Think Wider: Mitigating Latent Rank Collapse in Implicit Chain-of-Thought Reasoning

Yuwen Hao **Menglin Yang***
The Hong Kong University of Science and Technology (Guangzhou)
yhao481@connect.hkust-gz.edu.cn menglinyang@hkust-gz.edu.cn

Abstract

Chain-of-thought (CoT) reasoning improves the reasoning ability of large language models by introducing intermediate computation, but explicit rationales increase decoding length, latency, and context cost. Implicit CoT offers a more efficient alternative by moving intermediate reasoning into continuous latent states. However, latent reasoning can be unstable: successive latent states may become overly similar and collapse toward a shared dominant direction, reducing the diversity of the reasoning trajectory. In this work, we identify *latent rank collapse* and propose **WIDER**, a lightweight spectral regularizer for implicit CoT. During training, WIDER estimates the shared direction of each latent trajectory and penalizes projections onto this direction, encouraging latent states to span a broader representational subspace. The method is plug-and-play and leaves the backbone model, latent schedule, and inference-time decoding procedure unchanged. We further formulate this collapse as a geometric bottleneck in implicit reasoning, casting its mitigation as a training-time regularization problem rather than an inference-time decoding change. Extensive experiments show that WIDER improves matched implicit CoT baselines, while mechanistic analyses reveal higher effective rank, lower dominant-direction energy, and reduced redundancy among latent steps. These results highlight latent subspace utilization as an important factor for efficient continuous reasoning, providing a geometric perspective for analyzing and improving implicit CoT. Code is available at https://github.com/whitesweater/WIDER.

1 Introduction

Chain-of-thought (CoT) prompting is a powerful but costly interface for eliciting reasoning in large language models (LLMs). Prior work shows that explicit intermediate steps, from few-shot and zero-shot CoT to self-consistency and least-to-most reasoning, improve arithmetic, commonsense, symbolic, and compositional reasoning (Wei et al., 2022; Kojima et al., 2022; Zhou et al., 2022; Wang et al., 2022; Merrill and Sabharwal, 2024). Yet explicit CoT externalizes computation as natural-language tokens, increasing generated tokens, latency, context usage, and deployment cost (He et al., 2026; Wang et al., 2025), while constraining reasoning to the vocabulary space and mixing computation with communicative or textual-coherence tokens (Hao et al., 2024; Xu et al., 2025a; Shen et al., 2025; Wei et al., 2025). These limitations have motivated implicit reasoning, where intermediate computation is kept silent or latent rather than fully verbalized (Li et al., 2025).

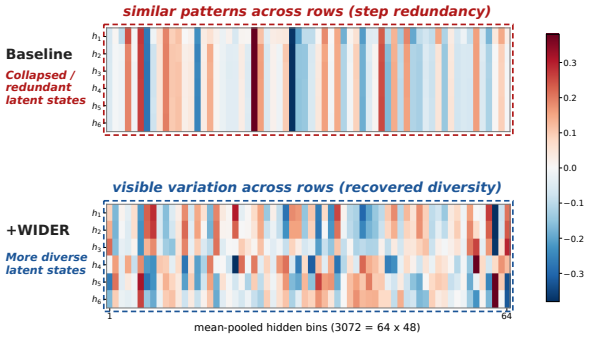


Figure 1: Latent rank collapse in implicit CoT. Each panel visualizes one latent trajectory: rows are six latent steps, and columns are mean-pooled hidden dimensions. The baseline (**top**) repeats similar patterns across steps, yielding a low-rank trajectory (effective rank 1.49). WIDER regularization (**bottom**) suppresses the shared dominant mode and restores stepwise variation (effective rank 3.81).

Implicit CoT addresses this tradeoff by preserving useful intermediate computation while avoiding full rationale decoding. Recent work in this direction can be roughly grouped into three categories according to how the latent computation is

* Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 37936–37951
October 24-29, 2026 ©2026 Association for Computational Linguistics

introduced. First, pause, filler, and thought-control token methods provide additional internal computation budget without exposing all intermediate reasoning as final text (Goyal et al., 2024; Zelikman et al., 2024). Second, continuous-thought and compressed-thought methods allow hidden states to directly participate in subsequent reasoning (Hao et al., 2024; Cheng and Van Durme, 2024; Shen et al., 2025). Third, distillation, assistant-model, and auxiliary-supervision methods compress the capability of explicit CoT into compact latent trajectories (Xu et al., 2025a,b; Wang et al., 2025; He et al., 2026; Wei et al., 2025).

Despite this progress, implicit CoT still has a reliability gap: useful latent computation is not necessarily organized into a stable trajectory. Although prior studies show that implicit or continuous reasoning can match or surpass explicit CoT in some settings, latent states may also suffer from training instability, semantic homogenization, and diversity collapse, becoming too similar to encode distinct reasoning steps (Shen et al., 2025; Xu et al., 2025a; Wang et al., 2025; He et al., 2026; Wei et al., 2025). **This motivates the central question: do multiple latent steps span distinct computational directions, or do they collapse toward a shared mode?**

Representation collapse provides a concise lens for this instability: latent states can concentrate in a narrow subspace instead of using distinct computational directions. In Transformers, low-rank behavior has been linked to token uniformity, degraded signal propagation, and vanishing query/key gradients (Dong et al., 2021; Noci et al., 2022; Nait Saada et al., 2025). In representation learning, related failures are commonly diagnosed as redundancy, reduced effective rank, covariance collapse, or dimensional collapse (Jing et al., 2021; Zbontar et al., 2021; Bardes et al., 2021; Chen et al., 2026). For implicit CoT, this suggests that adding latent steps does not guarantee added computation; multiple states may instead place excessive energy on the same dominant direction. We refer to this failure mode as *latent rank collapse*.

To mitigate latent rank collapse, we introduce WIDER, a lightweight spectral regularizer that improves the geometry of implicit CoT during training while leaving inference unchanged. First, it penalizes excessive alignment with the shared dominant direction of each latent trajectory, encouraging latent states to occupy a broader representational subspace. Second, it is plug-and-play: it adds no inference-time modules, tokens, or decoding steps, and keeps the backbone, latent schedule, and decoder interface unchanged. In matched comparisons, this regularization improves reasoning accuracy and produces latent trajectories with higher effective rank, lower dominant-direction energy, and reduced redundancy, suggesting that the gain comes from better latent organization rather than surface output shifts.

Our contributions are:

- We formulate *latent rank collapse* as a geometric failure mode of implicit CoT, where multiple latent states overuse a shared dominant direction instead of providing distinct computational directions.
- We propose WIDER, a plug-in spectral regularizer that suppresses this shared direction during training, encourages a broader latent subspace, and leaves the inference procedure unchanged.
- Extensive results show that WIDER improves matched implicit CoT baselines across multiple backbones and reasoning benchmarks, while focused ASDiv-Aug analyses connect the gains to higher effective rank, lower dominant-direction energy, and reduced latent-step redundancy.

2 Related Work

Explicit chain-of-thought reasoning. Chain-of-thought (CoT) reasoning shows that large language models benefit from generating intermediate reasoning steps before final answers (Wei et al., 2022). This idea has been extended by zero-shot reasoning prompts (Kojima et al., 2022), answer aggregation over multiple sampled paths (Wang et al., 2022), and problem decomposition (Zhou et al., 2022). Other work improves or restructures explicit reasoning through rationale bootstrapping (Zelikman et al., 2022), program-aided reasoning (Chen et al., 2022), reasoning-action interaction (Yao et al., 2022), and tree- or graph-structured search over thoughts (Yao et al., 2023; Besta et al., 2024). Although these methods differ in form, they share the same principle: intermediate computation is useful for complex reasoning. Their main limitation is that such computation is usually externalized as natural-language tokens, which increases latency and context usage while forcing internal states to be verbalized. This motivates latent reasoning methods that keep part of the reasoning process in continuous space.

