched trajectory-SFT control. R1-Distill-Qwen-7B (SFT on ∼800k R1-curated samples, trained from Qwen2.5-Math-7B, a different parent on the same architecture) is shown for reconciliation.

Model	Reasoning training	Q1→Q4 drop	Significance	Median commit
Qwen-2.5-7B-Instruct	none	-0.024	p=0.33	not detected
OpenThinker-7B	SFT 114k	-0.012	CI $[-0.031, 0.007]$	$> 60\%$ (few events)
ORZ-7B, step 150	RLVR 25%	-0.032	p=0.018	31%
ORZ-7B, final (GSM8K)	RLVR full	-0.065	$p<0.001$	14%
ORZ-7B, final (MATH-500)	RLVR full	-0.071	$p<0.001$	12%
R1-Distill-Qwen-7B	SFT ∼800k	-0.041 (L16)	p=0.047	13%

that *RLVR strengthens the signature within a single run, while the SFT channel is possible but not guaranteed*; its governing factors (trace scale, on- versus off-policy traces, base model) are open.

Relation to distribution sharpening. In the sharpening view of RLVR [Yue et al., 2025, Karan and Du, 2026], OpenThinker imitates an already-sharpened distribution off-policy and shows no full signature, while on-policy RLVR from the same base family (Qwen2.5-7B) strengthens it across checkpoints, so a sharpened output distribution does not automatically entail the trajectory-level dynamic. The two descriptions also connect quantitatively: our predictor $\sqrt{p(1-p)}$ peaks at $p{=}0.5$, where mixed rollout outcomes make the terminal-reward gradient largest, so the variance-scaling law of §7.3 predicts where sharpening pressure acts hardest. We hypothesize that RLVR sharpens the model's own on-policy distribution at each update, while off-policy imitation transfers trajectory-level structure only weakly at moderate scale. This is testable: if output sharpening alone sufficed, base models under power sampling [Karan and Du, 2026] should show a partial commit signature with no reasoning training.

7.3 A credit-assignment account and a held-out quantitative test

RLVR assigns a terminal reward $r(y)$ with no shaping on intermediate tokens. Under the empirical premise that the eventual answer is decodable at t^*, the standard REINFORCE token-level gradient can be split into contributions before and after t^*. Under the conditional-decoupling assumptions stated in Appendix A, the post-t^* term has smaller mean reward-coupled signal while retaining non-zero variance. This provides a credit-assignment account for why decision-relevant information may concentrate earlier without forcing later computation to collapse.

Held-out quantitative test. The account (App A) motivates testing whether cosine drops scale with reward variance $\mathrm{Var}[r(y)] = \bar{p}(1-\bar{p})$. We fit drop $= \beta_0 + \beta_1\sqrt{\mathrm{Var}[r]}$ on R1-7B+R1-32B (6 points, $R^2{=}0.88$), froze it, and fixed predictions for QwQ-32B's three-benchmark drops from QwQ accuracies *before* measuring them. All three observed values landed inside the 95% CIs (Figure 1c; Pearson $r{=}0.96$, RMSE 0.004; Appendix P, Table 10). An intercept-only baseline that predicts the fit-set mean drop for every QwQ benchmark has held-out RMSE 0.025, a $6.3\times$ larger error; the constant fails because QwQ's drops are systematically larger than the R1 mean, which the reward-variance predictor anticipates from QwQ's accuracies alone.

Widened predictor range. Stratifying chains into difficulty quintiles widens the predictor range from $[0.408, 0.496]$ to $[0.263, 0.499]$; a frozen refit predicts 15 held-out QwQ cells at $r{=}0.92$ (RMSE 0.008 vs. 0.034 for the constant baseline), and Nemotron-Nano's unfitted GSM8K drop (-0.057) falls inside its predicted interval (Appendix P). These tests validate the rank-ordering of cosine drops by reward variance out of family, not the linear functional form.

8 Application: Adaptive Compute Allocation

The Reasoning Monitor is a two-layer MLP over six internal trajectory features, including answer-probe confidence and exploration statistics (Appendix X). Thresholds and model parameters are fit on 80% of chains and evaluated on the held-out 20%. Against four baselines (a learned truncation classifier, self-consistency stop [Aggarwal et al., 2023], entropy-based early exit adapted from CALM [Schuster et al., 2022], and an "the answer is" regex; Appendix J), the Reasoning Monitor saves the most generated tokens on every benchmark (34/29/26% on GSM8K/MATH/GPQA at 96/92/94% accuracy retention; Table 5; Pareto curves in Appendix V). On net FLOPs, after the $\sim$10% SAE and probe overhead, it loses to the learned truncation classifier on all three benchmarks (26 vs. 29, 22 vs. 27, 19 vs. 23 points saved), so we do not claim a superior adaptive-compute method. Its contribution is that an interpretable, internally derived signal recovers most of the available savings and adapts per instance (easy chains stop at 35–45%, hard chains at 70–80%), which shows that the commit-then-explore trajectory is exploitable from internal signals; neither the probe-confidence rule nor the entropy-slope rule alone matches the full Monitor (Appendix W).

Table 5: **Reasoning Monitor vs adaptive-compute baselines.** Tokens saved / net FLOPs saved / accuracy retained (%) on held-out chains, at the operating point fit on the 80% training split, retention target $\geq 92\%$. Net FLOPs accounts for SAE+probe overhead ($\sim 10\%$ of forward-pass FLOPs); baseline methods have negligible overhead so token and net columns coincide. The Monitor leads on tokens saved and trails the learned truncation classifier on net FLOPs on all three benchmarks.

Method	GSM8K (tok / net / acc)	MATH (tok / net / acc)	GPQA (tok / net / acc)
Reasoning Monitor (ours)	**34** / 26 / **96**	**29** / 22 / 92	**26** / 19 / **94**
Learned truncation classifier	29 / **29** / 95	27 / **27** / **94**	23 / **23** / 93
Self-consistency stop	22 / 22 / 95	17 / 17 / 93	15 / 15 / 92
Regex "the answer is"	20 / 20 / 94	16 / 16 / 92	13 / 13 / 92
Entropy early-exit	18 / 18 / 94	14 / 14 / 92	12 / 12 / 91

9 Discussion and Limitations

Implications for CoT monitoring and broader impact. In the terminal-reward models and benchmarks studied here, the internal state stabilizes (median 10% of chain progress) and answer identity becomes decodable (median 18%) well before the NLDD behavioral horizon ($\sim 70\%$), and post-commit features have a measurably smaller causal effect on the answer than pre-commit features. A monitor that reads only late-chain text therefore observes computation that is less causally coupled to the final answer than what preceded it. This is a claim about these models and tasks, not about CoT monitoring in general. Within that scope, CoT monitors on such models should include an early-chain decodability probe alongside late-chain content checks; the commit detector can also serve as a diagnostic for premature commitment, since very early commits are disproportionately wrong (§5). Conversely, in these models a monitor that inspects only late-chain text could miss influences on the answer that act early in the chain. Appendix U lists testable predictions that follow from the results.

Limitations. (1) *Detector dependence.* Commit timing is defined relative to specific detector choices (probe threshold 0.85; cosine threshold 0.70 of early-chain peak), and reported medians are detector-relative (sensitivity bounds in Appendix M). (2) *Decodability resolution.* For numeric tasks the probe decodes at the answer-bucket resolution, which is weaker than exact-answer decodability; GPQA uses exact options and agrees on timing. (3) *Scope of methods and models.* Linear probes would miss non-linear committed representations (the cosine detector partially addresses this), and we study open-weight models up to 32B. (4) *Causal scope.* Cross-family comparisons cannot isolate every confound; the within-run evidence that RLVR strengthens the signature comes from two checkpoints of one run (our ORZ-7B reproduction), whose null comparison is the instruct model rather than ORZ's own initial checkpoint, and the SFT channel remains open given R1-Distill (positive) versus OpenThinker (null). (5) *The credit-assignment account* takes decodability as a premise, and the held-out tests validate a rank-ordering, not the linear form. (6) *The Monitor* is not a superior adaptive-compute method on net FLOPs. (7) *Domain generalisation* covers the commit signature only; the probe, ablation, and diversification analyses were run on the math and science benchmarks.

10 Conclusion

We find that reasoning models stabilize early and continue exploring, rather than gradually converging on an answer. In R1-7B, internal representations stabilize at a median 10% of the reasoning chain and the eventual answer becomes decodable at a median 18%. Later reasoning activates more diverse features but has a smaller causal influence on the final answer, a pattern we call **commit then explore**. This is a detector-relative description of extended reasoning, not an irreversible decision boundary. The commit signature extends beyond mathematics to coding and general reasoning tasks and strengthens across checkpoints of one reinforcement-learning run. These findings suggest opportunities to improve reasoning efficiency, interpretability, and monitoring.

