# DYNAMICS WITHIN LATENT CHAIN-OF-THOUGHT: AN EMPIRICAL STUDY OF CAUSAL STRUCTURE

**Zirui Li**[1], **Xuefeng Bai**[1,*], **Kehai Chen**[1], **Yizhi Li**[2,3], **Jian Yang**[2,4], **Chenghua Lin**[3], **Min Zhang**[1]
[1]Harbin Institute of Technology, Shenzhen (HITSZ)
[2]M-A-P
[3]University of Manchester
[4]Beihang University
zirui.li@stu.hit.edu.cn, baixuefeng@hit.edu.cn

## ABSTRACT

Latent or continuous chain-of-thought methods replace explicit textual rationales with a number of internal latent steps, but these intermediate computations are difficult to evaluate beyond correlation-based probes. In this paper, we view latent chain-of-thought as a manipulable causal process in representation space by modeling latent steps as variables in a structural causal model (SCM) and analyzing their effects through step-wise $\mathrm{do}$-interventions. We study two representative paradigms (i.e., Coconut and CODI) on both mathematical and general reasoning tasks to investigate three key questions: (1) which steps are causally necessary for correctness and when answers become decidable early; (2) how does influence propagate across steps, and how does this structure compare to explicit CoT; and (3) do intermediate trajectories retain competing answer modes, and how does output-level commitment differ from representational commitment across steps. We find that latent-step budgets behave less like homogeneous extra depth and more like staged functionality with non-local routing, and we identify a persistent gap between early output bias and late representational commitment. These results motivate mode-conditional and stability-aware analyses—and corresponding training/decoding objectives—as more reliable tools for interpreting and improving latent reasoning systems. Code is available at https://github.com/J1mL1/causal-latent-cot.

## 1 INTRODUCTION

Large language models (LLMs) have achieved strong performance on mathematical problem solving and logical question answering (Cobbe et al., 2021; Geva et al., 2021). A widely adopted technique is Chain-of-Thought (CoT) prompting, which improves accuracy by eliciting intermediate reasoning steps in natural language (Wei et al., 2022). Despite its empirical effectiveness, explicit CoT incurs substantial decoding cost, often produces verbose outputs, and may contain post-hoc rationalizations that do not faithfully reflect the computations driving model predictions (Pruthi et al., 2020; Turpin et al., 2023). These limitations motivate a shift from reasoning in tokens to reasoning in representations.

Recent work explores latent or continuous CoT, where multi-step inference is carried out in continuous hidden representations rather than long textual explanations (Hao et al., 2024; Shen et al., 2025; Zhang et al., 2025; Xu et al., 2025; Gozeten et al., 2025). This paradigm promises a higher-bandwidth internal workspace and reduced decoding overhead, but it faces two fundamental interpretability challenges: 1) intermediate computations are no longer exposed as discrete, human-editable steps; 2) reasoning-relevant information is often distributed across latent dimensions and iterative steps. Consequently, traditional analytical methods—such as step editing or ablation—cannot be directly applied to implicit CoT, leaving the causal and mechanistic analysis of implicit CoT remaining underexplored.

*Corresponding author.

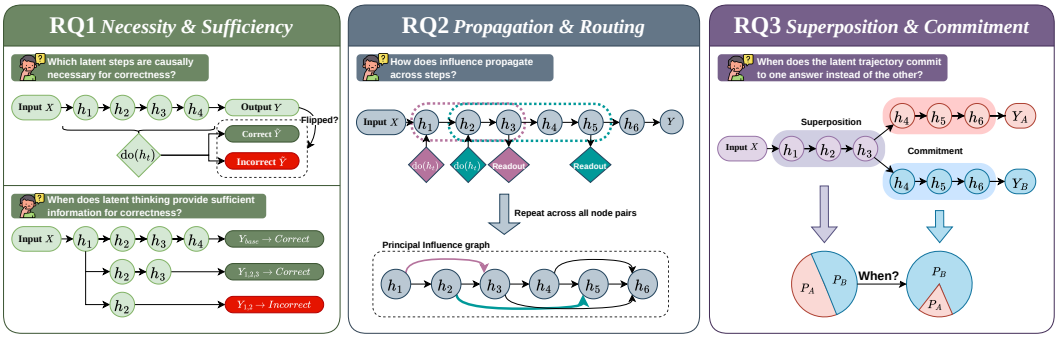



Figure 1: **Overview of step-centric research questions for latent CoT.** RQ1 tests step necessity and early decodability; RQ2 characterizes step-to-step influence propagation; RQ3 probes trajectory-level superposition and commitment across rollouts.

Motivated by this research gap, we conceptualize latent CoT as a **causal system** (Pearl, 2000; Yao et al., 2021) evolving over latent-step variables and evaluate it with intervention-based causal analysis. Specifically, we treat model-defined intermediate latent states as the unit of a reasoning "step" and perform step-wise do-interventions (Singh et al., 2022; Kaddour et al., 2022) that modify an intermediate state while keeping downstream computation unchanged. We then quantify the causal sensitivity of the model's output to these interventions. Furthermore, we aggregate step-to-step influences to construct a directed acyclic graph (DAG) (Thulasiraman & Swamy, 1992) of information flow, derived from intervention-induced shifts in teacher-forced output distributions. This graph characterizes how information propagates through the latent reasoning process. Figure 1 provides a high-level roadmap of the three questions we study, organized from *phenomenon* to *mechanism* to *nature*: (RQ1) which latent steps are causally necessary for correctness and when the answer becomes decodable, (RQ2) how influence propagates across steps, and (RQ3) whether intermediate trajectories retain competing hypotheses and how commitment evolves across steps.

We instantiate this causal evaluation framework on two representative latent-reasoning paradigms, Coconut (Hao et al., 2024) and CODI (Shen et al., 2025), across both mathematical and general reasoning tasks. Empirically, we find a phenomenon–mechanism–nature pattern: *(phenomenon)* causal leverage is highly heterogeneous across latent steps, with a small subset exerting outsized influence; *(mechanism)* step-to-step effects are often non-local, indicating routed propagation rather than purely chain-like transmission; and *(nature)* output-level preference can emerge earlier than representational consolidation, revealing a persistent gap between early bias and later commitment.

Our main contributions are threefold: **(1) the first causal, step-resolved evaluation view of latent CoT** that distinguishes when a solution becomes available from which steps remain causally necessary; **(2) an operator- and readout-conditioned influence analysis** that recovers dominant propagation routes while avoiding sparsity over-claims; and **(3) mode-conditional evidence that links latent-step budgets to practical design implications**: early "decision signals" need not imply early commitment, so improving latent reasoning likely requires shaping routing/commitment rather than simply adding more steps.

## 2 Evaluation Framework: Latent CoT as a Causal System

### 2.1 Scope and Causal Queries

We view latent chain-of-thought (latent CoT) as a manipulable causal process evolving in continuous representation space. Specifically, we treat intermediate latent reasoning steps as variables in a structural causal model (SCM) (Pearl, 2000; Yao et al., 2021), which allows us to pose intervention-based causal queries and compute reproducible effect estimates under a fixed intervention protocol. This viewpoint complements mechanistic analyses that intervene on internal activations to localize task-relevant computation (Elhage et al., 2022; Nanda et al., 2023).

Our analysis provides a unified intervention and readout protocol (Figure 2) and uses it to answer three step-centric causal questions. **(RQ1: necessity and sufficiency)** asks whether individual latent

