Unlocking the Black Box of Latent Reasoning: An Interpretability-Guided Approach to Intervention

Shuochen Chang[1,*], Tong Bai[2,*], Xiaofeng Zhang[1], Qianli Ma[1], Qingyang Liu[1], Zhaohe Liao[1], Yibo Miao[1], Li Niu[1,†]

[1]Shanghai Jiao Tong University [2]Fudan University

{csc1332741686,framebreak,mqlqianli}@sjtu.edu.cn

tbai22@m.fudan.edu.cn

{narumimaria,zhaoheliao,miaoyibo,ustcnewly}@sjtu.edu.cn

[*]Equal contribution [†]Corresponding author

Abstract

Latent reasoning enables Large Language Models (LLMs) to perform multi-step inference within continuous hidden states, offering efficiency gains over explicit Chain-of-Thought (CoT). However, the opacity of these continuous thought vectors hinders their reliability and controllability. This paper bridges the gap between mechanistic interpretability and actionable control. We first present a systematic analysis using structural, causal, and geometric probes, revealing that latent vectors encode compressed, faithful representations of reasoning steps, with early vectors acting as critical causal hubs. Building on this, we operationalize these interpretability insights into a suite of training-free, decode-time interventions that refine the latent reasoning process by imposing the identified geometric and semantic priors. Extensive experiments across multiple model scales and diverse task domains demonstrate that our approaches consistently improve reasoning accuracy. Our interpretability-guided interventions consistently unlock latent capabilities and improve reasoning accuracy without any parameter updates.

1 Introduction

Large Language Models (LLMs) have evolved from simple pattern completion to performing sophisticated multi-step reasoning (DeepSeek-AI, 2025; OpenAI, 2023). Much of this progress stems from techniques like Chain-of-Thought (CoT) prompting (Wei et al., 2023), where models generate explicit, token-based rationales to structure their inference process. While effective, this reliance on verbalized reasoning introduces significant drawbacks. Explicit CoTs can be excessively long (Wang et al., 2025c), incurring substantial latency and computational costs. Furthermore, their textual nature provides a low-bandwidth medium for complex computation (Hao et al., 2024), often containing redundant information and increasing susceptibility to compounding errors (etc., 2023).

These limitations motivate a shift toward latent reasoning (etc., 2025c; Li et al., 2025b; Chen et al., 2025), where multi-step inference unfolds within the model's continuous hidden states without generating intermediate text. This approach enables more efficient computation by bypassing the need for explicit token generation. Existing methods explore various paradigms, such as optimizing hidden state trajectories (Tan et al., 2025; Zhang et al., 2025b; Li et al., 2025a; Zelikman et al., 2024) or reusing network layers (Saunshi et al., 2025; etc., 2025b; Geiping et al., 2025; Wang et al., 2025b) to simulate deeper computation. Among these, the continuous-thought paradigm (Hao et al., 2024; Shen et al., 2025b; Zeng et al., 2025; Wei et al., 2025; Liu et al., 2025a), which employs special latent vectors to propagate information across reasoning steps. It has shown particular promise due to its simplicity and compatibility with standard decoding pipelines.

Despite its potential, the continuous-thought paradigm faces two critical gaps in understanding. First, its interpretability remains largely unverified: it is unclear whether these latent vectors genuinely encode semantic steps of a reasoning process or are merely artifacts of the training objective (Deng et al., 2024). Second, existing studies are typically limited to smaller-scale models, leaving the scalability and behavior of this paradigm at larger scales under-explored. This paper aims to bridge these gaps: Not by proposing a new training method, but by establishing rigorous principles and diagnostic tools for interpreting and intervening in the latent reasoning process.

To address the first gap whether latent thought vectors genuinely encode semantic reasoning steps,

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 34019–34032

we develop a suite of three complementary interpretability probes (Wei et al., 2025; Deng et al., 2025; etc., 2025a; Alain and Bengio, 2018), designed to examine their structure, content, and causal role in the reasoning process. First, through structural alignment analyses (Kornblith et al., 2019; Davari et al., 2022; Li et al., 2024a), we find that the geometry of latent thought vectors closely mirrors that of explicit CoT rationales, suggesting a shared representational scaffold. Second, we show that a simple linear map can reconstruct textual reasoning steps from these latent vectors with high fidelity, confirming they serve as compressed yet faithful encodings of intermediate reasoning. Finally, causal intervention experiments demonstrate that targeted edits to these vectors especially in early reasoning steps to predictably alter the model's final answer, establishing their functional role in the inference pipeline (Meng et al., 2023; Turner et al., 2024; Wu et al., 2024; Liu et al., 2025b; Li et al., 2023).

Building on these insights, we design a series of training-free, decode-time interventions. By manipulating cached latent states by projecting them along directions identified by our probes, we can both validate our interpretability hypotheses and measurably improve reasoning performance without any parameter updates. These consistent gains from simple, norm-preserving edits highlight the untapped potential residing within the model's existing latent pathways.

Our contributions are threefold:

1. **Unified Interpretability Framework:** We provide the first systematic analysis of the continuous-thought paradigm, establishing structural, pointwise, and causal evidence that latent vectors are not opaque artifacts but steerable, semantic reasoning blueprints.

2. **From Theory to Practice:** We translate our interpretability findings into a novel suite of training-free, decode-time interventions. By enforcing semantic consistency and geometric regularity, we demonstrate that model performance can be improved solely by manipulating inference dynamics.

3. **Scalable and Robust Validation:** We extend the evaluation of latent reasoning to large-scale models (up to 8B) and diverse domains. Our results on GSM8K, OOD benchmarks, and StrategyQA confirm that the identified causal mechanisms are fundamental and generalizable properties of latent reasoning.

2 Related Works

2.1 Paradigms for Latent Reasoning

Latent reasoning seeks to realize multi-step inference inside hidden states, avoiding explicit intermediate tokens while retaining standard decoding for the final answer. Prior work advances mechanisms as follows. Latent optimization refines hidden trajectories in place (Saunshi et al., 2025; etc., 2025b; Wang et al., 2025a; Goyal et al., 2024) during decoding or distills long textual rationales into compact continuous representations. Signal-guided control steers trajectories with auxiliary signals such as confidence estimates, verifier outputs, or decode-time selection criteria, and may bias states toward informative embedding subspaces (Zelikman et al., 2024; Du et al., 2025b; Liu et al., 2024a; Du et al., 2025a). Layer-recurrent (Geiping et al., 2025; Wang et al., 2025b; Zhang et al., 2025a; Shen et al., 2025a) execution emulates deeper computation without lengthening the sequence by reusing layers or activations, for example through architectural loops or by feeding a layer's output back as the next input. Our study focuses on the continuous-thought latent-span variant (Wei et al., 2025; Zhang et al., 2025b; Zeng et al., 2025; Hao et al., 2024; Liu et al., 2025a), in which reserved latent vectors are written into the prefix to carry information across steps, as popularized by CoConut-style models. Those most demonstrations target small models and their scalability and interpretability at larger scales remain underexplored.