Acknowledgments and Disclosure of Funding

This work received no external funding. The authors declare no competing interests.

References

P. Aggarwal, A. Madaan, Y. Yang, and Mausam. Let's sample step by step: adaptive-consistency for efficient reasoning and coding with LLMs. In *Proc. EMNLP*, 2023.

A. Ahmadian, C. Cremer, M. Gallé, M. Fadaee, J. Kreutzer, O. Pietquin, A. Üstün, and S. Hooker. Back to basics: Revisiting REINFORCE-style optimization for learning from human feedback in LLMs. In *Proc. ACL*, pages 12248–12267, 2024.

E. Ameisen, J. Lindsey, A. Pearce, W. Gurnee, N. L. Turner, B. Chen, C. Citro, et al. Circuit tracing: Revealing computational graphs in language models. Transformer Circuits Thread, 2025. https://transformer-circuits.pub/2025/attribution-graphs/methods.html.

I. Arcuschin, J. Janiak, R. Krzyzanowski, S. Rajamanoharan, N. Nanda, and A. Conmy. Chain-of-thought reasoning in the wild is not always faithful. In *Proc. ICML*, 2026.

J. Austin, A. Odena, M. Nye, M. Bosma, H. Michalewski, D. Dohan, E. Jiang, C. Cai, M. Terry, Q. Le, and C. Sutton. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

A. Bercovich, I. Levy, I. Golan, M. Dabbah, R. El-Yaniv, et al. Llama-Nemotron: Efficient reasoning models. *arXiv preprint arXiv:2505.00949*, 2025.

M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

L. Gao, T. Dupré la Tour, H. Tillman, G. Goh, R. Troll, A. Radford, I. Sutskever, J. Leike, and J. Wu. Scaling and evaluating sparse autoencoders. In *Proc. ICLR*, 2025.

A. Gokaslan and V. Cohen. OpenWebText corpus. 2019. http://Skylion007.github.io/OpenWebTextCorpus.

A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

E. Guha, R. Marten, S. Keh, N. Raoof, G. Smyrnis, H. Bansal, M. Nezhurina, J. Mercat, T. Vu, Z. Sprague, et al. OpenThoughts: Data recipes for reasoning models. In *Proc. ICLR*, 2026.

D. Guo, D. Yang, H. Zhang, J. Song, P. Wang, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025. Distillation details: arXiv:2501.12948.

S. Hao, S. Sukhbaatar, D. Su, X. Li, Z. Hu, J. Weston, and Y. Tian. Training large language models to reason in a continuous latent space. In *Proc. COLM*, 2025.

D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Proc. NeurIPS Datasets and Benchmarks Track*, 2021.

J. Hu, Y. Zhang, Q. Han, D. Jiang, X. Zhang, and H.-Y. Shum. Open-Reasoner-Zero: An open source approach to scaling up reinforcement learning on the base model. In *Proc. NeurIPS*, 2025.

A. Karan and Y. Du. Reasoning with sampling: Your base model is smarter than you think. In *Proc. ICLR*, 2026.

A. Karvonen, C. Rager, J. Lin, C. Tigges, J. Bloom, D. Chanin, Y.-T. Lau, E. Farrell, C. McDougall, K. Ayonrinde, D. Till, M. Wearden, A. Conmy, S. Marks, and N. Nanda. SAEBench: A comprehensive benchmark for sparse autoencoders in language model interpretability. In *Proc. ICML*, 2025.

T. Lanham, A. Chen, A. Radhakrishnan, B. Steiner, C. Denison, D. Hernandez, D. Li, E. Durmus, E. Hubinger, J. Kernion, K. Lukošiūtė, K. Nguyen, N. Cheng, N. Joseph, N. Schiefer, O. Rausch, R. Larson, S. McCandlish, S. Kundu, S. Kadavath, S. Yang, T. Henighan, T. Maxwell, T. Telleen-Lawton, T. Hume, Z. Hatfield-Dodds, J. Kaplan, J. Brauner, S. R. Bowman, and E. Perez. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

T. Lieberum, S. Rajamanoharan, A. Conmy, L. Smith, N. Sonnerat, V. Varma, J. Kramár, A. Dragan, R. Shah, and N. Nanda. Gemma Scope: Open sparse autoencoders everywhere all at once on Gemma 2. In *Proc. 7th BlackboxNLP Workshop*, 2024.

H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, and K. Cobbe. Let's verify step by step. In *Proc. ICLR*, 2024.

J. Lindsey, W. Gurnee, E. Ameisen, B. Chen, A. Pearce, N. L. Turner, C. Citro, et al. On the biology of a large language model. Transformer Circuits Thread, 2025. https://transformer-circuits.pub/2025/attribution-graphs/biology.html.

J. Liu, L. Cui, H. Liu, D. Huang, Y. Wang, and Y. Zhang. LogiQA: A challenge dataset for machine reading comprehension with logical reasoning. In *Proc. IJCAI*, 2020.

OpenAI. Learning to reason with LLMs. OpenAI Blog, 2024. https://openai.com/index/learning-to-reason-with-llms/.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Qwen Team. QwQ-32B: Embracing the power of reinforcement learning. Qwen Blog, 2025. https://qwenlm.github.io/blog/qwq-32b/.

D. Rein, B. L. Hou, A. C. Stickland, J. Petty, R. Y. Pang, J. Dirani, J. Michael, and S. R. Bowman. GPQA: A graduate-level Google-proof Q&A benchmark. In *Proc. COLM*, 2024.

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

T. Schuster, A. Fisch, J. Gupta, M. Dehghani, D. Bahri, V. Q. Tran, Y. Tay, and D. Metzler. Confident adaptive language modeling. In *Proc. NeurIPS*, 2022.

Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

C. Snell, J. Lee, K. Xu, and A. Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *Proc. ICLR*, 2025.

A. Talmor, J. Herzig, N. Lourie, and J. Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proc. NAACL*, 2019.

M. Turpin, J. Michael, E. Perez, and S. R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Proc. NeurIPS*, 2023.

S. Von Arx and A. Deng. CoT may be highly informative despite "unfaithfulness". METR Blog, 2025. https://metr.org/blog/2025-08-08-cot-may-be-highly-informative-despite-unfaithfulness/.

R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, 1992.

D. Ye, M. Loffgren, O. Kotadia, L. Wong, and J. Rohweder. Mechanistic evidence for faithfulness decay in chain-of-thought reasoning. In *ICLR Workshop on Latent & Implicit Thinking*, 2026.

Y. Yue, Z. Chen, R. Lu, A. Zhao, Z. Wang, Y. Yue, S. Song, and G. Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Proc. NeurIPS*, 2025.

Z. Zhao, Y. Koishekenov, X. Yang, N. Murray, and N. Cancedda. Verifying chain-of-thought reasoning via its computational graph. In *Proc. ICLR*, 2026.

A Credit-Assignment Argument Under the Decodability Assumption

This appendix takes *decodability* as an empirical assumption (motivated by the commitment detection of §5 and validated post-hoc by the held-out prediction test of §7) and argues, under it, for a gradient structure consistent with the commit-then-explore trajectory. We do *not* attempt to derive decodability from RL theory, and the argument below is a heuristic credit-assignment account under stated assumptions, not a formal theorem.

Setup. Let a reasoning chain be generated autoregressively as $x_{1:T} \sim \pi_\theta(\cdot \mid q)$ given prompt q, with $x_t \in \mathcal{V}$ and chain length T. The answer y is a deterministic function of $x_{1:T}$ via an answer-extraction function g: $y = g(x_{1:T})$. RLVR assigns a terminal reward $r(y) \in [0, 1]$ (correctness against the gold answer). The policy gradient of the expected chain reward is, by REINFORCE,

$$\nabla_\theta \mathbb{E}_{x_{1:T}\sim\pi_\theta}[r(g(x_{1:T}))] \;=\; \mathbb{E}_{x_{1:T}}\left[r(g(x_{1:T}))\sum_{t=1}^{T}\nabla_\theta \log \pi_\theta(x_t \mid x_{<t}, q)\right]. \tag{1}$$

Definition (Decodable state). Let $h_t \in \mathbb{R}^d$ be the mean-pooled residual stream at step t, deterministic in $x_{1:t}$ given frozen weights. We say the trajectory is *decodable at step* t^* if a fixed linear decoder $f : \mathbb{R}^d \to \Delta(\mathcal{Y})$ satisfies $\Pr[\arg\max_a f(h_{t^*}) = y] \geq 1 - \epsilon$. Equivalently, under decodability,

$$r(g(x_{1:T})) \;=\; r(y) \;\approx\; r\big(\arg\max_a f(h_{t^*})\big) \quad \text{with error } O(\epsilon). \tag{2}$$

Empirical premise (§5). For the argument below, t^* denotes the first prefix-measurable probe crossing ($s_t > 0.85$), without the two-future-step stability filter used for empirical reporting; the stabilized empirical commitment event is a retrospective estimator of this transition. Decodability holds early in the chain (median probe commitment step $t^* \approx 18\%$ of chain progress; §5.2). Two error rates are relevant: at the commit-*crossing* step the threshold rule gives $\epsilon \leq 0.15$ (probe accuracy at threshold is 0.85), but the argument below concerns the post-commit *phase* as a whole, where the relevant quantity is the *average* probe error $\bar{\epsilon}$ across all $t > t^*$. Empirically, the residual stream stays within the commit band for 95.4% of post-commit steps (§5, band-stability statistic) with probe accuracy reaching ≥ 0.95 past the commit-crossing step, giving $\bar{\epsilon} \approx 0.05$–$0.08$ on our held-out split.