Implicit and latent reasoning. Recent work

moves intermediate computation into continuous hidden states, avoiding the need to verbalize every reasoning step. Coconut feeds continuous thoughts back into the model, while CCoT compresses explicit rationales into compact latent representations (Hao et al., 2024; Cheng and Van Durme, 2024). CODI transfers explicit-CoT capability into latent trajectories through self-distillation, and SoftCoT generates instance-specific soft thoughts with an auxiliary model (Shen et al., 2025; Xu et al., 2025a). Other methods use step-level supervision, semantic alignment, or adaptive latent reasoning to improve stability and efficiency (Wei et al., 2025; He et al., 2026; Wang et al., 2025). These works mainly study how latent reasoning is constructed or supervised; we instead examine the geometry of the resulting latent trajectory.

Representation collapse and spectral regularization. Representation collapse provides a natural lens for this question. Transformer and neural representations can concentrate in low-dimensional subspaces (Dong et al., 2021; Noci et al., 2022), a phenomenon also studied as dimensional collapse in self-supervised learning (Jing et al., 2021). Anti-collapse methods reduce redundancy, preserve variance, or rebalance spectra (Zbontar et al., 2021; Bardes et al., 2021; Zhang et al., 2023), while related studies connect collapse to reasoning stability, spectral gaps, and architectural rank limits (Arefin et al., 2025; Saada et al., 2024; Chen et al., 2026). We therefore regularize the singular-value structure of the per-example latent-token matrix, encouraging implicit reasoning states to use a broader subspace without changing the architecture or inference procedure.

3 Understanding Latent Rank Collapse

We formalize latent rank collapse as a per-example geometric failure mode of implicit CoT: the model allocates multiple latent steps, but their hidden states occupy nearly the same direction. For an input $\mathbf{x}$, write the latent trajectory as

$$\mathbf{H}(\mathbf{x}) = [\mathbf{h}_1(\mathbf{x}); \ldots; \mathbf{h}_m(\mathbf{x})] \in \mathbb{R}^{m \times d}, \quad (1)$$

where each row $\mathbf{h}_i(\mathbf{x}) \in \mathbb{R}^d$ is the hidden state read out at the i-th implicit reasoning position before the answer token, following prior work on latent reasoning (Shen et al., 2025). Here m is the number of latent positions and d is the model hidden size. To focus on directional diversity rather than hidden-state norms, we also use the row-normalized trajectory

$$\mathbf{z}_i(\mathbf{x}) = \frac{\mathbf{h}_i(\mathbf{x})}{\|\mathbf{h}_i(\mathbf{x})\|_2 + \varepsilon}, \qquad \mathbf{Z}(\mathbf{x}) = [\mathbf{z}_1(\mathbf{x}); \ldots; \mathbf{z}_m(\mathbf{x})]. \quad (2)$$

The central geometric question is whether multiple allocated latent positions occupy a broad set of directions or mostly repeat a shared representation. Prior work reports related symptoms in implicit or continuous reasoning, including semantic homogenization across latent steps and reduced latent diversity (Wei et al., 2025).

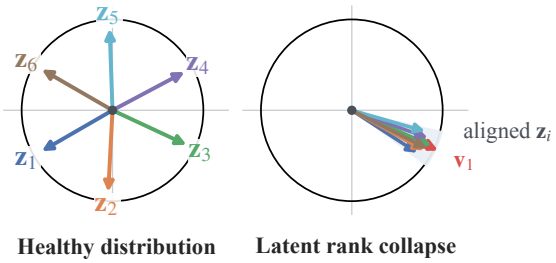


Figure 2: Conceptual illustration of latent rank collapse. Each arrow is a unit-length latent state. In the healthy distribution (**left**), states span many directions. In latent rank collapse (**right**), states concentrate near one dominant direction, so hidden states mainly differ by scalar coefficients.

Definition. We use *latent rank collapse* to describe the failure mode in which the normalized rows in $\mathbf{Z}(\mathbf{x})$ concentrate around a dominant direction $\mathbf{v}_1$. Equivalently, the unnormalized states can often be viewed as $\mathbf{h}_i(\mathbf{x}) \approx \alpha_i(\mathbf{x})\mathbf{v}_1$, so latent steps differ mostly by scale and add little new directional variation. The trajectory need not be exactly rank one; the key issue is that successive steps use only a small part of the available latent subspace.

Figure 2 illustrates this geometry: useful latent computation spreads across directions, whereas collapse makes additional steps align with a shared mode.

Why rank matters. $\mathbf{H}(\mathbf{x})$ is the channel through which the decoder receives input-specific computation from the implicit reasoning stage. If all latent steps vary mainly along one direction, the decoder receives more positions but not many independent directions of computation. The conditional family $\{p(y \mid \mathbf{x})\}_{\mathbf{x}}$ is then bottlenecked by the low-rank

coordinates carried by the latent trajectory, even when the number of latent steps m is large. Raising the effective rank of $\mathbf{H}(\mathbf{x})$ can therefore enlarge per-input conditional capacity while leaving the marginal $p(y)$ unchanged.

Proposition 1 (Subspace factorization). *Let the decoder factor through the latent trajectory as $p(y \mid \mathbf{x}) = g(\mathbf{H}(\mathbf{x}))$. If, for every $\mathbf{x} \in \mathcal{X}$, the rows of $\mathbf{H}(\mathbf{x})$ lie in a common r-dimensional subspace with basis $\mathbf{U} \in \mathbb{R}^{d \times r}$, so that $\mathbf{H}(\mathbf{x}) = \mathbf{A}(\mathbf{x})\mathbf{U}^\top$ with $\mathbf{A}(\mathbf{x}) \in \mathbb{R}^{m \times r}$, then there exists $\tilde{g} : \mathbb{R}^{m \times r} \to \Delta(\mathcal{Y})$ such that $p(y \mid \mathbf{x}) = \tilde{g}(\mathbf{A}(\mathbf{x}))$ for all $\mathbf{x}$. In particular, rank-one collapse reduces the family to a function of m scalar coefficients along one direction.*

The proof and detailed scope qualifications are given in Appendix A.1. Related links between rank and expressiveness also appear in Transformer attention and self-supervised representation learning (Merrill and Sabharwal, 2024; Dong et al., 2021; Bhojanapalli et al., 2020; Nait Saada et al., 2025; Jing et al., 2021; Bardes et al., 2021). Our experiments report shared-direction dominance and step-to-step redundancy as geometric diagnostics of within-trajectory concentration.

Why collapse can emerge. A compact local model provides one possible way in which this geometric pattern can arise. In implicit CoT, the model repeatedly updates continuous states before decoding the answer (Shen et al., 2025). If the local update around a trajectory amplifies one direction more than others, repeated applications can make successive latent states increasingly aligned, as in power iteration. We do not assume the model is globally linear; the point is to motivate measuring the dominant shared direction and penalizing it during training. Appendix A.1 gives the derivation, and Figure 1 shows the same pattern empirically: the baseline has nearly identical latent steps with effective rank 1.49, while the regularizer restores visible diversity and raises effective rank to 3.82.

4 Method

We propose WIDER, a training-time regularizer for latent rank collapse in implicit reasoning. For each example, it estimates the dominant shared direction of the latent trajectory and penalizes projections onto that direction.

The method follows the geometry in Section 3: if successive states overuse one direction, extra latent steps add little new computation. WIDER changes only the training objective; the backbone, latent schedule, and decoding procedure are unchanged.

4.1 Latent states and row normalization

For each input $\mathbf{x}$, we use the row-normalized trajectory $\mathbf{Z}(\mathbf{x})$ defined in Eq. (2). This makes the auxiliary loss depend on the direction of each latent state rather than on its absolute norm. The base training loss remains responsible for learning the answer; the auxiliary loss only controls redundancy among latent steps.

