Interpreting Latent CoT Reasoning as Dynamical Systems

Anonymous Authors[1]

Abstract

Recent latent reasoning methods such as CODI and COCONUT face a fundamental interpretability problem since they carry multiple superimposed candidate traces in hidden space at each step, obscuring how reasoning evolves, while explicit-CoT follows a single transparent reasoning trace at each step. Existing mechanistic methods show compression, shortcuts, and superposition without in-depth analysis, hence failing to explain how reasoning evolves across latent steps. To address this gap, we model the sequence of latent tokens as a trajectory in representation space and apply dynamical-system analysis to define the reasoning evolution. Using both quantitative (step-to-step change, direction consistency, Lyapunov sensitivity, etc.) and qualitative (UMAP and DMD/PHATE projections), we show that latent CoT exhibits structured, non-random dynamics with two distinct stability classes: CODI behaves as a stable attractor while COCONUT behaves as an unstable expanding system. SimCoT supervision tightens both behaviors handling latent instability in these methods without changing the underlying dynamics. This framework advances the interpretability of latent CoT reasoning dynamics and introduces actionable findings to catalyze further research into improving latent reasoning performance. All code, data, and other artifacts will be publicly released upon acceptance.

1. Introduction

Latent CoT paradigms such as CODI and COCONUT have consistently outperformed explicit CoT in the performance-compute tradeoff. However, the interpretability of Latent CoT remains underexplored and is still an active area of research. While existing mechanistic interpretation methods (logit lens, attention heatmaps, activation patching) show relationships between latent tokens and outputs, and the role of latent steps through causality, they do not show how reasoning evolves through latent steps (Liang & Pan, 2026; Goyal et al., 2025). Also, prior works show compression, shortcuts, and superposition in latent-CoTs (Liang & Pan, 2026; Li et al., 2026), but they do not highlight quantitative behavior, and quantitatively analyze reasoning evolution in latent steps.

Dynamical systems offers a principled approach to study how internal representations evolve during reasoning. When applied to explicit CoT, it helps evaluate whether models genuinely reason step-by-step or rely on memorization (Yu et al., 2025; Pham et al., 2026). While these works operate on explicit CoT, they analyze reasoning dynamics by how frequently the model shifts and visits other states, but they do not thoroughly measure the rate of change, direction, and stability of that change, which are crucial for understanding the actual reasoning dynamics. Faithfulness of latent CoT, viz. the extent to which latent steps reflect genuine, step-by-step reasoning rather than opaque computation, is currently underexplored and lacks rigorous verification. Dynamical systems provides the tools to address this gap directly.

To address this gap, we propose the first **dynamical system framework to interpret Latent-CoT reasoning**, evaluating its *stability, representational geometry, and underlying cognitive strategies*. The framework models latent tokens at each intermediate step as a trajectory of states and applies concepts from dynamical systems to examine reasoning evolution across latent steps. Using quantitative metrics and qualitative representations from dynamical systems, we hypothesize that latent reasoning exhibits structured, non-random dynamics that correlate with training strategies of the Latent-CoT technique.

2. Related Works

2.1. Chain-of-Thought and Latent Reasoning

Chain-of-Thought (CoT) has become the efficient paradigm for LLM reasoning (Wei et al., 2022), but explicit CoTs are computationally inefficient due to verbose reasoning traces. Latent CoTs were introduced to tackle this prob-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

lem by skipping the verbose reasoning rationales. Recent latent CoT frameworks such as CODI (Shen et al., 2025) and COCONUT (Hao et al., 2024) preserve the CoT reasoning in the latent space, achieving better token-efficient performance over explicit CoT.

COCONUT reasons in the latent space by feeding back the last hidden state as next input enabling a breadth-first search over candidate reasoning steps. It is trained using a multi-stage curriculum that progressively replaces explicit CoT tokens with continuous thoughts. CODI instead avoids curriculum learning altogether by jointly training a teacher on explicit CoT and a student on implicit CoT within a single self-distillation stage, transferring reasoning ability via hidden-state at the answer token.

While COCONUT and CODI represent reasoning with latent steps, both methods suffer from the latent instability problem where we see unstable training as the number of implicit reasoning tokens scale up, leading to latent representations collapsing into homogeneous states that lose semantic diversity. Sim-CoT (Wei et al., 2025) addresses this instability by adding auxiliary decoder during training to align each implicit latent token with its corresponding explicit reasoning step, enforcing structured semantics in the latent space.

Mechanistic interpretability has emerged as an important direction for understanding the nature of latent CoT and its reasoning behavior, which we discuss in the following section.

2.2. Mechanistic Interpretability of Latent CoT

(Liang & Pan, 2026) depicts CODI can retrieve bridge states in a multi-hop reasoning tasks, while relying on compression and short-cut like pathways for longer reasoning chain tasks. (Goyal et al., 2025) shows a "scratchpad-style" latent reasoning in CODI where latent states alternate between storage and compute steps. While these works establish that latent reasoning is functionally meaningful, they do not explain the reasoning dynamics.

(Li et al., 2026) provides a causal analysis of CODI showing that latent reasoning is a two-stream structure where latent states propagate information, while final inputs can bypass computation through direct copy pathways. (Liang & Pan, 2026) has also shown latent-CoTs enable shortcut-like paths to the final answer. These results suggest that latent reasoning contains compression, shortcut routing, and superposition, and they collapse during deeper reasoning tasks, but prior works still lack explanations of "why" and "how" they emerge in Latent CoTs.

2.3. Dynamical Systems for Reasoning Dynamics

Dynamical systems theory provides a mathematical framework for characterizing how states evolve over time, capturing concepts such as attractors, trajectory stability, and divergence. These tools have been widely applied in physics and neuroscience (Holmes, 1990; John et al., 2022) to study complex temporal processes, and have recently been adapted to analyze reasoning behavior in language models.

Viewing reasoning as trajectories in latent spaces has been proven effective for studying reasoning dynamics (Yu et al., 2025). Recent works have shown that explicit CoT reasoning can be analyzed using state-aware and recurrence-based measures to understand how often the model revisits, shifts, or stabilizes in particular reasoning states (Yu et al., 2025; Pham et al., 2026).

Spectral analysis can be used to represent and cluster reasoning steps semantically to evaluate reasoning dynamics (Yu et al., 2025). Similarly, Pham et al. (Pham et al., 2026) apply recurrence-based analysis to reasoning traces and show that metrics such as determinism, laminarity, and stalling reveal the reasoning process and the frequency of state visits. These results motivate treating reasoning as a dynamical system rather than as a sequence of independent steps.

However, these approaches have mainly been developed for explicit CoTs, which leaves the study of Latent-CoT reasoning as a dynamical system underexplored. In particular, they do not measure how latent hidden states evolve through representation space in terms of step size, direction consistency, and stability, nor what causes the emergence of phenomena such as compression, shortcuts, and superposition in the Latent-CoT reasoning paradigm.

Our work addresses this gap by treating latent CoT as a trajectory in latent space and studies its reasoning evolution using dynamical systems.

3. Methodology

We propose a framework (Figure 1) for analyzing latent chain-of-thought (CoT) reasoning as a dynamical system. The approach extracts intermediate latent representations from reasoning models, constructs trajectories in representation space, and applies dynamical systems analysis to characterize reasoning behaviors.

3.1. Problem Formulation

Given an input x, a latent CoT model produces a sequence of hidden representations across reasoning steps:

$$z_1, z_2, \ldots, z_T, \quad z_t \in \mathbb{R}^d,$$

where each z_t denotes the model's internal state at step t, and d is the hidden dimension of the model.

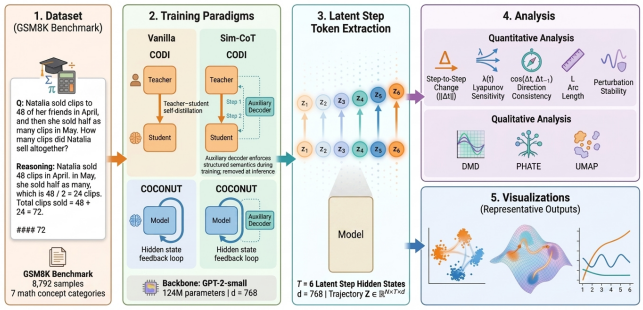


Figure 1. **Dynamical System Analysis of Latent-CoTs.** (1) GSM8K dataset. (2) Four model configs - CODI (teacher–student self-distillation) and COCONUT (hidden-state feedback loop) in Vanilla and Sim-CoT settings; auxiliary decoder enforces structured semantics. (3) T = 6 latent hidden states form a trajectory in representation space. (4) Quantitative metrics and qualitative projections. (5) Representative latent space and dynamics visualizations.

This sequence is treated as a trajectory in representation space.

3.2. Dimensionality Reduction

The trajectory tensor $Z \in \mathbb{R}^{N\times T\times d}$ lies in a high-dimensional space that makes direct visualization intractable. Five complementary methods are applied to project latent states into a low-dimensional space for analysis. For t-SNE, UMAP and PHATE, the tensor is first flattened to $[N{\cdot}T,\ d]$ so that all latent states are reduced and then reshaped to $[N, T, k]$ to preserve the temporal ordering of each trajectory.

t-SNE (van der Maaten & Hinton, 2008) preserves the local neighborhood structure through pairwise, similarity modeling, making it sensitive to fine-grained clustering among nearby reasoning states.

UMAP (McInnes et al., 2018) approximates the underlying data to balance local and global structure, producing stable embeddings suited for continuous trajectories.

Dimensionality reduction is used solely for visualization (Section 4.3.1). All quantitative metrics are computed directly on the original representations $z_t \in \mathbb{R}^d$.

3.3. Dynamical Systems Concepts

DMD (Tu et al., 2013) treats the latent sequence as a discrete-time dynamical system and identifies the spatial modes and eigenvalues via an SVD of the snapshot pair $(X_1, X_2) = (z_{0:T-2},\ z_{1:T-1})$. The decomposition is computed in a single batched operation across all N trajectories simultaneously. Each trajectory is then projected onto the leading two DMD modes for visualization. The resulting eigenvalues serve as a stability measure in the quantitative analysis (Section 4.3.2).