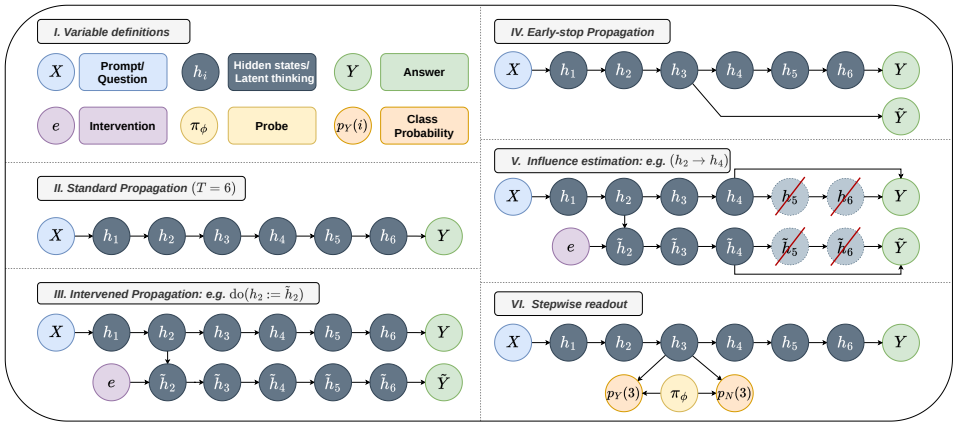



Figure 2: **Intervention-based protocol for latent CoT as a causal system. (I) Variables.** Input $X$ induces a latent trajectory $\{h_t\}_{t=1}^{T}$ and output $Y$; an intervention operator implements step-wise $\mathrm{do}(h_t \leftarrow \tilde{h}_t)$; a readout maps hidden states to answer support. **(II) Standard propagation.** Unperturbed dynamics from $X$ through steps to $Y$. **(III) Intervened propagation.** We replace a step's state while keeping downstream computation intact, yielding an intervened outcome $\tilde{Y}$ (RQ1). **(IV) Early-stop propagation.** We truncate latent computation after step $k$ and decode from $h_k$ to test when correctness becomes decodable (RQ1). **(V) Influence estimation.** Combining a step-$t$ intervention with an early readout at step $s$ yields directed propagation strengths $W_{t,s}$ summarized as an empirical influence structure (RQ2). **(VI) Step-wise readouts.** We read out the answer competition from $h_t$ (e.g., teacher forcing or a probe) to characterize superposition and commitment (RQ3).

steps are behaviorally necessary, and whether the full latent budget is required to make the correct answer decodable. **(RQ2: propagation and routing)** asks how a perturbation at an upstream step propagates to downstream computation, summarized as a step-to-step influence matrix and visualized as an empirical influence graph. **(RQ3: superposition and commitment)** asks whether intermediate trajectories toward answer retain competing answer hypotheses and how output-level commitment relates to representational commitment across steps.

### 2.2 Causal Variables, Minimal SCM, and Latent-step Interface

For an input problem $x$, we model latent reasoning as a sequence of continuous causal variables $H_{1:T} = (H_1, \ldots, H_T)$ with $H_t \in \mathbb{R}^d$, where each $H_t$ corresponds to the model's internal latent state at reasoning step $t$. We denote the task-level output by $Y$ (e.g., the final predicted answer/label), and treat it as a random variable whose conditional distribution is induced by the model's decoder given the latent trajectory. A minimal SCM consistent with this computation is

$$H_t = f_t(H_{<t}, x, \epsilon_t; \theta), \quad t = 1, \ldots, T, \tag{1}$$

$$Y = g(H_{1:T}, x, \epsilon_y; \theta), \tag{2}$$

where $f_t$ and $g$ are the model's transition and decoding mechanisms, and $\epsilon_t, \epsilon_y$ capture stochasticity.

In our experiments, we intervene on the intermediate states produced by a latent thinking rollout. We denote realized latent states by lowercase $h_t$ and write

$$h_{1:T} \sim p_\theta(H_{1:T} \mid x), \tag{3}$$

where $h_t \in \mathbb{R}^d$ instantiates the random variable $H_t$ at step $t$ under the model dynamics. Latent-reasoning models expose these variables through a fixed-length sequence of hidden states $h_{1:T} = (h_1, \ldots, h_T)$, where $h_t$ is the last-layer hidden representation associated with the $t$-th latent step (e.g., a continuous "thought token" in COCONUT or the designated reasoning position in CODI) and is used as the step-$t$ reasoning input embedding.

Given input $x$ and a realized trajectory $h_{1:T}$, the intervention $\mathrm{do}(h_t \leftarrow \tilde{h}_t)$ replaces the latent state at step $t$ by $\tilde{h}_t$ and then propagates the resulting change through all later steps using the same transition mechanism, yielding a counterfactual trajectory $\tilde{h}_{1:T}$. Formally, let $\tilde{h}_{<t} = h_{<t}$ and let $\tilde{h}_t$

be the overwritten state; for $t' > t$ we set

$$\tilde{h}_{t'} \;:=\; f_{t'}(\tilde{h}_{<t'}, x, \tilde{\epsilon}_{t'}; \theta), \tag{4}$$

where $\tilde{\epsilon}_{t'}$ matches the baseline randomness when applicable. The corresponding counterfactual output is obtained by the same readout mechanism,

$$\tilde{y} = g(\tilde{h}_{1:T}, x, \tilde{\epsilon}_y; \theta). \tag{5}$$

Unless otherwise stated, we use deterministic rollouts whenever possible; otherwise we control randomness (e.g., fixed seeds) and isolate propagation effects via teacher-forced readouts (Figure 2(V)) to reduce sampling noise.

### 2.3 Paradigms of Latent-reasoning Models

We instantiate it on two latent-reasoning paradigms diverging in their realization of latent steps.

**Coconut** (Hao et al., 2024) uses an explicit latent mode: it treats the final hidden state as a continuous reasoning token and feeds it back as input to the next step, rather than decoding a discrete token.

**CODI** (Shen et al., 2025) compresses discrete CoT into continuous space via self-distillation: a continuous-CoT student is trained to both produce the correct answer and align its hidden states, at specific reasoning steps, with those of a discrete-CoT teacher, encouraging the latent trajectory to inherit stepwise structure.

### 2.4 Models and Data

**Models.** We experiment with CODI and Coconut on multiple backbones. We use official CODI checkpoints for GPT-2 (Radford et al., 2019) and Llama3-1B (Grattafiori et al., 2024), and reproduce it on Qwen3-4B-Instruct (Yang et al., 2025). Coconut models are reproduced across the same three backbones (GPT-2, Llama3-1B, Qwen3-4B-Instruct). Implementation details are in Appendix A.

**Datasets.** We train and evaluate on CoT-augmented datasets that follow previous latent-reasoning methods. For GSM8K, we train on GSM8K-Aug (Deng et al., 2023) and evaluate on the original GSM8K test set (Cobbe et al., 2021). For CommonsenseQA, we use the CoT-augmented training set released by CODI (Shen et al., 2025) and evaluate on the original CommonsenseQA test set (Talmor et al., 2019). Dataset details are provided in Appendix B.

## 3 RQ1: Step-wise Necessity and Sufficiency

Latent reasoning replaces long textual rationales with a fixed-length sequence of intermediate hidden states, enabling step-wise manipulation under the intervention plus readout protocol introduced in Sec. 2 (Figure 2 (III–IV)). RQ1 asks two causal questions aligned with this protocol: **(i) step-wise necessity,** i.e., whether the final decoded decision depends on an intermediate state, and **(ii) latent-budget sufficiency,** i.e., how many latent steps are required before the answer becomes decodable.

### 3.1 Step-wise Interventions: Setting and Findings

#### 3.1.1 Experiment Setting

We evaluate step-wise necessity using the step-wise $\mathrm{do}$-intervention in Equation 4. For each example, we run a baseline rollout and an intervened rollout that modifies exactly one latent state $h_t$ while keeping all other components unchanged, including the input $x$, parameters $\theta$, and the downstream transition and readout mechanisms, as exemplified in Figure 2 (IV). Following Appendix A.2, we adopt the **zero intervention** for its simplicity and consistency across different models:

$$\mathrm{do}(h_t \leftarrow \tilde{h}_t), \quad \tilde{h}_t = \mathbf{0}. \tag{6}$$

We then decode the final prediction $\tilde{y}^{(t)}$ and compute the **flip rate** $\mathrm{Flip}(t)$, defined as the fraction of examples for which $\tilde{y}^{(t)}$ differs from the baseline output:

$$\mathrm{Flip}(t) \;=\; \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\Big[\tilde{y}_i^{(t)} \neq y_i\Big], \tag{7}$$

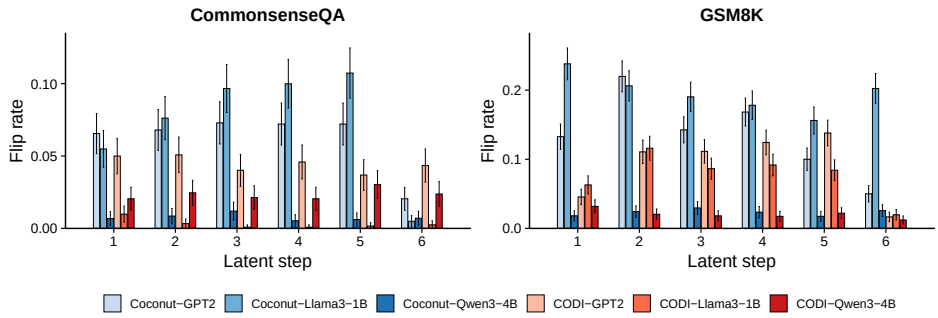



Figure 3: **Step-wise necessity measured by decision instability.** We intervene at a single latent step $t \in \{1, \ldots, 6\}$ by zeroing its state, $\mathrm{do}(h_t := \mathbf{0})$, and then decode the final answer. We report the flip rate $\mathrm{Flip}(t)$, i.e., the fraction of examples whose decoded prediction changes relative to the baseline, on CommonsenseQA (left) and GSM8K (right). Error bars indicate estimation uncertainty.

where $y_i$ is the baseline prediction for example $i$, $\tilde{y}_i^{(t)}$ is the prediction under $\mathrm{do}(h_t \leftarrow \mathbf{0})$, $N$ is the number of evaluated examples, and $\mathbb{I}[\cdot]$ is the indicator function.