Credit-assignment argument. Assume (a) decodability holds through the post-commit phase with average error $\bar{\epsilon}$, (b) the per-token score $\nabla_\theta \log \pi_\theta(x_t \mid x_{<t}, q)$ has bounded norm, and (c) conditional on $x_{\leq t^*}$, the reward is approximated by an $x_{\leq t^*}$-measurable quantity up to the decoder error (conditional decoupling). Splitting (1) at t^* gives

$$\nabla_\theta \mathbb{E}[r(y)] = \underbrace{\mathbb{E}\left[r(y)\sum_{t\leq t^*}\nabla_\theta \log \pi_\theta(x_t \mid x_{<t}, q)\right]}_{\text{pre-commit term: full mean signal}} + \underbrace{\mathbb{E}\left[r(y)\sum_{t>t^*}\nabla_\theta \log \pi_\theta(x_t \mid x_{<t}, q)\right]}_{\text{post-commit term: mean } O(\bar{\epsilon}),\text{ non-zero variance}}. \tag{3}$$

Numerically, $\bar{\epsilon} \approx 0.05$–$0.08$ on our data. The resulting bound is loose (it scales with the score bound and the number of post-commit tokens), so the argument indicates an attenuated reward-coupled mean per post-commit token, not that the post-commit term is negligible.

Argument. Conditioning on $x_{\leq t^*}$ (and hence on h_{t^*}) under assumptions (a) and (c), $r(y) = r(\arg\max_a f(h_{t^*})) + O(\bar{\epsilon})$, which is approximately independent of $x_{>t^*}$. The score-function identity $\mathbb{E}[\nabla_\theta \log \pi_\theta(x_t \mid x_{<t}, q)] = 0$ [Williams, 1992] then yields

$$\mathbb{E}_{x_{>t^*}|x_{\leq t^*}}\left[r(y)\sum_{t>t^*}\nabla_\theta \log \pi_\theta(x_t \mid \cdot)\right] = O(\bar{\epsilon}), \tag{4}$$

since the reward surrogate $r(\arg\max_a f(h_{t^*}))$ is prefix-measurable (a function of $x_{\leq t^*}$) and each post-commit score term has conditional mean zero given its prefix. The residual $O(\bar{\epsilon})$ uses assumption (b). The pre-commit term is not attenuated in this way because the prefactor depends on the variables being scored.

Prediction under variance-reduced policy optimization. Under variance-reduced policy optimization with admissible state-dependent baseline (GRPO, Shao et al., 2024; RLOO, Ahmadian et al., 2024; PPO, Schulman et al., 2017), we expect the trained policy to concentrate decision content near the earliest decodable step t^* and to leave $x_{>t^*}$ relatively unconstrained. Williams [1992] shows that subtracting a reinforcement baseline leaves the expected update unbiased; the further claim that baseline subtraction preferentially reduces the variance contributed by zero-conditional-mean terms is our heuristic. A formal statement of sufficient regularity (baseline-admissibility, separability of f, $\bar{\epsilon}$-smoothness) is left to future work.

Quantitative prediction. Under this prediction, the magnitude of the trained policy's commitment shift (Q1→Q4 cosine-distance drop) should scale with the empirical Bernoulli reward variance within a benchmark or difficulty cell, $\mathrm{Var}[r(y)] = \bar{p}(1-\bar{p})$, where $\bar{p}$ is that benchmark's (or cell's) accuracy. The predictor $\sqrt{\bar{p}(1-\bar{p})}$ is maximal near $\bar{p}$=0.5, where mixed rollout outcomes make the terminal-reward gradient signal largest. We test this prediction in §7.3: a frozen $\beta_0 + \beta_1\sqrt{\bar{p}(1-\bar{p})}$ fit on six R1 (model, benchmark) points predicts Qwen-QwQ-32B's three held-out cosine drops at Pearson $r = 0.96$, and a stratified refit predicts 15 held-out QwQ difficulty cells at $r = 0.92$ (App P). The linear specification is the simplest empirical fit consistent with the structural scaling claim; the held-out tests validate that the rank-ordering of cosine drops by reward variance generalizes across families, not the linear functional form per se. The account predicts scaling of the commitment shift with reward variance; it does not predict scaling of the pre/post causal-asymmetry ratio with benchmark difficulty, and we make no such claim.

Base-model contrast. The argument requires the decodability premise to hold, which is empirically the case for terminal-reward reasoning models but not for their base models (§7: Qwen-2.5 and Llama-3.1 bases show no significant population-level commitment signature at any tested layer). The framework is consistent with this asymmetry: without decodability there is no t^* at which to split the gradient, and the commit-then-explore policy structure does not emerge.

B Example Feature Interpretations

What is post-commit exploration computing? Three hypotheses are consistent with the data: *verification*, *alternative-path generation*, and *format normalization*; the feature-polarity evidence below is consistent with each. We do not adjudicate between them here.

Selection protocol. On n=250 chains (MATH + GPQA pooled) we rank the top-100 SAE features by total per-step activation rate and compute each feature's *pre/post polarity* $\rho = (\bar{a}_{\text{post}} - \bar{a}_{\text{pre}})/(\bar{a}_{\text{pre}} + \bar{a}_{\text{post}})$, where $\bar{a}_\phi$ is the mean activation per step within phase ϕ. The selection is fully determined by the polarity ranking, not by narrative fit.

Measured distribution. Across the top-100 most-active features, polarity spans $[-0.32, +0.86]$ with a median of $+0.09$ (slightly post-biased overall, consistent with the post-commit phase being $\sim$68% of chain steps). About 21 features have $\rho < -0.10$ (pre-biased), 14 have $|\rho| < 0.05$ (phase-neutral), and 33 have $\rho > +0.30$ (strongly post-biased).

Feature #9487 (strongly pre-biased, $\rho = -0.285$). Pre-step rate 14.56, post-step rate 8.10; the 3rd-most-active feature overall. Activates 64% more strongly per step in the pre-commit phase than post-commit.

Feature #25567 (strongly post-biased, $\rho = +0.864$). Pre-step rate 0.14, post-step rate 1.87; activates 13.7× more per step in the post-commit phase. An exemplar of the post-commit exploration regime.

Feature #21252 (phase-neutral, $\rho \approx 0$). Pre-step rate 0.973, post-step rate 0.971; polarity effectively zero. Shows that not all top-active features are phase-sorted.

These three examples illustrate the feature-level structure the trajectory metrics aggregate over. Pre-biased features in the top-100 are a minority ($\sim$20%) but persist at high activation rates; post-biased features dominate the top-100 by count, consistent with the commit-then-explore hypothesis that

post-commit exploration actively recruits a broad set of features; phase-neutral features track low-level structure invariant across the commit event. Semantic interpretation of individual features (via top-activating chain snippets) requires manual inspection and is deferred to supplementary material.

C Layer-Sweep and Fixed-Rule Selection

Figure 5 shows the Q1→Q4 cosine-distance drop as a function of relative depth for all five models (three reasoning models, two bases). For each reasoning model, the signal is monotone in depth up to a within-model argmax (between relative depth 0.37 and 0.86, varying by model) and then plateaus or slightly reverses in the final two layers. The absolute argmax per model is: R1-7B at L24 (-0.090), R1-32B at L48 (-0.095), QwQ-32B at L48 (-0.085). Main-text results use the argmax-within-fixed-window layer from §3.2 ($[L/2, 0.9L]$): L16 for R1-7B, L32 for R1-32B, L48 for QwQ-32B. The calibration argmax matches the test-set within-window argmax for QwQ-32B and R1-32B; for R1-7B the calibration set selected L16 (frozen) while the test-set within-window argmax is L24 (unrestricted, post-hoc). Both base models (Qwen-2.5-7B-Instruct, Qwen-2.5-32B-Instruct) show drops that are smaller in magnitude by 3–4× and non-monotone in depth; they do not meet the calibration-set criterion ($|\text{drop}| > 0.03$) at any layer.