PHATE (Moon et al., 2019) uses a multi-scale diffusion process to capture transitions in high-dimensional data. It uses a diffusion operator based on an affinity graph constructed by kernel methods to calculate the potentials that represent the distances between points, which represents the geometric structure of the underlying manifold. Such a low-dimensional embedding captures both the local structure and global dynamics, which is also ideal for representing sequential latent paths.

4. Experimental Configurations

4.1. Datasets

GSM8K Experiments are conducted on the GSM8K benchmark (Cobbe et al., 2021), a collection of grade-school math word problems requiring multi-step arithmetic reasoning. The full GSM8K dataset (train and test splits combined, 8,792 samples) is used across seven math concept categories: Geometry (210), Rates & Speed (675), Percentages & Ratios (1,266), Money & Pricing (2,741), Fractions & Decimals (1,045), Multiplication & Division (224), and Arithmetic & Multi-step (2,631). Questions are assigned to categories via priority-ordered keyword matching, with more specific categories checked before broader ones to prevent improper labeling. Five hundred stratified samples are taken per category. Categories with fewer available examples contribute all remaining samples. Ground-truth answers are parsed from the `"####"` delimiter in GSM8K annotations.

4.2. Models

CODI and COCONUT are both GPT-2-small backbones (124 M parameters, 12 layers, 12 heads, hidden size 768, vocabulary extended to 50 260 with three latent special tokens`<|start-latent|>`, `<|latent|>`, and`<|end-latent|>`) trained on GSM8K (Hao et al., 2024; Shen et al., 2025). Vanilla checkpoints are taken

from `ModalityDance/latent-tts-coconut`[1] and `ModalityDance/latent-tts-codi`[2]; CODI additionally loads a self-distillation projection module (`prj.pt`) at inference time that maps each latent hidden state back into the input-embedding space. Sim-CoT variants of both methods are taken from `internlm/SIM_COT-GPT2-Coconut`[3] and `internlm/SIM_COT-GPT2-CODI`[4] (Wei et al., 2025). Each model generates a fixed sequence of $T = 6$ latent reasoning steps per input. Text generation uses greedy decoding with a maximum of 512 output tokens. All experiments use a fixed random seed of 42.

4.3. Analysis of Latent CoT trajectories

Analysis is performed at 2 levels: qualitative inspection of geometric structure and quantitative characterization of dynamics.

4.3.1. Qualitative Analysis

Reduced representations from Section 3.2 are visualized as sequences of points in $\mathbb{R}^2$ and $\mathbb{R}^3$ with temporal ordering preserved. These plots expose geometric phenomena including smooth progression through state space, directional pivots (using centroids of latent representations), convergence, and separation between correct and incorrect trajectories.

4.3.2. Quantitative Analysis

Trajectory dynamics are characterized by two groups of metrics: *step-based metrics* that measure how the trajectory moves between consecutive steps, and *stability metrics* that characterize sensitivity and attraction to fixed points.

Step-based Metrics

Step-to-step change measures the magnitude of displacement at each transition:

$$\|\Delta_t\| = \|z_{t+1} - z_t\|_2. \tag{1}$$

Large values indicate significant representational transitions; a declining profile indicates the trajectory is decelerating. When reasoning process is near the end, we hypothesize that latent CoT tokens settle on a solution, hence decrease in the value of consecutive differences is expected.

Direction consistency measures whether consecutive displacements point in the same direction:

$$C_t = \cos(\Delta_t, \Delta_{t-1}). \tag{2}$$

Values near $+1$ indicate smooth forward movement; values near -1 indicate the trajectory is reversing direction at each step. The latent CoT tokens is expected to get aligned in a particular direction as it progresses through the reasoning process, showing consecutive tokens are aligning towards the final solution.

Arc length summarizes total path complexity as the cumulative displacement over all transitions:

$$L = \sum_{t=1}^{T-1} \|z_{t+1} - z_t\|_2. \tag{3}$$

It demonstrates the reasoning effort it takes to solve a problem in latent space. The change in latent representation from the start to end shows the compute capacity to transform the intermediate latent tokens to the final solution.

Stability Metrics

Lyapunov Sensitivity (Surrogate) To approximate local stability without re-running the model, a trajectory-based surrogate is computed from the ratio of consecutive step magnitudes:

$$\lambda(t) = \log \frac{\|z_{t+1} - z_t\|}{\|z_t - z_{t-1}\|}, \tag{4}$$

where $\|\cdot\|$ denotes the ℓ_2 norm. A positive $\lambda(t)$ indicates the trajectory is locally diverging. A negative $\lambda(t)$ indicates convergence. $\lambda(t) = 0$ indicates neutral stability, where the step magnitude remains constant and the trajectory is neither converging nor diverging. Diverging represents the latent reasoning space is still uncertain for a particular stage, while converging means latent CoT tokens are getting closer to a final solution. A typical reasoning process is expected to go from high uncertainty (diverging behavior) to low uncertainty (converging/stable behavior).

Perturbation stability re-runs inference with Gaussian noise injected into input embeddings and measures divergence from the clean trajectory at each step. Growing divergence indicates a sensitivity to initial conditions. This demonstrates the robustness of latent CoT tokens while reasoning.

5. Results

Results are reported for COCONUT and CODI methods for both Vanilla COT and SIM-COT training paradigms. The analysis is split into two parts: (1) *Qualitative dynamics*, where we visualize latent states using DMD, PHATE, and UMAP to understand how each model moves through

[1] https://huggingface.co/ModalityDance/latent-tts-coconut

[2] https://huggingface.co/ModalityDance/latent-tts-codi

[3] https://huggingface.co/internlm/SIM_COT-GPT2-Coconut

[4] https://huggingface.co/internlm/SIM_COT-GPT2-CODI

latent space across reasoning steps; and (2) *Quantitative dynamics*, where we measure step-to-step change, direction consistency, and fixed-point distances to compare how the two models behave differently.

Furthermore, we also present concept-wise plots for GSM8K are provided in Appendix A.

5.1. Quantitative Results

Step-to-Step Change The analysis here (as seen in Figure 2) measures the consecutive differences between latent reasoning steps. For the CODI method, we see an uniform curve with variance for vanilla CoT setting. In contrast, Sim-COT setting displays a monotonically decreasing curve. This decreasing characteristic demonstrates that latent CoT tokens start to become similar, a potential sign for convergence in the end phase (solution).

For the COCONUT method, the vanilla CoT setting shows a drop in the transition from 2^{nd} to 3^{rd} latent token, a possible premature convergence in the reasoning process. Whereas, we see a stable behavior in Sim-COT setting.

Step change takeaway

CODI portrays a more stable transition than COCONUT. Sim-COT handles latent stability irrespective of the method, improving the transition dynamics.

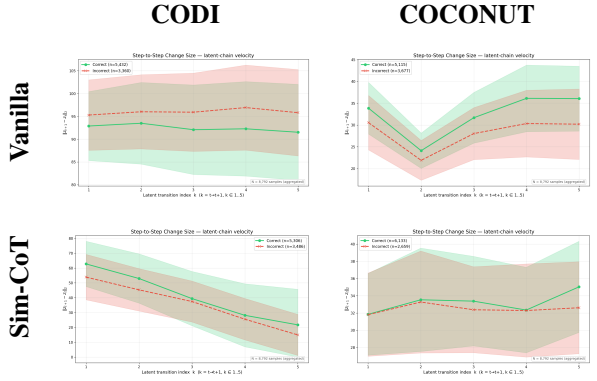


Figure 2. **Step-to-Step change for methods (CODI, COCONUT) under Vanilla + Sim-CoT settings**: Overall, the CODI method exhibits neutral or convergent behavior , while COCONUT shows premature convergence in the vanilla setting. Both the methods have more stable curves in Sim-COT setting compared to their Vanilla counterparts.

Lyapunov Sensitivity Lyapunov sensitivity (as seen in Figure 3) measures the local stability of the latent trajectory. Positive values indicate the trajectory is locally diverging (the model is exploring multiple candidate reasoning paths at that step), while negative values indicate it is locally contracting, i.e. committing to a single path.

For the CODI method, we see a monotonically decreasing curve with negative values across all steps in the Vanilla CoT setting. In the Sim-CoT setting, the curve descends deeper into the negative range, indicating that the supervision strengthens this convergence.

For the COCONUT method, the Vanilla CoT setting exhibits a sharp positive spike at $t = 3$, a localized expansion that captures the model branching across candidate reasoning paths at the mid-chain transition before settling, a genuine exploration phase native to COCONUT's curriculum-trained dynamics. Whereas, in the Sim-CoT setting, this spike disappears and the curve flattens, showing that the supervision replaces the exploratory mid-chain step with a deterministic transition by anchoring each latent token to its corresponding textual reasoning step.

Lyapunov Sensitivity takeaway

CODI shows stable converging behavior while COCONUT shows mid-chain divergence in vanilla. Sim-CoT improves local stability irrespective of the method.

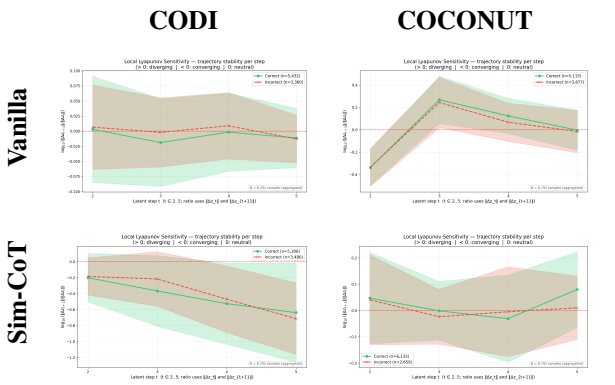


Figure 3. **Lyapunov Sensitivity for methods (CODI, COCONUT) under Vanilla + Sim-CoT settings**: CODI exhibits stable, near-neutral or converging behavior across both settings, while COCONUT shows a sharp divergence (local instability) at t=3 in the Vanilla setting. Sim-CoT supervision suppresses this instability in COCONUT and deepens the convergence in CODI.