4.2 Shared-direction proxy

We estimate the shared direction of the latent trajectory by summing the normalized states:

$$\mathbf{c}(\mathbf{x}) = \mathbf{Z}(\mathbf{x})^\top \mathbf{1}_m = \sum_{i=1}^{m} \mathbf{z}_i(\mathbf{x}). \tag{3}$$

The unit proxy is then

$$\mathbf{q}(\mathbf{x}) = \mathrm{sg}\left(\frac{\mathbf{c}(\mathbf{x})}{\|\mathbf{c}(\mathbf{x})\|_2 + \varepsilon} \right), \tag{4}$$

where $\mathrm{sg}(\cdot)$ denotes stop-gradient. Intuitively, if many latent states point in a similar direction, their sum identifies that shared component. If the states are already well diversified, the summed direction is weaker and less dominant.

The proxy in Eq. (4) is not introduced as an exact top singular-vector estimator. Its role is operational: it is a cheap direction that summarizes what the current latent steps have in common. Appendix A.2 relates this column-sum direction to average pairwise alignment among latent states, and Appendix A.2 gives the corresponding spectral interpretation. Appendix A.2 further shows that under a spiked latent-step model, the mean direction and the leading right singular direction identify the same shared component, which is why this lightweight proxy can replace an SVD inside the training loop.

4.3 Projection regularization

The auxiliary loss penalizes the average squared projection onto the shared direction:

$$\begin{aligned} \mathcal{L}_{\mathrm{spec}}(\mathbf{x}) &= \frac{1}{m} \sum_{i=1}^{m} \big(\mathbf{z}_i(\mathbf{x})^\top \mathbf{q}(\mathbf{x})\big)^2 \\ &= \frac{1}{m} \|\mathbf{Z}(\mathbf{x})\mathbf{q}(\mathbf{x})\|_2^2. \end{aligned} \tag{5}$$

The final training objective is

$$\mathcal{L} = \mathcal{L}_{\text{base}} + \lambda_{\text{spec}} \mathbb{E}_{\mathbf{x}\sim\mathcal{D}} \left[\mathcal{L}_{\text{spec}}(\mathbf{x})\right]. \quad (6)$$

Here $\mathcal{L}_{\text{base}}$ is the original latent-reasoning training objective. The spectral term does not replace the task loss and does not introduce additional inference steps. It only discourages the model from using multiple latent positions to encode the same shared component.

For batch size B, latent-step count m, and hidden dimension d, the added training cost is $O(Bmd)$. At inference time, the auxiliary loss is removed, so the latent schedule and decoding procedure are unchanged.

5 Experiments

We evaluate WIDER with two implicit-reasoning methods, CODI and SIM-CoT, on two language-model backbones and four reasoning benchmarks. The empirical study is organized to answer two questions in turn: (i) does adding the spectral loss to a matched implicit baseline give a measurable accuracy gain (§5.2)? and (ii) is that gain robust to the regularization strength (§5.3)? Section 6 examines how latent geometry changes alongside the accuracy differences.

5.1 Experimental Setup

Baselines. We use four reasoning configurations as our baselines. Since WIDER is a plug-in training objective, we apply it to the two implicit-reasoning methods and compare matched configurations that share the same training data and evaluation parser; the only intended difference within each matched pair is whether the WIDER loss is added. **no-CoT** (Wei et al., 2022) is supervised to produce the final answer directly, without generating intermediate reasoning steps. **SFT-CoT** (Wei et al., 2022; Kojima et al., 2022) is supervised on CoT-annotated data, where the model generates explicit intermediate reasoning steps before the final answer. **CODI** (Shen et al., 2025) is an implicit CoT baseline that compresses explicit reasoning into continuous thought states through self-distillation, transferring reasoning ability by aligning hidden activations at a designated answer-side token. **SIM-CoT** (Shen et al., 2025; Wei et al., 2025) further adds an auxiliary decoder during training to align each implicit token with its corresponding explicit reasoning step, providing step-level latent supervision while removing the decoder at inference. The no-CoT and SFT-CoT settings are reported as contextual references. Because they belong to the explicit-output reasoning paradigm, they are not treated as ablations of the WIDER loss. The causal comparisons are always made between a matched CODI or SIM-CoT baseline and its WIDER-regularized counterpart.

Backbones. Our main experiments use two open-source LLMs. **LLaMA-3.2-3B-Instruct** (Grattafiori et al., 2024) serves as the default backbone for the main matched comparisons, the ablation, and the mechanism analysis, while **Qwen3-1.7B** (Yang et al., 2025) probes whether the spectral signal transfers across backbone families and parameter scales. We additionally evaluate **LLaMA-3.1-8B** (Grattafiori et al., 2024) in the scaling experiment reported in Appendix D. Training and adaptation configurations for the main experiments are provided in Appendix C.

Datasets. We evaluate on four reasoning benchmarks spanning three task formats, with a shared prompt template and evaluation parser per backbone. **StrategyQA** (Geva et al., 2021) is a binary yes/no commonsense benchmark. **CommonsenseQA** (Talmor et al., 2019) is a five-way multiple-choice commonsense benchmark from ConceptNet. **ASDiv-Aug** (Miao et al., 2020) is an arithmetic word-problem benchmark augmented along the standard CODI/Coconut protocol; its clean numerical answer space makes it our focused mechanism-analysis setting. **AQuA** (Ling et al., 2017) is a five-way multiple-choice algebraic word-problem benchmark. Together, these benchmarks cover binary, multiple-choice, and open-numeric answer formats, allowing us to examine whether the effect of WIDER varies with task format in addition to backbone and decoder supervision.

5.2 Main Results

Table 1 reports the calibrated result matrix across backbones, implicit-reasoning variants, and benchmarks. The no-CoT and SFT-CoT rows are included as reference points for explicit-output reasoning; the causal comparisons are the matched CODI/SIM-CoT pairs with and without WIDER. We highlight three observations.

Matched averages improve in every implicit setting. Across the four backbone–method combinations (LLaMA-3B/Qwen3-1.7B $\times$ CODI/SIM-CoT), adding WIDER raises average accuracy by

Backbone	Method	StrategyQA		CommonsenseQA		ASDiv-Aug		AQuA		Avg.
		Acc	tok	Acc	tok	Acc	tok	Acc	tok	Acc
LLaMA-3B	no-CoT	68.01	3.00	79.12	3.00	88.73	4.07	49.61	3.00	71.37
	SFT-CoT	78.68	125.10	77.23	80.61	93.06	12.81	47.66	75.58	74.16
	CODI	75.74	6.00	78.30	6.00	80.73	7.06	28.74	6.00	65.88
	CODI + WIDER	78.31 (+3.39%)	6.00	78.95 (+0.83%)	6.00	82.47 (+2.16%)	7.07	33.46 (+16.42%)	6.00	68.30 (+3.67%)
	SIM-CoT	73.53	6.00	79.61	6.00	80.06	7.06	27.17	5.00	65.09
	SIM-CoT + WIDER	75.00 (+2.00%)	6.00	79.12 (-0.62%)	6.04	82.27 (+2.76%)	7.06	28.74 (+5.78%)	6.00	66.28 (+1.83%)
Qwen3-1.7B	no-CoT	61.40	3.00	73.63	3.00	85.07	5.08	44.92	3.00	66.26
	SFT-CoT	69.85	126.08	72.40	79.45	91.71	17.45	47.27	78.42	70.31
	CODI	66.54	6.00	67.98	6.00	77.55	8.16	37.01	6.00	62.27
	CODI + WIDER	68.38 (+2.77%)	6.00	69.37 (+2.04%)	6.00	77.94 (+0.50%)	8.16	42.13 (+13.83%)	9.41	64.46 (+3.52%)
	SIM-CoT	63.97	6.00	67.90	6.01	72.35	8.15	41.73	6.00	61.49
	SIM-CoT + WIDER	64.71 (+1.16%)	6.00	68.39 (+0.72%)	6.00	79.19 (+9.45%)	8.11	44.49 (+6.61%)	6.00	64.20 (+4.40%)

Table 1: Main results across backbones, methods, and reasoning benchmarks. Each task is shown as two side-by-side sub-columns: *Acc* (accuracy in %) and *tok* (the per-cell average generation length). *Avg.* is the unweighted mean of the four task accuracies. The first two rows of each backbone are explicit-reasoning references (no-CoT and SFT-CoT). Parentheses on the four WIDER-regularized rows give the relative improvement of the WIDER variant over its matched CODI / SIM-CoT baseline in the same backbone (computed as $(v_{\text{spec}} - v_{\text{base}})/v_{\text{base}}$).