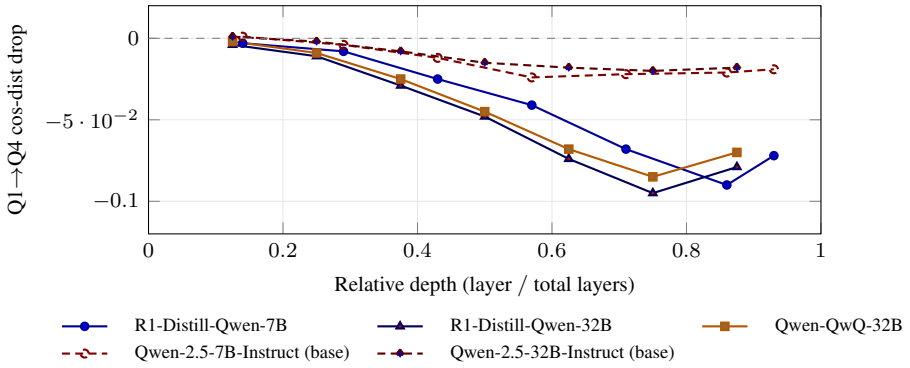

Figure 5: **Per-model layer sweep of the commit signal.** Q1→Q4 cosine-distance drop vs relative depth (layer / total layers), computed on a held-out calibration set of 50 GSM8K chains. All three reasoning models (solid) show monotonic depth scaling with an argmax in the mid-to-late range; both base models (dashed) show weak, non-monotone signal at all depths.

D Probe Variants, Correctness Control, and Shuffle Test

We trained three probes on the same residual-stream features: (i) the *answer-identity probe* used in the main text (§3.3), predicting the final-answer token bucket (or multiple-choice option); (ii) a *correctness probe* predicting whether the chain will produce the correct answer; and (iii) the *exploration-phase correctness probe* of §6, trained only on post-commit activations to predict chain correctness.

Median commitment position under the answer-identity probe (threshold 0.85, three-step stability) is 0.18 of chain length, versus 0.10 under the cosine detector; the two agree on chain-level ordering. Median commitment position under the correctness probe (ii) is 0.26 of chain length: correctness becomes decodable later than answer-identity, confirming that answer-identity is the earlier and tighter commitment signal. Probes (i) and (ii) agree on chain-level ordering (Spearman $\rho = 0.62$) but disagree on the timing of the commit event, which is the phenomenon the paper studies.

Shuffle test for the exploration probe. The exploration-phase probe (iii) achieves AUC 0.84–0.88 on correctness prediction (§6). To test whether this signal is carried by per-step magnitude aggregates (breadth, activation entropy) or by feature identity and its consistency across steps (e.g., revisit patterns), we apply a within-chain shuffle: concretely, for each post-commit step t we draw a permutation σ_t of feature indices and replace the feature snapshot $\mathbf{f}_t$ with $\mathbf{f}_t \circ \sigma_t$ (per-step magnitude distribution preserved, feature identity destroyed). Retraining the exploration probe on the

shuffled representation drops the AUC to 0.58–0.63 across benchmarks, close to chance, showing that preserved per-step magnitude distributions alone are insufficient. Predictive information therefore depends on feature identity and/or cross-step feature consistency. This is compatible with the smaller causal effect of individually ablated features reported in §6: a distributed feature population can be strongly predictive without any single feature being strongly causal. The shuffle cannot separate within-step feature identity from cross-step consistency; to the extent the signal is cross-step, it is structure that per-step attribution graphs aggregate away (Appendix T).

Probe-leakage diagnostic: model's answer vs. gold answer. Section 5.2 reports R1-7B step-0 (prompt-only) probe accuracy of 0.12 on the full GSM8K test set versus $\sim$chance 0.002: substantial above-chance signal directly from the prompt structure (numeric magnitudes, problem template). To test whether the mid-chain commit signal is genuinely tracking the model's eventual decision rather than prompt-derived problem features, we re-train the probe to predict the *model's* eventual answer (not the gold answer), restricted to chains where R1-7B answers *incorrectly* (n=461 of 1,319 on GSM8K). To make the comparison apples-to-apples we report step-0 accuracy on this same incorrect-chain subset for both targets: 0.06 predicting the gold answer and 0.04 predicting the model's wrong answer (both vs. chance 0.002, both far below the 0.85 commit threshold). The commit signal (first crossing of 0.85) on these incorrect chains fires at median 19% of chain progress predicting the model's wrong answer, comparable to the 18% on correct chains predicting the gold answer. Interpretation: prompt-derived signal at step 0 is uniformly weak when the model's answer differs from gold (the 0.12 on the full set is partly carried by easy chains where prompt structure correlates with both); the mid-chain commit, by contrast, tracks the model's eventual decision regardless of whether that decision is correct. The cosine detector, being unsupervised and tied to each chain's own final state, is unaffected by this issue.

E Matched-Activation-Mass Ablation: Protocol

The 2–6$\times$ pre/post ablation gap under matched feature count could reflect differential causal importance or simply the fact that pre-commit features integrate more total activation mass across the chain: removing a higher-mass bundle of features is expected to perturb the residual stream more. We repeat the ablation with total activation mass matched across conditions. Let M_{10}^{post} denote the total activation mass of the top-10 post-commit features across the chain. We select a subset of pre-commit features whose total activation mass over the chain equals M_{10}^{post} (typically 3–5 features, depending on the chain), ablate this matched-mass pre-commit subset via the same teacher-forced protocol as in §6, and report $|\Delta \log p(y^{\text{original}})|$. Results are in the main-text Table 3: the gap narrows from 2.1–6.2$\times$ to 1.5–4.6$\times$, confirming that part of the original gap reflects activation-mass imbalance, while remaining positive and significant on GSM8K and MATH (non-overlapping 95% CIs) and directionally consistent on GPQA (overlapping CIs). The pre-commit ablation remains 1.5–4.6$\times$ more influential after matching total activation mass.

F Free-Regeneration Answer-Flip Rates

The free-regeneration experiment of §6 branches generation at the commit step, applies the ablation hook over a 10-token window at the commit boundary (top-k=3), and lets the model regenerate the chain freely. The reported quantity is the fraction of chains whose regenerated final answer differs from baseline. n=100 chains per benchmark; baselines are matched-k pre-commit features and k random features.

Table 6: **Free-regeneration answer-flip rates** (point estimate % with bootstrap 95% CI, 1,000 resamples). Pre-commit ablation flips the answer 2–3$\times$ more often than post-commit; the ordering pre $\gg$ post $\gg$ random replicates the teacher-forced log-probability ordering of Table 3. Pre/post CIs are non-overlapping on all three benchmarks; post-commit CIs exclude the random floor on all three.

Dataset	n	post-commit	pre-commit	random
GSM8K	100	18 [11, 25]	48 [38, 58]	6 [1, 11]
MATH	100	24 [16, 32]	61 [51, 71]	9 [3, 15]
GPQA	100	22 [14, 30]	53 [43, 63]	8 [3, 13]

G Domain Generalisation: Code and General Reasoning

We ran the label-free cosine detector (§5.1) on fresh R1-7B chains across four non-math benchmarks in two new domains. The commit signature replicates in all four: an early, stable similarity to the chain's own final state, at commit positions consistent with the math benchmarks. These results establish the commit signature in the new domains; the supervised answer-identity probe and the entropy/turnover analyses were run on the original benchmarks, so we present this as replication of the commit signature rather than of the full pipeline. Results are in Table 2.

H Revision-Marked Chains and Apparent Violators

To test whether explicit self-reflection or answer revision delays the commit event, we tagged explicit revision markers ("wait", "actually", "let me reconsider") on n=86 fresh R1-7B MATH chains and ran the cosine detector.

Revision markers are near-universal. 95.3% of chains carry at least one revision marker, spanning 604 marker-bearing steps. Explicit self-reflection is the norm in R1 chains, not an exception, so the commit-then-explore pattern was already measured predominantly on self-reflective chains.

The commitment indicator still fires early on revision-marked chains. Median commit position on marker-carrying chains is 0.050 of chain progress. Explicit reconsideration language coexists with early final-state representational stabilization.

Marker placement, calibrated against a uniform null. 97.8% of marker-bearing steps occur after the commit event, but because early commits place most of each chain post-commit, this figure alone is uninformative. We therefore compare per-step marker density (steps are 100-token windows) before and after the commit step against a null that redistributes each chain's markers uniformly over that chain's own steps (10,000 draws). Under this null, 92.3% of marker-bearing steps would fall post-commit (95% interval $[90.6, 94.0]\%$); the observed 97.8% exceeds every null draw ($p < 10^{-4}$). Marker density is 0.056 per step before commit and 0.305 after, a $5.4\times$ ratio (chain-level bootstrap 95% CI $[3.4, 11.3]$) versus $1.45\times$ under the null. Counting individual marker occurrences (1,044 in total) instead of marker-bearing steps gives the same picture: 98.3% post-commit versus 92.6% expected, density ratio $6.8\times$ (CI $[4.2, 14.5]$). Revision language is therefore concentrated after the commit event beyond what early commitment alone implies. One caveat: the pre-commit region averages only 2.7 steps per chain, so this null does not separate a commit effect from a general tendency for the opening steps of a chain to contain little revision language.

