Decoding Self-Knowledge from Latent Reasoning Models for Selective Prediction and Adaptive Routing

Anonymous ACL submission

Abstract

Latent reasoning models compress chain-of-thought into hidden states to reduce inference cost. A natural question is whether these compressed representations retain decodable self-knowledge about reasoning correctness, and if so, how such signals can be exploited. Recent work on recurrent latent architectures (LTO) shows that hidden states support latent best-of-N selection, but whether this generalizes to *compression-based* paradigms remains open. Studying two such paradigms, Latent-SFT and CoLaR, we find a sharp dichotomy in how self-knowledge is encoded. A lightweight attention probe recovers a strong correctness signal that transfers zero-shot across paradigms, compression ratios, and datasets. Yet the signal is **problem-level, not trajectory-level**: it discriminates *which problems* the model will solve, but cannot rank rollouts of the *same* problem, and this limitation is not recoverable through distribution-matched retraining. This dichotomy precisely predicts which applications are viable. Problem-level methods succeed: selective prediction yields substantial accuracy gains, and adaptive routing between latent and explicit CoT outperforms either solver alone while reducing token cost. Trajectory-level selection fails despite the existence of better rollouts, indicating the bottleneck lies in encoding rather than diversity. These results suggest a paradigm-conditional viability boundary: latent best-of-N may depend on architectural stochasticity absent in compression-based systems, while problem-level applications generalize more broadly.[1]

1 Introduction

Latent reasoning models compress explicit chain-of-thought (Wei et al., 2022) into sequences of hidden states, dramatically reducing inference cost while preserving reasoning capability (Hao et al., 2024; Tan et al., 2025; Deng et al., 2025; Goyal et al., 2024). A natural question arises: do these compressed representations retain *self-knowledge*—internal signals about whether the model's reasoning is likely correct? Du et al. (2026) answer affirmatively for Huginn-3.5B (Geiping et al., 2025), a *recurrent* latent model with stochastic initialization, showing that a trained classifier enables best-of-N selection (+5.9pp on GSM8K). However, Huginn's architecture is fundamentally different from *compression-based* paradigms that deterministically compress explicit reasoning into shorter latent sequences (Deng et al., 2024). Whether such paradigms also encode exploitable self-knowledge, and whether the same algorithmic approach applies, remain open questions.

We study two compression-based paradigms: Latent-SFT (Deng et al., 2025) (vocab-space soft tokens, fixed compression ratio) and CoLaR (Tan et al., 2025) (continuous hidden states, dynamic termination). On both, the signal exists, but is mechanistically different from prior work. A lightweight attention probe ($\sim$1M parameters) extracts a strong correctness signal from latent trajectories. It achieves AUC 0.805 on Latent-SFT and 0.853 on CoLaR, with three-axis zero-shot transfer across paradigms, tasks (GSM8k$\rightarrow$SVAMP), and compression ratios. Crucially, this signal is *problem-level*: the probe discriminates which problems the model will solve correctly (pooled AUC$>$0.93) but cannot rank rollouts of the same problem (within-problem AUC 0.51–0.68). Distribution-matched retraining does not fix this.

This problem-level nature precisely predicts which downstream applications succeed and which fail (Figure 3). Problem-level decisions work: selective prediction at 50% coverage yields +20–25pp; adaptive routing achieves 52.5$\pm$1.5% accuracy at 43% token saving, exceeding both pure solvers. Trajectory-level decisions fail: reproducing LTO's best-of-N algorithm (Du et al., 2026) on our paradigms yields $\leq$+0.3pp, while oracle

[1] Code is included in the Supplementary Material.

ceilings show that +3.8–10.2pp gains are achievable. Together with LTO's positive result on Huginn, this delineates a *paradigm-conditional viability boundary*—the first evidence that algorithm–paradigm pairing, not signal strength alone, determines whether latent self-knowledge is practically exploitable.

2 Related Work

Latent reasoning and test-time compute. Chain-of-thought prompting (Wei et al., 2022) and self-consistency (Wang et al., 2023) established that explicit intermediate reasoning improves LLM performance, motivating latent alternatives that preserve quality at reduced token cost. Several paradigms compress explicit CoT into latent representations: Coconut (Hao et al., 2024) uses hidden states as latent tokens; CoLaR (Tan et al., 2025) introduces dynamic compression with KL-aligned distribution matching; Latent-SFT (Deng et al., 2025) compresses into vocab-space superposition tokens at a fixed ratio; pause tokens (Goyal et al., 2024) provide additional latent computation before generation. Huginn-3.5B (Geiping et al., 2025) takes a different approach, employing recurrent depth with stochastic initialization, building on adaptive computation (Graves, 2016; Dehghani et al., 2019). Concurrent analyses include causal studies of continuous thought (Zhang et al., 2025b) and depth-accuracy failure modes (Sahoo et al., 2026). Prior work focuses on the accuracy–efficiency tradeoff; we focus on what these models internally "know" about their own reasoning quality.

Confidence estimation and hidden-state probing. Selective prediction (El-Yaniv and Wiener, 2010; Geifman and El-Yaniv, 2017) requires reliable confidence scores, yet last-token probabilities are poorly calibrated for reasoning tasks (Kadavath et al., 2022; Guo et al., 2017). Hidden-state probing (Conneau et al., 2018; Belinkov, 2022) has been applied to factual knowledge (Burns et al., 2023), truthfulness (Azaria and Mitchell, 2023), and representation structure (Zou et al., 2023). Recent work probes *explicit* reasoning models: Zhang et al. (2025a) show that DeepSeek-R1 hidden states encode correctness signals for early-exit; LYNX (Akgul et al., 2025) uses conformal prediction for early-exit on explicit CoT; ReProbe (Ni et al., 2025) trains a probe for best-of-N selection on explicit traces. Our work targets *compression-based latent reasoning*, which is architecturally distinct in that explicit cue tokens are absent and standard token-level confidence is either unreliable, as in Latent-SFT, or undefined, as in CoLaR.