2.2 Interpretability of Latent Reasoning

Interpretability efforts examine where and how hidden computation acquires stepwise structure and causal influence. Mechanistic analyses (Wei et al., 2025; Deng et al., 2025; Liu et al., 2025a; Wang et al., 2025a) track the emergence of staged computation within transformer internals, relating attention flow and layer specialization to progressive subgoal formation. Behavioral studies (Li et al., 2025a; Tan et al., 2025; Yu, 2025; Lu et al., 2025; Du et al., 2025a) infer latent processing from macroscopic signatures such as abrupt memorization-to-generalization transitions and the internalization of intermediate steps that manifest as step skipping at inference. Representational analyses (Shen et al., 2025b; Zhu et al., 2025;

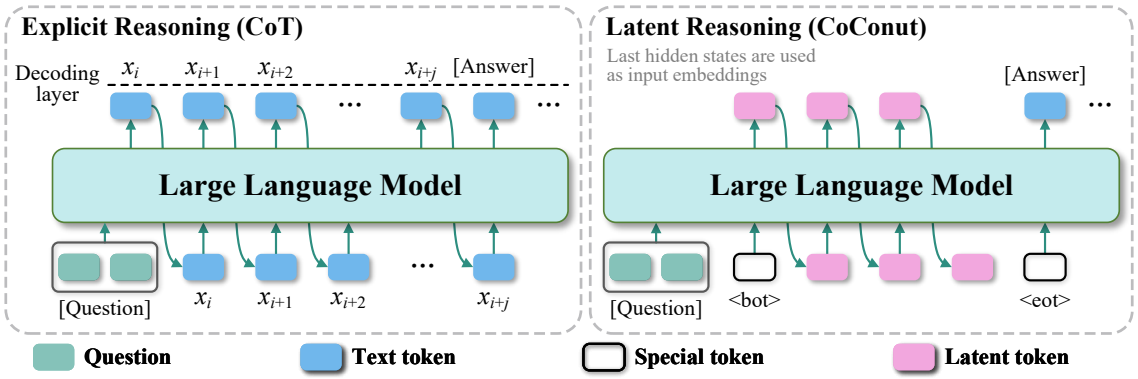


Figure 1: **Explicit Reasoning versus Latent Reasoning**. Left: Explicit Reasoning externalizes intermediate reasoning as a token sequence $(x_i, x_{i+1}, \ldots, x_{i+j})$.Right: Latent Reasoning reserves a latent span and at each reserved slot, feeds the preceding final-layer hidden state back as the next input, yielding a continuous thought that conditions subsequent decoding.

Zhang et al., 2025a; Liu et al., 2024b; Yan et al., 2024) link hidden trajectories to explicit reasoning via centered kernel alignment, linear decoding, and lightweight mappers, and corroborate these relations through targeted causal edits that ablate, steer, or transplant states (Wu et al., 2024; Li et al., 2024b). Our method contributes the interpretability by integrating structural alignment, then operationalizing the resulting priors as training-free, decode-time interventions. The protocol both supports interpretability and yields consistent gains in reasoning accuracy, connecting explanation with capability improvement.

3 Interpretation of Latent Reasoning

To investigate whether latent reasoning genuinely performs structured, multi-step inference, we conduct a three-part analysis. We first probe the semantic content of the latent vectors by measuring their correspondence with explicit Chain-of-Thought (CoT) representations (Section 3.2). Next, we perform a series of causal interventions to test whether manipulating these vectors predictably controls the model's output (Section 3.3). Finally, we explore the architectural and geometric properties that enable this emergent computational structure (Section 3.4).

3.1 Preliminary

Our work focuses on a mode of reasoning where inference occurs within the model's continuous hidden states, rather than through verbalized CoT. A standard causal language model with parameters θ defines the probability of a sequence $w_{1:T}$ as:

$$p_\theta(w_{1:T}) = \prod_{t=1}^{T} p_\theta(w_t \mid w_{<t}). \tag{1}$$

While explicit CoT involves minimizing cross-entropy over a sequence of textual reasoning steps, latent reasoning reserves a special span of K positions within the prompt that do not correspond to vocabulary tokens. The computation within this span is realized through a sequence of hidden vectors we term continuous thoughts.

Formally, given an input embedding matrix $\mathbf{E} \in \mathbb{R}^{T\times d}$, a latent span of length K is introduced. The model deterministically populates this span by sequentially generating K continuous thought vectors, $\mathbf{z}_{1:K}$. At each latent position ℓ_k, the vector is derived from the previous position's final hidden state and then used as the input embedding for the current position:

$$\mathbf{z}_k \triangleq \mathbf{h}^{(L)}_{\ell_k-1}, \quad \text{and} \quad \tilde{\mathbf{E}}_{\ell_k} \leftarrow \mathbf{z}_k, \tag{2}$$

where $\mathbf{h}^{(L)}_t$ is the last-layer hidden state at position t and $\tilde{\mathbf{E}}$ is the resulting "filled" embedding matrix. Conditioned on $\tilde{\mathbf{E}}$, the model proceeds with standard autoregressive decoding to generate the final answer y:

$$p_\theta(y \mid x) \equiv p_\theta\big(y \mid \tilde{\mathbf{E}}(x)\big). \tag{3}$$

Crucially, the latent vectors $\mathbf{z}_k$ are not directly supervised. Their values are learned implicitly through the standard next-token prediction loss,

which is applied only to the visible target tokens $\mathcal{V}_{\text{vis}}$ (i.e., the answer):

$$\mathcal{L}(\theta) = -\sum_{t \in \mathcal{V}_{\text{vis}}} \log p_\theta\big(w_t \mid w_{<t};\, \tilde{\mathbf{E}}(x)\big). \quad (4)$$

3.2 Probing the Semantic Content of Latent Vectors

Our first hypothesis is that the sequence of continuous thoughts, $\mathbf{z}_{1:K}$, serves as a compressed, continuous analogue to a textual CoT. To test this, we assess the correspondence between the latent representations and the hidden states generated by an equivalent model executing an explicit CoT.

Establishing Correspondence. Let $\mathcal{T}$ be the set of textual step boundaries in a teacher-forced CoT. We extract the final-layer hidden state at the end of each step $t \in \mathcal{T}$ to serve as its canonical representation: $\mathbf{c}_t(x) \triangleq \mathbf{h}^{(L)}_{\text{end}(t)}(x)$. To create a comparable latent representation, we locally average a small, contiguous block of continuous thoughts that are heuristically aligned with step t, forming a step-level latent surrogate:

$$\bar{\mathbf{z}}_t(x) \triangleq \frac{1}{|\mathcal{K}(t)|} \sum_{k \in \mathcal{K}(t)} \mathbf{z}_k(x). \quad (5)$$

This aggregation allows us to test for step-wise correspondence without making strong assumptions about a one-to-one mapping between latent vectors and reasoning steps.

Multi-faceted Probing Analysis. We employ three complementary methods to probe the relationship between the latent surrogates $\{\bar{\mathbf{z}}_t\}$ and their explicit counterparts $\{\mathbf{c}_t\}$. First, we measure their **structural alignment** using linear Centered Kernel Alignment (CKA), which quantifies the similarity of the geometric arrangement of data points across different instances. CKA is defined as:

$$\text{CKA}(\tilde{\mathbf{Z}}_t, \tilde{\mathbf{C}}_t) = \frac{\|\tilde{\mathbf{Z}}_t^\top \tilde{\mathbf{C}}_t\|_F^2}{\|\tilde{\mathbf{Z}}_t^\top \tilde{\mathbf{Z}}_t\|_F \cdot \|\tilde{\mathbf{C}}_t^\top \tilde{\mathbf{C}}_t\|_F}, \quad (6)$$

where $\tilde{\mathbf{Z}}_t$ and $\tilde{\mathbf{C}}_t$ are matrices whose rows are the centered latent surrogates and CoT step representations, respectively, for a set of problems.

Second, we test for pointwise recoverability by training a simple linear map f_ϕ to predict a CoT representation $\mathbf{c}_t(x)$ from its corresponding latent surrogate $\bar{\mathbf{z}}_t(x)$. The model is trained to maximize cosine similarity, a stringent test of linear decodability:

$$\mathcal{L}_{\cos}(\phi) = \mathbb{E}_{t,x}\left[1 - \cos\big(f_\phi(\bar{\mathbf{z}}_t(x)), \mathbf{c}_t(x)\big)\right]. \quad (7)$$

Finally, we conduct a non-parametric lexical probe to see if a latent surrogate contains information about the tokens in the corresponding reasoning step. We feed $\bar{\mathbf{z}}_t(x)$ through the model's language modeling head and measure the probability mass assigned to the top tokens of the actual CoT step:

$$p_{\text{vocab}}(\cdot \mid \mathbf{z}) = \text{softmax}\big(\mathbf{W} \times \text{LN}(\mathbf{z}) + \mathbf{b}\big). \quad (8)$$

Metric	Value	vs. Baseline
I. Structural Alignment *(Geometry)*		
Linear CKA Score	**0.72**	–
II. Pointwise Recoverability *(Content)*		
Cosine Similarity	**0.75**	+0.61 (Identity)
III. Lexical Probe *(Vocabulary)*		
Top-1 Token Accuracy	**0.38**	> Random

Table 1: **Semantic Correspondence Analysis.** Latent vectors exhibit high structural similarity (CKA) and linear recoverability relative to explicit CoT steps, confirming they are compressed semantic states.