Direction Consistency Direction consistency measures the cosine between consecutive latent transitions, $C_t = \cos(\Delta_t, \Delta_{t-1})$. Values near $+1$ indicate smooth forward motion, values near 0 indicate orthogonal pivots, and values near -1 indicate the trajectory reverses direction at every step.

For the CODI method, the vanilla CoT setting shows consistently negative values across all transitions with low variance, meaning the latent CoT tokens reverse direction at

every step. In the Sim-CoT setting, the curve sits near zero, showing that the supervision replaces the reversals with near-orthogonal pivots.

For the COCONUT method, the vanilla CoT setting starts directionally opposing, becomes near-orthogonal at $t = 3$, and reverts to opposing transitions toward the end. In the Sim-CoT setting, the curve remains consistently opposing across all transitions, indicating that supervision stabilizes the directional behavior into a single oscillatory pattern.

Direction Consistency takeaway

Sim-CoT shifts CODI toward orthogonal pivots and steadies COCONUT's mid-chain instability, but neither paradigm produces the smooth forward alignment expected of a directionally converging reasoner.

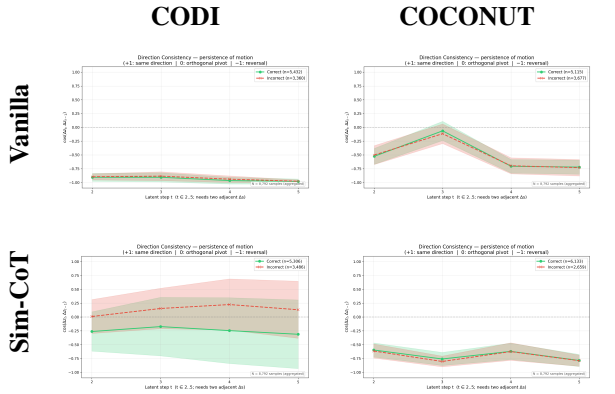


Figure 4. **Direction consistency $C_t = \cos(\Delta_t, \Delta_{t-1})$ across latent steps for methods (CODI and COCONUT) under Vanilla + Sim-CoT settings**: In the Vanilla setting, CODI shows consistently opposing transitions, while COCONUT transitions become near orthogonal at t=3 before reversing again. Under Sim-CoT, CODI shifts to near orthogonal transitions and COCONUT maintains consistent opposition.

Arc length and Perturbation stability Arc length summarizes the total displacement traversed by the latent trajectory, reflecting overall reasoning effort. Perturbation stability measures trajectory divergence $\|z_t^{\text{perturbed}} - z_t^{\text{clean}}\|_2$ under Gaussian noise injected into the input embeddings, tested at $\sigma \in \{0.01, 0.1, 1.0\}$.Results and additional discussions are added in the Appendix A.3.1 and A.3.2.

5.2. Qualitative Analysis

DMD (as seen in Figure 5) projects each latent trajectory onto its dominant modes of variation, showing how the latent representations are spatially organized across the reasoning chain. Tightly clustered latent states indicate contraction toward a bounded region, while spreading latent states indicate expansion away from it.

For the CODI method, the vanilla CoT setting shows a two-lobe pattern with the latent CoT tokens tightly packed into both lobes across all steps, a sign of stable attractor-like organization where the trajectory remains bounded throughout the chain. In the Sim-CoT setting, the two-lobe pattern is preserved with even tighter clustering, indicating that the supervision reinforces this bounded behavior.
For the COCONUT method, the vanilla CoT setting exhibits a butterfly pattern where the latent states start near the center at $t = 0$ and spread outward by $t = 5$, a sign of expanding behavior where the model explores across latent space as the chain progresses. Whereas, in the Sim-CoT setting, the butterfly geometry is retained but with reduced spread at later steps, showing that the supervision constrains the exploration without changing the underlying geometric structure

DMD takeaway

CODI shows a converging two-lobe geometry while COCONUT shows an expanding butterfly geometry, reflecting fundamentally different reasoning dynamics: bounded contraction for CODI versus directional expansion for COCONUT.

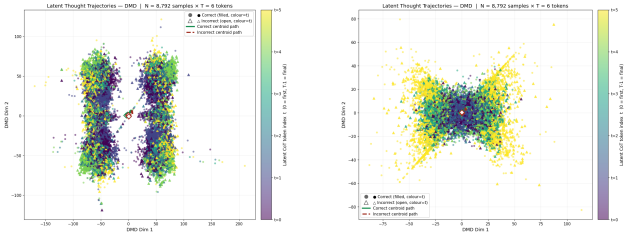


Figure 5. **DMD trajectory projections across latent steps for methods (CODI (left) and COCONUT(right))**: COCONUT exhibits a butterfly pattern with latent states diverging outward across steps, reflecting an unstable system exploring multiple reasoning paths simultaneously. CODI shows tightly packed representations clustering into two dense, well-separated regions across all latent steps, indicating a convergent and stable reasoning dynamic.

PHATE (as seen in Figure 6, right) embeds the latent trajectory using a multi-scale diffusion process that preserves both local neighborhoods and the global manifold structure, making it suitable for showing how latent representations move through different regions of the reasoning manifold.
For the CODI method, the vanilla CoT setting shows a two-lobe pattern with compact yet interleaved representations across latent steps, a sign of convergent organization where the trajectory remains bounded throughout the chain. In the Sim-CoT setting, the structure becomes more compact and the temporal separation between latent steps reduces further, indicating that the supervision tightens the convergence onto a single region.
For the COCONUT method, the vanilla CoT setting shows

each latent step occupying a distinct region of the manifold, with the beginning tokens (purple) and end tokens (yellow/green) positioned in close proximity, a possible indication of shortcut pathways where the model bridges early and late representations directly. Whereas, in the Sim-CoT setting, the distinct-region structure is preserved but the proximity between beginning and end tokens reduces, showing that the supervision retains the regional separation while reducing the shortcut tendency.

PHATE takeaway

COCONUT's beginning and end latent tokens cluster close in PHATE space, hinting at a possible shortcut pathway through the reasoning chain. CODI's representations remain interleaved across all steps with no such signature.

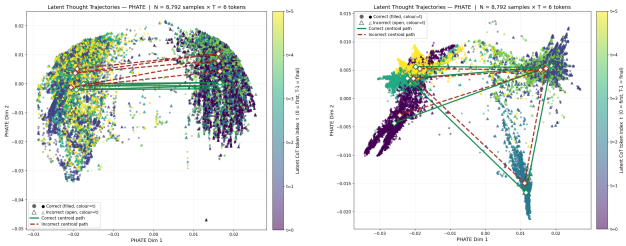


Figure 6. **PHATE trajectory projections across latent steps for methods (CODI (left) and COCONUT(right))**: COCONUT exhibits each latent step occupying distinct regions of the manifold, with beginning and end tokens positioned in closer proximity-suggesting the formation of shortcut pathways in the reasoning process. CODI shows interleaved representations across steps, consistent with its convergent behavior observed in the DMD projection.

UMAP (as seen in Figure 7, right) projects the latent trajectory while preserving local neighborhood relationships and balancing global structure, showing how latent representations cluster by reasoning step.

For the CODI method, the vanilla CoT setting shows two main clusters: middle tokens (3^{rd} and 4^{th}) scattered across one cluster while beginning and end tokens are contained in the other, a sign of role separation where mid-chain steps explore broadly and terminal steps remain bounded. In the Sim-CoT setting, the cluster boundaries tighten and the middle-token spread reduces, indicating that the supervision concentrates the latent representations more uniformly across steps.

For the COCONUT method, the vanilla CoT setting shows each latent step occupying a distinct region of the embedding space, a sign that latent representations carry step-specific roles in the reasoning process. Whereas, in the Sim-CoT setting, the step-wise regional separation is preserved with tighter cluster boundaries per step, showing that the supervision sharpens the role separation without changing the underlying organization.

UMAP takeaway

COCONUT separates each latent step into its own UMAP region, signaling distinct roles per step. CODI groups tokens into two role-based clusters (mid-chain vs. terminal), suggesting it relies on fewer but more functionally specialized positions.

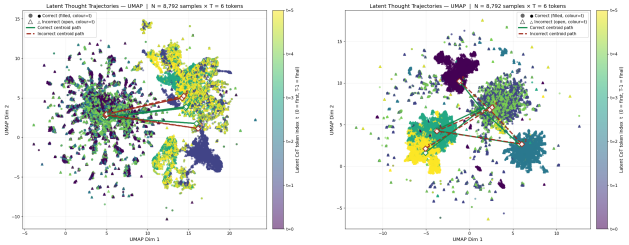


Figure 7. **UMAP trajectory projections across latent steps for methods (CODI (left) and COCONUT(right))**: COCONUT shows each latent step occupying distinct regions, reinforcing step-specific reasoning roles. CODI forms two clusters — middle tokens diffused broadly, while beginning and end tokens remain densely contained.

We also show the t-SNE and PCA plots for the above methods (GSM8k + Vanilla CoT setting) in the Appendix A.2.2 and A.2.1 respectively.

6. Discussion

We present our findings based on the above analysis as follows,

Dynamics of CODI and COCONUT exhibit different latent stability types - Lyapunov metric analysis (see Figure 3) highlights CODI behaves as a *stable attractor* type, where latent representations converge as timestep approaches infinity, which is consistent with its two lobe pattern (see Figure 6 (left)). COCONUT, in contrast, exhibits *unstable expanding* dynamics, where it starts with stable latent space and diverges as time progresses reflecting its butterfly pattern (see Figure 5 (right)).