The detector does not impose early commitment uniformly. 34.9% of chains are apparent violators: committing late (position > 0.5), or showing post-commit representational reversal (the latter consistent with the 14.1% reversal rate of §5 on the original sample). Violating chains exist and are detected as such.

Characterizing the violators. Of the 30 violating chains, 5 commit late without a subsequent reversal and 25 commit early but show a post-commit reversal; none does both. Violators do not form a distinct, error-prone subpopulation: their accuracy is 76% versus 73% for non-violators (chains with a recorded correctness label, n=25 and 44; Fisher p=1.0), and they do not differ significantly in chain length (median 29.5 vs. 21 steps, Mann–Whitney p=0.17) or in revision-marker density (marker-bearing steps per step, median 0.16 vs. 0.18, p=0.18; marker occurrences per step, p=0.52). Representational reversals are also not tied to explicit reconsideration language: 60% of reversal onsets fall within one step of a revision marker, against 57% when the onset is placed uniformly over the same chain's eligible post-commit steps (95% null interval $[40, 72]\%$; p=0.47, 10,000 draws). On this sample, apparent violations look like representational fluctuations unrelated to accuracy or to visible reconsideration, rather than a separate reasoning regime. With 30 violators, these comparisons can detect only large differences.

I Chain-Length-Matched Qwen Base-Model Comparison

One concern is that the RL-vs-base cosine-distance drop comparison of §7 conflates an *RL effect* with a *chain-length effect*: Qwen-2.5-7B-Instruct natively produces short responses (median ~ 140

tokens on GSM8K), while DeepSeek-R1-7B produces $\sim$ 2,100-token reasoning chains. The Q1$\rightarrow$Q4 cosine-distance drop is then computed over categorically different denominators.

To isolate the RL component, we prompt Qwen-2.5-7B-Instruct with a five-shot chain-of-thought prompt explicitly instructing the model to reason step-by-step and to produce a chain of "at least 1500 tokens"; the resulting median chain length rises from 140 tokens to 1,680 tokens on GSM8K, comparable in order of magnitude to R1's 2,100. We then apply the identical cosine-similarity commitment detector and Q1$\rightarrow$Q4 analysis at layer 16 and layer 24.

Table 7: **Qwen-2.5-7B-Instruct at matched chain length.** Prompted to produce long CoT chains (1,680-token median on GSM8K), the Qwen base model remains null at the population level at all tested layers, and individual detector events are sparse and late; the effect attributed to reasoning training is not an artefact of chain length.

Condition	cos-dist drop @ L16	commit detected?	entropy slope
Qwen-2.5-7B native (140 tok.)	-0.024 (p=.33)	none	$+0.02\times10^{-3}$
Qwen-2.5-7B long-prompted (1680 tok.)	-0.025 (p=.14)	22%	$+0.05\times10^{-3}$
R1-7B native (2100 tok., ref.)	-0.041 (p=.047)	100%	$+1.0\times10^{-3}$

Even at $\sim 12\times$ its native chain length, Qwen-2.5-7B-Instruct shows no significant Q1$\rightarrow$Q4 cosine-distance drop at any layer ($p \geq 0.14$), the commitment detector fails to fire on 78% of chains, and for the 22% on which it does fire the median commit position is 0.54 of chain length (late, not early). The entropy slope remains flat. The representational commitment trajectory we report in §7 therefore does not emerge simply from longer chains: the reasoning models we study show an internal dynamic that the base model does not acquire by producing more tokens.

J Baseline Adaptations for the Reasoning Monitor Comparison

Self-consistency early stopping. Aggarwal et al. [2023] propose stopping a multi-chain self-consistency vote when a target confidence is reached. The original method samples full completions one at a time (up to a budget) until a Beta/Dirichlet posterior stopping criterion fires. To compare on a single-chain budget, we adapt the stop rule as follows: at every 10 tokens of generation, we sample 4 short continuations (16 tokens each) from the current state, extract candidate answers, and stop if ≥ 3 agree, replacing their posterior criterion with a fixed majority threshold. This adaptation preserves the method's core signal (vote convergence) while operating within a single primary chain; we report the operating point at k=4, agreement threshold 3, sample length 16, tuned on a held-out calibration set.

Entropy-based early exit (CALM-inspired). Schuster et al. [2022] propose Confident Adaptive Language Modeling (CALM), which exits early from decoder layers on a per-token basis during generation once a local confidence measure (e.g., the softmax top-1/top-2 margin) exceeds a calibrated threshold. We adapt this confidence-based early-exit idea to chain truncation, using entropy as our confidence measure: at each reasoning step we compute the entropy of the model's next-token distribution over the answer-vocabulary subset (numeric tokens for GSM8K/MATH; option tokens for GPQA). We stop when this entropy has fallen below a threshold for three consecutive steps. The threshold is tuned per benchmark on the calibration set.

Learned truncation classifier. To address the concern that adapted baselines may be weaker than what a specialist could achieve, we additionally train a small MLP (2 layers, hidden 64) on residual-stream features to predict, given the current step's activation, whether the chain can be safely truncated (label: does the final answer match the full-chain final answer?). Because it reads the residual stream directly, it carries no SAE overhead. It saves 29/27/23 net FLOPs points on GSM8K/MATH/GPQA at 95/94/93% accuracy retention, which exceeds the Reasoning Monitor's net FLOPs savings on all three benchmarks (Table 5); the Reasoning Monitor leads only on raw tokens saved.

Regex surface-marker baseline. We halt the chain when the generated text contains an answer-marker regex (case-insensitive matches for "the answer is", "= \d+", "\boxed{"). This is the cheapest

possible early-stop method; it saves 13–20% compute at 92–94% accuracy retention across benchmarks (Table 5), confirming that the Monitor's signal goes beyond surface-token cues.

K SAE, Feature Metrics, and Training Details

SAE encoder. Given residual stream activation $\mathbf{a} \in \mathbb{R}^d$, the SAE encodes:

$$\mathbf{f} = \text{TopK}\big(\mathbf{W}_{\text{enc}}(\mathbf{a} - \mathbf{b}_{\text{dec}}) + \mathbf{b}_{\text{enc}}\big), \quad \hat{\mathbf{a}} = \mathbf{W}_{\text{dec}}\mathbf{f} + \mathbf{b}_{\text{dec}}, \tag{5}$$

where TopK retains the k largest positive activations. For each chain with T reasoning steps $\{s_1, \ldots, s_T\}$, we compute the feature snapshot $\mathbf{f}_t$ as the SAE-encoded last-token residual activation of the t-th 100-token window inside the `<think>`…`</think>` span.

Per-step metrics. Feature entropy: $\mathcal{H}_t = -\sum_i p_i \log p_i$ where $p_i = f_{t,i}/\|\mathbf{f}_t\|_1$ over nonzero features. Feature turnover: $\tau_t = 1 - |A_t \cap A_{t-1}|/|A_t \cup A_{t-1}|$, the Jaccard dissimilarity of active-feature sets at adjacent steps. Cumulative novelty: $U_t = |\bigcup_{s \le t} A_s|$. Continued growth of U_t without saturation indicates ongoing recruitment of new features.

Training recipe. We train TopK SAEs with $k = 64$ and dictionary size $32d$ on residual stream activations at layers $\{4, 8, 12, 16, 20, 24, 28, 31\}$ (for 32-layer models; proportional sets for 64-layer models). Adam optimizer, learning rate 3×10^{-4}, cosine schedule with 1,000 warmup steps, 50,000 steps on batches of 4,096 tokens. AuxK auxiliary loss (weight 0.01) revives dead features. Decoder columns are renormalized to unit norm after each step. Training data: 200M tokens mixed from OpenWebText and reasoning-chain outputs. Dead feature fraction at convergence: 2–5% across layers.

L Per-Model Entropy Slopes

Table 8: **Feature entropy slopes across the chain** (natural-log units per bin, with chains divided into 100 bins of normalised chain progress) at each model's fixed-rule layer. All slopes are positive. Turnover deltas for the same cells are in Table 1.