Results. Table 1 confirms a high degree of semantic fidelity. The strong CKA scores indicate a significant geometric alignment between latent and explicit reasoning spaces, while the high cosine similarity verifies that CoT steps are linearly recoverable. Additionally, the lexical probe demonstrates robust token-level decodability. As visualized in Figure 2, these results establish that latent vectors function not as opaque artifacts, but as faithful, compressed encodings of the reasoning process.

3.3 Analyzing the Causal Role of Latent Thought Sequences

Having established a correlational link between latent vectors and reasoning steps, we now investigate their causal role. If continuous thoughts truly guide the reasoning process, then targeted interventions on them should predictably alter the model's final answer. We measure the effect of an intervention $\mathcal{I}$ on a latent sequence $\mathbf{Z}$ by the change in log-probability of the target answer $\mathbf{y}^\star$:

$$\Delta \log p_{\text{tgt}} \triangleq \log p(\mathbf{y}^\star | \mathcal{I}(\mathbf{Z})) - \log p(\mathbf{y}^\star | \mathbf{Z}). \quad (9)$$

Intervention Strategies. We probe causal influence through three complementary intervention

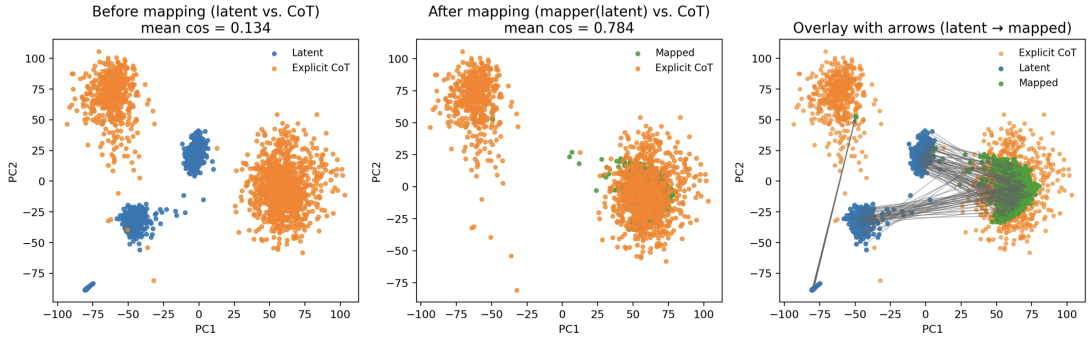


Figure 2: **Mapper-based alignment of latent thoughts to explicit CoT**. (a) Before mapping, latent and CoT points are separated in PCA space. (b) A linear mapper f_ϕ aligns latents with CoT clusters. (c) Overlay with arrows (latent→mapped) visualizes the systematic displacement toward CoT.

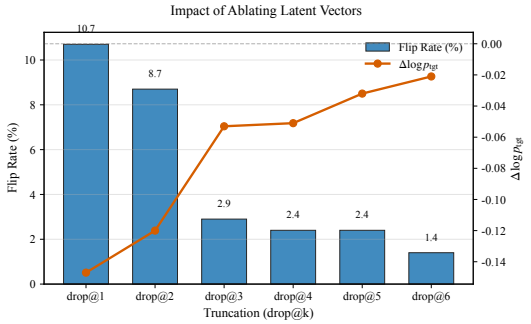


Figure 3: Impact of ablating latent vectors. The answer flip rate decreases as more of the initial latent vectors are preserved (from drop@0 to drop@K), confirming that early latent thoughts are causally critical for the final outcome.

Intervention Type	**Flip %**	$\Delta \log p$	**Dir. %**
A. Ablation (Positional Importance)			
Drop@0 (Remove all)	10.7	-0.15	–
Drop@1 (Keep first)	8.7	-0.12	–
B. Targeted Edits (Steerability)			
Slerp (to Answer)	21.5	**+0.85**	–
Gradient Update	24.1	**+1.02**	–
C. Transplant (Plan Transfer)			
Latent Sequence	2.5	-0.05	**48.3**
Explicit CoT (Ref)	95.0	-4.51	30.0

Table 2: **Causal Intervention Results.** Early ablation harms performance, while norm-preserving edits steer the outcome. Notably, latent transplants transfer "reasoning plans" (high Directional %) better than explicit text.

families applied to cached latent vectors. To assess positional salience, we truncate the latent sequence after the k-th vector, denoted $\mathrm{drop@}k$, thereby removing downstream stages of latent computation and measuring the resulting change in prediction. To test fine-grained controllability, we introduce small, norm-preserving edits to an early latent state: a spherical interpolation that reorients the vector toward an answer-supporting boundary (Eq. 13), and a single normalized gradient step that nudges the vector in the direction that increases the likelihood of the correct answer (Eq. 14). To evaluate transferability, we transplant the entire latent sequence from one instance into another and examine whether the model's answer shifts toward the donor's target, treating the sequence as a putative internal reasoning plan. Together, these interventions jointly characterize where information is concentrated, whether it can be steered with minimal edits, and to what extent latent trajectories encode reusable guidance for decoding.

Results. Table 2 summarizes the causal effects. First, ablation experiments (Figure 3) show that early latent vectors are critical; removing them causes a sharp drop in performance. Second, targeted edits prove that these vectors are steerable: small directional updates reliably shift the output probabilities and can flip the final answer. Finally, the transplant experiment demonstrates effective "plan transfer." Unlike explicit CoT, transplanting latent sequences successfully biases the model toward the donor's answer. Together, these results confirm that latent sequences encode robust and transferable reasoning trajectories.

3.4 Uncovering Architectural and Geometric Underpinnings

What properties of the Transformer architecture enable this stable, structured latent computation? We hypothesize two key factors: the geometry induced by tied input-output embeddings and the emergence of a monotonically decreasing energy function along the latent chain.

Weight Tying as a Nearest-Neighbor Projector. In models with tied embeddings, the output projection matrix is the transpose of the input embedding matrix ($W_o = E^\top$). Consequently, selecting the next token $\hat{\imath}$ at low temperature is equivalent to finding the token embedding $\mathbf{e}_{\hat{\imath}}$ with the highest cosine similarity to the final hidden state $\mathbf{h}$:

$$\hat{\imath} = \arg\max_i \langle \mathbf{h}, \mathbf{e}_i \rangle. \tag{10}$$

The standard "decode-then-re-embed" cycle can thus be viewed as a nearest-neighbor projection $\mathcal{T}(\mathbf{h}) = \mathbf{e}_{\hat{\imath}}$, which maps $\mathbf{h}$ onto the discrete codebook of token embeddings. Our latent update mechanism, which directly feeds $\mathbf{h}$ as the next input, is a continuous approximation of this cycle. This approximation is accurate when $\mathbf{h}$ is already close to an embedding vector $\mathbf{e}_{\hat{\imath}}$, i.e., when $\|\mathbf{h} - \mathcal{T}(\mathbf{h})\|^2$ is small. Untied embeddings introduce a tying gap, weakening this geometric correspondence and potentially destabilizing latent computation.