Routing, adaptive computation, and LTO. Adaptive computation (Graves, 2016; Schuster et al., 2022; Raposo et al., 2024) dynamically allocates computation at the *token level* within a single forward pass. Model routing (Chen et al., 2023; Ong et al., 2024) directs inputs to different models based on predicted difficulty. Our setting differs: we route between *reasoning strategies* (latent vs. explicit CoT) of the same base model at the *problem level*, exploiting solver complementarity. Most closely related, LTO (Du et al., 2026) trains a classifier on Huginn-3.5B's latent thoughts for best-of-N trajectory selection (+5.9pp on GSM8K), relying on inter-rollout diversity from stochastic initialization. We reproduce LTO's algorithm on compression-based paradigms, demonstrate it fails ($\leq$+0.3pp; §7), and propose routing and selective prediction as alternatives matched to the problem-level signal structure. Concurrently, inference-time interventions on discrete-CoT models (Soft Thinking (Zhang et al., 2025c), SwiReasoning (Shi et al., 2026), LaTER (Li et al., 2026)) operate intra-trajectory via soft embeddings or entropy-based mode switching, complementary to our problem-level analysis of compression-trained latent paradigms.

3 Experimental Setup

Latent reasoning paradigms. We study two compression-based paradigms built on Llama-3.2-1B-Instruct (Grattafiori et al., 2024): **Latent-SFT** (Deng et al., 2025) compresses explicit CoT into vocab-space soft tokens at a fixed compression ratio r. We use $r{=}2$ (mean $T{\approx}13$) and $r{=}4$ (mean $T{\approx}8$). **CoLaR** (Tan et al., 2025) produces continuous hidden states with dynamic termination governed by compression factor $c{=}5$ (mean $T{\approx}5.8$). These two paradigms cover the dominant representational classes for compression-based latent reasoning.

Datasets. Our primary evaluation uses **GSM8k-Aug**, derived from GSM8K (Cobbe et al., 2021), and **SVAMP** (Patel et al., 2021) as the cross-task transfer testbed. All reported AUC values measure binary classification of correct vs. incorrect model outputs.

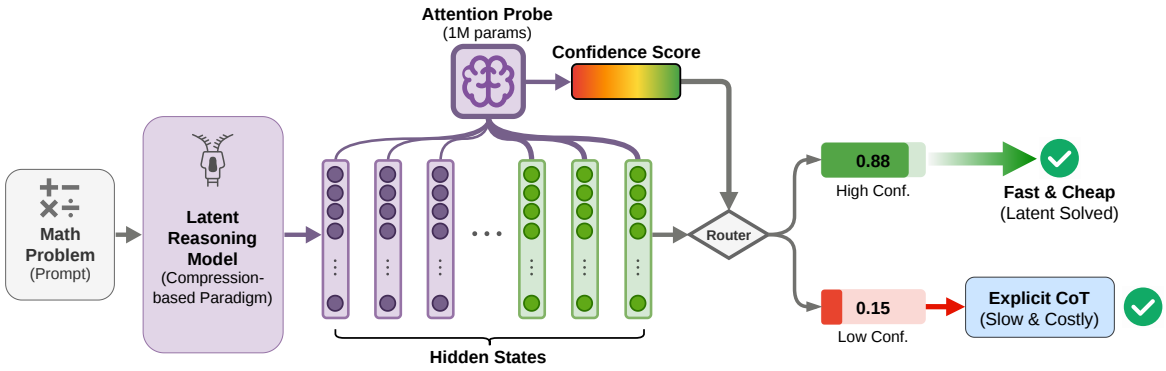

Figure 1: Overview of our approach. A lightweight attention probe (∼1M params) extracts a confidence score from the hidden-state trajectory of a compression-based latent reasoning model. A router uses this score to select between fast latent inference when confidence is high and explicit Chain-of-Thought when confidence is low, achieving accuracy gains while reducing token cost.

Probe architecture. We train an **AttentionConfidenceHead**: a lightweight (∼1M parameter) attention-pooling module over the latent trajectory followed by an MLP classifier, following the probing methodology established in prior work (Conneau et al., 2018; Belinkov, 2022). The probe takes as input the last-layer hidden states $\{\mathbf{h}_1, \ldots, \mathbf{h}_T\}$ and outputs a scalar confidence score. Training uses binary cross-entropy on correctness labels with nested cross-validation (§5). All CV splits are at the *problem level*: no problem appears in both train and test folds, ensuring no leakage of problem identity.

Key metrics. **Pooled AUC**: binary classification across all problems (cross-problem discrimination). **Within-problem AUC**: AUC computed per-problem over rollouts, then averaged (trajectory-level discrimination). **Oracle utilization**: LTO gain / oracle-best-of-N gain $\times$ 100%.

Paradigm	Representation	T (mean)	Acc.
Latent-SFT(2)	Vocab softmax	12.9	50.0%
Latent-SFT(4)	Vocab softmax	7.6	45.0%
CoLaR (c=5)	Continuous $\mathbb{R}^{2048}$	5.8	28.1%

Table 1: Latent reasoning paradigms studied. All share the Llama-3.2-1B-Instruct backbone. Representation indicates the form of latent states: vocab-space soft tokens for Latent-SFT, continuous hidden vectors for CoLaR. T is the mean trajectory length in latent steps.

Unless otherwise noted, all reported probe AUCs use 5-fold nested cross-validation at the problem level on GSM8k-Aug (N=1319), with mean±std over independent CV seeds.

4 The Signal Exists But Not Where Naïve Methods Look

We motivate the trained probe by first showing that hand-crafted scalar signals, while informative, are not robust across paradigms (§4.1), and then demonstrating that a single closed-form normalization partially resolves this instability but leaves a substantial gap that only learned representations can close (§4.2).

4.1 Hand-Crafted Signals Exist But Are Not Robust

Length direction is paradigm-dependent; raw entropy is length-confounded. Trajectory length T carries strong correctness signal, but its *direction* reverses across paradigms (Figure 2). CoLaR's dynamic termination yields short T for correct solutions, with AUC 0.750. Latent-SFT's fixed compression shows the opposite direction, with AUC 0.331–0.346. A length-based classifier trained on one paradigm will *anti-predict* on the other. Raw spectral entropy, a natural paradigm-agnostic candidate, correlates *inversely* with correctness (AUC 0.356–0.409) due to a Simpson's paradox: incorrect samples cluster at high T, mechanically inflating entropy. Length is the primary confound (ρ=−0.386) that any robust scalar must account for.