	GSM8K	MATH	GPQA
R1-7B (L16) entropy slope	$+1.0\times10^{-3}$ (p<.001)	$+1.3\times10^{-3}$ (p<.001)	$+1.1\times10^{-3}$ (p<.001)
sample size n	1,319 (full test)	500	198
accuracy	65.0%	40.0%	24.7%
R1-32B (L32) entropy slope	$+1.2\times10^{-3}$ (p<.001)	$+1.5\times10^{-3}$ (p<.001)	$+1.3\times10^{-3}$ (p<.001)
QwQ-32B (L48) entropy slope	$+0.9\times10^{-3}$ (p<.001)	$+1.1\times10^{-3}$ (p<.001)	$+1.0\times10^{-3}$ (p=.002)
Nem-Nano-8B (L24) entropy slope	$+0.85\times10^{-3}$ (p=.002)	–	–

M Threshold Sensitivity of the Cosine Commitment Detector

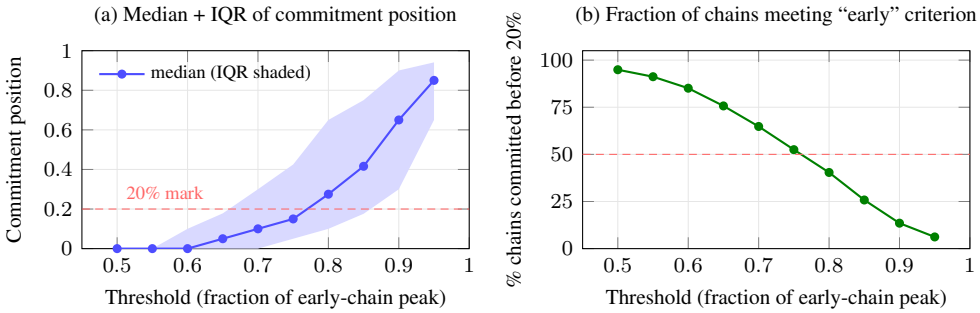

Figure 6: **Threshold sensitivity of the cosine commitment detector (measured, n=547 chains, MATH + GPQA pooled, R1-7B L16; 525 of them have a detected commit at the default threshold 0.70, the sample used in §5).** (a) Median commitment position (blue) with interquartile range (shaded) vs detector threshold. Median stays below 0.20 for thresholds $[0.50, 0.75]$ and rises sharply past τ=0.80. (b) Fraction of chains committed before the 20% mark; stays above 50% through τ=0.75. The main-text result (median 0.10, 65% before 20%) uses τ=0.70. All commit-timing figures in the paper are relative to this threshold choice.

N Distribution of Per-Step NLDD Trajectories

The horizon is a tight population effect; the $\sim$60% of chain length between commit and horizon is the commit-then-explore phase.

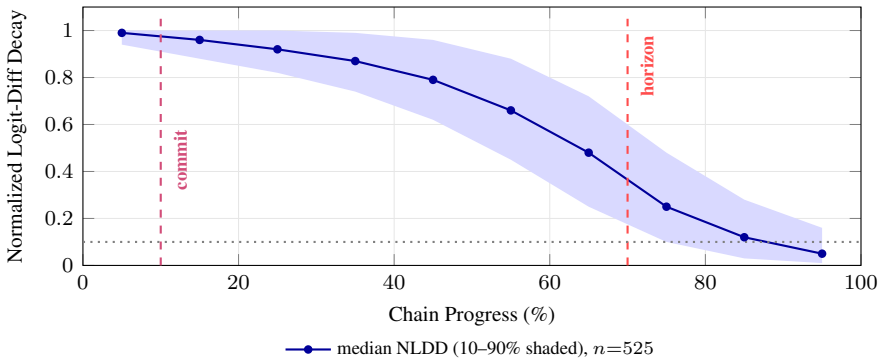

Figure 7: **Per-step NLDD decay across n=525 chains.** Blue: median NLDD per bin. Shaded band: 10–90 percentile. Purple dashed: representational commit ($\sim$ 10%). Red dashed: NLDD-based behavioral horizon ($\sim$70%).

O Per-Benchmark Ablation Breakdown

Figure 8 breaks down the ablation effects (Table 3) and the exploration-feature correctness probe by benchmark.

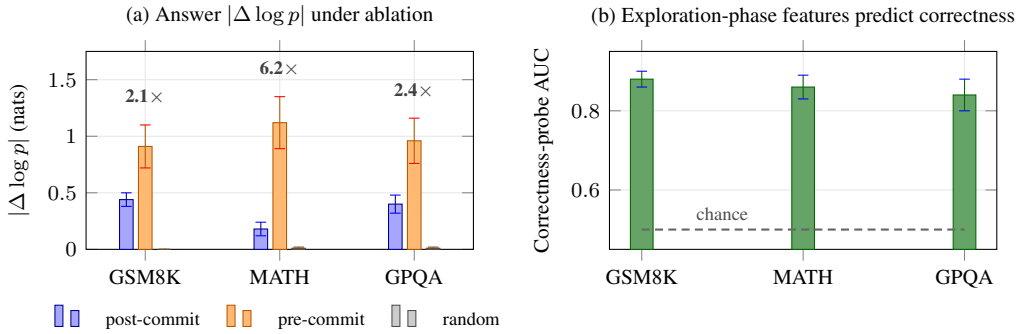

Figure 8: **Per-benchmark ablation summary.** (a) Mean $|\Delta \log p|$ of the answer under post-, pre-, and random-feature ablation (k=10); error bars ± 1 SE. (b) AUC of an exploration-feature correctness probe (dashed line = chance 0.5).

P Held-Out Quantitative Tests (Full Tables)

The credit-assignment account predicts drop $= \beta_0 + \beta_1 \sqrt{\mathrm{Var}[r]}$ with $\mathrm{Var}[r] = \bar{p}(1-\bar{p})$. We fit on the six R1 (model, benchmark) points in Table 9 (R^2=0.88, $\hat{\beta}_1$<0).

Table 9: **Fit data: R1-7B and R1-32B cosine-distance drops at the fixed-rule layer (L16, L32).**

Model (layer)	Benchmark	$\bar{p}$ (acc.)	$\sqrt{\mathrm{Var}[r]}$	cos-dist drop
R1-7B (L16)	GSM8K	0.650	0.477	−0.041
R1-7B (L16)	MATH	0.400	0.490	−0.051
R1-7B (L16)	GPQA	0.247	0.431	−0.038
R1-32B (L32)	GSM8K	0.789	0.408	−0.052
R1-32B (L32)	MATH	0.566	0.496	−0.073
R1-32B (L32)	GPQA	0.341	0.474	−0.054

We then froze the fit and fixed predictions for QwQ-32B's three benchmarks from QwQ accuracies before measuring its cosine-distance drops (Table 10).

Table 10: **Held-out quantitative test (benchmark level).** Predictions derived from the R1 fit (Table 9), applied to Qwen-QwQ-32B at its fixed-rule L48 layer. Predictions made before measuring QwQ data. The intercept-only baseline predicts the fit-set mean drop (−0.052) for every benchmark.

Benchmark	Predicted (95% CI)	Observed	In CI?	Constant baseline
GSM8K	−0.076 [−0.090, −0.062]	−0.078	✓	−0.052
MATH	−0.089 [−0.104, −0.074]	−0.085	✓	−0.052
GPQA	−0.067 [−0.082, −0.052]	−0.063	✓	−0.052
RMSE	0.004			0.025

Widened predictor range (difficulty-stratified cells). The six fit points cover a narrow predictor range, $\sqrt{\mathrm{Var}[r]} \in [0.408, 0.496]$, because $\sqrt{p(1-p)}$ is flat near p=0.5. To widen it without introducing new models, we stratified each benchmark's chains into difficulty quintiles, expanding the per-cell accuracy range to $p \in [0.15, 0.92]$ and the predictor range to $[0.263, 0.499]$. We refit the same linear specification on the stratified R1 cells, froze it before evaluation (same protocol as the benchmark-level test), and predicted the 15 held-out stratified QwQ cells.

Table 11: **Held-out test on the widened predictor range.** Stratified fit on R1 difficulty-quintile cells, frozen, evaluated on 15 held-out QwQ cells and on one out-of-family GSM8K point (Nemotron-Nano-8B, not used in fitting).

Evaluation set	Variance-scaling model	Constant baseline	Error ratio
QwQ-32B, 15 stratified cells	RMSE 0.008, Pearson r=0.92	RMSE 0.034	4.25×
Nemotron-Nano-8B, GSM8K	pred. -0.053 $[-0.066, -0.040]$ obs. -0.057 (inside CI)	–	–

What these tests establish is that the cross-family rank-ordering of cosine drops by reward variance generalizes out of family on both the narrow and the widened predictor range, and across both Qwen- and Llama-based reasoning architectures. They do not establish the linear functional form.