Sim-COT shows better latent stability than Vanilla Latent COT paradigm - From a dynamical perspective, we see how Sim-COT version of CODI and COCONUT show more stable behavior than their vanilla CoT counterparts in Step-to-Step change (Figure 2) and Lyapunov stability (Figure 3).

Latent geometry for CODI and COCONUT mirrors the training objective - In Figure 7, COCONUT exhibits more coherent and well-separated latent CoT

structure, consistent with its curriculum training which transitions from explicit to implicit tokens. By contrast, CODI shows more diffuse latent trajectories, consistent with its training objective centered on distillation rather than enforcing structure across individual latent CoT tokens.

Underlying cognitive mechanisms for CODI and COCONUT are fundamentally different - Based on the latent representation structures seen in DMD plots, we notice the following,

- COCONUT follows a *computation* strategy where it **persists** all reasoning trajectories simultaneously (denoted by the spread seen in later stages) and continues to expand on it.

- CODI follows a *classification* strategy where all the reasoning trajectories **converge to two modes** and the reasoning process moves between these modes before finalizing the solution.

Overall, the two methods solve problems through fundamentally opposite dynamics. In COCONUT, doing more work in latent space is associated with getting the answer right. In CODI, shorter and simpler paths are the ones that lead to correct answers.

7. Conclusion

This paper provides an insight into the dynamics for the latent CoT reasoning methods. Leveraging an analytical framework grounded in dynamical systems, we demonstrate how these methods differ fundamentally in their stability profiles, training procedures, and the cognitive strategies used in problem solving. We show that interpreting latent CoT tokens as dynamical reasoning states provides the community with another axis of interpretability for further improvements in this space.

Our experiments mainly focus on the GSM8k dataset and the GPT-2 Small model. We also focus mainly on the CODI and COCONUT latent CoT reasoning methods, and additionally use the Sim-COT training paradigm to show how are analysis is paradigm agnostic in nature.

Future works should investigate on the stability types for CODI and COCONUT on long CoT chains, and whether interventions in latent space can improve downstream performance. Other directions include extending this framework to larger reasoning models like Llama, Deepseek-distill-Llama3-8b, etc. and other datasets like MATH and Strategy-QA.

Impact Statement

Our framework establishes a trajectory-based analytical paradigm grounded in dynamical systems theory to formalize the interpretability of latent reasoning processes. This research facilitates the interpretability of reasoning dynamics in latent CoT reasoning technique, enabling the identification of dynamical structures and stability modes in explicit CoT. This enhances the reliability of latent autonomous reasoning in high-stakes environments but also provides an avenue for analyzing the improvements in computational efficiency and inferential accuracy of these methods. Ultimately, these findings offer a robust foundation for the deployment of transparent, verifiable, and resource-efficient latent reasoning architectures.

References

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

Goyal, S., Peters, B., Granda, M. E., Narmadha, A. V., Yugeswardeenoo, D., McDougall, C. S., O'Brien, S., Panda, A., Zhu, K., and Blondin, C. Scratchpad thinking: Alternation between storage and computation in latent reasoning models. In *NeurIPS 2025 Workshop on Regularization in Machine Learning (RegML)*, 2025. Available at https://openreview.net/forum?id=rdmnQIAdBq.

Hao, S., Sukhbaatar, S., Su, D., Li, X., Hu, Z., Weston, J., and Tian, Y. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024. Accepted at COLM 2025.

Holmes, P. Poincaré, celestial mechanics, dynamical-systems theory and "chaos". *Physics Reports*, 193(3): 137–163, September 1990. doi: 10.1016/0370-1573(90)90012-Q.

John, Y. J., Sawyer, K. S., Srinivasan, K., Müller, E. J., Munn, B. R., and Shine, J. M. It's about time: Linking dynamical systems with human neuroimaging to understand the brain. *Network Neuroscience*, 6(4):960–979, 2022. doi: 10.1162/netn_a_00230.

Li, Z., Bai, X., Chen, K., Li, Y., Yang, J., Lin, C., and Zhang, M. Dynamics within latent chain-of-thought: An empirical study of causal structure. *arXiv preprint arXiv:2602.08783*, 2026.

Liang, J. and Pan, L. Do latent-cot models think step-by-step? a mechanistic study on sequential reasoning tasks. *arXiv preprint arXiv:2602.00449*, 2026.

McInnes, L., Healy, J., and Melville, J. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Moon, K. R., van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D. B., Chen, W. S., Yim, K., van den Elzen, A., Hirn, M. J., Coifman, R. R., et al. Visualizing structure and transitions in high-dimensional biological data. *Nature Biotechnology*, 37:1482–1492, 2019.

Pham, Q. T., Jafari, M., and Salim, F. Is my model 'mind blurting'? interpreting the dynamics of reasoning tokens with recurrence quantification analysis (RQA). *arXiv preprint arXiv:2602.06266*, 2026.

Shen, Z., Yan, H., Zhang, L., Hu, Z., Du, Y., and He, Y. Codi: Compressing chain-of-thought into continuous space via self-distillation. *arXiv preprint arXiv:2502.21074*, 2025.

Tu, J. H., Rowley, C. W., Luchtenburg, D. M., Brunton, S. L., and Kutz, J. N. On dynamic mode decomposition: Theory and applications. *arXiv preprint arXiv:1312.0041*, 2013.

van der Maaten, L. and Hinton, G. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., and Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Wei, X., Liu, X., Cao, Y., Wang, J., and Lin, D. Sim-cot: Supervised implicit chain-of-thought. *arXiv preprint arXiv:2509.20317*, 2025.

Yu, S., Xiong, Y., Wu, J., Li, X., Yu, T., Chen, X., Sinha, R., Shang, J., and McAuley, J. Explainable chain-of-thought reasoning: An empirical analysis on state-aware reasoning dynamics. *arXiv preprint arXiv:2509.00190*, 2025.

A. Appendix

A.1. Hyperparameters

A.1.1. Dimensionality Reduction Hyperparameters

All reduction methods project to $k = 2$ and $k = 3$ dimensions for visualization. The following hyperparameters are fixed across all runs:

- **t-SNE:** Perplexity $= 5.0$ (auto-adjusted to $\min\big(5.0,\ \max(2.0,\ (N-1)/3)\big)$ for small batches).
- **UMAP:** $n_{\text{neighbors}} = 5$, min_dist $= 0.1$, Euclidean metric.
- **DMD:** Full-rank SVD (svd_rank $= -1$, effective rank $= \min(D,\ T-1) = 5$ for T=6 step trajectories); batched GPU computation across all N samples simultaneously. Eigenvalue magnitudes $|\lambda|$ classify modes as stable ($|\lambda| < 1$) or unstable ($|\lambda| > 1$).
- **PHATE:** k-nearest neighbors $k = 5$, diffusion time $t =$ auto.

A.1.2. Perturbation Stability Hyperparameters

Perturbation experiments inject Gaussian noise directly into input embeddings and re-run inference. Noise standard deviation $\sigma = 0.01$ and $n = 3$ independent perturbation runs per sample are used. Divergence is measured as the mean ℓ_2 distance between clean and perturbed trajectories at each step, alongside a scale-invariant relative divergence $\|z_t^{\text{perturbed}} - z_t\|/\|z_t\|$.

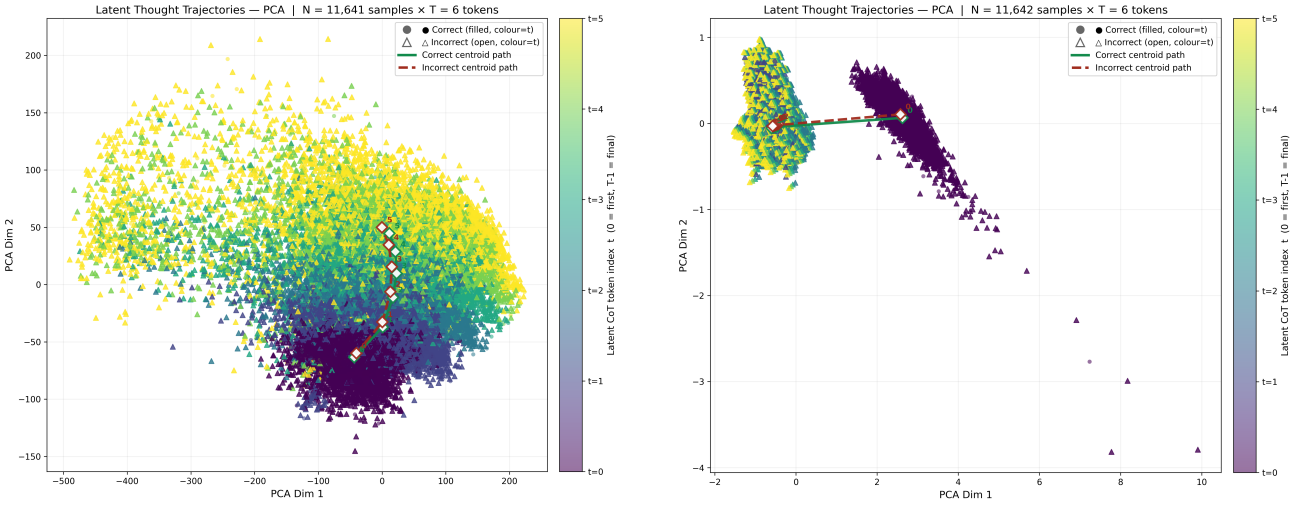
A.1.3. Metric Plot Conventions

All metric plots display mean curves $\pm$ one standard deviation across samples, split by prediction correctness (correct and incorrect). In trajectory plots, each point represents one sample colored by latent step index t; prediction correctness is encoded by marker shape (filled circle = correct, open triangle = incorrect), with mean centroid paths overlaid.

A.2. Additional Trajectory Visualizations

A.2.1. PCA Projections

Linear PCA projects the latent trajectory tensor onto the two directions of greatest variance. Color encodes the latent step index from t=0 (purple) to t=5 (yellow); filled circles mark correct outputs and open triangles mark incorrect outputs. Solid green and dashed red lines show the correct and incorrect centroid paths respectively.