A Monotone Energy Over the Latent Chain. We further hypothesize that the latent trajectory follows a directed path from a high-energy "unresolved" state to a low-energy "resolved" state. We test this by learning a simple scalar energy function $H : \mathbb{R}^d \to \mathbb{R}$ optimized with a margin ranking loss to ensure energy decreases at each step along the chain from the last question token to the first answer token:

$$\mathcal{L}_{\text{rank}} = \mathbb{E}\left[\max\{0, \gamma - (H(\mathbf{x}_t) - H(\mathbf{x}_{t+1}))\}\right]. \tag{11}$$

Results. Table 3 validates both architectural hypotheses. First, **weight tying** proves critical; models with tied embeddings achieve significantly higher accuracy ratios (r_{Acc}) compared to untied variants. Second, the learned energy landscape exhibits monotonicity along the latent chain, with rank correlations approaching 1.0. This confirms that the latent process is not a random walk, but a **structured, contractive descent** toward the final answer.

In summary, this section establishes a unified interpretability framework for the continuous-thought paradigm. We demonstrate that latent vectors are not arbitrary states, but semantically aligned representations that precise causal control over the thinking process. Our findings validate latent reasoning as a robust and interpretable alternative to explicit CoT.

Architectural Property	Result
I. Impact of Weight Tying	
Accuracy Ratio (r_{Acc}) – **Tied (Ours)**	**2.97**
Accuracy Ratio (r_{Acc}) – Untied	1.15
II. Energy Landscape Monotonicity	
Strictly Monotonic Chains	**99.0%**
Spearman Rank Correlation	> 0.99

Table 3: **Geometric Underpinnings.** Weight tying is crucial for latent efficacy (r_{Acc}), and the learned energy function confirms a highly structured, monotonic descent towards the solution.

4 Interventions on Latent Reasoning

Building upon the interpretability analysis in Section 3, we now operationalize our findings. We introduce a series of novel training-free interventions that manipulate cached latent vectors at decode-time. Formally, given a latent state $\mathbf{z}_k$, we seek a steered state $\mathbf{z}'_k$ that maximizes alignment with a prior-defined reasoning manifold while minimizing deviation from the original context. As detailed in Algorithm 1, this process is governed by a generalized update rule, $\mathbf{z}'_k \leftarrow \mathcal{I}(\mathbf{z}_k; \Phi, \Omega)$, where the *guidance prior* Φ and *manifold constraint* Ω are instantiated based on the specific interpretability source (Semantic, Causal, or Geometric). Our goal is to demonstrate that by leveraging a theoretical understanding of the latent reasoning process, we can unlock and enhance the model's inherent inferential capabilities without any parameter updates.

4.1 Experimental Setup

Latent Reasoning Paradigms. We evaluate our interventions on two distinct paradigms within the continuous-thought framework: CoConut employs a latent span optimization to fill reserved placeholders with continuous thoughts, CODI distills a long, explicit Chain-of-Thought into a compressed sequence of continuous vectors prefixed to the input.

Models and Tasks. We conduct experiments across Qwen3-8B, Llama-3.1-8B and the smaller Llama-3.2-3B. To rigorously evaluate efficacy and scalability, Our evaluation covers three distinct domains: (1) **In-Domain Mathematical Reasoning**: We use the standard GSM8K (Cobbe et al., 2021) test set. (2) **Out-of-Domain (OOD) Robustness**: We evaluate on GSM-Hard (Gao et al., 2023) and SVAMP (Patel et al., 2021) to test generalization to higher complexity and varying linguistic patterns. (3) **Implicit Commonsense Reasoning**: We in-

Algorithm 1 Unified Interpretability-Guided Latent Steering

Require: Latent State $\mathbf{z}_k$, Variant $\mathcal{M}$, Hyperparams α, λ, η

Configuration (Define Prior Φ and Constraint Ω via $\mathcal{M}$)

A: $\Phi(\mathbf{z}) = F_{\text{map}}(\mathbf{z}), \quad \Omega(\mathbf{z}) = P_{\text{subspace}}\mathbf{z}$

B: $\Phi(\mathbf{z}) = \mathbf{z} - \eta\nabla\mathcal{L}, \quad \Omega(\mathbf{z}) = \mathbf{z}/\|\mathbf{z}\|$

C: $\Phi(\mathbf{z}) = \mathbf{z} - \eta\nabla H, \quad \Omega(\mathbf{z}) = \text{WT-Proj}(\mathbf{z})$

Execution

$\mathbf{v}^* \leftarrow \Phi(\mathbf{z}_k)$ ▷ Extract guidance signal

$\mathbf{d} \leftarrow \mathbf{v}^* - \mathbf{z}_k$

$\mathbf{z}^{\text{steer}} \leftarrow \mathbf{z}_k + \alpha \cdot \frac{\mathbf{d}}{\|\mathbf{d}\|} \cdot \|\mathbf{z}_k\|$ ▷ Apply directional update

$\mathbf{z}'_k \leftarrow (1-\lambda)\mathbf{z}^{\text{steer}} + \lambda \cdot \Omega(\mathbf{z}^{\text{steer}})$ ▷ Manifold regularization

return $\mathbf{z}'_k$

clude StrategyQA (Geva et al., 2021) to verify that our findings extend beyond mathematical tasks.

4.2 Intervention A: Semantic Structure Transport

Drawing from our discovery in Section 3.2 that latent vectors are semantically aligned with explicit CoT steps, we propose **Mapper-Guided Transport** to refine the latent trajectory. We identify the mean $\mathbf{m}_0$ of the terminal latent pair and map it to a target semantic destination $\mathbf{t}_0 = F(\mathbf{m}_0)$ using the linear mapper trained in Section 3.

We then steer the latent state using spherical linear interpolation (slerp) and norm mixing:

$$\begin{aligned} \widehat{\mathbf{u}} &= \text{slerp}\Big(\frac{\mathbf{m}_0}{\|\mathbf{m}_0\|}, \frac{\mathbf{t}_0}{\|\mathbf{t}_0\|}; \alpha\Big), \\ \widehat{\mathbf{m}} &= \Big[(1-\eta)\|\mathbf{m}_0\| + \eta\|\mathbf{t}_0\|\Big]\ \widehat{\mathbf{u}}, \end{aligned} \tag{12}$$

where α controls the directional shift and η regulates the norm influence. To preserve instance-specific information, we re-apply the original residual vectors to the updated mean. Finally, to ensure the modified vectors remain on the valid token manifold, we apply **Embedding-Subspace Alignment** via a projector P_r derived from the principal components of the embedding matrix.

4.3 Intervention B: Causal Hub Editing

Our causal analysis in Section 3.3 identified early latent vectors (e.g., $\mathbf{z}_2$) as "causal hubs" that exert disproportionate control over the final output. We introduce two strategies to strictly edit these critical states.

Answer-Anchored Slerp. We reorient the critical vector $\mathbf{z}_2$ toward a boundary vector $\mathbf{t}$ (derived from a retrieved exemplar) to steer the reasoning direction:

$$\mathbf{z}'_2 = \|\mathbf{z}_2\|\ \text{slerp}\Big(\frac{\mathbf{z}_2}{\|\mathbf{z}_2\|}, \frac{\mathbf{t}}{\|\mathbf{t}\|}; \alpha\Big). \tag{13}$$

Answer-Directed Gradient Update. Alternatively, we apply a single gradient step to maximize the likelihood of the correct answer direction while strictly enforcing a norm constraint to prevent energy explosion:

$$\mathbf{z}'_2 = \mathbf{z}_2 - \eta\,\|\mathbf{z}_2\|\ \frac{\nabla_{\mathbf{z}_2}\mathcal{L}}{\|\nabla_{\mathbf{z}_2}\mathcal{L}\| + \varepsilon}. \tag{14}$$

4.4 Intervention C: Geometric Priors

Based on the architectural findings in Section 3.4, we propose interventions that enforce the geometric consistency of the latent space.