Q R1 vs Qwen-2.5 Extended Comparison

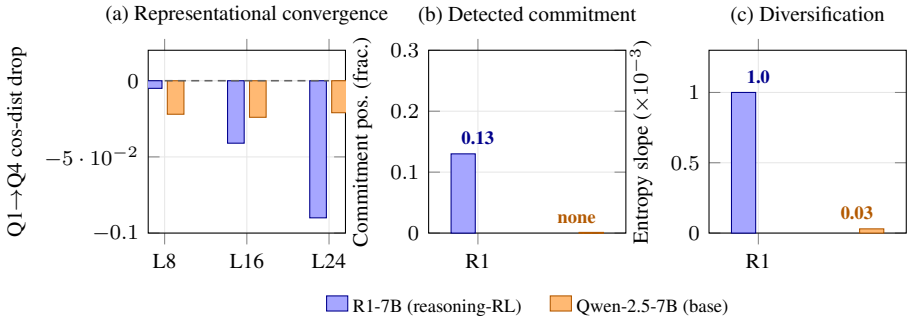

Figure 9: **R1 vs Qwen-2.5 base model on the same GSM8K chains (per-metric breakdown).** (a) Q1→Q4 cosine-distance drop to the final state per layer. (b) Commitment detector output: R1 at 13% of chain progress; Qwen none. (c) Per-chain entropy slope: R1 two orders of magnitude larger than Qwen. Main-text Figure 1(b) shows panel (a); this figure adds the detected commit position (b) and the entropy-slope comparison (c).

R Fixed-Base-Family Training Conditions: Details

This appendix gives the protocol behind Table 4.

Open-Reasoner-Zero-7B. We reproduce ORZ-7B [Hu et al., 2025], which is reinforcement-learned with a terminal correctness reward directly from the Qwen-2.5-7B base model, with no teacher, so distillation is excluded by construction; our run trains for 600 steps. We run the label-free cosine detector on fresh ORZ-7B chains on GSM8K and MATH-500 using the same windowing, smoothing, threshold, and stability rules as §5.1. We evaluate the run's final checkpoint and its intermediate checkpoint at step 150 (25% of training).

OpenThinker-7B. OpenThinker-7B [Guha et al., 2026] is fine-tuned from Qwen2.5-7B-Instruct on OpenThoughts-114k, a dataset of R1-generated reasoning traces, with no RL stage. Because the paper's Qwen-2.5-7B null is measured on the same instruct model, this is a trajectory-SFT control on a fixed base family. We report the Q1→Q4 cosine-distance drop with a 95% CI (n=500) rather than a point p-value, since the question is whether the full effect is excluded.

Positive control. The identical pipeline applied to ORZ-7B final-checkpoint chains recovers the full signature, ruling out a pipeline failure as the explanation for the OpenThinker null.

Reading the four conditions. Instruct (null), trajectory-SFT at 114k traces (no full signature, late events), 25% RLVR (partial, commit 31%), full RLVR (full, commit 12–14%). Together with R1-Distill-Qwen-7B (trajectory-SFT on ∼800k R1-curated samples, full signature), the pattern supports

the scoped claim of §7.2: RLVR strengthens the signature and moves it earlier within a single run; trajectory imitation can transmit it but does not do so automatically at moderate scale.

S Nemotron-Nano Architecture-Generalisation Check

Llama-3.1-Nemotron-Nano-8B (a Llama-3.1-based reasoning-distilled model, 32 layers) evaluated on 200 GSM8K chains. Argmax layer L24 (relative depth 0.75): Q1→Q4 cosine-distance drop -0.057, p=0.003; commitment detected on 81% of chains at median position 0.16; per-chain entropy slope $+0.85\times10^{-3}$ (p=0.002); turnover Q1→Q4 $+0.16$. The Llama-3.1-8B-Instruct base model shows no significant population-level commitment signature at any layer ($p\geq0.21$), commits on only 9% of chains (median position 0.62), flat entropy slope. Non-Qwen reasoning training replicates the full commit-and-explore signature; its non-RL base does not. The observed GSM8K drop of -0.057 was not used in any fit and serves as the out-of-family check in Table 11.

T Relation to Circuit-Based Reasoning Verification

CRV [Zhao et al., 2026] and our work are complementary on two axes. *Level of analysis*: CRV is step-level and circuit-level, treating the attribution graph of an individual reasoning step as an execution trace and classifying whether that step is computed correctly; our analysis is chain-level and trajectory-level, characterizing when the answer becomes decodable and how the feature population evolves around that event. *Scope*: CRV restricts to standard instruction-tuned models (its experiments use Llama-3.1-8B-Instruct), noting that reasoning models' convoluted paths obscure single-step computation; our analysis targets exactly those terminal-reward reasoning models. Llama-3.1-8B-Instruct is one of the base models that shows no significant population-level commitment signature at any tested layer (§7, Appendix S), so the commit-then-explore dynamic is absent in the regime CRV has studied, and neither paper's claims constrain the other's as published.

The two make direct contact through a testable joint prediction, which requires extending CRV's methodology to a terminal-reward reasoning model (Llama-3.1-Nemotron-Nano-8B is the natural candidate, sharing CRV's base architecture): CRV's step-level correctness signal should be a stronger predictor of final-answer error for pre-commit steps than for post-commit steps. If it holds, CRV-style verification can concentrate its per-step attribution-graph budget on pre-commit steps, addressing the computational intensity CRV's authors identify as a primary limitation; if it fails, our causal-asymmetry claim would need revisiting in that setting. Our shuffle-test result (Appendix D) also illustrates chain-level structure that per-step analysis would not surface: the exploration probe's signal depends on feature identity and/or cross-step feature consistency (e.g., revisit patterns); its cross-step component, if present, is what per-step attribution graphs aggregate away. This motivates using the two methodologies together, with step-level verification gated by chain-level commit position.

U Open Questions and Testable Predictions

Three predictions follow from the results and can be tested with the machinery in this paper. (1) *Sharpening versus on-policy training.* If output-distribution sharpening alone sufficed for the dynamic, base models under power-style sampling [Karan and Du, 2026] should show a partial commit signature with no reasoning training; our label-free detector can test this directly. (2) *Step-level verification gated by commit position.* Extending CRV [Zhao et al., 2026] to a terminal-reward reasoning model (Llama-3.1-Nemotron-Nano-8B shares CRV's base architecture), CRV's step-level correctness signal should predict final-answer error more strongly for pre-commit steps than for post-commit steps; if so, per-step attribution budget can be concentrated pre-commit (Appendix T). (3) *Governing factors of the SFT channel.* R1-Distill shows the signature and OpenThinker does not; trace scale, on- versus off-policy traces, and base model are the candidate factors, and a scale sweep on one base would separate them.

V Monitor Pareto Frontier and Stop-Position Distribution

Sweeping the operating threshold τ from 0.70 to 0.95 traces a Pareto curve above the simpler baselines (Figure 10).

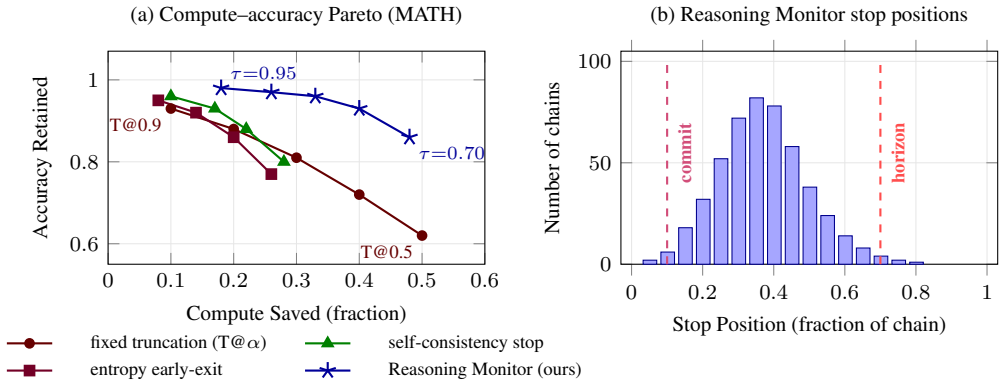

Figure 10: **Reasoning Monitor vs fixed-truncation baselines (MATH).** (a) Compute–accuracy Pareto. (b) Distribution of monitor-chosen stop positions: the monitor concentrates stops between the commitment point ($\sim 10\%$ of chain) and the behavioral horizon ($\sim 70\%$), with mode near 30–40%.

W Monitor Stop-Rule Ablation

The probe-only rule is too conservative (saves little compute at 100% retention); the entropy-slope-only rule is too aggressive (saves more compute but loses 32% of correct chains). The full Monitor, a learned MLP over six features including probe confidence and feature entropy (Appendix X), balances compute savings and retention: unlike the aggressive entropy-slope rule, it meets the 92% retention target while saving substantially more tokens than the conservative probe-only rule.