4.2 LENS: A Closed-Form Cross-Paradigm Baseline

The diagnosis above identifies two problems: length direction reverses across paradigms, and raw entropy is confounded by length. We resolve

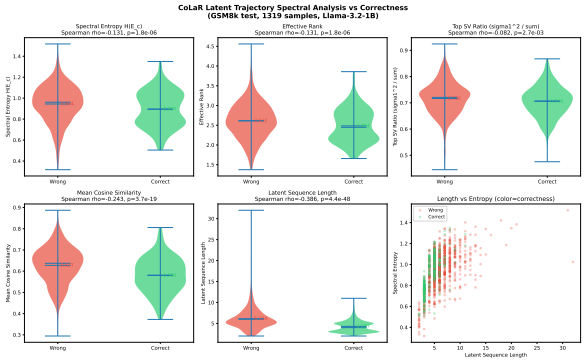

Figure 2: Spectral feature distributions by correctness on CoLaR (GSM8k test, N=1319, Llama-3.2-1B). Bottom-right: Length vs. Entropy scatter colored by correctness.

both with a single normalization, defining LENS (**L**ength-normalized **EN**tropy of **S**ingular values):

$$\mathrm{LENS}(x) = \frac{H(\text{singular values of } [\alpha_1; \ldots; \alpha_T])}{\log T}, \tag{1}$$

where $[\alpha_1; \ldots; \alpha_T] \in \mathbb{R}^{T \times d}$ is the matrix of latent hidden states along the trajectory for input x, $H(\cdot)$ denotes the Shannon entropy of the normalized singular value distribution (i.e., treating $\sigma_i / \sum_j \sigma_j$ as a probability), and T is the trajectory length.

Dividing by $\log T$ accomplishes two things simultaneously: (a) it removes the length confound from entropy, and (b) it absorbs the paradigm-dependent length direction. After normalization, higher per-step representational diversity consistently indicates correctness regardless of whether the paradigm produces short-correct or long-correct trajectories. The correlation reverses to positive across all three settings (Table 6, Appendix B).

LENS achieves AUC $\geq$0.70 across all three paradigm configurations (0.714–0.754), representing the ceiling of what a closed-form scalar statistic can extract. Yet the partial correlation $\rho(\mathrm{LENS}|T)$ remains modest (0.125–0.307), indicating that LENS captures only part of the available signal. The remaining gap to the trained probe motivates learning directly from the hidden-state sequence (§5), where attention over the full trajectory can exploit richer structure inaccessible to any single scalar.

5 Trained Probe: Method, Performance, and Transfer

We establish three properties of the trained probe: (i) it substantially exceeds closed-form and feature-engineering baselines on both paradigms (§5.1); (ii) it generalizes zero-shot across paradigms, compression ratios, and tasks (§5.2); and (iii) the signal it captures is not answer-encoding, exceeds prompt-derived difficulty, and concentrates near the trajectory endpoint (§5.2).

5.1 Architecture and Main Results

Architecture. The AttentionConfidenceHead applies query-key attention over the T trajectory steps to produce a fixed-dimensional summary, followed by a two-layer MLP outputting a scalar confidence score. Attention pooling handles variable-length trajectories naturally. Training minimizes binary cross-entropy on correctness labels over 1,319 labeled trajectories from GSM8k-Aug (5-fold nested CV at problem level). Hyperparameters and training details are in Appendix A.

Setting	LENS	Probe	Δ
LSFT GSM8k	0.713	**0.805±0.011**	+0.092
CoLaR GSM8k	0.650	**0.853±0.004**	+0.203

Table 2: Main probe results on GSM8k-Aug. We compare our trained AttentionConfidenceHead against LENS, a closed-form baseline given by spectral entropy normalized by trajectory length. Seeds: 6 runs for Latent-SFT, 4 for CoLaR.

Main results. Table 2 presents the main result. The trained probe achieves 0.805±0.011 on Latent-SFT and 0.853±0.004 on CoLaR, substantially exceeding the LENS baseline. The asymmetric improvement (+0.092 vs. +0.203) reflects representational differences: Latent-SFT's vocab-space simplex is already well-captured by hand-crafted entropy features, while CoLaR's continuous hidden states contain richer learnable structure inaccessible to scalar statistics. Standard token-level confidence (last-token argmax, verbalized elicitation) and feature-engineering baselines (multi-feature LR) all substantially underperform; full comparison in Appendix C.

Why attention? Sequence-aware processing is critical. A pointwise MLP, which processes each step independently, barely exceeds LENS (+0.007 on LSFT, +0.096 on CoLaR). The attention mechanism, which can attend across trajectory steps, provides the dominant improvement (Table 7, Appendix B), consistent with the signal being a trajectory-level property rather than a per-step statistic.

5.2 Generalization and Signal Characterization

The probe generalizes robustly and encodes meaningful structure beyond simple baselines.

Cross-task transfer. The probe trained on GSM8k transfers zero-shot to SVAMP, achieving AUC 0.810 (Latent-SFT) and 0.785 (CoLaR), substantially exceeding LENS (0.478 and 0.646). AUC$\geq$0.78 across all four paradigm$\times$dataset combinations. Both datasets are arithmetic reasoning; broader domain generalization remains to be evaluated.

Cross-paradigm and cross-compression-ratio. The same architecture applies to both paradigms without modification. The probe trained on Latent-SFT(r=4) is applied zero-shot to Latent-SFT(r=2) for routing (§6.2) with no retraining, validating cross-r generalization.

Not answer encoding. A natural concern is that the probe implicitly decodes the answer (Belinkov, 2022). An answer-prediction probe on the same last-step hidden state reaches only 25.8%/10.2% token-level top-1 accuracy (LSFT/CoLaR), with numeric exact match at **0.0% on both paradigms**. The model "knows when it is right" without "knowing what is right" in a directly readable form.

Beyond prompt-derived difficulty. A prompt-only MLP probe (last hidden state before latent generation) achieves AUC 0.769/0.728, showing that latent-trajectory states contribute Δ+0.036/+0.125 above prompt-derived difficulty (paired bootstrap 95% CI excludes zero). The routing-gain gap (§6.2) is more pronounced: +0.5pp vs. +2.0pp.

Endpoint dominance. The last single step alone recovers 93–98% of full-trajectory AUC on both paradigms (Table 10, Appendix F), foreshadowing the problem-level nature analyzed in §7: the final representation summarizes the *entire reasoning outcome* rather than encoding step-by-step trajectory dynamics.

6 Problem-Level Applications

The problem-level nature of the signal precisely predicts which applications succeed. We evaluate two: selective prediction (abstaining on hard problems) and adaptive routing (choosing between reasoning strategies).

6.1 Selective Prediction

Selective prediction, i.e., abstaining on low-confidence problems (El-Yaniv and Wiener, 2010; Geifman and El-Yaniv, 2017, 2019; Madhusudhan et al., 2024), is the most direct application of a problem-level signal. We evaluate at 50% coverage (the model answers only the top-50% most confident problems).