(a) COCONUT · GSM8K *(b)* CODI · GSM8K

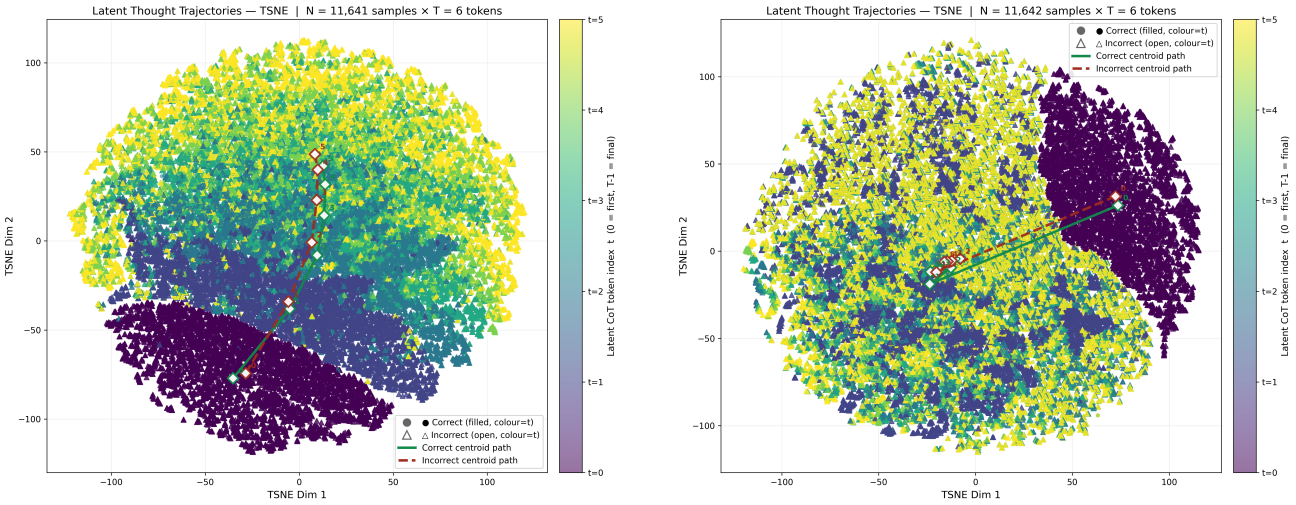
Figure 8. PCA projections of latent trajectories for COCONUT (left) and CODI (right).

For COCONUT, the point cloud splits bimodally along PC2 (the vertical axis): late steps t=4 and 5 occupy an upper lobe above PC2 > 0, while early steps t=0 and 1 occupy a lower lobe below PC2 < -10. Mid-steps t=2 and 3 bridge the two lobes. Both centroid paths trace nearly identical zig-zag routes through this structure, sitting on top of one another throughout the chain. PC2 is acting as a temporal axis: the principal directions of variance encode where in the chain a vector lives, not whether the chain succeeds. Linear PCA cannot separate correct from incorrect trajectories on this model.

For CODI, the bimodal structure is preserved but rotated: the split is along PC1 (the horizontal axis) rather than PC2. Late steps t=3 to 5 form a right lobe at PC1 > 30, and early steps t=0 and 1 form a left lobe at PC1 < -30, with a quieter central gap between them. Centroid paths sweep horizontally from left to right. The two-lobe horizontal axis in CODI mirrors COCONUT's two-lobe vertical axis, but the orientation reflects a different dominant mode structure. In both models, the principal directions of variance encode reasoning phase rather than problem type.

A.2.2. t-SNE Projections

t-SNE preserves local neighborhood structure and is used here to assess whether correct and incorrect trajectories occupy distinct regions at each step. Unlike PCA, t-SNE is nonlinear and does not preserve global distances.



(a) COCONUT · GSM8K *(b)* CODI · GSM8K

Figure 9. t-SNE projections of latent trajectories for COCONUT (left) and CODI (right).

For COCONUT, the projection produces a roughly disc-shaped cloud spanning approximately ± 100 in both dimensions. A pronounced yellow island at t=5 sits in the upper center of the disc, indicating that late-step representations collapse into a single localized attractor region. Early steps (t=0) scatter across the periphery in distributed pockets. Both centroid paths walk overlapping zig-zag routes through the dense middle of the disc. The correctness signal in t-SNE coordinates is weak. The plot is best read as evidence that the late chain commits to a small region while the early chain explores broadly — the local-neighborhood signature of the same compression visible in the UMAP and DMD projections in the main paper.

For CODI, the disc shape is similar in overall envelope but the internal structure is more diffuse: sub-clusters bleed into each other rather than forming sharp islands. Yellow t=5 vectors appear in scattered local pockets without a single clean breakaway region. A dark purple concentration of early steps sits in the upper portion of the disc; mid-step teal forms a diffuse band through the middle. Centroid paths run as short, heavily overlapping trajectories through the dense center. The correctness signal is weak, as in COCONUT. This diffuseness is consistent with CODI's distillation procedure flattening the latent geometry: the model has many equally important local structures rather than a few dominant attractor basins.

A.2.3. 3D Qualitative Trajectory plots of Vanilla CODI and COCONUT with GSM8K

DMD For COCONUT, the latent states start near the center for t=0 and spread outward by t=5 showing that the model starts diverging through latent space at later reasoning steps proving its "chaotic" behaviour as described by direction consistency (Figure 10, right).

Unlike COCONUT, CODI shows a unclear similar divergence in latent steps as they increase (Figure 10, left).

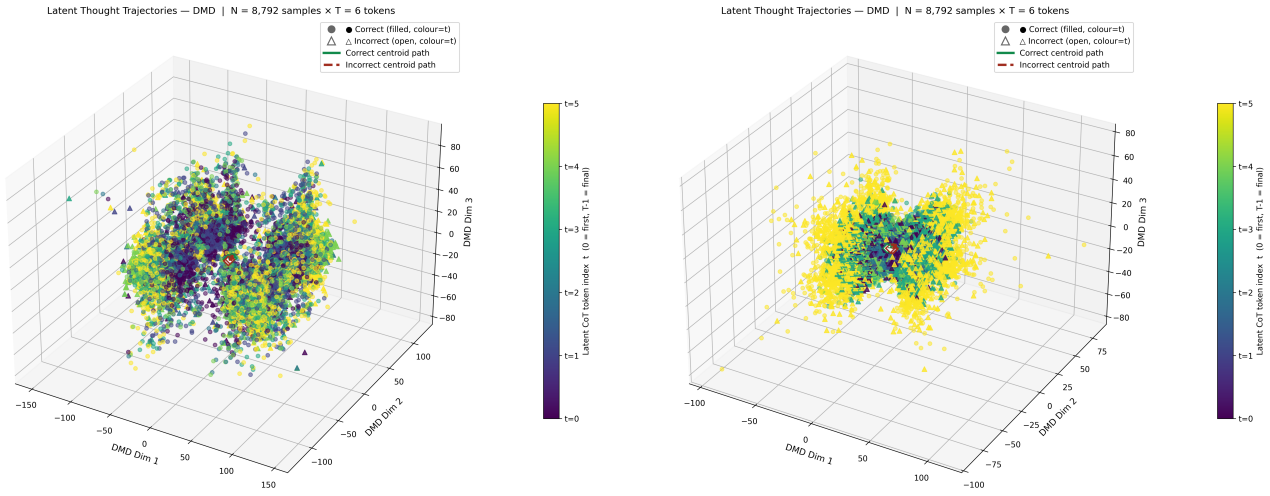


Figure 10. 3D DMD trajectory projections across latent steps for CODI (left) and COCONUT(right).

PHATE We can see from the PHATE projections of COCONUT that each latent step explores different latent regions demonstrating their distinct role in the reasoning process (Figure 11, right).

Unlike COCONUT, CODI shows a unclear similar clustering between latent steps (Figure 11, left).

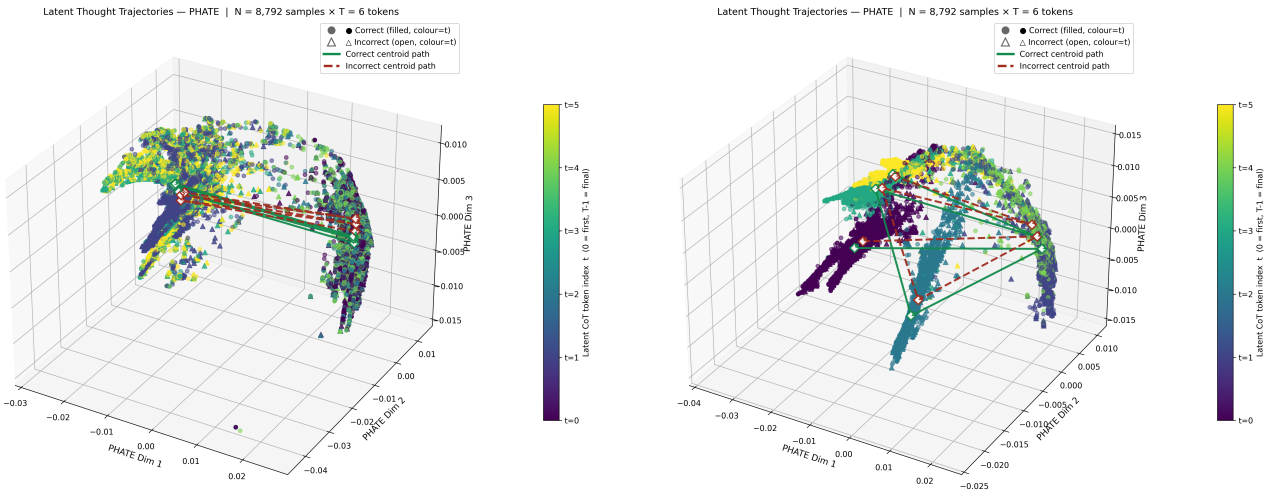


Figure 11. 3D PHATE trajectory projections across latent steps for CODI (left) and COCONUT(right).