$+3.67\%$, $+1.83\%$, $+3.52\%$, and $+4.40\%$ relative to the corresponding matched baseline. This suggests that the signal is not tied to a single backbone family or to one particular implicit-reasoning interface.

The gains are broad. At the task-cell level, fifteen of the sixteen matched comparisons improve. The single exception is small: LLaMA-3B SIM-CoT on CommonsenseQA decreases by 0.49 accuracy points. We therefore treat WIDER as a broadly useful plug-in objective.

The largest point gains occur on the arithmetic benchmarks. The biggest single-cell gain is Qwen SIM-CoT on ASDiv-Aug ($72.35 \rightarrow 79.19$, $+6.84$ points), followed by Qwen CODI on AQuA ($37.01 \rightarrow 42.13$, $+5.12$ points) and LLaMA CODI on AQuA ($28.74 \rightarrow 33.46$, $+4.72$ points).

Scaling to 8B. We additionally evaluate WIDER on LLaMA-3.1-8B with CODI. WIDER improves accuracy on all four benchmarks , with gains of $+2.57$, $+0.32$, $+4.05$, and $+1.18$ percentage points on StrategyQA, CommonsenseQA, ASDiv-Aug, and AQuA, respectively. Effective rank also increases consistently, from 1.715–2.092 under CODI to 3.999–5.523 with WIDER. Full results are reported in Appendix D.

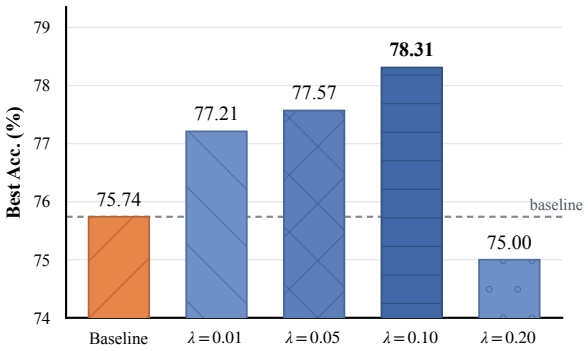


Figure 3: Sensitivity to the regularization coefficient λ on LLaMA-3B + CODI, StrategyQA. The dashed line marks the baseline ($\lambda = 0$); the peak at $\lambda = 0.10$ is highlighted.

5.3 Ablation Experiments

We sweep the spectral coefficient λ on LLaMA-3B + CODI on StrategyQA to verify that the gain is not an artifact of a single hyperparameter choice and to locate the operating regime. Figure 3 reports the best calibrated accuracy at each value.

Accuracy improves monotonically as λ grows from 0.01 to 0.10 and then drops at $\lambda = 0.20$, where the spectral penalty starts to over-regularize the latent trajectory. The peak at $\lambda = 0.10$ is the value used for the matched comparisons in Table 1; the curve is smooth on both sides of the peak, in-

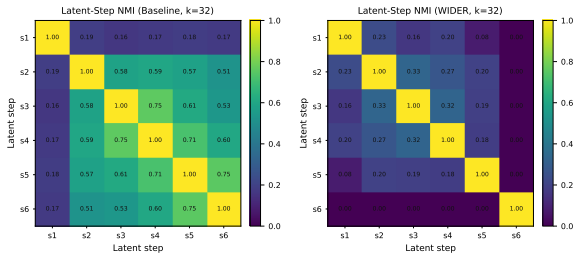


Figure 4: Latent-step NMI heatmaps on ASDiv-Aug (SIM-CoT, k=32). The diagonal is fixed at 1.0; signal lies in the off-diagonals. The baseline (left) shows a dense high-dependence block among later steps; WIDER (right) sharply reduces this block, leaving s_6 nearly independent of earlier states.

Codebook	Baseline	+WIDER	Δ	Ratio
k=16	0.306	0.142	−0.164	0.46
k=32	0.472	0.144	−0.328	0.31

Table 2: Mean pairwise latent-step NMI on ASDiv-Aug (SIM-CoT). WIDER roughly halves NMI at k=16 and cuts it by ∼70% at k=32.

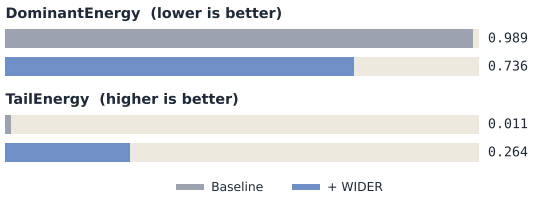


Figure 5: Dominant and tail energy on ASDiv-Aug (SIM-CoT, LLaMA-3B). WIDER redistributes energy from the first singular direction into the spectral tail.

dicating that the gain is robust to moderate perturbations of λ rather than tuned to a knife-edge value.

6 Mechanistic Analysis

The main results show that WIDER improves the matched baselines; this section asks *why*. Unless otherwise specified, the analysis in this section uses ASDiv-Aug with LLaMA-3B + SIM-CoT. We organize the analysis around four candidate explanations — token-level repetition, dominant-direction shortcut, shortcut content, and a shifted answer prior — and ask which of them, if any, the spectral loss actually moves.

6.1 Are Latent Tokens Repetitive?

The most direct failure mode is token-level repetition: later latent steps reproduce earlier states. We measure pairwise dependence between the six latent steps with normalized mutual information (NMI). Latent vectors are standardized, PCA-projected, and discretized with a shared MiniBatchKMeans codebook of size k; NMI is then computed between latent-step cluster assignments. Higher off-diagonal NMI means stronger repetition.

Figure 4 shows the qualitative shift, and Table 2 the averages: mean pairwise NMI drops from 0.472 to 0.144 at k=32, and from 0.306 to 0.142 at k=16. The baseline therefore spends part of its latent-token budget revisiting earlier states, and WIDER reclaims that budget by making the steps carry distinguishable cluster assignments. A natural next question is whether those now-distinguishable steps actually span independent directions of the latent space, or merely re-weight a single shared one.

6.2 Does WIDER Release the Trajectory from a Dominant Shortcut?

NMI tests pairwise step repetition; a complementary failure is that all six steps, while pairwise distinct, still concentrate their energy on a single shortcut direction. We track this with $\mathrm{DominantEnergy} = \sigma_1^2 / \sum_i \sigma_i^2$ and $\mathrm{TailEnergy} = 1 - \mathrm{DominantEnergy}$ over the latent trajectory.

Figure 5 shows the baseline is effectively rank-one: the top direction carries 0.989 of the energy, leaving only 0.011 in the tail. WIDER moves the split to 0.736/0.264 — the latent trajectory now uses non-dominant directions in earnest rather than routing all computation through one channel.