Weight-Tying Consistent Projection. To strengthen the link between hidden states and the output vocabulary space, we nudge the hidden state $\mathbf{h}_\ell$ towards its expected embedding representation $\tilde{\mathbf{e}}_\ell$:

$$\tilde{\mathbf{e}}_\ell = E^\top \text{softmax}(\mathbf{o}_{\ell-1}/\tau), \quad \mathbf{h}'_\ell = (1-\alpha)\mathbf{h}_\ell + \alpha\,\tilde{\mathbf{e}}_\ell. \tag{15}$$

Energy-Guided Local Descent. Using the monotonicity energy function $H(\cdot)$ trained in Section 3, we apply a trust-region gradient descent step. This ensures the reasoning process progresses "downhill" in the energy landscape, stabilizing the inference trajectory:

$$\mathbf{h}'_\ell = \mathbf{h}_\ell - \text{Proj}_{\mathbb{B}(\mathbf{0}, \rho\|\mathbf{h}_\ell\|)}\Big(\eta\,\nabla H(\mathbf{h}_\ell)\Big). \tag{16}$$

4.5 Experimental Results and Analysis

We present a unified evaluation of our interventions. Table 4 summarizes the performance across all mathematical reasoning benchmarks (In-Domain and OOD), organized hierarchically by intervention family. Table 5 details the results on commonsense reasoning.

Efficacy on Mathematical Reasoning. Table 4 demonstrates that our interpretability-guided interventions yield consistent improvements across all settings. Semantic Transport (Method A)

		Qwen3-8B			Llama-3.1-8B			Llama-3.2-3B		
		In-domain	OOD		In-domain	OOD		In-domain	OOD	
Variant	**Paradigm**	**GSM8K**	**GSM-H**	**SVAMP**	**GSM8K**	**GSM-H**	**SVAMP**	**GSM8K**	**GSM-H**	**SVAMP**
Baselines										
No Interv.	CoConut	50.4	14.1	68.5	44.8	12.2	63.2	42.1	11.5	58.9
	CODI	62.1	17.9	80.3	60.7	14.5	76.4	58.9	13.4	72.3
A. Semantic Structure (Sec. 4.2)										
Mapper	CoConut	51.9 (+1.5)	15.3 (+1.2)	69.6 (+1.1)	46.2 (+1.4)	13.5 (+1.3)	64.2 (+1.0)	43.2 (+1.1)	12.4 (+0.9)	60.1 (+1.2)
	CODI	**63.5** (+1.4)	18.9 (+1.0)	**81.6** (+1.3)	**62.2** (+1.5)	**15.9** (+1.4)	77.7 (+1.3)	59.6 (+0.7)	**14.7** (+1.3)	**73.6** (+1.3)
B. Causal Hub Editing (Sec. 4.3)										
Slerp (B.1)	CoConut	51.6 (+1.2)	**15.5** (+1.4)	**69.7** (+1.2)	46.2 (+1.4)	13.3 (+1.1)	**64.4** (+1.2)	43.0 (+0.9)	12.6 (+1.1)	59.6 (+0.7)
	CODI	62.5 (+0.4)	18.6 (+0.7)	81.4 (+1.1)	61.3 (+0.6)	15.4 (+0.9)	**77.8** (+1.4)	59.8 (+0.9)	14.4 (+1.0)	73.3 (+1.0)
Gradient (B.2)	CoConut	**52.2** (+1.8)	15.1 (+1.0)	69.4 (+0.9)	**46.4** (+1.6)	13.1 (+0.9)	64.2 (+1.0)	43.3 (+1.2)	**12.7** (+1.2)	60.0 (+1.1)
	CODI	62.9 (+0.8)	19.2 (+1.3)	81.1 (+0.8)	61.1 (+0.4)	15.7 (+1.2)	77.0 (+0.6)	59.7 (+0.8)	14.2 (+0.8)	73.4 (+1.1)
C. Geometric Priors (Sec. 4.4)										
WT-Proj (C.1)	CoConut	51.0 (+0.6)	14.9 (+0.8)	69.5 (+1.0)	45.5 (+0.7)	13.0 (+0.8)	64.1 (+0.9)	43.1 (+1.0)	12.3 (+0.8)	**60.1** (+1.2)
	CODI	62.8 (+0.7)	18.8 (+0.9)	81.3 (+1.0)	61.6 (+0.9)	15.3 (+0.8)	77.6 (+1.2)	59.6 (+0.7)	14.4 (+1.0)	72.7 (+0.4)
Energy (C.2)	CoConut	51.4 (+1.0)	15.2 (+1.1)	69.4 (+0.9)	45.4 (+0.6)	13.1 (+0.9)	**64.4** (+1.2)	42.9 (+0.8)	**12.7** (+1.2)	59.8 (+0.9)
	CODI	62.6 (+0.5)	**19.3** (+1.4)	81.5 (+1.2)	61.9 (+1.2)	15.7 (+1.2)	77.5 (+1.1)	59.8 (+0.9)	14.5 (+1.1)	73.3 (+1.0)

Table 4: **Unified Performance Analysis on Mathematical Reasoning.** We report the accuracy (%) on GSM8K (In-Domain), GSM-Hard and SVAMP (OOD) across three model scales. The results are grouped by intervention family (A, B, C). Boldface highlights the best performance for each model-paradigm configuration.

proves particularly robust for OOD tasks, validating that aligning latents with semantic CoT clusters improves generalization. Causal Hub Editing (Method B) is highly effective as shown, the Gradient-based update (B.2) achieves the highest gains on the standard GSM8K benchmark, corroborating our finding that early latent vectors act as steerable control points. Geometric Priors (Method C) offer stable gains, confirming that enforcing architectural constraints regularizes the inference process effectively.

Scalability to Model-Scales. Table 4 show that our interventions remain highly effective cross different scales. For instance, Mapper-Guided Transport improves CODI performance on GSM-Hard by +1.3%. This indicates that the latent structures we exploit are fundamental to the paradigm.

Generalization to Commonsense Reasoning. We report results on StrategyQA in Table 5in order to assess domain universality. All interventions provide positive gains, with Causal Hub Editing showing particular strength (up to +1.5% for Qwen-CODI). This suggests that the causal mechanisms are task-agnostic properties of continuous thought.

		Qwen3-8B	Llama-3.1-8B	Llama-3.2-3B
Variant	**Paradigm**	**StrategyQA**	**StrategyQA**	**StrategyQA**
Baselines				
No Interv.	CoConut	75.5	71.4	67.6
	CODI	80.9	76.3	70.2
A. Semantic Structure				
Mapper	CoConut	76.6 (+1.1)	72.4 (+1.0)	68.3 (+0.7)
	CODI	82.1 (+1.2)	77.4 (+1.1)	**71.3** (+1.1)
B. Causal Hub Editing				
Slerp (B.1)	CoConut	76.7 (+1.2)	72.6 (+1.2)	**68.8** (+1.2)
	CODI	**82.4** (+1.5)	77.1 (+0.8)	71.1 (+0.9)
Gradient (B.2)	CoConut	**76.8** (+1.3)	**72.7** (+1.3)	68.5 (+0.9)
	CODI	81.9 (+1.0)	**77.5** (+1.2)	71.2 (+1.0)
C. Geometric Priors				
WT-Proj (C.1)	CoConut	76.4 (+0.9)	72.2 (+0.8)	68.7 (+1.1)
	CODI	81.9 (+1.0)	77.3 (+1.0)	**71.3** (+1.1)
Energy (C.2)	CoConut	76.1 (+0.6)	72.4 (+1.0)	68.0 (+0.4)
	CODI	82.2 (+1.3)	77.2 (+0.9)	**71.3** (+1.1)

Table 5: **Impact on Commonsense Reasoning (StrategyQA).** We apply the same interventions to evaluate cross-domain applicability. Values report accuracy (%) with absolute improvement in parentheses.

5 Conclusion

This paper establishes a unified framework of interpretability and intervention in latent reasoning. We demonstrate that continuous thought vectors are not artifacts but structured semantic trajectories governed by distinct geometric and causal priors. Translating these insights into training-free interventions leads to consistent performance gains across mathematics, commonsense domains, and different model scales. These improvements serve as a strong validation of our interpretability hypotheses, demonstrating that latent reasoning is interpretably controllable, offering a robust foundation for future latent reasoning paradigms.