#### 3.1.2 Findings

**Single-step ablation produces clear, step-specific flip patterns (not uniformly sensitive).** Within the same model and dataset, $\mathrm{Flip}(t)$ changes noticeably with the intervened step $t$ (Figure 3), with several settings exhibiting mid-step peaks rather than a flat or monotone pattern. This indicates that different latent steps contribute differently to the final decision, and that single-step removal can selectively disrupt the decision more at some steps than others.

**Arithmetic exhibits substantially higher decision volatility than commonsense.** Flip rates on GSM8K are higher than on CommonsenseQA under the same intervention protocol, with several backbones reaching around $0.1$–$0.2$ or higher on GSM8K, while CommonsenseQA stays mostly below $\sim 0.1$ (Figure 3). This gap appears across both Coconut and CODI variants. This suggests that arithmetic solutions rely more on intermediate latent computation under our protocol, whereas commonsense decisions are more stable to the same step-wise perturbations.

**Coconut shows larger flips than CODI under matched backbones, while stronger backbones suppress flips.** Under the same backbone, Coconut variants generally yield higher flip rates than CODI, especially on GSM8K (Figure 3). Moreover, while stronger backbones substantially reduce flip rates across both paradigms, the flipping profile remains step-dependent even when absolute rates are low. These results point to paradigm-dependent robustness: scaling reduces absolute sensitivity, but does not erase structured, step-specific dependence.

### 3.2 Early-stop Decoding: Setting and Findings

#### 3.2.1 Experiment Setting

We perform early-stop decoding by truncating latent computation after step $k$ and decoding directly from $h_k$ (Figure 2 (IV)), determining when correctness first becomes readable from the latent trajectory. For example $i$, let $\hat{y}_i^{(\leq k)}$ be the decoded answer when we truncate latent computation after step $k$ and decode from $h_{i,k}$. We define the earliest step at which the correct answer becomes decodable:

$$k_i = \min\big(\{k \in \{1, \ldots, T\} : \hat{y}_i^{(\leq k)} = y_i^*\} \cup \{\infty\}\big). \tag{8}$$

Let $k_i$ be defined in Equation 8. We define then the cumulative solved fraction $S(k)$ as

$$S(k) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\{k_i \leq k\}. \tag{9}$$

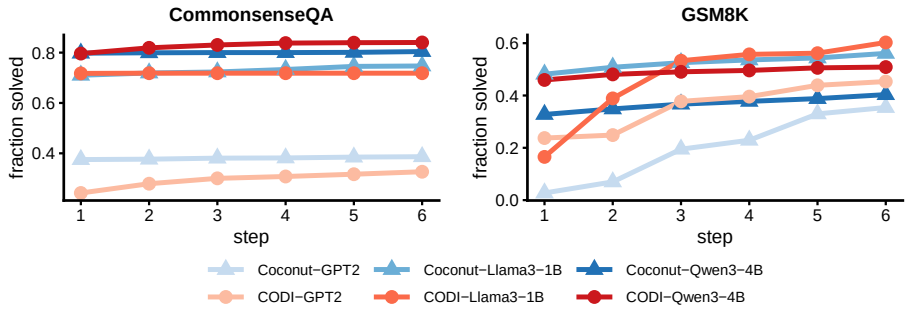



Figure 4: **Early-stop decoding reveals when correctness becomes decodable.** We report the cumulative solved fraction $S(k) = \mathbb{P}(k_i \leq k)$ (Equation 9) under early-stop decoding on CommonsenseQA (left) and GSM8K (right), where $k_i$ is the earliest step at which the correct answer becomes decodable (Equation 8).

### 3.2.2 Findings

**Early-stop curves differ across datasets.** On CommonsenseQA, $S(k)$ typically rises rapidly within the first few steps and then saturates (Figure 4, left), suggesting most solvable instances become decodable within few steps. In contrast, on GSM8K $S(k)$ often continues to increase toward later steps (Figure 4, right), with several settings showing gains up to $k$=6, indicating that additional latent computation can expand the set of instances for which the correct answer becomes decodable.

**Backbone strength shapes when correctness becomes decodable, while paradigm-level similarity is weak.** Across both datasets, stronger backbones tend to have higher $S(1)$ and earlier saturation of $S(k)$, whereas weaker backbones improve more gradually with $k$ (Figure 4). At the same time, the step-wise growth profiles do not cluster by training paradigm: Coconut and CODI do not consistently exhibit similar curve shapes within each paradigm across backbones, suggesting that "when correctness becomes decodable" is not a stable paradigm-level signature in this experiment.

## 4 RQ2: Information Flow and Stepwise Influence Structure

RQ1 establishes that intervening on a single latent step can change the final decision, but it does not reveal where this perturbation propagates along the reasoning trajectory. In RQ2, we compare step-to-step propagation of explicit CoT and latent CoT using an influence matrix $W$ (Eq. 11) and its principal influence graph rendering for readability. We couple a single-step intervention at step $t$ with an early readout at a downstream step $s$, and measure how strongly the intervention changes the teacher-forced output distribution for the gold answer; see Figure 2 (IV). Unless otherwise stated, we report GSM8K in the main text; CommonsenseQA results are provided in Appendix C.2.

### 4.1 Experiment Setting

**Nodes.** For latent-reasoning models, each node corresponds to a latent step $t \in \{1, \ldots, T\}$ with state $h_t$ (Sec. 2). For explicit CoT baselines (CoT-SFT), we segment the generated rationale into at most $T$=6 steps and represent step $t$ by the last-layer hidden state at the final token of that segment.[1] This yields matched step-level trajectories for latent and explicit reasoning.

**Edges via intervention + early decoding.** To probe directed propagation from step $t$ to a downstream step $s > t$, we run (i) a baseline rollout and (ii) an intervened rollout that modifies exactly one step state while keeping downstream computation unchanged. We then *decode at step* $s$ using teacher forcing to obtain output distributions $p^{(s)}_{\text{base}}(\cdot)$ and $p^{(s)}_{\text{do}(t)}(\cdot)$, and define the example-level propagation strength as a position-averaged KL shift:

$$\mathrm{KL}^{(i)}_{t\to s} = \frac{1}{|y^*_i|} \sum_{u=1}^{|y^*_i|} \mathrm{KL}\Big(p^{(s)}_{\text{base}}(\cdot \mid y^*_{i,<u}) \,\|\, p^{(s)}_{\text{do}(t)}(\cdot \mid y^*_{i,<u})\Big). \tag{10}$$

[1] If an example has fewer than $T$ CoT steps, the remaining nodes are absent.

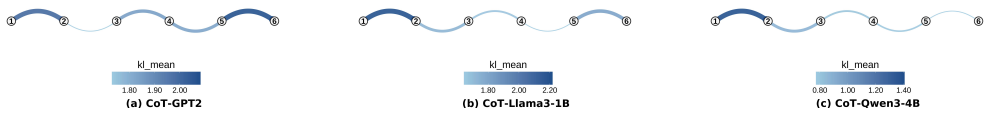



Figure 5: **Explicit CoT principal influence graphs (GSM8K; CoT-SFT baselines).** Nodes denote the first $T$=6 segmented CoT steps. Edge $t \to s$ indicates propagation strength $W_{t,s}$ from Eq. 11 (teacher-forced KL shift on the gold answer under a single-step intervention at $t$ and readout at $s$). For readability we show only top-1 outgoing edges after thresholding at $\alpha$=$0.1 \cdot \max(W)$.