Where the gain accrues across latent steps. The same redistribution shows up in effective rank, which rises from 1.487 to 3.817. To localize *which* steps gain, we compute prefix effective rank — the effective rank of the first t states as t grows from 1 to 6. Figure 6 shows the baseline saturates near 1.5 after the first few steps, while WIDER keeps accumulating rank through s_6, so later steps contribute new directions instead of re-tracing the early trajectory. The fact that the rank gain accrues at later steps, rather than being a uniform offset, is consistent with collapse being a step-wise drift onto the dominant direction that WIDER resists from s_2 onward.

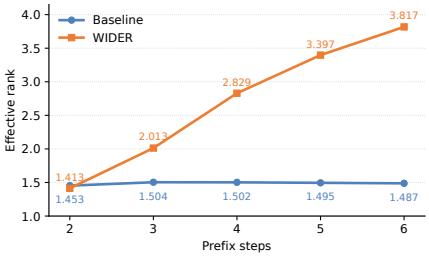


Figure 6: Prefix effective rank on ASDiv-Aug (SIM-CoT), over the first six latent states.

Setting	Shortcut ↓	Content ↑
Baseline	16.03%	83.97%
+WIDER	0.00%	100.00%

Table 3: Dominant-direction vocabulary projection on ASDiv-Aug (SIM-CoT, LLaMA-3B). Shortcut share is the complement of content share, i.e. everything that is not a content-like token.

6.3 What Is Encoded in the Dominant Direction?

Reducing dominant-direction concentration is only meaningful if that direction carries low-value signal in the first place. To probe its content, we project the leading right singular vector of each latent trajectory through the language-model head, take the top vocabulary items (using absolute scores, since singular-vector sign is arbitrary), and split them into two buckets: *content-like* tokens and everything else (punctuation, function words, numeric literals, and whitespace/special tokens). We report $\mathrm{ContentShare}$ together with $\mathrm{ShortcutShare} = 1 - \mathrm{ContentShare}$.

Table 3 shows the baseline dominant direction is 16.03% shortcut-like; after WIDER this drops to 0.00%, with content share rising correspondingly. So WIDER does not merely weaken the dominant direction in magnitude; it also clears away the shortcut-flavored vocabulary items aligned with it. We treat this as a diagnostic rather than a causal statement, but it is consistent with the broader picture from §6.2. The full categorization protocol is given in Appendix E.

6.4 Is the Gain a Shifted Answer Prior?

A final alternative is that WIDER simply reshapes the marginal answer distribution $p(y)$ over the test set — for instance by flattening a few high-frequency answers — and that accuracy moves as a side effect.

Figure 7 rules this out: top-1 prediction share, normalized entropy, unique-prediction rate, and prediction–gold cosine all move by less than one percentage point in absolute terms and are all smaller than the accuracy gap. The marginal $p(y)$ is essentially preserved, while per-example accuracy rises — so the gain must come from a more input-discriminative latent trajectory rather than from a redistributed answer prior.

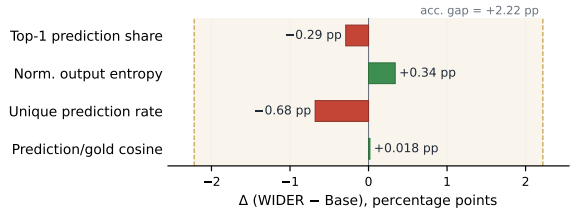


Figure 7: Output-distribution shifts on ASDiv-Aug (SIM-CoT). All four marginal-$p(y)$ metrics move by less than one percentage point and are all smaller than the $+2.22$ pp accuracy gap (dashed reference band).

A three-way consistency argument. Marginal $p(y)$ unchanged, per-example accuracy up, and effective rank from 1.487 to 3.817 together admit one reading: the gain is a drop in average $-\log p(y^\star \mid \mathbf{x})$ enabled by a higher-rank, more input-discriminative $\mathbf{H}(\mathbf{x})$, not a reshuffled prior. This matches the pattern in §5.2: the larger gains appear on AQuA and ASDiv-Aug, whose answer space most stresses the conditional bottleneck, while binary StrategyQA — where one bit of input-specific signal often suffices — shows smaller deltas.

7 Conclusion

Implicit chain-of-thought moves intermediate computation into continuous latent states, but successive states can collapse toward a shared direction, leaving additional latent positions underused. We identified this as *latent rank collapse* — a per-example geometric failure that bottlenecks $\{p(y \mid \mathbf{x})\}$ by the rank of the latent trajectory — and proposed WIDER, a training-time spectral regularizer that penalizes projections onto this shared direction. Across multiple backbones and reasoning benchmarks, WIDER delivers consistent accuracy gains over matched implicit-CoT baselines. These results highlight latent subspace utilization as a key factor for efficient implicit reasoning, and offer a geometric perspective for analyzing and improving implicit chain-of-thought.

Limitations

This study has several limitations. First, our experiments evaluate models up to 8B parameters across a limited set of reasoning benchmarks. Whether the observed benefits persist for substantially larger models and larger-scale training regimes remains to be explored. Second, our current experiments use a fixed trajectory of six latent tokens. Investigating how latent rank collapse and WIDER behave with longer or dynamically allocated latent trajectories is an interesting direction for future work. Third, because WIDER is designed as a plug-in objective for latent reasoning, it inherits the interpretability limitations of latent reasoning itself. Although our geometric diagnostics characterize the organization of the latent trajectory, they do not make individual latent tokens directly human-interpretable or identify the specific reasoning role performed by each token.

Ethics Statement

This paper presents work whose goal is to advance the field of Machine Learning by addressing *latent rank collapse* in implicit chain-of-thought reasoning. The proposed spectral regularizer is a training-time mechanism that encourages latent trajectories to span a broader subspace, and it leaves the backbone architecture and inference procedure unchanged.

Because the method adds no extra tokens or modules at inference, it does not increase deployment-time compute, latency, or energy use relative to the implicit-CoT baseline. However, it targets only the geometry of latent reasoning and does not by itself address factual correctness, social bias, or harmful content; models trained with it may still produce incorrect or biased answers. All datasets used are publicly available reasoning benchmarks and involve no human subjects or personally identifiable information.

Acknowledgments

We thank the anonymous reviewers and area chairs for their constructive feedback. This work was partially supported by the Guangdong Basic and Applied Basic Research Foundation (Grant No. 2025A1515140110) and the Research Travel Grant from the Base of Red Bird MPhil (RBM) at The Hong Kong University of Science and Technology (Guangzhou). We would like to thank our RBM Project Supervisor Lanhui Li for academic support.

References

Md Rifat Arefin, Gopeshh Raaj Subbaraj, Nicolas Gontier, Yann LeCun, Irina Rish, Ravid Shwartz-Ziv, and Christopher Pal. 2025. Seq-vcr: Preventing collapse in intermediate transformer representations for enhanced reasoning. In *International Conference on Learning Representations*, volume 2025, pages 72939–72960.

Adrien Bardes, Jean Ponce, and Yann LeCun. 2021. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and 1 others. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 17682–17690.

Srinadh Bhojanapalli, Chulhee Yun, Ankit Singh Rawat, Sashank Reddi, and Sanjiv Kumar. 2020. Low-rank bottleneck in multi-head attention models. In *International conference on machine learning*, pages 864–873. PMLR.

Jin Chen, Shangyu Zhang, Bin Hu, Chao Zhou, Junwei Pan, Gengsheng Xue, Wentao Ning, Gengyu Weng, Wang Zheng, Shaohua Liu, and 1 others. 2026. Rankup: Towards high-rank representations for large scale advertising recommender systems. *arXiv preprint arXiv:2604.17878*.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W Cohen. 2022. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*.

Jeffrey Cheng and Benjamin Van Durme. 2024. Compressed chain of thought: Efficient reasoning through dense representations. *arXiv preprint arXiv:2412.13171*.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. 2021. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International conference on machine learning*, pages 2793–2803. PMLR.