Limitations

While our training-free interventions significantly enhance reasoning performance without parameter updates, they introduce a marginal computational overhead during inference due to the additional calculation of gradients and projections. Future work aims to eliminate this latency by incorporating these interpretability priors as supervisory signals, thereby developing improved training algorithms for latent reasoning.

Acknowledgments

The work was supported by the National Natural Science Foundation of China (Grant No. 62471287).

References

Guillaume Alain and Yoshua Bengio. 2018. Understanding intermediate layers using linear classifier probes. *Preprint*, arXiv:1610.01644.

Xinghao Chen, Anhao Zhao, Heming Xia, Xuan Lu, Hanlin Wang, Yanjun Chen, Wei Zhang, Jian Wang, Wenjie Li, and Xiaoyu Shen. 2025. Reasoning beyond language: A comprehensive survey on latent chain-of-thought reasoning. *Preprint*, arXiv:2505.16782.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *Preprint*, arXiv:2110.14168.

MohammadReza Davari, Stefan Horoi, Amine Natik, Guillaume Lajoie, Guy Wolf, and Eugene Belilovsky. 2022. Reliability of cka as a similarity measure in deep learning. *Preprint*, arXiv:2210.16156.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Jingcheng Deng, Liang Pang, Zihao Wei, Shichen Xu, Zenghao Duan, Kun Xu, Yang Song, Huawei Shen, and Xueqi Cheng. 2025. Latent reasoning in llms as a vocabulary-space superposition. *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. *Preprint*, arXiv:2405.14838.

Hanwen Du, Yuxin Dong, and Xia Ning. 2025a. Latent thinking optimization: Your latent reasoning language model secretly encodes reward signals in its latent thoughts. *Preprint*, arXiv:2509.26314.

Hung Du, Srikanth Thudumu, Rajesh Vasa, and Kon Mouzakis. 2025b. A survey on context-aware multi-agent systems: Techniques, challenges and future directions. *Preprint*, arXiv:2402.01968.

Andy Zou etc. 2025a. Representation engineering: A top-down approach to ai transparency. *Preprint*, arXiv:2310.01405.

Lanham Tamera etc. 2023. Measuring faithfulness in chain-of-thought reasoning. *Preprint*, arXiv:2307.13702.

Rui-Jie Zhu etc. 2025b. Scaling latent reasoning via looped language models. *Preprint*, arXiv:2510.25741.

Rui-Jie Zhu etc. 2025c. A survey on latent reasoning. *Preprint*, arXiv:2507.06203.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Pal: Program-aided language models. *Preprint*, arXiv:2211.10435.

Jonas Geiping, Sean McLeish, Neel Jain, John Kirchenbauer, Siddharth Singh, Brian R. Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, and Tom Goldstein. 2025. Scaling up test-time compute with latent reasoning: A recurrent depth approach. *Preprint*, arXiv:2502.05171.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies.

Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Krishna Menon, Sanjiv Kumar, and Vaishnavh Nagarajan. 2024. Think before you speak: Training language models with pause tokens. *Preprint*, arXiv:2310.02226.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. 2024. Training large language models to reason in a continuous latent space. *Preprint*, arXiv:2412.06769.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. *Preprint*, arXiv:1905.00414.

Hengli Li, Chenxi Li, Tong Wu, Xuekai Zhu, Yuxuan Wang, Zhaoxin Yu, Eric Hanchen Jiang, Song-Chun Zhu, Zixia Jia, Ying Nian Wu, and Zilong Zheng. 2025a. Seek in the dark: Reasoning via test-time instance-level policy gradient in latent space. *Preprint*, arXiv:2505.13308.

Jindong Li, Yali Fu, Li Fan, Jiahong Liu, Yao Shu, Chengwei Qin, Menglin Yang, Irwin King, and Rex Ying. 2025b. Implicit reasoning in large language models: A comprehensive survey. *Preprint*, arXiv:2509.02350.

Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2024a. Emergent world representations: Exploring a sequence model trained on a synthetic task. *Preprint*, arXiv:2210.13382.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2024b. Inference-time intervention: Eliciting truthful answers from a language model. *Preprint*, arXiv:2306.03341.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023. Contrastive decoding: Open-ended text generation as optimization. *Preprint*, arXiv:2210.15097.

Bingshuai Liu, Chenyang Lyu, Zijun Min, Zhanyu Wang, Jinsong Su, and Longyue Wang. 2024a. Retrieval-augmented multi-modal chain-of-thoughts reasoning for large language models. *Preprint*, arXiv:2312.01714.

Jiayu Liu, Zhenya Huang, Anya Sims, Enhong Chen, Yee Whye Teh, and Ning Miao. 2025a. Marcos: Deep thinking by markov chain of continuous thoughts. *Preprint*, arXiv:2509.25020.

Sheng Liu, Tianlang Chen, Pan Lu, Haotian Ye, Yizheng Chen, Lei Xing, and James Zou. 2025b. Fractional reasoning via latent steering vectors improves inference time compute. *Preprint*, arXiv:2506.15882.

Tengxiao Liu, Qipeng Guo, Xiangkun Hu, Cheng Jiayang, Yue Zhang, Xipeng Qiu, and Zheng Zhang. 2024b. Can language models learn to skip steps? *Preprint*, arXiv:2411.01855.

Wenquan Lu, Yuechuan Yang, Kyle Lee, Yanshu Li, and Enqi Liu. 2025. Latent chain-of-thought? decoding the depth-recurrent transformer. *Preprint*, arXiv:2507.02199.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2023. Locating and editing factual associations in gpt. *Preprint*, arXiv:2202.05262.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? *Preprint*, arXiv:2103.07191.

Nikunj Saunshi, Nishanth Dikkala, Zhiyuan Li, Sanjiv Kumar, and Sashank J. Reddi. 2025. Reasoning with latent thoughts: On the power of looped transformers. *Preprint*, arXiv:2502.17416.

Xuan Shen, Yizhou Wang, Xiangxi Shi, Yanzhi Wang, Pu Zhao, and Jiuxiang Gu. 2025a. Efficient reasoning with hidden thinking. *Preprint*, arXiv:2501.19201.

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. 2025b. Codi: Compressing chain-of-thought into continuous space via self-distillation. *Preprint*, arXiv:2502.21074.

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

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2024. Steering language models with activation engineering. *Preprint*, arXiv:2308.10248.

Jiaqi Wang, Binquan Ji, Haibo Luo, Yiyang Qi, Ruiting Li, Huiyan Wang, Yuantao Han, Cangyi Yang, jiaxu Zhang, and Feiliang Ren. 2025a. Lta-thinker: Latent thought-augmented training framework for large language models on complex reasoning. *Preprint*, arXiv:2509.12875.

Xiaoqiang Wang, Suyuchen Wang, Yun Zhu, and Bang Liu. 2025b. System-1.5 reasoning: Traversal in language and latent spaces with dynamic shortcuts. *Preprint*, arXiv:2505.18962.

Zhaoyang Wang, Jinqi Jiang, Tian Qiu, Hui Liu, Xianfeng Tang, and Huaxiu Yao. 2025c. Efficient long cot reasoning in small language models. *Preprint*, arXiv:2505.18440.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Xilin Wei, Xiaoran Liu, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Jiaqi Wang, Xipeng Qiu, and Dahua Lin. 2025. Sim-cot: Supervised implicit chain-of-thought. *Preprint*, arXiv:2509.20317.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. 2024. Reft: Representation finetuning for language models. *Preprint*, arXiv:2404.03592.

Hanqi Yan, Yanzheng Xiang, Guangyi Chen, Yifei Wang, Lin Gui, and Yulan He. 2024. Encourage or inhibit monosemanticity? revisit monosemanticity from a feature decorrelation perspective. *Preprint*, arXiv:2406.17969.