Setting	All	@50%	Δ
LSFT GSM8k	45.0%	**69.7%**	+24.7pp
LSFT SVAMP	55.6%	79.2%	+23.6pp
CoLaR GSM8k	28.1%	49.3%	+21.2pp
CoLaR SVAMP	47.2%	67.6%	+20.4pp

Table 3: Selective prediction at 50% coverage. `All`: accuracy on the full test set without abstention. `@50%`: accuracy on the top-50% most confident problems ranked by the probe.

Table 3 shows +20–25pp improvement across all four settings with striking consistency. The effect size is robust: from +20.4pp (CoLaR SVAMP, the hardest setting with cross-task transfer) to +24.7pp (Latent-SFT GSM8k, in-distribution). This consistency directly reflects the problem-level nature of the signal—the probe reliably identifies which problems the model will solve, regardless of paradigm or dataset.

Coverage–accuracy tradeoff. The selective signal extends smoothly across operating points. At 30%/50%/70% coverage, Latent-SFT GSM8k accuracy is 80.1%/69.7%/59.3%, tracing a monotonic risk-coverage curve. Even at 70% coverage—where the probe abstains on only the bottom 30%—accuracy already gains +14.3pp over the no-abstention baseline (45.0%). This suggests the probe identifies a long tail of confidently-wrong problems rather than only the most extreme failures, making the abstention budget tunable to deployment cost.

6.2 Adaptive Inference Routing

Setup. Adaptive routing between reasoning strategies has been explored for cross-model settings (Chen et al., 2023; Ong et al., 2024; Dekoninck et al., 2024); we apply it to intra-model strategy selection. For each problem, we run latent reasoning with Latent-SFT(2), obtain a confidence score from the probe, and route: if confidence exceeds threshold τ, output the latent answer; otherwise fall back to CoT-SFT. Total amor-

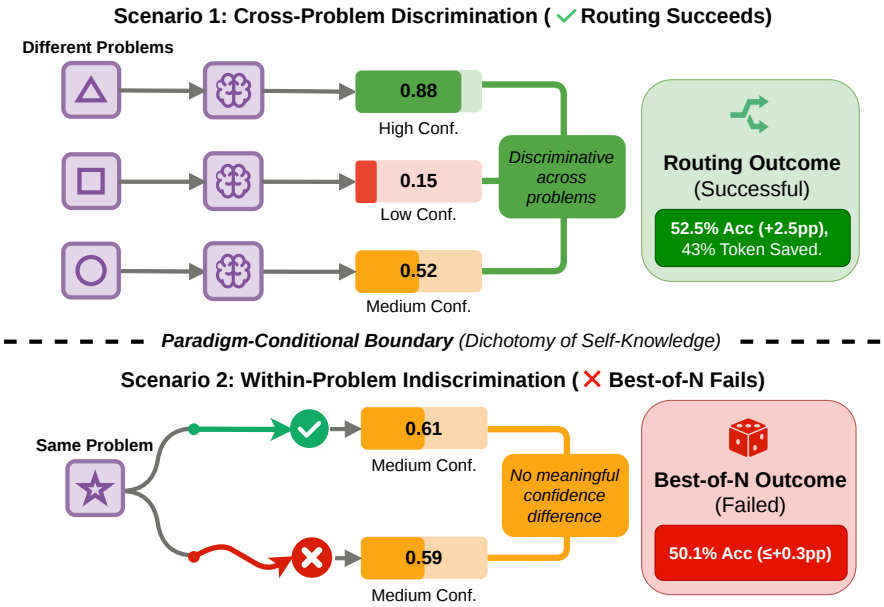

Figure 3: The dichotomy of self-knowledge. **Scenario 1**: the probe assigns well-separated scores to different problems, enabling routing. **Scenario 2**: rollouts of the same problem receive nearly identical scores regardless of outcome, causing best-of-N to fail.

tized cost per problem includes the latent forward pass (mean 12.9 tokens) plus any CoT fallback: cost $= T_{\text{latent}} + \mathbb{1}[\text{conf} \leq \tau] \cdot T_{\text{CoT}}$. The probe was trained on Latent-SFT(4) and is applied zero-shot to Latent-SFT(2)—a simultaneous cross-compression-ratio transfer. Threshold τ is selected by 5-fold cross-validation with argmax-accuracy criterion (alternative criteria yield consistent results; Appendix F).

τ	Latent%	Acc.	Tokens	Saving
0.00 (pure latent)	100%	50.0%	12.9	84%
0.05	49%	52.8%	56.5	28%
0.10	37%	52.1%	65.4	17%
0.20	28%	51.1%	72.9	8%
1.00 (pure CoT)	0%	50.5%	78.8	0%
5-fold CV	**59%**	**52.5±1.5%**	**45.2**	**43%**

Table 4: Routing Pareto sweep on GSM8k-Aug. The router outputs the latent answer when confidence exceeds τ and falls back to explicit CoT otherwise. Latent% is the fraction of problems handled by latent reasoning. Tokens reports the amortized cost per problem, summing the latent forward pass and any CoT fallback. Saving is the reduction relative to pure CoT. The bottom row is the deployable operating point: τ selected by 5-fold CV, with mean±std across folds.

Main results. Table 4 and Figure 4 present the routing Pareto frontier. The CV-deployable operating point (52.5±1.5% accuracy, 43% token saving) exceeds both pure baselines. Oracle τ-selection on the full test set gives 52.8%, only 0.3pp above CV, confirming that threshold selection is not overfit. Statistical significance: routing produces correct answers on 107 problems where pure CoT fails, while CoT wins on 71 problems (McNemar χ^2=7.28, p=0.007; paired bootstrap 95% CI: [+0.7pp, +4.7pp]). Among all baselines tested (Table 8, Appendix C), only the full-trajectory probe significantly exceeds pure CoT. The strongest non-trajectory baseline (multi-feature LR: +1.4pp) narrows but does not close the gap to our probe (+2.0pp, p=0.007).

Mechanistic explanation: solver complementarity. Bin analysis reveals why routing exceeds both baselines: on high- and mid-confidence problems (upper 2/3, N=878), latent reasoning outperforms CoT by +3.9pp (72.0% vs 68.1%). On low-confidence problems (bottom 1/3), CoT wins by +9.1pp (31.5% vs 22.4%). The probe correctly identifies each solver's specialization region—a problem-level decision the signal directly supports. Full bin table in Appendix F.