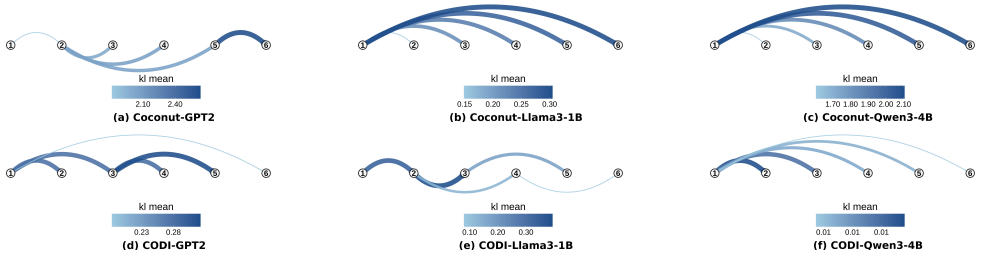



Figure 6: **Latent principal influence graphs (GSM8K; COCONUT/CODI).** Nodes are latent steps $t \in \{1, \ldots, 6\}$. Edge weights follow Eq. 11 under single-step interventions, rendered with the same sparsification protocol as Figure 5.

Aggregating over evaluation examples yields the influence matrix $W \in \mathbb{R}^{T \times T}$:

$$W_{t,s} = \mathbb{E}_i \big[ \mathrm{KL}^{(i)}_{t \to s} \big], \quad t < s, \tag{11}$$

**Principal influence graph rendering.** For compact comparison across models, we visualize $W$ as a sparsified principal influence graph: we drop edges below $\alpha \cdot \max(W)$ with $\alpha$=0.1 and retain only the top-1 outgoing edge per node. Edge thickness scales with $W_{t,s}$. Dense heatmaps are provided in Appendix C.1.

**Structural summaries.** To quantify the heatmap-level patterns beyond visual inspection, we compute four normalized structure metrics on the dense $W$ (definitions in Appendix C.3): **locality** (mass near the diagonal), **span** (expected hop distance), **early-out** (influence originating from early steps), and **late-in** (influence terminating at late steps). We normalize $W$ to remove scale effects across backbones before computing these metrics.

### 4.2 FINDINGS

**Explicit CoT influence graphs remain near-chain across backbones.** CoT-SFT exhibits a consistently sequential topology: the dominant edges follow adjacent transitions, with only minor deviations across backbones (Figure 5). This stability is also reflected in the structure summaries on GSM8K (Figure 7): CoT-SFT has uniformly high locality (all $\geq 0.6$) with low span, matching the intuition that textual steps induce predominantly local dependencies.

**Latent influence graphs are dominated by skip connections, revealing non-chain propagation.** Latent graphs contain substantially more skip connections than explicit CoT (Figure 6), indicating that influence often bypasses intermediate steps rather than accumulating strictly along a local chain. This is also captured by the structure summaries on GSM8K (Figure 7): latent models are markedly less local and have larger spans than CoT-SFT, and they place substantially more normalized influence into late-step targets (late-in). Within this skip-dominant regime, COCONUT tends to exhibit more pronounced early→late routing (often connecting early steps directly to the final step), while CODI departs from a strict chain but is generally less dominated by early→final shortcuts and shows greater variation across backbones (Figure 6). Despite this difference between the two paradigms, the skip connections together with the witnessed non-uniform sensitivity in Figure 3 indicate the difference in functionality across the nodes in the latent trajectory.

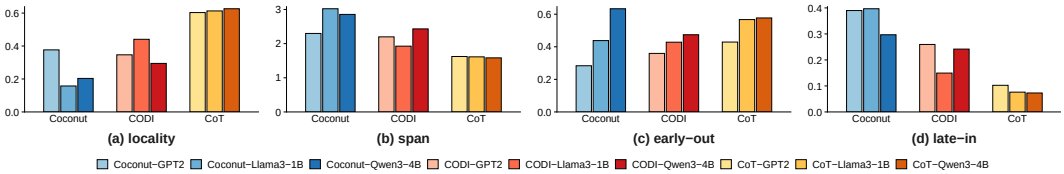



Figure 7: **Structural summaries of influence matrices (GSM8K).** We report locality, span, early-out, and late-in computed on dense normalized $W$ (Appendix C.3).

## 5 RQ3: Superposition and Commitment in Latent Dynamics

RQ1 demonstrates that removing a single latent step can alter the final decision, while RQ2 reveals step-to-step influence is non-local: early computation can directly affect multiple later steps, bypassing adjacent ones. A remaining ambiguity is whether such non-locality reflects (i) *early commitment* to one answer mode, which is then propagated, or (ii) *sustained competition* among multiple hypotheses within the latent trajectory. Prior work offers contrasting views: some argue continuous "soft thinking" remains effectively greedy and single-threaded (Wu et al., 2025b), while others suggest latent reasoning may retain competing hypotheses in shared subspaces (Zhu et al., 2025a).

RQ3 therefore asks a trajectory-level question: **when stochastic rollouts of the same prompt lead to different final answers, do intermediate steps exhibit superposition, and how does this evolve across steps?** We instantiate this analysis on **StrategyQA** (Geva et al., 2021), whose binary label space (`Yes`/`No`) clearly defines two modes and allows direct probability-based readout at each step. We do not apply the same two-cluster filtering to GSM8K, as its open-ended numeric answers yield many distinct output strings under sampling, making prompts with exactly two dominant modes too sparse for reliable analysis.

### 5.1 Experiment Setting

**Two-mode prompts from stochastic rollouts.** For each prompt $x$, we enable stochastic decoding and sample $K$ rollouts. Each rollout produces a latent trajectory $\{h_t^{(k)}\}_{t=1}^{T}$ and a final answer $\hat{y}^{(k)} \in \{\texttt{Yes}, \texttt{No}\}$. We retain prompts whose rollouts contain both answers, and partition rollouts into $\mathcal{C}_Y$ and $\mathcal{C}_N$ accordingly. Filtering thresholds and strategies are reported in Appendix D.1.

**Intermediate-step readouts (teacher-forced vs. probe).** At each latent step $t$, we quantify the model's relative support for `Yes` vs. `No` using two log-probability readouts: (i) a **teacher-forced** scoring under a fixed answer template, and (ii) a fixed **probe** protocol that maps $h_t$ to next-token probabilities in a manual probe context. Both yield a step-wise binary distribution $p_Y(t), p_N(t)$. Templates and probing details are provided in Appendix D.2.

**Superposition score.** Given $p_Y(t), p_N(t)$, we define the step-wise superposition score

$$\mathrm{SS}(t) = \min\big(p_Y(t), p_N(t)\big), \tag{12}$$

which is high when both answers retain non-trivial support and low when one answer dominates.

### 5.2 Findings

**Teacher-forced readout suggests early output-level commitment.** In Figure 8 (right), $\mathrm{SS}(t)$ is uniformly low and varies only modestly across steps, indicating that the model's answer distribution becomes strongly skewed toward either `Yes` or `No` early in the latent budget. A minimal implication is that the non-local edges observed in RQ2 can be compatible with early mode selection followed by propagation, without requiring prolonged output-level ambiguity (cf. the non-local principal influence graphs in Figure 6).

**Probe readout reveals more sustained competition and a late collapse.** In Figure 8 (left), the probe panel shows substantially higher $\mathrm{SS}(t)$ throughout the trajectory, with a clear drop at the final step, implying that intermediate states can retain decodable support for the alternative mode even when teacher forcing appears committed. This readout gap highlights operator dependence:

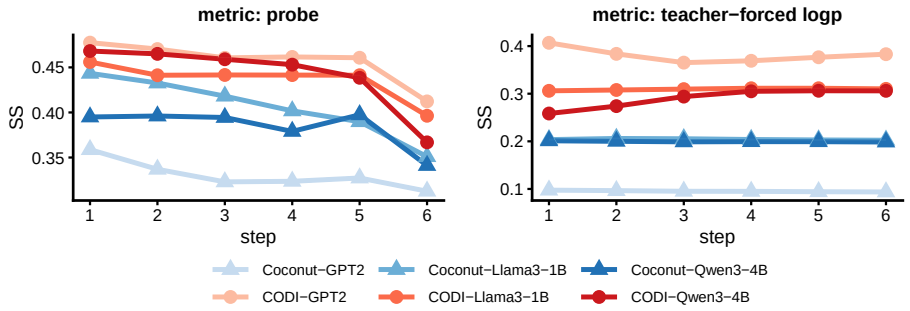



Figure 8: **RQ3: Step-wise probing readout on StrategyQA.** Left: probe readout; right: teacher-forced log-probability readout. The superposition score SS measures how much support is simultaneously retained for `Yes` and `No` at each latent step.

intermediate representations may remain "multi-mode available" beyond what is expressed by the default answer distribution.

**Paradigm-level separation aligns with RQ2's structure differences.** Under probe readout, CODI variants exhibit higher superposition than COCONUT variants across steps (Figure 8), matching the qualitative separation in RQ2 where COCONUT more often shows dominant early→late principal edges while CODI is less extreme (Figure 6 and Figure 7).