A.2.4. 3D Qualitative Trajectory Plots of Sim-CoT CODI and COCONUT with GSM8K

DMD Unlike Vanilla CODI (Figure 10, left), CODI under Sim-CoT paradigm (Figure 12, left) shows that the latent states start near the center for $t{=}0$ and spread outward by $t{=}5$ showing that the model starts diverging through latent space at later reasoning steps like Vanilla COCONUT (Figure 10, right) and Sim-CoT COCONUT (Figure 12, right).

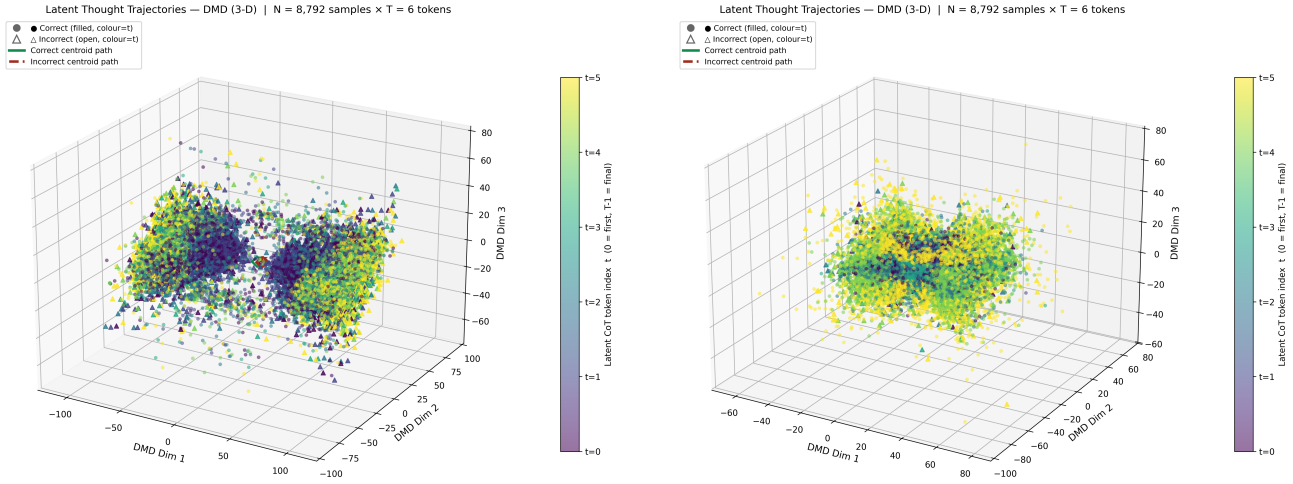


Figure 12. 3D DMD trajectory projections across latent steps for Sim-CoT CODI (left) and Sim-CoT COCONUT(right).

PHATE We can see from the COCONUT PHATE projections that each latent step explores different latent regions, demonstrating their distinct role in the reasoning process (Figure 13, right).

Unlike COCONUT, CODI shows a unclear similar clustering between latent steps (Figure 13, left).

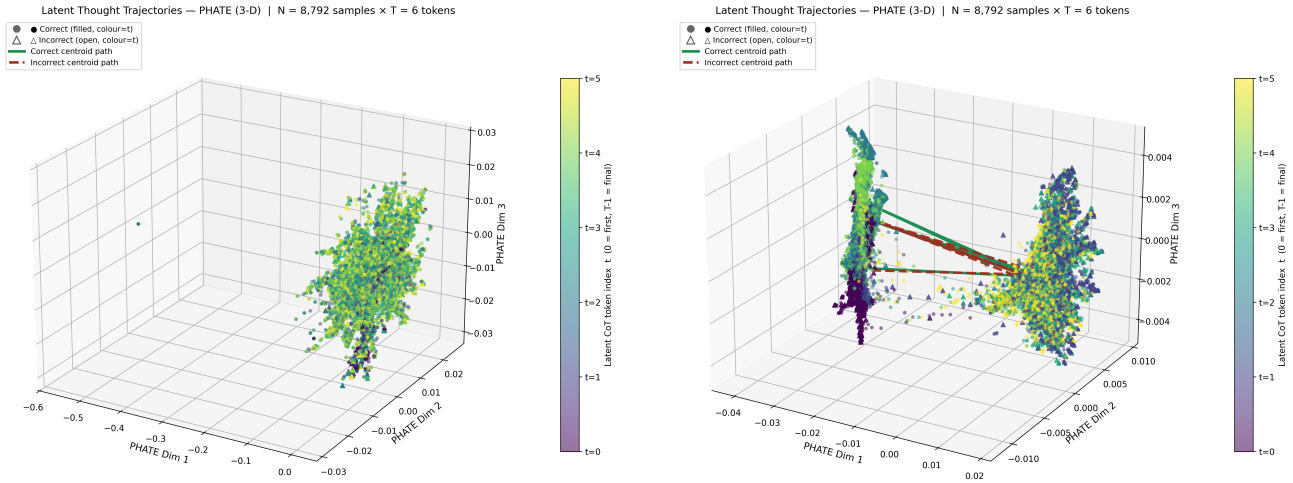


Figure 13. 3D PHATE trajectory projections across latent steps for Sim-CoT CODI (left) and Sim-CoT COCONUT(right).

A.3. Metric Plots

A.3.1. Arc Length

Figures 14–17 show the arc length distributions for CODI and COCONUT under Vanilla and Sim-CoT settings on GSM8K ($N = 8{,}792$ each). Each figure shows three panels: (a) the arc-length histogram split by correct (green) and incorrect (red) predictions with mean lines, (b) a boxplot of arc length by correctness, and (c) a per-concept strip plot with mean $\pm 1\sigma$ across the seven GSM8K concept categories. For both methods and both training paradigms, correct predictions show slightly longer arc lengths than incorrect ones, with CODI's distribution shifted to a higher absolute range than COCONUT's; Sim-CoT compresses CODI's distribution while leaving COCONUT's largely unchanged, indicating that the supervision reduces CODI's reasoning effort without altering COCONUT's.

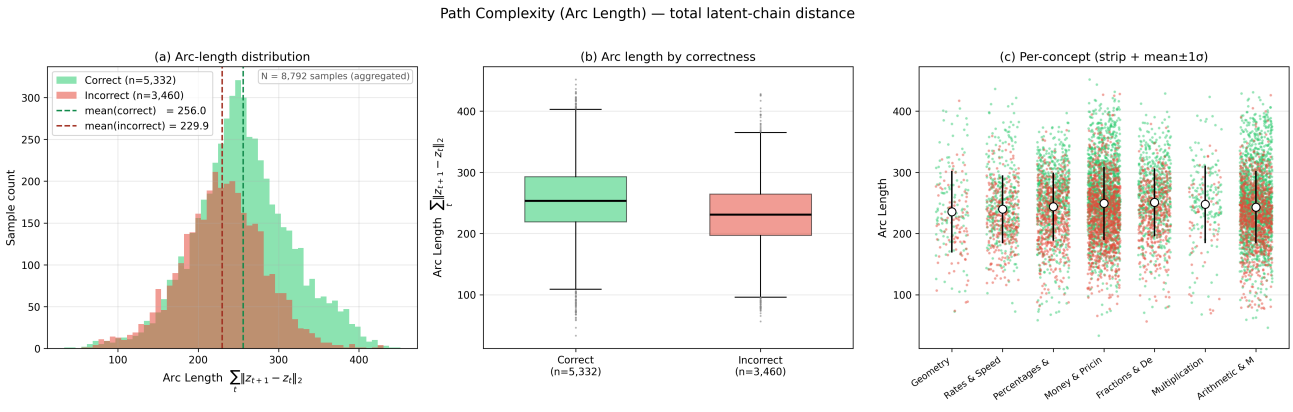


Figure 14. Arc length distribution for CODI on GSM8K (Vanilla setting,$N = 8{,}792$).

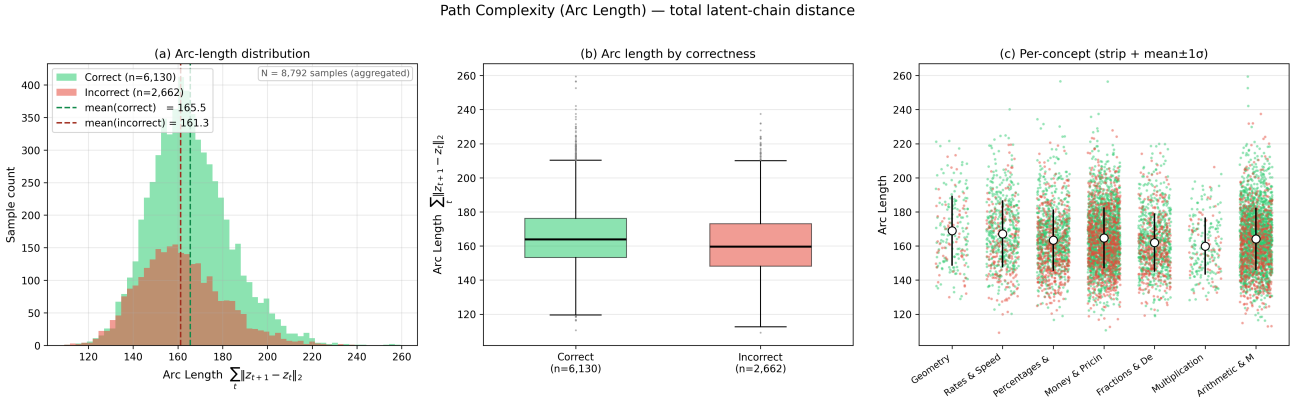


Figure 15. Arc length distribution for COCONUT on GSM8K (Vanilla setting,$N = 8{,}792$).