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. *Transactions of the Association for Computational Linguistics*, 9:346–361.

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. In *International Conference on Learning Representations*, volume 2024, pages 27896–27923.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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. *arXiv preprint arXiv:2412.06769*.

Yinhan He, Wendy Zheng, Yaochen Zhu, Zaiyi Zheng, Lin Su, Sriram Vasudevan, Qi Guo, Liangjie Hong, and Jundong Li. 2026. Semcot: Accelerating chain-of-thought reasoning through semantically-aligned implicit tokens. *Advances in Neural Information Processing Systems*, 38:43455–43485.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. 2021. Understanding dimensional collapse in contrastive self-supervised learning. *arXiv preprint arXiv:2110.09348*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. 35:22199–22213.

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

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 158–167.

William Merrill and Ashish Sabharwal. 2024. The expressive power of transformers with chain of thought. In *International Conference on Learning Representations*, volume 2024, pages 7690–7706.

Shen-Yun Miao, Chao-Chun Liang, and Keh-Yih Su. 2020. A diverse corpus for evaluating and developing english math word problem solvers. In *Proceedings of the 58th annual meeting of the Association for Computational Linguistics*, pages 975–984.

Thiziri Nait Saada, Alireza Naderi, and Jared Tanner. 2025. Mind the gap: a spectral analysis of rank collapse and signal propagation in attention layers. In *Proceedings of the 42nd International Conference on Machine Learning*, pages 45561–45587. PMLR.

Lorenzo Noci, Sotiris Anagnostidis, Luca Biggio, Antonio Orvieto, Sidak Pal Singh, and Aurelien Lucchi. 2022. Signal propagation in transformers: Theoretical perspectives and the role of rank collapse. 35:27198–27211.

Thiziri Nait Saada, Alireza Naderi, and Jared Tanner. 2024. Mind the gap: a spectral analysis of rank collapse and signal propagation in attention layers. *arXiv preprint arXiv:2410.07799*.

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. 2025. Codi: Compressing chain-of-thought into continuous space via self-distillation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 677–693.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158.

Jianwei Wang, Ziming Wu, Fuming Lai, Shaobing Lian, and Ziqian Zeng. 2025. Synadapt: Learning adaptive reasoning in large language models via synthetic continuous chain-of-thought. *arXiv preprint arXiv:2508.00574*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. 35:24824–24837.

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. *arXiv preprint arXiv:2509.20317*.

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. 2025a. Softcot: Soft chain-of-thought for efficient reasoning with llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23336–23351.

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. 2025b. Softcot++: Test-time scaling with soft chain-of-thought reasoning. *arXiv preprint arXiv:2505.11484*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. 2021. Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning*, pages 12310–12320. PMLR.

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. *arXiv preprint arXiv:2403.09629*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. Star: Bootstrapping reasoning with reasoning. 35:15476–15488.

Yifei Zhang, Hao Zhu, Zixing Song, Piotr Koniusz, and Irwin King. 2023. Spectral feature augmentation for graph contrastive learning and beyond. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 11289–11297.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. 2022. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

A Additional Mathematical Details

This appendix collects the short derivations and scope qualifications behind the geometric interpretations in the main text. Section A.1 gives the common-subspace factorization and a local power-iteration illustration used in Section 3; Section A.2 justifies the mean-direction proxy and the projection loss in Section 4. None of the arguments assumes that the full nonlinear model is globally linear.

A.1 Common-subspace factorization and a local concentration scenario

Common-subspace factorization. Proposition 1 in the main text states an exact reduced-coordinate factorization under a single fixed subspace shared across inputs. In the rank-one setting $\mathbf{h}_i(\mathbf{x}) = \alpha_i(\mathbf{x})\mathbf{v}_1$, the trajectory matrix becomes $\mathbf{H}(\mathbf{x}) = \boldsymbol{\alpha}(\mathbf{x})\mathbf{v}_1^\top$ and is represented relative to the fixed direction $\mathbf{v}_1$ by the m scalar coordinates $\boldsymbol{\alpha}(\mathbf{x})$, regardless of how large d is.

Proof of Proposition 1. $\mathbf{U}$ is fixed, so $\mathbf{H}(\mathbf{x})$ is determined by $\mathbf{A}(\mathbf{x})$ via $\mathbf{H}(\mathbf{x}) = \mathbf{A}(\mathbf{x})\mathbf{U}^\top$; take $\tilde{g}(\mathbf{A}) = g(\mathbf{A}\mathbf{U}^\top)$. □

Effect of row normalization. With the ε-stabilized normalization in Eq. (2), the normalized trajectory can be written as $\mathbf{Z}(\mathbf{x}) = \mathbf{D}(\mathbf{x})\mathbf{H}(\mathbf{x})$, where $\mathbf{D}(\mathbf{x})$ is a positive invertible diagonal matrix. This rescaling preserves the row span and algebraic rank while reducing the diagnostics' sensitivity to variation in row norms.

The proposition is an exact reduced-coordinate reparameterization under a single fixed subspace shared across inputs, not an information-theoretic bound on real-valued coordinates. Our trajectory diagnostics (effective rank, dominant-energy ratio, and latent-step cosine) are computed separately for each example; NMI is instead an across-example statistic of dependence between latent-step assignments. The concentrated subspace may vary with $\mathbf{x}$. The trajectory metrics diagnose within-trajectory geometric concentration; they do not directly measure semantic reasoning capacity. Related rank–expressiveness arguments appear in the Transformer-expressiveness line (Merrill and Sabharwal, 2024; Dong et al., 2021; Bhojanapalli et al., 2020; Nait Saada et al., 2025) and in the self-supervised representation-collapse line (Jing et al., 2021; Bardes et al., 2021; Zbontar et al., 2021).

Local power-iteration picture. For a fixed $\mathbf{x}$, write deviations $\boldsymbol{\delta}_t = \mathbf{h}_t - \mathbf{h}^\star$ around a local reference state $\mathbf{h}^\star$ and suppose the local update is approximated by $\boldsymbol{\delta}_{t+1} = \mathbf{A}_\mathbf{x}\boldsymbol{\delta}_t$, with $\mathbf{A}_\mathbf{x}$ diagonalizable, eigenvalues $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_d|$, and $\boldsymbol{\delta}_1$ having a nonzero component on $\mathbf{v}_1$. Expanding $\boldsymbol{\delta}_1 = \sum_k c_k\mathbf{v}_k$ gives $\boldsymbol{\delta}_t = \sum_k c_k\lambda_k^{t-1}\mathbf{v}_k$, so after normalization

$$\frac{\boldsymbol{\delta}_t}{\|\boldsymbol{\delta}_t\|_2} \to \pm\frac{\mathbf{v}_1}{\|\mathbf{v}_1\|_2}, \tag{7}$$

at rate controlled by $|\lambda_2/\lambda_1|$ per step. Under these stated assumptions, repeated application of the same local update is one possible geometric route by which additional latent steps occupy few new directions in representation space.

A.2 Justifying the mean-direction proxy and the projection loss

Let $\mathbf{Z} \in \mathbb{R}^{m\times d}$ be the row-normalized latent trajectory and $\bar{\mathbf{z}} = \frac{1}{m}\sum_{i=1}^m \mathbf{z}_i$. Three short arguments show why penalizing projections onto the normalized mean direction is principled.