## 6 DISCUSSION

Our experiments offer a unified, step-centric perspective on latent-token reasoning by connecting three complementary views: step-wise causal necessity and early decodability (RQ1), directed propagation summarized at the step level (RQ2), and trajectory-level mode dynamics under stochastic rollouts (RQ3). Taken together across COCONUT and CODI, these analyses suggest that a fixed latent budget functions less like homogeneous extra depth and more like a structured interface: steps have unequal causal leverage, influences can route non-locally across the trajectory, and apparent output-level commitment need not coincide with the underlying representational state.

**Latent steps are causally functional, with heterogeneous leverage.** RQ1 indicates that latent computation is broadly engaged: intervening on a single step can change the decoded decision, but the effect is not evenly distributed across the budget. A useful lens is to treat the step index as an implicit interface for *division of labor*: certain steps act as high-leverage intervention sites whose removal disrupts downstream computation, while others appear to contribute more conditionally, surfacing as sensitivity only on specific inputs or reasoning modes. This helps interpret the non-monotonic profiles without assuming that the model simply "refines the same state" at every step; instead, latent reasoning may introduce step-specific updates whose downstream effect is later amplified, transformed, or gated. The fact that COCONUT and CODI allocate leverage differently on matched backbones further suggests that training paradigm shapes where decision-relevant dependence concentrates along the trajectory.

**Minimal computation towards the correct answer is distinct from the corresponding commitment.** Early-stop decoding offers a complementary notion of "how much reasoning is needed": it measures when the correct answer first becomes readable from the latent state, rather than whether a step remains behaviorally necessary when removed. Seen together with intervention sensitivity, this separates *availability* from *stability*: a solution can become decodable at an intermediate step while later computation still consolidates the final decision, reduces volatility or mitigates unstable mode switching. This distinction also clarifies how RQ1 and RQ3 coexist: early readability does not imply irreversible commitment, and late-step sensitivity can reflect stabilization even when the correct answer is already representationally present.

**Superposition and commitment: shared computation versus early collapse.** Interpreted through the lens of RQ2, superposition can be interpreted as *shared prefix computation*: competing solutions can reuse substantial intermediate processing before divergence becomes externally visible. While separation between probe-based and teacher-forced readouts in RQ3 suggests that "commitment" is

not a single event: output distributions can collapse earlier than intermediate representations cease to carry information about alternatives. This gap is consistent with late-step stabilization effects seen in RQ1, and indicates that intermediate states may retain mode-relevant structure even when the default decoding path is already biased. This is consistent with recent evidence that behavioral/preference directions can be isolated and steered in representation space via lightweight inference-time interventions (Rimsky et al., 2024; Li et al., 2023; Zou et al., 2025; Turner et al., 2024; Zhang et al., 2026). Together, RQ1–RQ3 point to a division of labor across steps, with shared intermediate computation and paradigm-dependent mechanisms for robust mode dominance in the final decision. Additional discussion of principal influence structure is provided in Appendix E.

## 7 Related Work

**Latent and continuous chain-of-thought reasoning.** To avoid the cost and potential unfaithfulness of explicit CoT, recent methods move reasoning into continuous representations (Zhu et al., 2025b). A first family performs depth-iterative latent reasoning, where the model recurrently updates a hidden state or continuous "thought token" before decoding, as in Coconut (Hao et al., 2024) and CODI (Shen et al., 2025). Related approaches incorporate additional supervision or alignment to stabilize latent reasoning (Wei et al., 2025; He et al., 2025), while hybrid methods mix latent and textual steps to balance efficiency and interpretability (Su et al., 2025). A second family of studies test-time compute and recurrent-depth paradigms that scale computation without emitting long text, including recurrent depth approaches and parallel continuous updates (Geiping et al., 2025; Wu et al., 2025a; Zhu et al., 2025c). Most literature emphasizes accuracy and efficiency; our work instead focuses on the internal causal organization of latent reasoning and how it relates to explicit CoT.

**Chain-of-thought faithfulness and causal tests.** A growing literature questions whether model-provided explanations—including CoT rationales—faithfully reflect the computations that determine a model's final prediction. Models can generate convincing yet unfaithful artifacts—from attention explanations and CoT rationales to structured parses (Pruthi et al., 2020; Turpin et al., 2023; Bai et al., 2025). To assess faithfulness, prior work employs causal tests via context- or rationale-level interventions (e.g., deleting, shuffling, or editing rationales) and measures the resulting changes in downstream answers (Tutek et al., 2025; Wang et al., 2023; Yang et al., 2023; Yu et al., 2025). Complementarily, parameter-level interventions such as unlearning remove information associated with specific reasoning steps from model weights, offering another causal probe of whether a step is internally represented and used (Tutek et al., 2025).

**Causal and mechanistic analysis of internal representations.** Causal perspectives on neural networks view hidden activations as mediators that can be intervened upon to test necessity and sufficiency (Pearl, 2000; Feder et al., 2022; Schölkopf et al., 2021). Mechanistic approaches (e.g., ablation and activation patching) have revealed algorithmic structure in model internals (Elhage et al., 2022; Nanda et al., 2023). Recent work also shows that models can internalize reasoning strategies while concealing them from surface text under process supervision, further decoupling latent computation from explanation (Skaf et al., 2025). These causal and mechanistic toolkits are broadly applicable to our setting as well, providing concrete intervention operators for probing the necessity and functional role of latent states.

## 8 Conclusion

We framed latent chain-of-thought as a step-indexed causal system and evaluated it across Coconut and CODI with interventions, influence-structure estimation, and trajectory-level readouts. Beyond characterizing these models, our results offer design-relevant insights: latent-step budgets should be treated as an allocatable interface rather than homogeneous "extra depth," since causal leverage concentrates unevenly and propagates along a few dominant long-range routes; and training/decoding should account for a gap between early output bias and late representational commitment, where alternatives remain latent-available even after the output distribution tilts. Together, this suggests that future latent-reasoning models can be improved by explicitly shaping where information is written and how it is consolidated across steps (e.g., encouraging more stable bottlenecks or controllable routing), rather than only scaling the number of latent steps.

## REFERENCES

Xuefeng Bai, Jialong Wu, Yulong Chen, Zhongqing Wang, Kehai Chen, Min Zhang, and Yue Zhang. Constituency Parsing Using LLMs. *IEEE Transactions on Audio, Speech and Language Processing*, 33:3762–3775, 2025. doi: 10.1109/TASLPRO.2025.3600867.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems, November 2021. URL http://arxiv.org/abs/2110.14168. arXiv:2110.14168 [cs].

Yuntian Deng, Kiran Prasad, Roland Fernandez, Paul Smolensky, Vishrav Chaudhary, and Stuart Shieber. Implicit Chain of Thought Reasoning via Knowledge Distillation. 2023. doi: 10.48550/ARXIV.2311.01460. URL https://arxiv.org/abs/2311.01460. arXiv:2311.01460 [cs].

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy Models of Superposition, September 2022. URL http://arxiv.org/abs/2209.10652. arXiv:2209.10652 [cs].

Amir Feder, Katherine A. Keith, Emaad Manzoor, Reid Pryzant, Dhanya Sridhar, Zach Wood-Doughty, Jacob Eisenstein, Justin Grimmer, Roi Reichart, Margaret E. Roberts, Brandon M. Stewart, Victor Veitch, and Diyi Yang. Causal Inference in Natural Language Processing: Estimation, Prediction, Interpretation and Beyond. *Transactions of the Association for Computational Linguistics*, 10:1138–1158, 2022. doi: 10.1162/tacl_a_00511. URL https://aclanthology.org/2022.tacl-1.66/.

Jonas Geiping, Sean McLeish, Neel Jain, John Kirchenbauer, Siddharth Singh, Brian R. Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, and Tom Goldstein. Scaling up Test-Time Compute with Latent Reasoning: A Recurrent Depth Approach, February 2025. URL http://arxiv.org/abs/2502.05171. arXiv:2502.05171 [cs].

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did Aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021. doi: 10.1162/tacl_a_00370. URL https://aclanthology.org/2021.tacl-1.21/.

Halil Alperen Gozeten, M. Emrullah Ildiz, Xuechen Zhang, Hrayr Harutyunyan, Ankit Singh Rawat, and Samet Oymak. Continuous chain of thought enables parallel exploration and reasoning, 2025. URL https://arxiv.org/abs/2505.23648. arXiv:2505.23648 [cs].