A.3.2. Perturbation Stability

Figures 18 and 19 show the perturbation relative divergence

$$\|z_t^{\text{perturbed}} - z_t^{\text{clean}}\|_2 / \|z_t^{\text{clean}}\|_2$$

for SIM-CoT CODI and SIM-CoT COCONUT on GSM8K across three Gaussian noise levels ($\sigma \in \{0.01, 0.1, 1.0\}$). Each panel shows correct (green) and incorrect (red) mean curves with ± 1 standard-deviation bands across $n = 3$ perturbation runs per sample. The incorrect-above-correct ordering is preserved at $\sigma \in \{0.01, 0.1\}$ for both methods,a sign that uncertain reasoning paths are more sensitive to input-embedding perturbations; COCONUT's relative divergence sits consistently higher than CODI's at every σ, reflecting COCONUT's smaller clean-state magnitudes amplifying the proportional noise effect.

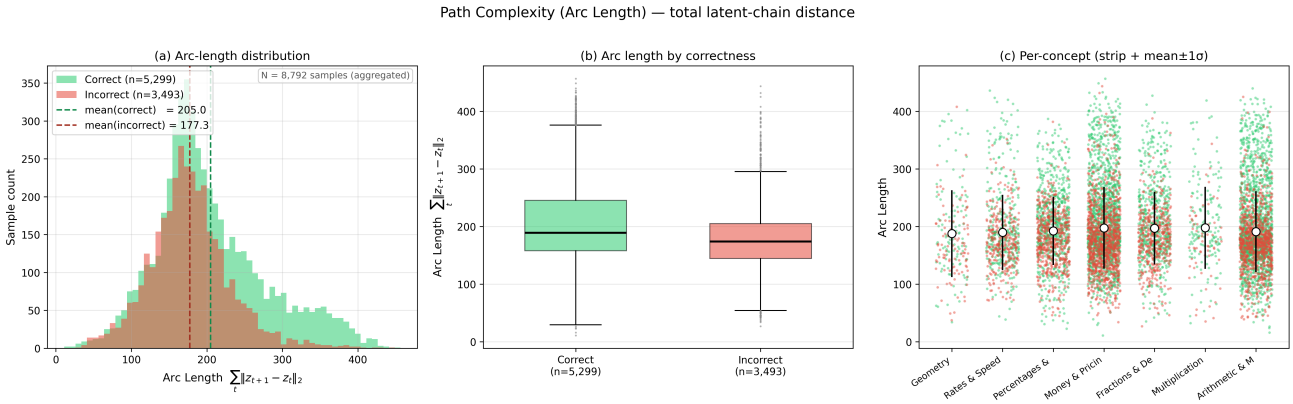


Figure 16. Arc length distribution for SIM-CoT CODI on GSM8K($N = 8{,}792$).

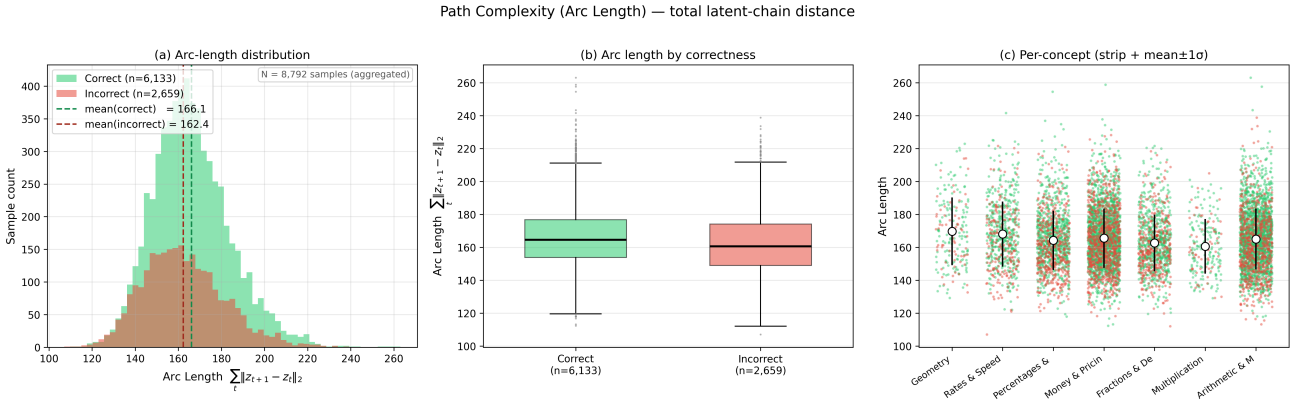


Figure 17. Arc length distribution for SIM-CoT COCONUT on GSM8K ($N = 8{,}792$).

A.3.3. Concept-wise Metric Plots

Figures 20–22 show the same metrics broken down by the seven GSM8K math concept categories. Each panel shows correct (green) and incorrect (red) mean curves with standard deviation bands for one concept. The per-concept profiles are consistent with the aggregated results above, confirming that the observed dynamics are a property of each model's architecture rather than the distribution of problem types.

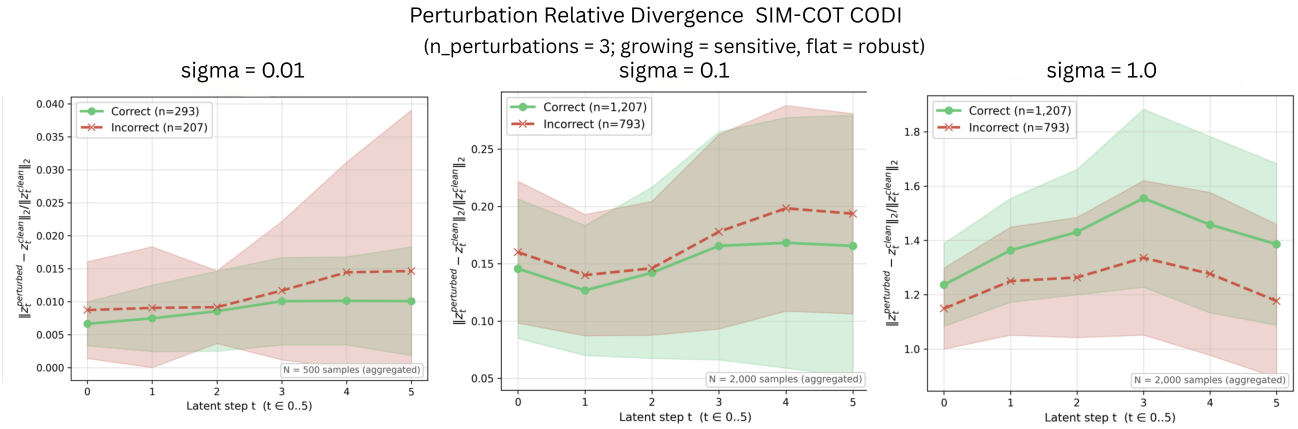


Figure 18. Perturbation relative divergence for SIM-CoT CODI on GSM8K at $\sigma = 0.01$ (left), $\sigma = 0.1$ (middle),and $\sigma = 1.0$ (right).

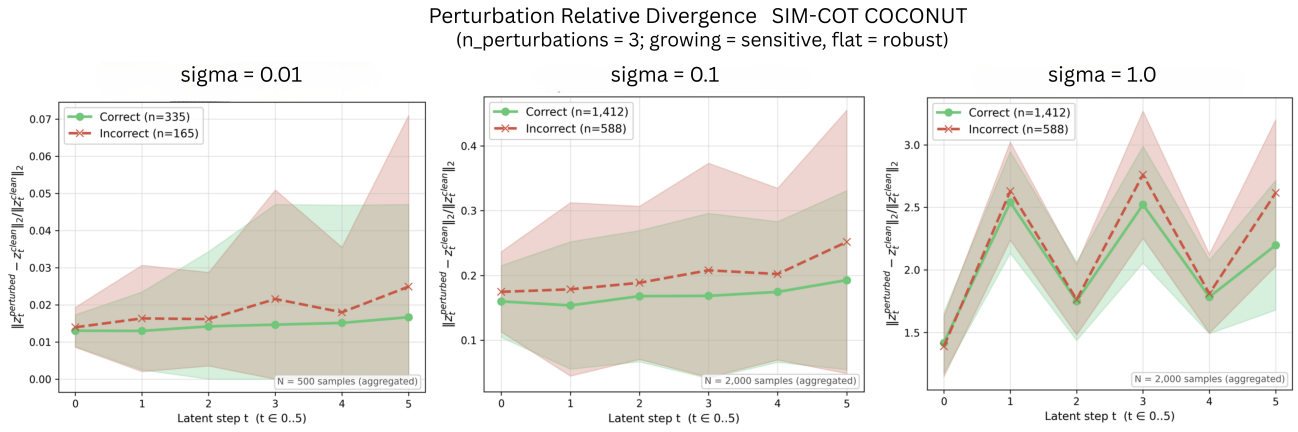


Figure 19. Perturbation relative divergence for SIM-CoT COCONUT on GSM8K at $\sigma = 0.01$ (left), $\sigma = 0.1$ (middle),and $\sigma = 1.0$ (right).