Cross-task and cross-paradigm validation. Under double transfer (cross-task + cross-r), SVAMP routing achieves 64.0% (+0.1pp over CoT's 63.9%) while preserving Pareto dominance over random routing. The modest gain reflects CoT's large ad-

vantage on SVAMP; the meaningful result is preserved dominance under transfer. CoLaR routing (28.1% vs. CoT 50.5%) cannot exceed CoT due to the accuracy gap, but Pareto dominance over random is preserved—confirming paradigm-invariant ranking validity.

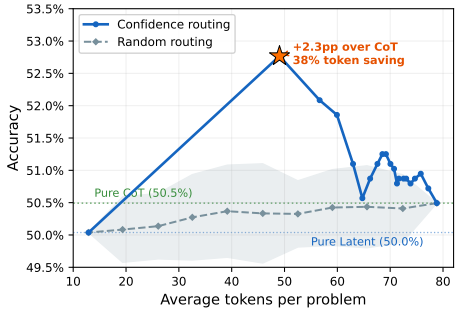

Figure 4: Routing Pareto frontier on Latent-SFT, GSM8k-Aug, N=1319, cross-r probe. Confidence routing achieves +2.3pp over pure CoT at 38% token saving. The solid curve is confidence-based routing at varying τ. The dashed curve is random routing where each problem is sent to latent with probability f swept over $f \in \{0, 0.1, \ldots, 1.0\}$; the shaded band shows ± 1 std over 20 seeds. Confidence routing Pareto-dominates random across all operating points.

7 Latent Best-of-N Is Paradigm-Conditional

We now directly test whether the self-knowledge signal supports trajectory-level selection. This is the setting Du et al. (2026) demonstrated on Huginn-3.5B, analogous to best-of-N approaches in explicit reasoning (Wang et al., 2023; Cobbe et al., 2021; Lightman et al., 2023).

7.1 LTO Best-of-N Reproduction

We reproduce LTO's KL-regularized rejection sampling on our paradigms, using our trained probe as the Latent Reward Model. For each problem, we generate N rollouts and select using the LTO criterion with inverse-temperature β scanned over [0.001, 5.0] (200 repeated trials per setting).

Table 5 presents the central finding. LTO Algorithm 1 achieves $\leq$+0.3pp on *all* compression-based configurations, despite substantial oracle ceilings (+3.8–10.2pp indicating that better rollouts exist). On deterministic Latent-SFT(2), all N=20 rollouts are identical (no rollout diversity to exploit), so LTO returns the base accuracy unchanged. Even with injected stochasticity (ε=0.001 noise), the probe cannot select among rollouts—oracle utilization remains at 1.6–8.4%.

Method	Paradigm	Acc.	Δ	Oracle	Util.
Base	LSFT(2)	50.0	—	—	—
LTO	LSFT(2), det.	50.0	+0.0	0.0	—
LTO	LSFT(2), ε	50.1	+0.2	+10.2	1.6%
LTO	LSFT(4), ε	45.1	+0.3[1]	+3.8	8.4%
LTO	CoLaR, σ	30.2	−0.5	+5.3	—
LTO[2]	Huginn-3.5B	38.5	**+5.9**	—	—
Routing	**LSFT(2)**	**52.5**	**+2.5**	—	—

Table 5: LTO Best-of-N reproduction on GSM8k-Aug. For each problem we generate N=20 rollouts and select one using LTO; Acc. is the resulting accuracy. Δ is the gain over the single-rollout base. Oracle gives the gain a perfect selector would achieve, i.e., oracle-best-of-N minus the base. Util. is Δ/Oracle, the fraction of available headroom that LTO captures. [1]Δ relative to stochastic base (44.8%); deterministic LSFT(4) base is 45.0%. [2]Numbers from Du et al. (2026); their oracle ceiling not reported.

Candidate diversity does exist (8–14% of problems produce mixed outcomes, oracle ceilings +3.8–10.2pp); the failure is specifically within-problem ranking, not absence of exploitable diversity.

7.2 Intervention Test: The Signal Cannot Be Retrained

We test whether retraining on stochastic rollouts enables within-problem discrimination. Across five magnitudes of latent-space perturbation on CoLaR ($\sigma \in \{1, \ldots, 10000\}$), oracle ceiling rises from +4.0pp to +11.5pp (diversity is created), but within-problem AUC remains at 0.49–0.56. Distribution-matched retraining (σ=1000) achieves pooled AUC 0.975 (near-perfect cross-problem ranking) but within-problem AUC = **0.510**—indistinguishable from random (Figure 5, Appendix F). Training cannot create rollout-level signal that the architecture does not encode; problem-level encoding is an architectural property, not a training artifact (Zou et al., 2023).

7.3 Cross-Configuration Universality

Within-problem AUC remains in [0.51, 0.68] across all three configurations spanning two paradigms, three compression ratios $r \in \{2, 4, \sim 5\}$, and three trajectory lengths T=5.8–13.9. This range is insufficient for best-of-N in every case (Table 9, Appendix D).

7.4 The 2×2 Paradigm-Algorithm Boundary

The central diagnostic: the confidence probe decodes problem-level difficulty (pooled AUC >0.93) but does not differentiate rollouts of the same problem (within-problem AUC 0.51–0.68). This yields a clear viability pattern. LTO's best-of-N succeeds on recurrent Huginn (+5.9pp; Du et al., 2026) but fails on compression paradigms (≤+0.3pp). Conversely, our routing (+2.5pp) and selective prediction (+20pp) succeed precisely because they operate at the problem level. This comparison confounds architecture, scale, and stochasticity; the boundary is an empirical observation, not a proven architectural law.

8 Discussion

Our findings suggest a design principle: systems targeting best-of-N (Snell et al., 2024; Tang et al., 2025) must preserve rollout stochasticity, while systems targeting problem-level applications work with deterministic generation and pooled classifiers. Verifier-based test-time scaling favors recurrent/stochastic architectures; cost-sensitive inference with abstention favors compression paradigms.

We propose a mechanistic interpretation of why compression erases trajectory-level signal. Compression training objectives (KL alignment, vocab-space distillation) require the trajectory to match the *expected output* of explicit CoT, integrating over trajectory-internal variation. The endpoint dominance (§5.2, last-1 step recovers 93–98% AUC) is the fingerprint: the final representation encodes *whether the problem is solved*, not which path was taken. Huginn's stochastic initialization preserves rollout-specific variation by design. The distinction is not deterministic vs. stochastic, but *what variation the training objective retains*.