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, and et al. The Llama 3 Herd of Models, November 2024. URL http://arxiv.org/abs/2407.21783. arXiv:2407.21783 [cs].

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training Large Language Models to Reason in a Continuous Latent Space, December 2024. URL http://arxiv.org/abs/2412.06769. arXiv:2412.06769 [cs] version: 1.

Yinhan He, Wendy Zheng, Yaochen Zhu, Zaiyi Zheng, Lin Su, Sriram Vasudevan, Qi Guo, Liangjie Hong, and Jundong Li. SemCoT: Accelerating Chain-of-Thought Reasoning through Semantically-Aligned Implicit Tokens, October 2025. URL http://arxiv.org/abs/2510.24940. arXiv:2510.24940 [cs].

Jean Kaddour, Aengus Lynch, Qi Liu, Matt J. Kusner, and Ricardo Silva. Causal Machine Learning: A Survey and Open Problems, July 2022. URL http://arxiv.org/abs/2206.15475. arXiv:2206.15475 [cs].

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: eliciting truthful answers from a language model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability, October 2023. URL http://arxiv.org/abs/2301.05217. arXiv:2301.05217 [cs].

Judea Pearl. *Causality: models, reasoning, and inference*. Cambridge University Press, Cambridge New York, NY Port Melbourne New Delhi Singapore, second edition, reprinted with corrections edition, 2000. ISBN 978-0-521-89560-6.

Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig, and Zachary C. Lipton. Learning to Deceive with Attention-Based Explanations. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4782–4793, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.432. URL https://aclanthology.org/2020.acl-main.432/.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.828. URL https://aclanthology.org/2024.acl-long.828/.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Towards Causal Representation Learning, February 2021. URL http://arxiv.org/abs/2102.11107. arXiv:2102.11107 [cs].

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. CODI: Compressing Chain-of-Thought into Continuous Space via Self-Distillation. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 677–693, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.36. URL https://aclanthology.org/2025.emnlp-main.36/.

Rahul Singh, Liyuan Xu, and Arthur Gretton. Kernel Methods for Causal Functions: Dose, Heterogeneous, and Incremental Response Curves, October 2022. URL http://arxiv.org/abs/2010.04855. arXiv:2010.04855 [econ].

Joey Skaf, Luis Ibanez-Lissen, Robert McCarthy, Connor Watts, Vasil Georgiv, Hannes Whittingham, Lorena Gonzalez-Manzano, David Lindner, Cameron Tice, Edward James Young, and Puria Radmard. Large language models can learn and generalize steganographic chain-of-thought under process supervision, December 2025. URL http://arxiv.org/abs/2506.01926. arXiv:2506.01926 [cs].

DiJia Su, Hanlin Zhu, Yingchen Xu, Jiantao Jiao, Yuandong Tian, and Qinqing Zheng. Token Assorted: Mixing Latent and Text Tokens for Improved Language Model Reasoning, September 2025. URL http://arxiv.org/abs/2502.03275. arXiv:2502.03275 [cs].

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *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)*, pp. 4149–4158, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1421. URL https://aclanthology.org/N19-1421/.

K. Thulasiraman and M. N. S. Swamy. *Graphs: theory and algorithms*. John Wiley & Sons, Inc., USA, 1992. ISBN 0471513563.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering, 2024. URL https://arxiv.org/abs/2308.10248. arXiv:2308.10248 [cs].

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: unfaithful explanations in chain-of-thought prompting. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, pp. 74952–74965, Red Hook, NY, USA, December 2023. Curran Associates Inc.

Martin Tutek, Fateme Hashemi Chaleshtori, Ana Marasovic, and Yonatan Belinkov. Measuring Chain of Thought Faithfulness by Unlearning Reasoning Steps. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9946–9971, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.504. URL https://aclanthology.org/2025.emnlp-main.504/.

Fei Wang, Wenjie Mo, Yiwei Wang, Wenxuan Zhou, and Muhao Chen. A causal view of entity bias in (large) language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 15173–15184, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.1013. URL https://aclanthology.org/2023.findings-emnlp.1013/.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, pp. 24824–24837, Red Hook, NY, USA, November 2022. Curran Associates Inc. ISBN 978-1-7138-7108-8.

Xilin Wei, Xiaoran Liu, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Jiaqi Wang, Xipeng Qiu, and Dahua Lin. SIM-CoT: Supervised Implicit Chain-of-Thought, September 2025. URL http://arxiv.org/abs/2509.20317. arXiv:2509.20317 [cs].

Haoyi Wu, Zhihao Teng, and Kewei Tu. Parallel Continuous Chain-of-Thought with Jacobi Iteration. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 914–926, Suzhou, China, November 2025a. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.47. URL https://aclanthology.org/2025.emnlp-main.47/.

Junhong Wu, Jinliang Lu, Zixuan Ren, Gangqiang Hu, Zhi Wu, Dai Dai, and Hua Wu. LLMs are Single-threaded Reasoners: Demystifying the Working Mechanism of Soft Thinking, October 2025b. URL http://arxiv.org/abs/2508.03440. arXiv:2508.03440 [cs].

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. SoftCoT: Soft Chain-of-Thought for Efficient Reasoning with LLMs. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23336–23351, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1137. URL https://aclanthology.org/2025.acl-long.1137/.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 Technical Report, May 2025. URL http://arxiv.org/abs/2505.09388. arXiv:2505.09388 [cs].

Zhen Yang, Yongbin Liu, and Chunping Ouyang. Causal intervention-based few-shot named entity recognition. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 15635–15646, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.1046. URL https://aclanthology.org/2023.findings-emnlp.1046/.

Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. A survey on causal inference. *ACM Trans. Knowl. Discov. Data*, 15(5), May 2021. ISSN 1556-4681. doi: 10.1145/3444944. URL https://doi.org/10.1145/3444944.

Xiangning Yu, Zhuohan Wang, Linyi Yang, Haoxuan Li, Anjie Liu, Xiao Xue, Jun Wang, and Mengyue Yang. Causal sufficiency and necessity improves chain-of-thought reasoning, 2025. URL https://arxiv.org/abs/2506.09853. arXiv:2506.09853 [cs].

Yu Zhang, Jinlong Ma, Yongshuai Hou, Xuefeng Bai, Kehai Chen, Yang Xiang, Jun Yu, and Min Zhang. Evaluating and steering modality preferences in multimodal large language model, 2026. URL https://arxiv.org/abs/2505.20977. arXiv:2505.20977 [cs].

Zhen Zhang, Xuehai He, Weixiang Yan, Ao Shen, Chenyang Zhao, Shuohang Wang, Yelong Shen, and Xin Eric Wang. Soft Thinking: Unlocking the Reasoning Potential of LLMs in Continuous Concept Space, May 2025. URL http://arxiv.org/abs/2505.15778. arXiv:2505.15778 [cs].

Hanlin Zhu, Shibo Hao, Zhiting Hu, Jiantao Jiao, Stuart Russell, and Yuandong Tian. Reasoning by Superposition: A Theoretical Perspective on Chain of Continuous Thought, November 2025a. URL http://arxiv.org/abs/2505.12514. arXiv:2505.12514 [cs].

Rui-Jie Zhu, Tianhao Peng, Tianhao Cheng, Xingwei Qu, Jinfa Huang, Dawei Zhu, Hao Wang, Kaiwen Xue, Xuanliang Zhang, Yong Shan, Tianle Cai, Taylor Kergan, Assel Kembay, Andrew Smith, Chenghua Lin, Binh Nguyen, Yuqi Pan, Yuhong Chou, Zefan Cai, Zhenhe Wu, Yongchi Zhao, Tianyu Liu, Jian Yang, Wangchunshu Zhou, Chujie Zheng, Chongxuan Li, Yuyin Zhou, Zhoujun Li, Zhaoxiang Zhang, Jiaheng Liu, Ge Zhang, Wenhao Huang, and Jason Eshraghian. A Survey on Latent Reasoning, July 2025b. URL http://arxiv.org/abs/2507.06203. arXiv:2507.06203 [cs] version: 1.

Rui-Jie Zhu, Zixuan Wang, Kai Hua, Tianyu Zhang, Ziniu Li, Haoran Que, Boyi Wei, Zixin Wen, Fan Yin, He Xing, Lu Li, Jiajun Shi, Kaijing Ma, Shanda Li, Taylor Kergan, Andrew Smith, Xingwei Qu, Mude Hui, Bohong Wu, Qiyang Min, Hongzhi Huang, Xun Zhou, Wei Ye, Jiaheng Liu, Jian Yang, Yunfeng Shi, Chenghua Lin, Enduo Zhao, Tianle Cai, Ge Zhang, Wenhao Huang, Yoshua Bengio, and Jason Eshraghian. Scaling latent reasoning via looped language models, 2025c. URL https://arxiv.org/abs/2510.25741. arXiv:2510.25741 [cs].