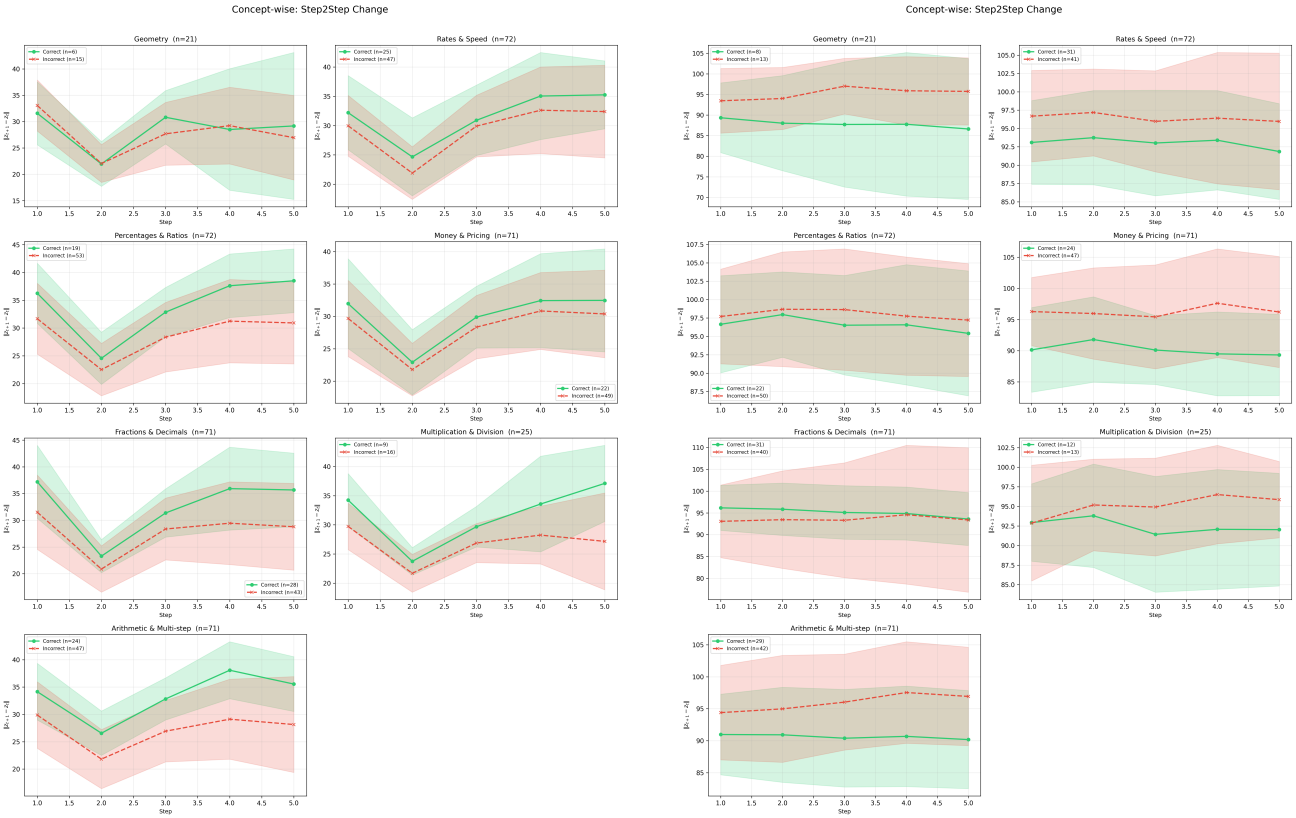


Figure 20. Concept-wise step-to-step change $\|\Delta_t\|$ on GSM8K across seven math concept categories. Left: COCONUT. Right: CODI.

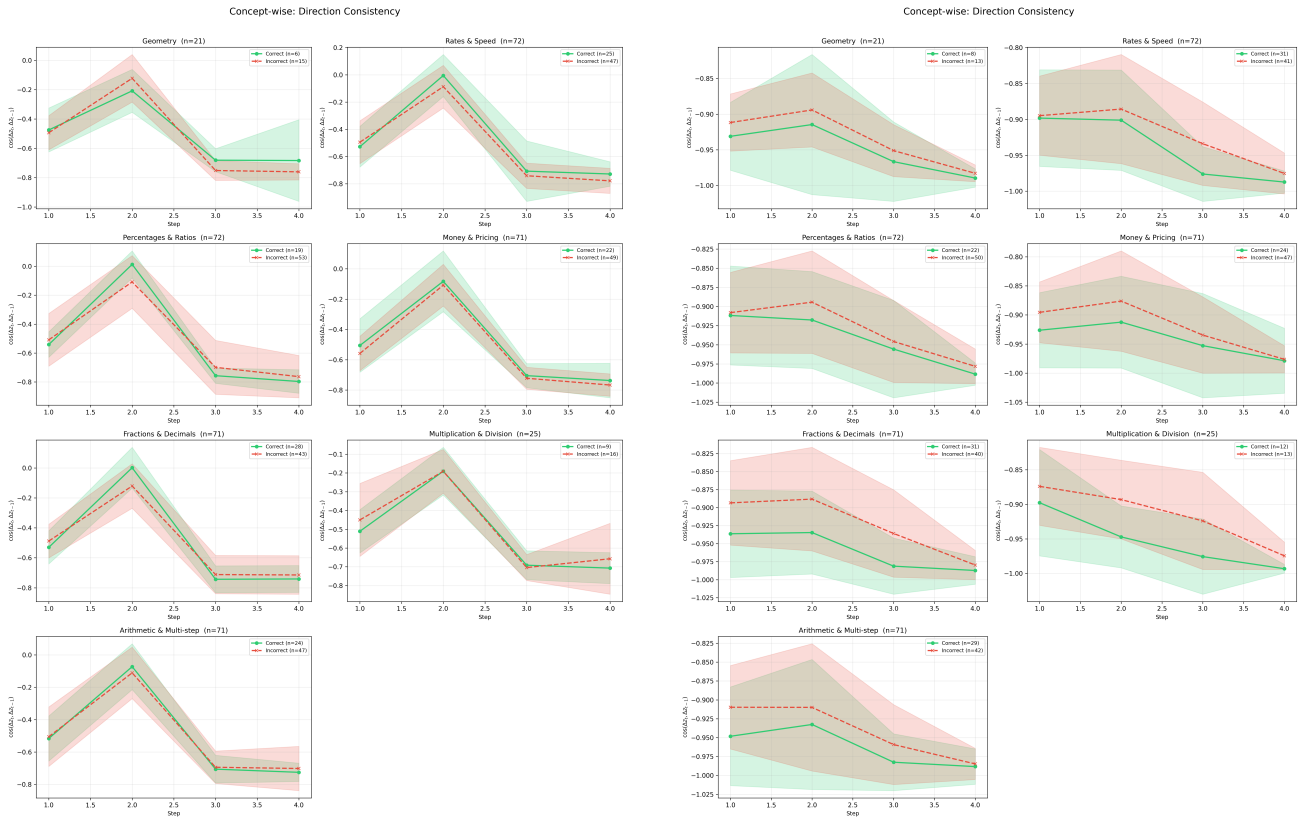


Figure 21. Concept-wise direction consistency C_t on GSM8K. Left: COCONUT. Right: CODI.

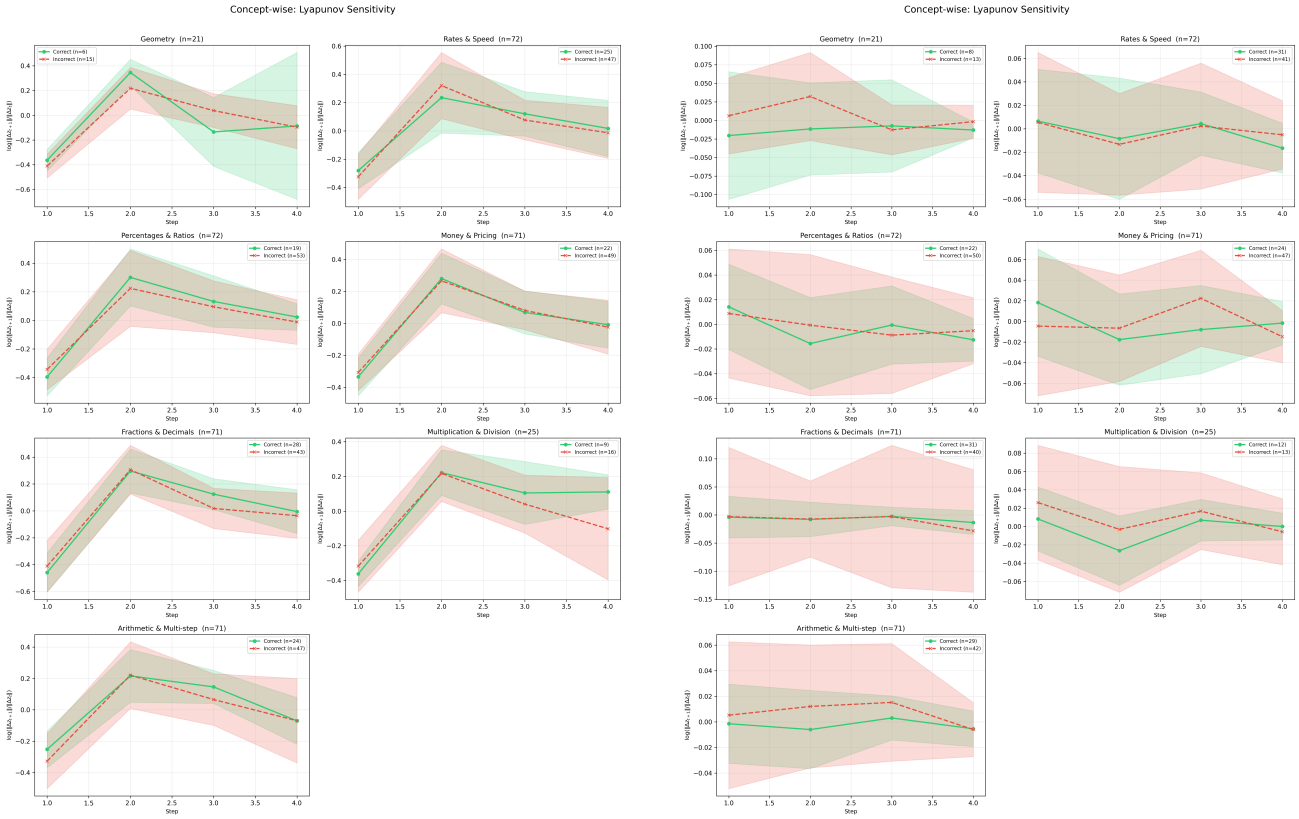


Figure 22. Concept-wise Lyapunov sensitivity $\lambda(t)$ on GSM8K. Left: COCONUT. Right: CODI.