These findings yield a diagnostic protocol for new latent paradigms. Two cheap held-out measurements—pooled AUC and within-problem AUC over a handful of stochastic rollouts—predict application viability without a full pipeline: high pooled + low within-problem admits selective prediction and routing but rules out best-of-N; high on both additionally admits trajectory selection; low pooled rules out all three. This reframes self-knowledge as a structured property to be measured rather than a scalar to be optimized.

Beyond this dichotomy, our routing analysis reveals that latent reasoning is not simply a cheaper CoT. Stratifying by confidence (Table 11, Appendix F), latent reasoning outperforms CoT by +3.9pp on the upper two-thirds but underperforms by 9.1pp on the lowest third. Quality is not uniformly preserved but *redistributed* along the difficulty spectrum (Hao et al., 2024; Deng et al., 2025; Tan et al., 2025). This complementarity enables routing to exceed both pure solvers, and suggests the probe encodes *solver-fit* (which reasoning mode suits a given problem) rather than raw difficulty. Concurrently, SwiReasoning (Shi et al., 2026) finds that purely latent decoding on discrete-CoT models diffuses probability mass and requires switching back to explicit reasoning to consolidate, mirroring our solver-complementarity finding.

9 Conclusion

Compression-based latent reasoning retains decodable self-knowledge that is *problem-level, not trajectory-level*: a trained probe achieves strong cross-paradigm AUC while within-problem discrimination remains near chance, and this limitation persists under distribution-matched retraining. This dichotomy predicts application viability: selective prediction (+20–25pp) and adaptive routing (+2.5pp, 43% token saving) succeed where latent best-of-N fails (≤+0.3pp). The resulting paradigm-conditional boundary suggests that algorithm–architecture pairing, not signal strength alone, determines whether latent self-knowledge is practically exploitable, and that problem-level methods may generalize more broadly across compression-based systems.

10 Limitations

We evaluate two compression-based paradigms on arithmetic reasoning; broader architectures (e.g., Coconut, CCOT) and domains remain untested. Huginn's within-problem AUC is inferred from LTO's reported viability rather than directly measured. The 2×2 paradigm–algorithm boundary confounds architecture, scale, and stochasticity with paradigm class. Part of the probe signal reflects prompt-derived problem difficulty (§5.2), though latent processing contributes meaningfully. Routing gains depend on solver complementarity and vanish when one solver dominates. Our approach operates per-problem; within-problem dynamic switching (Li et al., 2026) is complementary and not addressed here.

References

Omer Faruk Akgul, Yusuf Hakan Kalayci, Rajgopal Kannan, Willie Neiswanger, and Viktor Prasanna. 2025. LYNX: Learning dynamic exits for confidence-controlled reasoning. *arXiv preprint arXiv:2512.05325*.

Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it's lying. *arXiv preprint arXiv:2304.13734*.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2023. Discovering latent knowledge in language models without supervision. In *International Conference on Learning Representations (ICLR)*.

Lingjiao Chen, Matei Zaharia, and James Zou. 2023. FrugalGPT: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Alexis Conneau, German Kruszewski, Guillaume Lample, and 1 others. 2018. What you can cram into a single $\$vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, and 1 others. 2019. Universal transformers. In *International Conference on Learning Representations (ICLR)*.

Jasper Dekoninck and 1 others. 2024. A unified approach to routing and cascading for LLMs. *arXiv preprint arXiv:2410.10347*.

Jingcheng Deng, Liang Pang, Zihao Wei, Shicheng Xu, Zenghao Duan, Kun Xu, Yang Song, Huawei Shen, and Xueqi Cheng. 2025. LLM latent reasoning as chain of superposition. *arXiv preprint arXiv:2510.15522*.

Yuntian Deng, Yejin Choi, and Stuart Shieber. 2024. From explicit CoT to implicit CoT: Learning to internalize CoT step by step. *arXiv preprint arXiv:2405.14838*.

Shrey Desai and Greg Durrett. 2020. Calibration of pre-trained transformers. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Hanwen Du, Yuxin Dong, and Xia Ning. 2026. Latent thinking optimization: Your latent reasoning language model secretly encodes reward signals in its latent thoughts. In *International Conference on Learning Representations (ICLR)*.

Ran El-Yaniv and Yair Wiener. 2010. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11:1605–1641.

Yonatan Geifman and Ran El-Yaniv. 2017. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Yonatan Geifman and Ran El-Yaniv. 2019. SelectiveNet: A deep neural network with an integrated reject option. In *International Conference on Machine Learning (ICML)*.

Jonas Geiping and 1 others. 2025. Scaling up test-time compute with latent reasoning: A recurrent depth approach. *arXiv preprint arXiv:2502.05171*.

Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, and 1 others. 2024. Think before you speak: Training language models with pause tokens. *International Conference on Learning Representations (ICLR)*.

Aaron Grattafiori and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Alex Graves. 2016. Adaptive computation time for recurrent neural networks. *arXiv preprint arXiv:1603.08983*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *International Conference on Machine Learning (ICML)*.

Shibo Hao and 1 others. 2024. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*.

Saurav Kadavath and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Xuan Li, Yining Wang, Yuchen Liu, Guanjun Liu, Delai Qiu, Shengping Liu, Jiaen Liang, Wei Huang, Jun Yu, and Junnan Zhu. 2026. LaTER: Efficient test-time reasoning via latent exploration and explicit verification. *arXiv preprint arXiv:2605.07315*.

Hunter Lightman, Vineet Kosaraju, Yura Burda, and 1 others. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Nishanth Madhusudhan and 1 others. 2024. Do LLMs know when to not answer? investigating abstention abilities of large language models. *arXiv preprint arXiv:2407.16221*.

Jingwei Ni, Ekaterina Fadeeva, Tianyi Wu, Mubashara Akhtar, Jiaheng Zhang, Elliott Ash, Markus Leippold, Timothy Baldwin, See-Kiong Ng, Artem Shelmanov, and Mrinmaya Sachan. 2025. ReProbe: Efficient test-time scaling of multi-step reasoning by probing internal states of large language models. *arXiv preprint arXiv:2511.06209*.

Isaac Ong, Amjad Almahairi, Vincent Wu, and 1 others. 2024. RouteLLM: Learning to route LLMs with preference data. *arXiv preprint arXiv:2406.18665*.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are NLP Models Really Able to Solve Simple Math Word Problems? In *Proceedings of NAACL*.

David Raposo, Sam Ritter, Blake Richards, and 1 others. 2024. Mixture-of-depths: Dynamically allocating compute in transformer-based language models. *arXiv preprint arXiv:2404.02258*.