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency, 2025. URL https://arxiv.org/abs/2310.01405. arXiv:2310.01405 [cs].

# A Implementation Details

## A.1 Training settings for reproduced Coconut and CODI

We reproduce COCONUT across three backbones (GPT-2, Llama-1B-Instruct, and Qwen-4B-Instruct) and three datasets (GSM8K, CommonsenseQA, and StrategyQA) using the standard stage-wise latent-replacement curriculum. On GSM8K, we follow the official two-stage recipe (*CoT-SFT* $\rightarrow$ *Coconut*): the model is first trained with explicit CoT supervision, and then continued with Coconut-style latent reasoning initialized from the CoT checkpoint. For GPT-2, both stages are trained for 25 epochs (25+25). For larger backbones, we use a shorter schedule (5+10) consistent with the large-model setting. On CommonsenseQA and StrategyQA, we adopt a uniform latent curriculum that progressively increases the latent-step budget up to $T$=6 using one epoch per latent stage (6 epochs total), followed by 4 additional epochs at the final stage (i.e., 6+4). Across backbones, we use AdamW-style optimization with weight decay $0.01$ and a backbone-dependent learning rate: GPT-2 runs use $\mathrm{lr} = 10^{-4}$, while Llama/Qwen runs use $\mathrm{lr} = 10^{-5}$.

For CODI, we use *officially released* checkpoints for GSM8K on GPT-2 and Llama3-1B.[2] All other CODI results in this paper are obtained from our reproduced checkpoints trained on the remaining dataset/backbone combinations. We follow the official CODI training pipeline (LoRA-based fine-tuning with a fixed number of latent tokens, cosine learning-rate scheduling with warmup, mixed-precision training, and the projection head enabled in our runs), while allowing dataset/backbone-dependent learning rates for stable optimization. Concretely, our reproduced CODI runs use learning rates in the range $[5 \times 10^{-6},\ 3 \times 10^{-3}]$ depending on the backbone and dataset (e.g., $3 \times 10^{-3}$ for GPT-2, $8 \times 10^{-4}$ for Llama-1B, and $2 \times 10^{-4}/10^{-5}/5 \times 10^{-6}$ for Qwen-4B on GSM8K-Aug/CommonsenseQA/StrategyQA, respectively). This yields CODI checkpoints matched to our Coconut reproductions in latent-step budget ($T$=6) and task coverage, enabling controlled comparisons under identical intervention and readout protocols.

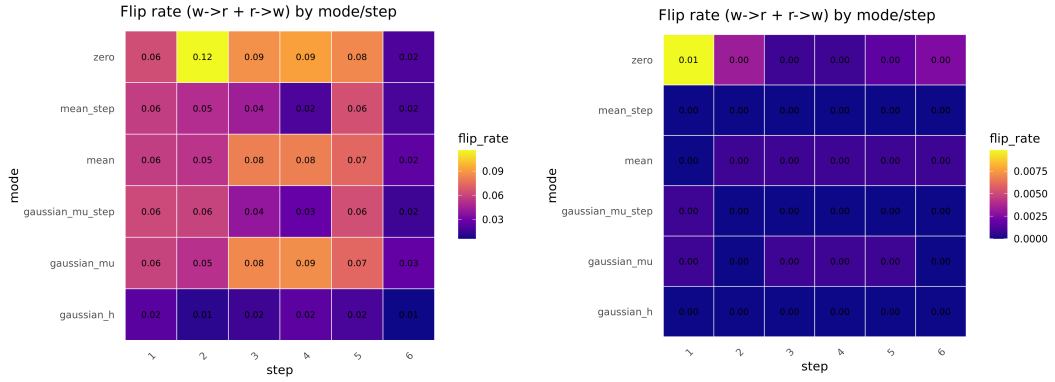
## A.2 Intervention operators: robustness and choice of zero overwrite

Our step-wise necessity analysis (RQ1) instantiates the single-step $\mathrm{do}$-intervention (Equation 4) via a concrete *intervention operator* that maps a realized latent state $h_t$ to an edited state $\tilde{h}_t$. To verify that our conclusions are not tied to a particular operator, we compare six commonly used perturbations that preserve the same causal interface (overwrite one step, then recompute downstream computation under fixed $x$ and $\theta$): `zero` (*replace latent with zeros*), `mean` (*replace with global mean*), `mean_step` (*replace with step-specific mean*), `gaussian_h` (*add Gaussian noise to* $h_t$), `gaussian_mu` (*add Gaussian noise around the global mean*), and `gaussian_mu_step` (*add Gaussian noise around the step-specific mean*). Letting $\mu$ denote the global mean latent state and $\mu_t$ the mean at step $t$ (estimated for a given model/dataset), these operators correspond to: $\tilde{h}_t = \mathbf{0}$ (`zero`), $\tilde{h}_t = \mu$ (`mean`), $\tilde{h}_t = \mu_t$ (`mean_step`), $\tilde{h}_t = h_t + \sigma\epsilon$ (`gaussian_h`), $\tilde{h}_t = \mu + \sigma\epsilon$ (`gaussian_mu`), and $\tilde{h}_t = \mu_t + \sigma\epsilon$ (`gaussian_mu_step`), where $\epsilon \sim \mathcal{N}(0, I)$ and $\sigma$ is a fixed noise scale.

Figures 9a and 9b show two representative examples on CODI (Llama3-1B) for GSM8K and CommonsenseQA, respectively. Each heatmap cell reports the **flip rate** $\mathrm{Flip}(t)$ used throughout RQ1, computed as the fraction of examples whose final decoded prediction changes under the intervention at step $t$. Following our main metric, we aggregate both wrong$\rightarrow$right and right$\rightarrow$wrong flips (i.e., any decision change relative to the baseline rollout), so larger values indicate stronger decision-level dependence on the intervened step. Across operators, the qualitative step-wise sensitivity patterns are stable: operators that substantially perturb the latent state yield similar relative trends over steps, while weaker/noisier operators typically reduce absolute flip rates without altering the overall step profile. This robustness indicates that our RQ1 findings are not driven by a specific perturbation choice.

We adopt **zero overwrite** (`zero`) as the default operator for the main paper for two practical reasons. First, it is deterministic and parameter-free, eliminating tuning choices (e.g., the noise scale $\sigma$) and reducing variance across runs. Second, it applies uniformly across architectures and training recipes, making cross-model comparisons more fair: the intervention strength does not depend on backbone-

[2] We use the official CODI GSM8K checkpoints as provided by the authors.




(a) **GSM8K.** (b) **CommonsenseQA.**

Figure 9: **Intervention-operator comparison (CODI Llama3-1B).** Each cell shows the flip rate $\mathrm{Flip}(t)$, the fraction of examples whose decoded final prediction changes when intervening at step $t$ aggregated over wrong→right and right→wrong flips.

specific hidden-state norms or distributional statistics beyond the shared representation space. We therefore use `zero` throughout the main experiments for reproducibility and interpretability, and treat the remaining operators as sanity checks that validate the stability of our conclusions.

### A.3 Teacher-forced readouts and influence matrix

To measure propagation effects with minimal sampling noise, we use teacher-forced readouts on a canonical gold-answer string. The answer template follows each method's training paradigm (rather than the dataset): for COCONUT we use "[prefix] ### {answer}", while for CODI we use "[prefix] The answer is {answer}". We denote the resulting gold answer token sequence by $a_{1:L}$.

**Teacher-forced distributions.** Given a trajectory (baseline or intervened) and a designated readout step $s$, we compute teacher-forced logits for the gold answer tokens $a_{1:L}$ and obtain token-level predictive distributions

$$p^{(s)}(a_\ell \mid a_{<\ell}, x) \;=\; \mathrm{softmax}\big(\mathbf{z}_\ell^{(s)}\big), \qquad \ell = 1, \ldots, L, \tag{13}$$

where $\mathbf{z}_\ell^{(s)}$ denotes the teacher-forced logit vector at gold token position $\ell$ when reading out from step $s$ (with the same $x$ and fixed parameters $\theta$). We compute these distributions for both the baseline rollout ($p_{\text{base}}^{(s)}$) and the intervened rollout ($p_{\text{int}}^{(s)}$).