Mean direction summarizes pairwise alignment. For the ε-stabilized rows,

$$\|\bar{\mathbf{z}}\|_2^2 = \frac{1}{m^2}\sum_{i,j}\mathbf{z}_i^\top\mathbf{z}_j. \tag{8}$$

The squared norm of the mean direction is therefore an exact summary of average pairwise dot products. Because $\|\mathbf{z}_i\|_2 = \|\mathbf{h}_i\|_2/(\|\mathbf{h}_i\|_2 + \varepsilon)$, this is a norm-weighted cosine identity; it coincides with average pairwise cosine alignment for exactly unit-normalized nonzero rows and approximates it when ε is negligible relative to their norms. This is why the column-sum direction in Eq. (4) is a natural proxy for what the latent steps share. For any unit $\mathbf{q}$ and $a_i = \mathbf{z}_i^\top\mathbf{q}$, Jensen's inequality gives

$$\frac{1}{m}\sum_i a_i^2 \geq \left(\frac{1}{m}\sum_i a_i\right)^2, \tag{9}$$

which equals $\|\bar{\mathbf{z}}\|_2^2$ when $\mathbf{q}$ is the normalized mean. The projection loss therefore upper-bounds the squared strength of the shared mean direction.

Spectral interpretation. Define $L_\mathbf{q}(\mathbf{Z}) = \frac{1}{m}\|\mathbf{Z}\mathbf{q}\|_2^2 = \frac{1}{m}\mathbf{q}^\top\mathbf{Z}^\top\mathbf{Z}\mathbf{q}$, let $\mathbf{v}_1$ be a top right singular vector of $\mathbf{Z}$, and set $\rho_1(\mathbf{Z}) = \sigma_1(\mathbf{Z})^2/m$ and $\gamma_\mathbf{q} = \langle\mathbf{q}, \mathbf{v}_1\rangle^2$. Then

$$\gamma_\mathbf{q}\rho_1(\mathbf{Z}) \leq L_\mathbf{q}(\mathbf{Z}) \leq \rho_1(\mathbf{Z}), \tag{10}$$

where the upper bound is the Rayleigh quotient bound and the lower bound follows by expanding $\mathbf{q}$ in the right-singular basis and retaining the $\mathbf{v}_1$ term. When the proxy $\mathbf{q}(\mathbf{x})$ aligns with $\mathbf{v}_1$, the loss suppresses the leading spectral component; when alignment is weak, the loss still penalizes the mean shared direction but should not be read as a complete surrogate for the full spectrum.

Spiked model: mean direction equals top singular direction. Let $\mathbf{z}_i = a_i\mathbf{u} + \boldsymbol{\xi}_i$, with $\mathbf{u}$ a unit shared direction, $\mathbf{u}^\top \boldsymbol{\xi}_i = 0$, $\mathbb{E}[\boldsymbol{\xi}_i] = 0$, and isotropic residual covariance in the orthogonal complement. Then $\mathbb{E}[\bar{\mathbf{z}}] = \mathbb{E}[a_i]\mathbf{u}$, and

$$\mathbb{E}\left[\tfrac{1}{m}\mathbf{Z}^\top\mathbf{Z}\right] = \mathbb{E}[a_i^2]\mathbf{u}\mathbf{u}^\top + \frac{\tau^2}{d-1}(\mathbf{I} - \mathbf{u}\mathbf{u}^\top). \tag{11}$$

Whenever $\mathbb{E}[a_i^2] > \tau^2/(d-1)$, the population covariance has a leading spike along $\mathbf{u}$, and the normalized mean direction and the leading right singular direction identify the same shared component. This is why a lightweight mean-direction proxy can replace an SVD inside the training loop.

A.3 Signed-mean cancellation as a boundary case

A natural concern is a theoretical boundary case in which opposite directions cancel in the signed mean despite a rank-one trajectory. Let $\mathbf{z}_i = s_i\mathbf{u}$, where $s_i \in \{-1, +1\}$ and $\mathbf{u}$ is a unit vector. Since $\mathbf{Z} = \mathbf{s}\mathbf{u}^\top$, the trajectory remains rank one and SVD recovers $\mathbf{u}$ up to sign. For even m, however, equal numbers of positive and negative signs give $\sum_i \mathbf{z}_i = \mathbf{0}$; the underlying unit mean direction is then undefined, and the ε-stabilized proxy in Eq. (4) becomes the zero vector. With small off-axis perturbations, near balance can instead make the mean direction dominated by the perturbation residual and therefore unrepresentative of the leading rank-one axis.

This construction requires exact sign symmetry and is therefore an idealized boundary case; whether an approximate version occurs in practical trajectories is an empirical question. To distinguish spectral concentration from signed-mean cancellation, we use two quantities. First, $\rho_1 = \sigma_1(\mathbf{Z})^2/m$ is the leading singular energy per state; for exactly unit-normalized rows, it equals the fraction of total trajectory energy captured by the leading singular direction, so ρ_1 near one indicates near-rank-one structure. Second, $R = \|m^{-1}\sum_i \mathbf{z}_i\|_2$ is the magnitude of the signed mean, so R near zero indicates strong directional cancellation. We audit all 1,038 ASDiv-Aug trajectories from the same LLaMA-3B + SIM-CoT checkpoint pair used in Section 6. The audit uses all row-normalized states seen by the training loss.

We operationally flag near-rank-one cancellation when $\rho_1 \geq 0.9$ and $R \leq 0.1195$: the first condition requires at least approximately 90% of the trajectory energy to lie in one direction, whereas the second requires the signed mean to be unusually small. For seven exactly unit-normalized rows, $R^2 = (1 + 6\bar{c})/7$, where $\bar{c}$ is the average signed cosine over the 21 unordered state pairs; hence $R \leq 0.1195$ is approximately equivalent to $\bar{c} \leq -0.15$, or 90% of the way from zero correlation to the general zero-mean limit $-1/6$. No trajectory qualifies in either the baseline checkpoint or the trained WIDER checkpoint (0/1,038 in each). Under this criterion, no near-rank-one cancellation failure is observed among the audited trajectories.

B Datasets

Table 4 summarizes the train/test sizes of the four reasoning datasets used in Section 5, and Table 6 shows one representative example per dataset to illustrate the input–output formats. The same splits are used across all backbones and method variants. Table 5 lists these datasets together with the pretrained models used in this work and their licenses or terms of use; we use each asset in a manner consistent with its license, restrict usage to noncommercial academic research where required, and do not redistribute any underlying asset.

Dataset	Train	Test
StrategyQA	1,537	272
CommonsenseQA	8,196	1,221
ASDiv-Aug	4,183	1,038
AQuA	97,467	254

Table 4: Training and test sizes for the four reasoning datasets.

C Backbone and Training Configuration

The main experiments are organized as matched comparisons: within each CODI or SIM-CoT pair the backbone, dataset split, prompt template, parser, latent-step schedule, training data, and checkpoint-selection rule are held fixed, and the only intended difference is whether the spectral auxiliary loss in

Table 5: Existing assets used in this work and their licenses or terms of use.

Asset	License / Terms
StrategyQA	MIT License
CommonsenseQA	MIT License
ASDiv-Aug	CC BY-NC-ND 4.0 for the ASDiv-Aug release; CC BY-NC 4.0 for the original ASDiv dataset
AQuA	Apache-2.0 License
LLaMA-3.2-3B-Instruct	Llama 3.2 Community License and Acceptable Use Policy
LLaMA-3.1-8B	Llama 3.1 Community License and Acceptable Use Policy
Qwen3-1.7B	Apache-2.0 License

Eq. (6) is added. The no-CoT and SFT-CoT rows in Table 1 are contextual references, not ablations of the spectral objective. We use **LLaMA-3.2-3B-Instruct** as the default backbone for the main matched comparisons, the coefficient ablation, and the ASDiv-Aug mechanism analysis, and **Qwen3-1.7B** as a second backbone family and scale. For each backbone–dataset setting, the same evaluation parser is used across rows so that accuracy differences are not caused by answer extraction. The spectral coefficient is selected from the sweep in Section 5.3. The auxiliary loss is active only during training; at inference, the latent schedule and decoding procedure are unchanged, so token counts in Table 1 reflect the underlying reasoning configuration rather than an additional spectral module. The concrete training recipe below records the optimizer, schedule, and batching shared by each baseline and its WIDER variant, so that the matched comparisons in Section 5 can be reproduced end-to-end.