Subramanyam Sahoo, Aman Chadha, Vinija Jain, and Divya Chaudhary. 2026. When shallow wins: Silent failures and the depth-accuracy paradox in latent reasoning. *arXiv preprint arXiv:2603.03475*.

Tal Schuster, Adam Fisch, Jai Gupta, and 1 others. 2022. Confident adaptive language modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Dachuan Shi, Abedelkadir Asi, Keying Li, Xiangchi Yuan, Leyan Pan, Wenke Lee, and Wen Xiao. 2026. SwiReasoning: Switch-thinking in latent and explicit for pareto-superior reasoning LLMs. *arXiv preprint arXiv:2510.05069*.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*.

Wenhui Tan, Jiaze Li, Jianzhong Ju, Zhenbo Luo, Ruihua Song, and Jian Luan. 2025. Think silently, think fast: Dynamic latent compression of LLM reasoning chains. *arXiv preprint arXiv:2505.16552*.

Yung-Chen Tang, Pin-Yu Chen, and Andrea Cavallaro. 2025. CarBoN: Calibrated best-of-N sampling improves test-time reasoning. *arXiv preprint arXiv:2510.15674*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, and 1 others. 2023. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, and 1 others. 2024. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. *International Conference on Learning Representations (ICLR)*.

Anqi Zhang, Yulin Chen, Jane Pan, Chen Zhao, Aurojit Panda, Jinyang Li, and He He. 2025a. Reasoning models know when they're right: Probing hidden states for self-verification. *arXiv preprint arXiv:2504.05419*.

Yuyi Zhang, Boyu Tang, Tianjie Ju, Sufeng Duan, and Gongshen Liu. 2025b. Do latent tokens think? a causal and adversarial analysis of chain-of-continuous-thought. *arXiv preprint arXiv:2512.21711*.

Zhen Zhang, Xuehai He, Weixiang Yan, Ao Shen, Chenyang Zhao, Shuohang Wang, Yelong Shen, and Xin Eric Wang. 2025c. Soft thinking: Unlocking the reasoning potential of LLMs in continuous concept space. *arXiv preprint arXiv:2505.15778*.

Andy Zou, Long Phan, Sarah Chen, and 1 others. 2023. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*.

A Experimental Protocol

Probe architecture details. The AttentionConfidenceHead consists of: (1) a learned query vector $\mathbf{q} \in \mathbb{R}^{256}$ and key projection $W_K \in \mathbb{R}^{2048\times256}$; (2) attention weights $\alpha_t = \mathrm{softmax}(\mathbf{q}^\top W_K \mathbf{h}_t/\sqrt{256})$; (3) pooled representation $\mathbf{z} = \sum_t \alpha_t \mathbf{h}_t$; (4) MLP classifier: Linear(2048, 256) $\rightarrow$ ReLU $\rightarrow$ Dropout(0.1) $\rightarrow$ Linear(256, 1) $\rightarrow$ Sigmoid. Total parameters: $\sim$1.1M.

Training. Binary cross-entropy loss. Adam optimizer (lr=1e-4, weight decay 1e-5). Batch size 64. Early stopping on validation AUC (patience 10). Training data: 1,319 trajectories from GSM8k-Aug with binary correctness labels (5-fold CV at problem level).

Nested cross-validation. Outer loop: 5 folds (stratified by correctness). Inner loop: 4/5 train, 1/5 validation for early stopping. Final AUC reported on held-out outer fold. To quantify run-to-run variance, we repeat the full nested CV with different random seeds: 6 runs for Latent-SFT, 4 runs for CoLaR.

Data scope justification. We deliberately use only the GSM8k-Aug test set (N=1,319) under nested CV to ensure that probe evaluation is unbiased and that no correctness labels from the model's training distribution leak into probe training. While larger probe training sets could potentially improve absolute AUC, the consistency of results across nested CV folds (std 0.004–0.011) indicates the reported signal is robust at this sample size.

Compute. All experiments run on a single NVIDIA A100 (80GB). Probe training: $\sim$5 minutes per fold. Trajectory generation: $\sim$2 hours per paradigm on GSM8k-Aug. LTO reproduction (200 repeats $\times$ 11 β values): $\sim$1 hour.

Artifact licenses. GSM8k (Cobbe et al., 2021) is released under the MIT License. SVAMP (Patel et al., 2021) is released under the MIT License. Llama-3.2-1B-Instruct (Grattafiori et al., 2024) is released under the Llama 3.2 Community License Agreement. The Latent-SFT (Deng et al., 2025) and CoLaR (Tan et al., 2025) codebases are publicly available for research use. Our use of all artifacts is consistent with their intended research-only purposes.

B Hand-Crafted Signals and Architecture Ablation

Signal	CoLaR	LSFT(2)	LSFT(4)
Raw H_{spec}	0.409	0.356	0.391
Length T	**0.750**	0.331	0.346
LENS	**0.754**	0.702	0.714
Partial $\rho(\text{LENS}\|T)$	+0.125	+0.273	+0.307

Table 6: AUC of scalar signals for correct/incorrect classification on GSM8k-Aug, N=1319. AUC <0.5 indicates the signal is inverted (higher value correlates with incorrectness).

Architecture	LSFT	CoLaR
LENS (closed-form)	0.713	0.650
MLP (pointwise)	0.720±0.009	0.746±0.006
Attention (ours)	**0.805±0.011**	**0.853±0.004**

Table 7: Architecture ablation for the confidence probe on GSM8k-Aug. LENS is a closed-form baseline with no learned parameters. MLP processes each trajectory step independently and averages. Attention uses query-key attention to pool across the full trajectory before classification. Learned probes report mean±std over nested CV seeds: 6 runs for LSFT, 4 for CoLaR.

C Baseline Confidence Methods

We compare against standard confidence estimation approaches applicable to reasoning models. Last-token confidence (argmax probability) achieves only AUC 0.527 on Latent-SFT and is *undefined* for CoLaR's continuous hidden states—standard calibration methods (Guo et al., 2017; Desai and Durrett, 2020) do not apply to hidden-state paradigms, and verbalized confidence elicitation (Xiong et al., 2024) requires explicit token generation. A multi-feature logistic regression (length + entropy + last-token) reaches 0.738/0.770. The trained probe (0.805/0.853) dominates all baselines, with the largest gains on CoLaR where prompt and scalar features miss the most signal.

Table 8 further compares these methods in the downstream routing setting under the same cost model and 5-fold CV threshold selection. Only the full-trajectory probe significantly exceeds pure CoT (McNemar p=0.007). Length-based routing hurts because LSFT's length–correctness direction is inverted.