Yijiong Yu. 2025. Do llms really think step-by-step in implicit reasoning? *Preprint*, arXiv:2411.15862.

Eric Zelikman, Georges Harik, Yijia Shao, Varuna Jayasiri, Nick Haber, and Noah D. Goodman. 2024. Quiet-star: Language models can teach themselves to think before speaking. *Preprint*, arXiv:2403.09629.

Boyi Zeng, He Li, Shixiang Song, Yixuan Wang, Ziwei He, Xinbing Wang, and Zhouhan Lin. 2025. Ponderlm-2: Pretraining llm with latent thoughts in continuous space. *Preprint*, arXiv:2509.23184.

Guibin Zhang, Fanci Meng, Guancheng Wan, Zherui Li, Kun Wang, Zhenfei Yin, Lei Bai, and Shuicheng Yan. 2025a. Latentevolve: Self-evolving test-time scaling in latent space. *Preprint*, arXiv:2509.24771.

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

Hanlin Zhu, Shibo Hao, Zhiting Hu, Jiantao Jiao, Stuart Russell, and Yuandong Tian. 2025. Reasoning by superposition: A theoretical perspective on chain of continuous thought. *Preprint*, arXiv:2505.12514.

A Detailed Experimental Setup

A.1 Model Configurations and Training

We conduct our experiments using the CoConut (Hao et al., 2024) and CODI (Shen et al., 2025b) paradigms. To ensure a fair comparison and reproducibility, we strictly adhere to the model architectures and training protocols specified in their respective official repositories[1], with specific adjustments for model scaling.

Hyperparameters. For all experiments, we maintain the latent span length at $K = 6$. We use the AdamW optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.95$. The specific learning rate schedules differ by model size to ensure stability: For 8B Models (Qwen3-8B, Llama-3.1-8B), We use a peak learning rate of 1×10^{-5} with cosine annealing schedule. For Llama-3.2-3B, we use a peak learning rate of 2×10^{-5}. The training batch size is set to 16.

All other hyperparameters, including warm-up steps, epochs, and distillation loss weights for CODI, are kept the same as the default settings in the original papers.

A.2 Compute Resources

All training and inference experiments were conducted on a cluster equipped with 8 NVIDIA A100 (80GB) GPUs. Under this hardware configuration, training the latent reasoning models (both CoConut and CODI variants) for the 3B and 8B models required approximately 24 hours.

B Details of Interpretability Probes

B.1 Probe Architectures and Training

To analyze the semantic content of latent vectors (Section 3.2), we trained specific probe networks.

Dataset Construction. We constructed a probing dataset using the GSM8K training set. We randomly sampled 1,000 instances and generated both the latent thought sequences and the ground-truth explicit Chain-of-Thought (CoT) paths. This resulted in a paired dataset of aligned latent vectors and text representations.

Linear Mapper (f_ϕ). The mapper is designed to test linear recoverability. It is a simple transformation $f_\phi : \mathbb{R}^d \to \mathbb{R}^d$, where d is the model's hidden dimension. It is trained to minimize the

[1]We refer to the official implementations of CoConut and CODI.

Intervention Method	Hyperparameter	Search Range	Optimal	Acc. (%)
Baseline (No Interv.)	–	–	–	50.4
A. Semantic Structure				
Mapper-Guided	Slerp factor α	$[0.05, 0.25]$	0.15	51.9
	Norm mix η	fixed	0.25	
B. Causal Hub Editing				
Anchored Slerp (B.1)	Slerp factor α	$[0.05, 0.25]$	0.10	51.6
Gradient Edit (B.2)	Step size η	$[0.05, 0.25]$	0.20	**52.2**
C. Geometric Priors				
WT-Proj (C.1)	Mix ratio α	$[0.05, 0.25]$	0.20	51.0
	Temp. τ	fixed	1.0	
Energy Descent (C.2)	Step size η	[1e-3, 5e-3]	2e-3	51.4
	Trust region ρ	fixed	0.25	

Table 6: **Hyperparameter sensitivity analysis.** Grid search results on the Qwen3-8B (CoConut) model using the GSM8K validation set. We identify optimal configurations for each intervention family.

cosine distance between the mapped latent vector and the corresponding CoT hidden state.

Energy Function (H). The monotonicity energy function $H(\cdot)$ is parameterized as a two-layer Multi-Layer Perceptron (MLP). The architecture consists of:

$$\text{Input} \xrightarrow{W_1} \text{Hidden} \xrightarrow{\text{ReLU}} \text{Hidden} \xrightarrow{W_2} \text{Output} \tag{17}$$

The hidden dimension is set to equal the model's embedding dimension. We train this network using the margin ranking loss described in Eq. 11 to assign lower energy values to latent states closer to the solution.

B.2 Details on Structural Probes (CKA)

For the Centered Kernel Alignment (CKA) analysis, we employed Linear CKA to measure the similarity between the geometry of the latent space and the explicit CoT space. We computed the similarity matrix using a batch size of 64 reasoning steps sampled from the probing dataset, ensuring a robust estimation of the structural correspondence.

C Intervention Implementation Details

C.1 Hyperparameter Sensitivity and Selection

Our interventions involve specific hyperparameters that control the strength of the edit (e.g., interpolation factor α, step size η). We performed a grid search to identify optimal values on a held-out validation set.

Table 6 presents the sensitivity analysis for the Qwen3-8B model trained with CoConut on the GSM8K In-domain task. The "Optimal" column corresponds to the results reported in the main paper. Crucially, the optimal hyperparameters identified here were kept fixed and applied consistently across all other model scales (Llama-3.1-8B, Llama-3.2-3B) and datasets (OOD, StrategyQA) to demonstrate the robustness and transferability of our approach.

C.2 Retrieval Mechanism for Causal Editing

For the Answer-Anchored Slerp intervention (Eq. 13), we require a target vector $\mathbf{t}$ that represents a correct reasoning direction. We utilize the Latent LLM (CoConut or CODI) itself to encode the inputs. We extract the final hidden state of the input prompt to serve as the query vector, then compute the cosine similarity with the pre-cached latent vectors of the training exemplars. The top-1 retrieved instance ($k = 1$) is selected, and its final latent thought vector is used as the anchor $\mathbf{t}$ to verify solution path within latent space.

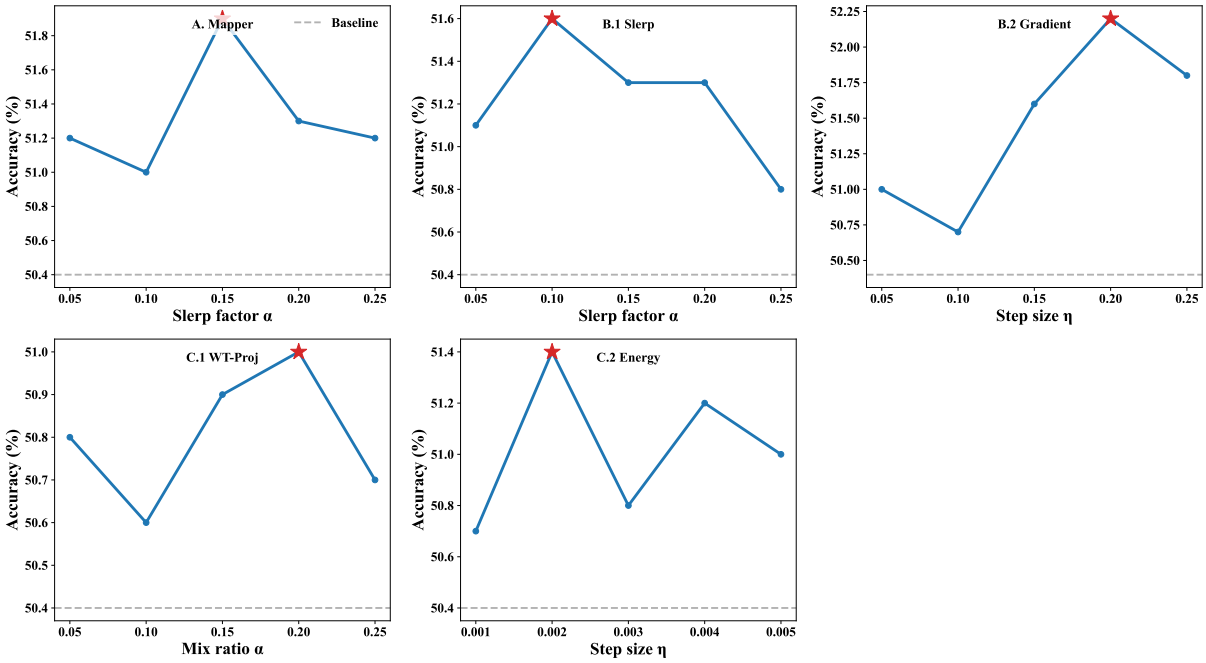


Figure 4: **Hyperparameter sensitivity analysis.** Each subplot varies one specific hyperparameter. The red stars indicate the optimal configurations. The baselines represent performance without intervention.