Table 12: **Stop-rule ablation on MATH-500 (measured, n=497).**

Stop rule	Tokens Saved	Accuracy Retained
Probe confidence > 0.85, 3-step stability only	16%	100%
Entropy-slope-flat only	45%	68%
Full Monitor (learned MLP, six features)	**29%**	**92%**

X Reasoning Monitor Architecture

The Reasoning Monitor is a two-layer MLP: Linear($6 \to 32$) $\to$ ReLU $\to$ Linear($32 \to 1$). Input features are z-score normalized. The six input features are: (1) feature entropy $\mathcal{H}_t$, (2) feature turnover τ_t, (3) top-K stability σ_t, (4) commitment probe confidence, (5) activation L2 norm, and (6) normalized step position t/T. Training uses Adam with lr $= 10^{-3}$ for 100 epochs with early stopping.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction (§1) state five contributions plus the Reasoning Monitor, each tied to an experimental section and each stated at the scope the evidence supports: early internal stabilization (label-free final-state similarity at median 10%, supervised answer decodability at median 18%), defined as detector-relative transitions rather than irreversible decisions (§5); post-commit feature diversification (Cohen's d=0.61–0.68, §4); a quantified causal asymmetry in which post-commit computation is less influential than pre-commit computation but not inert (2–6$\times$ teacher-forced, 1.5–4.6$\times$ under matched activation mass, with overlapping CIs on GPQA, §6); replication of the commit signature on code and general reasoning (signature only, §5.3); and, on a fixed base family, evidence that RLVR strengthens the signature and moves it earlier between an intermediate and final checkpoint while a matched trajectory-SFT control does not reproduce the full effect, with cross-family comparisons described as correlational and a held-out test validating a rank-ordering by reward variance rather than the linear form (§7). The Reasoning Monitor is presented as evidence that the trajectory is exploitable, not as a superior adaptive-compute method; it trails a learned truncation classifier on net FLOPs (§8).

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: §9 contains a dedicated Limitations paragraph with seven items: (1) commit timing is detector-relative (probe 0.85, cosine 0.70 of peak; sensitivity bounds in Appendix M); (2) numeric-task decodability is at answer-bucket resolution; (3) linear probes and open-weight models up to 32B; (4) cross-family comparisons cannot isolate every confound, the within-run RLVR evidence comes from one base family, and the SFT channel remains open; (5) the credit-assignment account takes decodability as a premise, and the held-out tests validate a rank-ordering, not the linear form; (6) the Reasoning Monitor is not superior on net FLOPs (Table 5); (7) domain generalisation covers the commit signature only.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not include formal theoretical results. Appendix A gives a heuristic credit-assignment argument under explicitly stated assumptions (decodability as an empirical premise, bounded score norms, and conditional decoupling), presented as an argument rather than a theorem with proof. Its quantitative prediction is tested empirically on held-out models (§7.3, Appendix P).

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: §3 specifies all models, benchmarks, SAE architecture and hyperparameters, probe formulation (§3.3), and the fixed-rule per-model layer-selection protocol (§3.2). Appendix K provides full SAE training details and formal metric definitions. Appendix C reports the per-model layer sweep backing the layer selection. Appendix D documents probe variants (answer-identity vs. correctness) and the shuffle test. Appendix J details the adaptations of the Monitor baselines (self-consistency stop, entropy early-exit, learned truncation classifier; the regex baseline uses a simple "`the answer is`" pattern match

requiring no adaptation). Appendix X specifies the Monitor architecture and training procedure.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide the complete codebase (SAE training, activation extraction, trajectory analysis, ablation, probe training, Monitor) as supplementary material with a README containing installation and execution instructions. All models used are open-weight. All benchmarks are publicly available.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: §3 describes models, benchmark sizes and splits, SAE architecture and hyperparameters (dictionary size, sparsity k), the probe formulation and thresholds (§3.3: 0.85 confidence, 3-step stability), and the fixed-rule layer-selection protocol; cosine-detector thresholds are in §5.1 (0.7 of early-chain peak, 2-step stability) and SAE layer sets in Appendix K. Appendix K provides the complete SAE training recipe (Adam, learning rate 3×10^{-4}, cosine schedule, warmup steps, total steps, auxiliary loss weight). Appendix J provides hyperparameters for the Monitor baselines (self-consistency sample size k=4 and agreement threshold, entropy exit threshold per benchmark, learned classifier architecture). Appendix X specifies the Monitor architecture and training.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Table 1 reports Q1→Q4 turnover deltas and Appendix Table 8 reports per-benchmark entropy-slope p-values from OLS regressions, across $n\geq198$ chains per condition. Table 3 reports mean ± standard error of the mean (SE) across chains; the main text asserts and Table 3 supports that the resulting 95% CIs for pre- vs post-commit do not overlap on any of the three benchmarks under matched feature count; under matched activation mass they do not overlap on GSM8K and MATH and overlap on GPQA. Appendix figures add further error-bar reporting: Figure 8 uses SE error bars; Figure 6 shades the interquartile range across chains; Figure 7 shades the 10–90 percentile band across n=525 chains. §7 reports p-values from two-sided t-tests on the Q1→Q4 cosine-distance drop. The held-out quantitative test (Appendix P, Table 10) reports 95% CIs on the prediction and verifies that all three observed values land inside. All error statistics represent variability across reasoning chains; factors of variability (dataset sampling) are stated where reported.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Main experiments run on a single contemporary data-center-class GPU with ≥80 GB unified memory (e.g. H100 or GB10). Per-model SAE training at dictionary size $32d$ and sparsity k=64 takes approximately 10–20 GPU-hours for the 7B models and 40–60 GPU-hours for the 32B models. Trajectory extraction across all three benchmarks and all five models requires approximately 30–60 GPU-hours. Ablation and probe training are negligible by comparison. Full project compute including failed or exploratory runs was approximately 4× the reported-experiments total. Individual experiment runtimes and the fixed-rule calibration-set selection cost are reported in the supplementary code documentation.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: This research uses only publicly available open-weight models and standard academic benchmarks. No human subjects are involved. We discuss potential dual-use concerns in §9 (paragraph on implications for CoT monitoring and broader impact).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: §9 (paragraph on implications for CoT monitoring and broader impact) discusses both positive impacts (early-chain decodability probes as a complement to CoT monitoring, and a diagnostic for premature commitment) and negative impacts (in these models, monitors that inspect only late-chain text could miss influences on the answer that act early in the chain).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This paper does not release new pre-trained language models or scraped datasets. The released artifacts are sparse autoencoders (analytical tools) and analysis code, which pose minimal misuse risk. The underlying language models (DeepSeek-R1, Qwen-2.5) are already publicly available under permissive licenses.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All models are cited with their original papers: DeepSeek-R1 and DeepSeek-R1-Distill-Qwen 7B/32B (MIT; the distills derive from Qwen-2.5, originally Apache 2.0), Qwen-2.5-Instruct 7B/32B (Apache 2.0), Qwen-QwQ-32B (Apache 2.0), Llama-3.1-8B-Instruct (Llama 3.1 Community License), Llama-3.1-Nemotron-Nano-8B (NVIDIA Open Model License, with the Llama 3.1 Community License), Open-Reasoner-Zero-7B (MIT), and OpenThinker-7B (Apache 2.0). Benchmarks are cited: GSM8K (MIT), MATH-500 (the 500-problem test split released in OpenAI's PRM800K repository, MIT, drawn from the MATH dataset, MIT), GPQA Diamond (CC BY 4.0; we follow its request not to reveal examples online), HumanEval (MIT), MBPP (CC-BY-4.0), CommonsenseQA (MIT), and LogiQA (no license is stated in the authors' public release; we use it only for research evaluation, as distributed by the authors). Tooling: we implement our own TopK SAEs; SAEBench and Gemma Scope are cited as methodological references, and none of their code or weights is used or redistributed. Self-consistency early-stopping and confident adaptive language modeling baselines are cited (Aggarwal et al. 2023; Schuster et al. 2022).

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We release trained SAE checkpoints (one per model $\times$ layer), the probe weights (answer-identity and correctness variants), the Reasoning Monitor, and the complete analysis codebase. The code includes a README with installation instructions, usage examples, the held-out calibration split used for the fixed-rule layer selection, and documentation of

all configuration options. SAE checkpoints include metadata (model, layer, dictionary size, sparsity, training steps, dead-feature fraction).

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing or research with human subjects. All experiments are conducted on publicly available benchmarks using automated evaluation.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not involve research with human subjects. No IRB approval is required.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs (DeepSeek-R1, Qwen-2.5, Llama-3.1) are the primary objects of study in this paper. Their use is described in §3, including model names and sizes. We analyze the internal representations of these models during chain-of-thought reasoning.