Method	AUC	CV Acc	Saving	Δ CoT
Pure CoT	—	50.5%	0%	—
Random (f=0.59)	0.500	50.2±0.7%	43%	−0.3pp
Length T	0.329	49.6±1.1%	17%	−0.9pp
Multi-feature LR	0.729	51.9±1.5%	17%	+1.4pp
Prompt-only probe	0.769	51.0±2.0%	23%	+0.5pp
Full traj. probe (ours)	**0.805**	**52.5±1.5%**	**43%**	**+2.0pp**

Table 8: Routing baselines under the same cost model and 5-fold CV threshold selection. Δ CoT is the accuracy gap vs. pure CoT.

D LTO Best-of-N Reproduction Details

We implement LTO Algorithm following Du et al. (2026): for each problem x, generate N rollouts $\{z_1, \ldots, z_N\}$, compute reward $r_i = f(z_i)$ from the probe, and select $z^* = \arg\max_i r_i \cdot \exp(-\beta \cdot D_{\text{KL}}(\pi_{z_i} \| \pi_{\text{base}}))$. Since our paradigms produce deterministic rollouts (Latent-SFT uses argmax decoding), we inject minimal noise: ε=0.001 Gaussian perturbation to logits (Latent-SFT) or σ-scaled noise to hidden states (CoLaR).

Config.	r	Pooled	Within	LTO	Oracle
LSFT(2)	2	0.755	0.594	+0.2	+10.2
LSFT(4)	4	0.929	0.680	+0.3	+3.8
CoLaR	∼5	0.975	0.510	−0.5	+5.3

Table 9: Within-problem AUC across configurations. LSFT(2) pooled AUC is lower than Table 2's 0.805 due to cross-r OOD and ε-perturbation.

β-scan protocol. $\beta \in \{0.001, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0\}$. For each β, 200 independent trials with different random seeds. Best β selected by mean accuracy across trials.

LSFT(2) pooled AUC discrepancy. Table 9 reports LSFT(2) pooled AUC of 0.755 (on ε-perturbed 8-rollout trajectories with the probe applied cross-r, trained on r=4). This differs from Table 2's 0.805±0.011 (deterministic single-rollout, in-distribution r=4). The degradation reflects calibration drift under cross-r OOD. However, threshold decisions for routing (§6.2) require only coarse

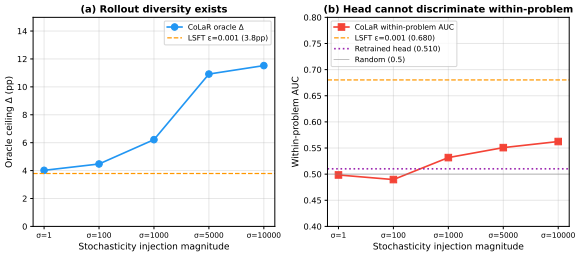

Figure 5: Intervention test on CoLaR. **Left**: oracle ceiling rises with stochasticity, confirming that diversity exists. **Right**: within-problem AUC remains near 0.5 across all σ scales. Even a retrained head achieves pooled AUC 0.975 but within-problem AUC 0.510.

discrimination (e.g., top-49% achieves 69% accuracy vs. bottom-51% at 32%, gap = 37pp), which remains feasible at this pooled AUC, and the routing accuracy in Table 4 confirms this empirically.

E CoLaR Routing Pareto Frontier

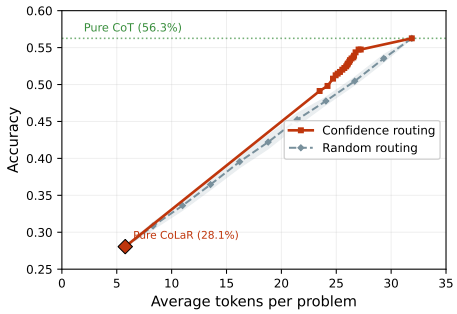

Figure 6: CoLaR routing Pareto frontier on GSM8k-Aug. Although the latent solver is much weaker than CoT, 28.1% vs. 56.3%, confidence routing still dominates random routing across all operating points. This confirms that the probe's ranking validity is paradigm-invariant, even when absolute accuracy precludes exceeding pure CoT.

F Additional Results

Quintile-level routing. The strongest single-bin routing advantage appears in quintile Q2 (second-highest confidence): latent 62.9% vs. CoT 56.8% (+6.1pp), with the latent advantage peaking in the "confident but not trivial" regime. Q1 shows a smaller gap because both solvers handle the easiest problems well, and Q4–Q5 reverse direction as difficulty exceeds latent reasoning's capacity. This non-monotone profile is consistent with the solver-fit interpretation: the probe identifies which problems lie in latent reasoning's competence region rather than ranking raw difficulty.

Variant	Latent-SFT		CoLaR	
	AUC	% of full	AUC	% of full
Full trajectory	0.820	100%	0.856	100%
Last-3 steps	0.823	100%	0.854	99.8%
Last-1 step	0.806	98.3%	0.813	93.3%
Fixed-length	0.801	97.7%	0.858	100%

Table 10: Truncation analysis. This table uses a single nested CV run to control for run-to-run variance; the LSFT full-trajectory value of 0.820 falls within the 0.805±0.011 band reported in Table 2.

Confidence bin	N	Latent	CoT	Δ
High (top 1/3)	439	**72.0%**	68.1%	+3.9pp
Mid (middle 1/3)	439	**55.8%**	51.9%	+3.9pp
Low (bottom 1/3)	441	22.4%	**31.5%**	−9.1pp

Table 11: Solver complementarity by confidence tertile. Columns show accuracy of each reasoning strategy within the bin.

Risk–coverage analysis. The selective prediction signal degrades smoothly across operating points. Latent-SFT GSM8k accuracy traces 80.1% / 69.7% / 59.3% at 30% / 50% / 70% coverage—an approximately linear ∼10pp decay per 20pp coverage increase. Two implications follow. First, the confidence ordering remains informative across the full distribution rather than isolating a small extremely-wrong tail, suggesting a continuous difficulty estimate rather than a binary failure detector. Second, the abstention budget is genuinely tunable: targeting high precision at ∼30% coverage yields 80.1% accuracy (+35pp over the 45.0% no-abstention baseline), while 70% coverage still gains +14.3pp. The same monotonic behavior holds on CoLaR and SVAMP, indicating the curve shape is a property of the signal structure rather than any single paradigm–dataset pairing.