D Additional Experimental Results

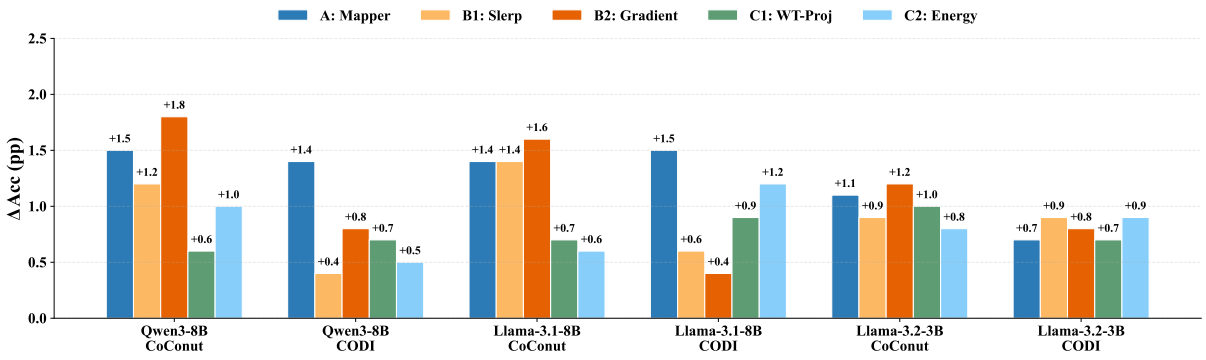
In this section, we provide supplementary visualizations to further substantiate our findings.

D.1 Hyperparameter Sensitivity

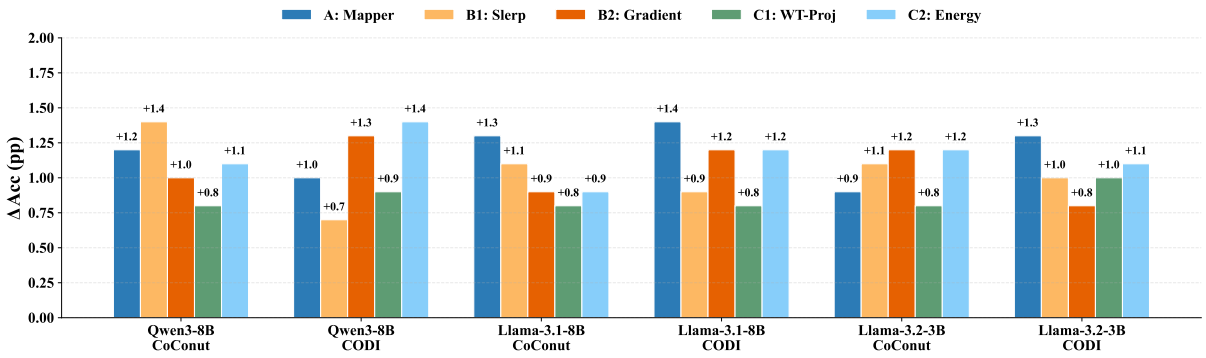
Figure 4 illustrates the impact of varying hyperparameter values on the in-domain GSM8K performance.

D.2 Unified Performance Analysis

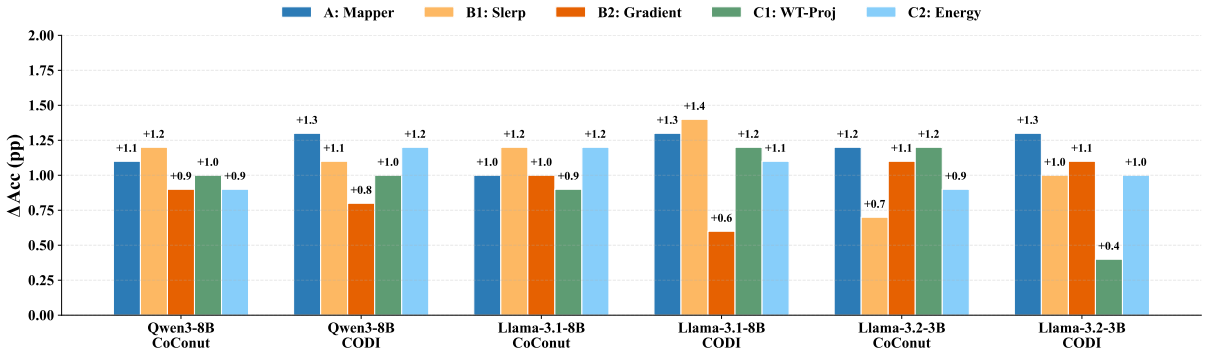
We visualize the accuracy gains (Δ Accuracy) across all tasks. Figure 5 details the improvements on mathematical reasoning benchmarks, separated by domain. Figure 6 presents the results for commonsense reasoning.



(a) In-Domain: GSM8K



(b) Out-of-Domain: GSM-Hard



(c) Out-of-Domain: SVAMP

Figure 5: **Unified Accuracy Improvements on Mathematical Reasoning.** Detailed breakdown of performance gains on (a) GSM8K, (b) GSM-Hard, and (c) SVAMP. By visualizing each dataset independently, we observe that Method B.2 (Gradient) excels in-domain, while Method A (Mapper) shows consistent robustness across OOD tasks.

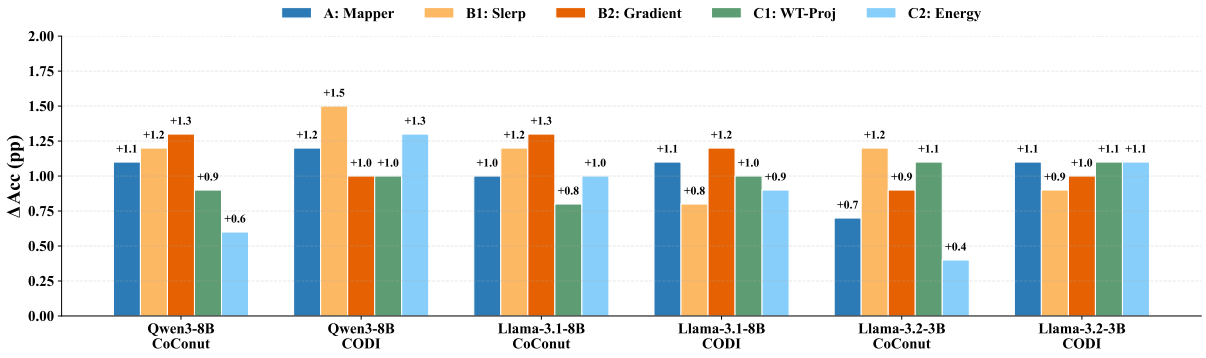


Figure 6: **Accuracy Improvements on StrategyQA (Commonsense Reasoning).** This visualization confirms that our interventions generalize effectively beyond mathematics, with consistent gains across model scales.