CODI Training Setup. For CODI, we follow the configuration in Shen et al. (2025). The effective batch size is set to 128, and all experiments are conducted using `bfloat16` precision. We train for 10 epochs. The learning rate is set according to the backbone: $1e{-}5$ for LLaMA-3.2-3B-Instruct and $1e{-}4$ for Qwen3-1.7B.

SIM-CoT Training Setup. We reproduce SIM-CoT by implementing its method within the CODI framework, keeping the effective batch size, precision, backbone-specific learning rates, learning-rate schedule, and number of epochs identical to the CODI setup above.

Hardware Information. We perform all experiments on multiple machines equipped with NVIDIA H800 80 GB GPUs. The software stack uses CUDA version 12.8.

D Results on LLaMA-3.1-8B

We extend our experiments to LLaMA-3.1-8B using CODI as the implicit-CoT baseline and the same matched training and evaluation protocol. As shown in Table 7, WIDER improves accuracy on all four benchmarks, with gains ranging from 0.32 to 4.05 percentage points.

We further test whether the same geometric pattern appears in this larger backbone by computing the effective rank of the first six latent states. Table 8 shows that WIDER substantially raises effective rank on every dataset.

These results show that latent rank collapse persists beyond compact models. A larger backbone provides a wider representation space, but successive latent reasoning states can still concentrate along only a few directions. The higher effective ranks under WIDER indicate broader trajectory subspaces, rather than an effect obtained merely by adding model capacity. This finding supports our original analysis: latent rank collapse arises when successive latent states align with a shared dominant direction, rather than merely when a model has too few parameters.

E Dominant-Direction Vocabulary Projection Protocol

For each example we stack the first six implicit states into $\mathbf{H}(\mathbf{x}) \in \mathbb{R}^{6\times d}$, take its first right singular vector $\mathbf{v}_1(\mathbf{x})$, and score the vocabulary by $\mathbf{s}(\mathbf{x}) = \mathbf{W}_{\mathrm{LM}}\mathbf{v}_1(\mathbf{x})$. Since the sign of $\mathbf{v}_1(\mathbf{x})$ is arbitrary, we rank by $|\mathbf{s}(\mathbf{x})|$ and keep the top $N = 30$ tokens per sample. Each token is deterministically assigned to one of five surface-form categories—*punctuation*, *function word* (closed-class English), *number* (pure numeric literal), *whitespace/special* (whitespace, byte-level artifacts, control tokens), and *content-like* (everything else)—using the same categorizer for baseline and Spectral models. We report per-sample slot shares averaged across samples, and summarize via $\mathrm{ContentShare}$ (the content-like share) and $\mathrm{ShortcutShare} = 1 - \mathrm{ContentShare}$, where shortcut share combines punctuation, function-word, number, and whitespace/special shares, so the two summaries sum to 100%. The diagnostic indicates which vocabulary region the dominant singular direction occupies; it is not a claim that individual

Dataset	Question	Ans.	CoT
StrategyQA	Are more people today related to Genghis Khan than Julius Caesar?	True	(1) Genghis Khan is known to have had many children and thousands of descendants across Asia. (2) Genetic studies suggest millions of men carry a Y-chromosome traceable to him. (3) Julius Caesar does not have a similarly documented wide-spanning direct lineage. (4) Historical and genetic evidence does not indicate a comparable descendant impact for Caesar.
Commonsense-QA	The sanctions against the school were a punishing blow, and they seemed to ___ what efforts the school had made to change? Choices: (A) ignore (B) enforce (C) authoritarian (D) yell at (E) avoid	A	The sanctions are undermining or dismissing the school's efforts. "Ignore" fits best, conveying that the sanctions do not acknowledge those efforts.
ASDiv-Aug	gino has 64 popsicle sticks. i have 100 popsicle sticks. what is the sum of our popsicle sticks?	164	$\langle\langle 64 + 100 = 164 \rangle\rangle$
AQuA	Two friends walk along a 43-km trail, starting at opposite ends at the same time. If P's rate is 15% faster than Q's, how many km will P have walked when they pass? Choices: (A) 21 (B) 21.5 (C) 22 (D) 22.5 (E) 23	E	If Q walks x km, P walks $1.15x$ km. Then $x + 1.15x = 43 \Rightarrow 2.15x = 43 \Rightarrow x = 20$, so P walks $1.15 \times 20 = 23$ km. The answer is E.

Table 6: Representative examples from each training dataset, illustrating the input–output formats spanned by the four benchmarks.

Dataset	CODI	+WIDER	Δ
StrategyQA	77.94	**80.51**	+2.57
CommonsenseQA	82.56	**82.88**	+0.32
ASDiv-Aug	85.55	**89.60**	+4.05
AQuA	27.17	**28.35**	+1.18

Table 7: Accuracy (%) on LLaMA-3.1-8B with CODI as the matched implicit-CoT baseline. Δ denotes the absolute gain in percentage points.

Dataset	**CODI** erank	**+WIDER** erank
StrategyQA	1.774	**4.057**
CommonsenseQA	2.092	**5.523**
ASDiv-Aug	1.715	**3.999**
AQuA	1.960	**4.771**

Table 8: Effective rank of the first six latent states on LLaMA-3.1-8B.

top-projected tokens are semantically meaningful.

Aggregation and reporting. For every checkpoint we compute the five surface-form shares on a held-out subset of evaluation prompts, then average across prompts to obtain a single per-checkpoint share vector. Reported figures pair each baseline with its matched WIDER variant on the same backbone, dataset, and evaluation subset, so the absolute share values are comparable across rows. We do not threshold or filter tokens beyond the top-N rank, and we do not re-weight categories; differences between paired rows therefore reflect a direct shift in where the dominant singular direction concentrates, rather than any post-hoc selection of which tokens to count.

Scope and caveats. The protocol is a diagnostic of where the dominant latent direction projects in the vocabulary, not a token-level interpretation. Two cautions apply. First, the sign of $\mathbf{v}_1(\mathbf{x})$ is arbitrary, so the projection magnitude $|\mathbf{s}(\mathbf{x})|$ is the only meaningful quantity at the token level; individual rankings are not invariant under unit-norm reparameterizations of nearby singular directions. Second, the surface-form categorizer is deterministic but coarse: it does not distinguish between, e.g., content-like tokens that are stop-word collocations and those that carry task-specific signal. We therefore interpret the diagnostic as a population-level statement about which vocabulary region absorbs latent energy, rather than as an account of which specific tokens drive any given prediction.

Implementation and comparability. We compute $\mathbf{s}(\mathbf{x})$ in full precision after casting the latent states and the unembedding $\mathbf{W}_{\mathrm{LM}}$ to `float32`, so that ranking decisions do not depend on the `bfloat16` training precision used elsewhere in the pipeline. The top-N cutoff is fixed at N=30 across all models and datasets, and the same prompt subset is reused for every checkpoint of a given

backbone–dataset pair, so paired rows in the main tables differ only in which model produced the latent states rather than in which prompts or ranks were inspected. When a token spans multiple byte-pair pieces, each piece is categorized independently and contributes to the slot share of its own surface form; we do not merge subword pieces back into whole words, because doing so would require a tokenizer-specific reassembly step that behaves differently across backbones and would obscure the backbone-agnostic comparison we report. The full categorizer, including the regular expressions used to detect numeric literals and the closed list of function words, is released with the code so that all share values in the main text can be recomputed from a released checkpoint without reimplementing any heuristics.