**Influence matrix from token-averaged KL.** For an intervention applied at step $t$ and a readout at step $s$, we define the influence weight $W_{t,s}$ (Eq. 11) as the token-averaged KL divergence between the baseline and intervened teacher-forced distributions on the gold answer:

$$W_{t,s} \;=\; \frac{1}{L} \sum_{\ell=1}^{L} D_{\mathrm{KL}}\Big(p_{\text{base}}^{(s)}(\cdot \mid a_{<\ell}, x) \;\Big\|\; p_{\text{int}}^{(s)}(\cdot \mid a_{<\ell}, x)\Big)\,. \tag{14}$$

Intuitively, $W_{t,s}$ measures how much an intervention at step $t$ changes the model's predictive distribution over the gold answer when the trajectory is read out at step $s$. We aggregate $W_{t,s}$ over examples by averaging across the evaluation set.

**Visualization and sparsification.** To visualize influence graphs, we apply the same sparsification protocol as in the main text. Specifically, we threshold edges at $\alpha \cdot \max(W)$ with $\alpha = 0.1$, and for each source step we retain only the top-1 outgoing edge for readability. Edge weights in figures correspond to the (dataset-averaged) dense matrix entries $W_{t,s}$ prior to sparsification.

**Why teacher forcing.** Teacher forcing isolates propagation effects from output sampling variability: it evaluates changes in the model's conditional distribution along a fixed gold answer path, rather than changes in a sampled decoded string. This makes $W$ a more stable proxy for step-to-step influence, especially when the decoded answer is short or when sampling induces high variance across rollouts.

# B Dataset Information

## B.1 Dataset Statistics

Table 1 reports the dataset split sizes used in our experiments. All evaluations use the original benchmark test sets from the corresponding dataset papers. For training, we use CoT-augmented variants when required by the training paradigms: for GSM8K we train on GSM8K-Aug from (Deng et al., 2023) while evaluating on the original GSM8K test set (Cobbe et al., 2021); for CommonsenseQA and StrategyQA we use CoT-augmented training data released by the CODI (Shen et al., 2025) and evaluate on the original test sets (Talmor et al., 2019; Geva et al., 2021). These augmentations affect training-time supervision only; we do not modify any benchmark test set.

Table 1: **Dataset statistics used in our experiments.** Train sizes correspond to the training splits actually used for model training (GSM8K-Aug (Deng et al., 2023); CommonsenseQA-CoT and StrategyQA-CoT from CODI (Shen et al., 2025)). Test sizes correspond to the original benchmark test splits (Talmor et al., 2019; Cobbe et al., 2021; Geva et al., 2021).

| Dataset | Train | Test |
|---|---|---|
| CommonsenseQA | 8,096 | 1,221 |
| GSM8K | 385,620 | 1,319 |
| StrategyQA | 1,809 | 229 |

## B.2 Dataset Examples

The following examples illustrate the CoT-augmented training format used in our runs. CoT rationales are *not* part of the original benchmarks; they are training-time supervision taken from the corresponding augmented variants (GSM8K-Aug (Deng et al., 2023) and CODI-released CoT data (Shen et al., 2025)). We evaluate on the original benchmark test sets without modification.

**GSM8K-Aug (training)**

**Question:** Out of 600 employees in a company, 30% got promoted while 10% received bonus. How many employees did not get either a promotion or a bonus?
**CoT:** 600*30/100=180, 600*10/100=60, 600-240=360
**Answer:** 360

**CommonsenseQA-CoT (training)**

**Question:** Google Maps and other highway and street GPS services have replaced what?
Choices: A: united states B: mexico C: countryside D: atlas E: oceans
**CoT:** Google Maps and other GPS services are tools used for navigation and mapping, which have largely replaced traditional paper maps and atlases that people used in the past for similar purposes. The other options, such as countries and geographic features, do not relate to navigational tools. Thus, the best answer is D: atlas.
**Answer:** D

**StrategyQA-CoT (training)**

**Question:** Would a dog respond to bell before Grey seal?
**CoT:** To determine if a dog would respond to a bell before a Grey seal, we need to consider the context and conditioning involved. Dogs are often trained to respond to bells due to their domestication and close relationships with humans. Classic conditioning experiments, like Pavlov's dogs, indicate that dogs can be trained to associate a bell with specific actions, such as being fed. Grey seals, on the other hand, are wild marine mammals that are not typically subjected to the same kind of training or conditioning that dogs regularly undergo. Therefore, without specific conditioning, a Grey seal is unlikely to naturally associate or respond to a bell in the same way a trained dog would. Thus, a dog, particularly one that has been conditioned, is expected to respond to a bell before a Grey seal.
**Answer:** true

# C Additional RQ2 Details

## C.1 Dense adjacency matrices for influence graphs

The main text visualizes sparsified influence graphs for readability (threshold $\alpha$=0.1 and top-1 outgoing edge per node). Here we provide the corresponding dense influence matrices $W$ (Eq. 11) as heatmaps, which are the objects used to compute all structure metrics.

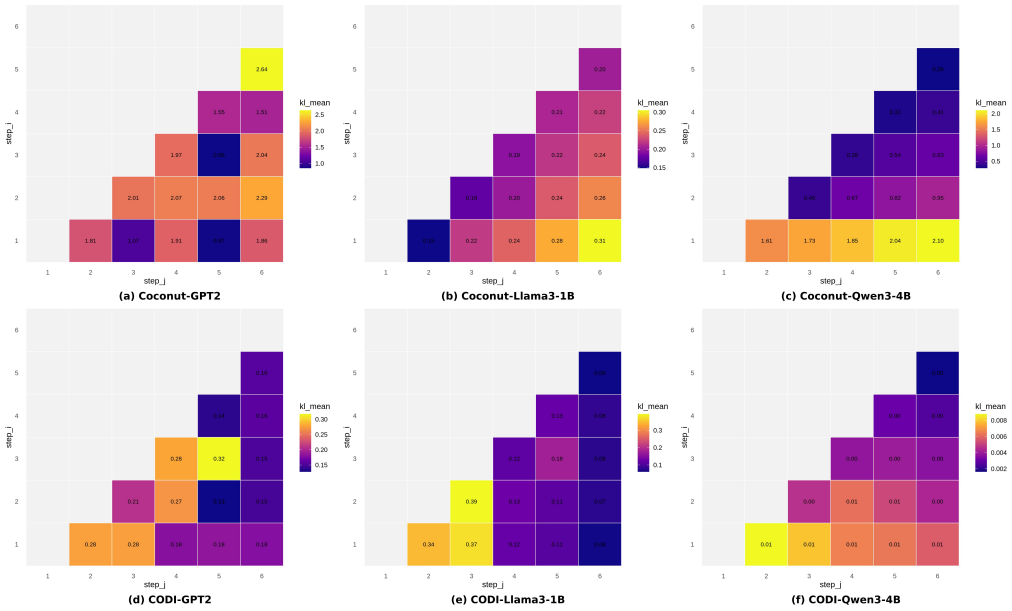



Figure 10: **Dense influence matrices for latent reasoning (GSM8K).** Each cell $(t, s)$ shows $W_{t,s}$ from Eq. 11 (teacher-forced KL shift on the gold answer when intervening at $t$ and reading out at $s$).

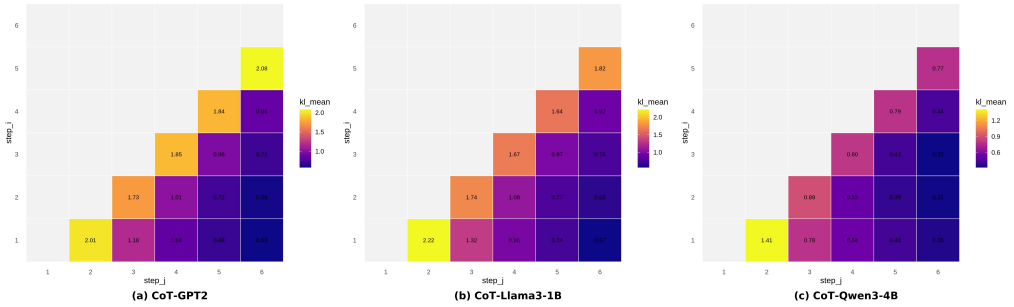



Figure 11: **Dense influence matrices for explicit CoT (GSM8K; CoT-SFT).** Each cell $(t, s)$ shows $W_{t,s}$ from Eq. 11 computed on segmented CoT-step states.

### C.2 Additional CommonsenseQA results

This subsection provides additional RQ2 results on CommonsenseQA, including dense influence matrices (latent and explicit), sparsified principal influence graphs, and the corresponding structure metrics. All quantities follow the same construction as in Sec. 4, based on the dense matrix $W$ from Eq. 11.

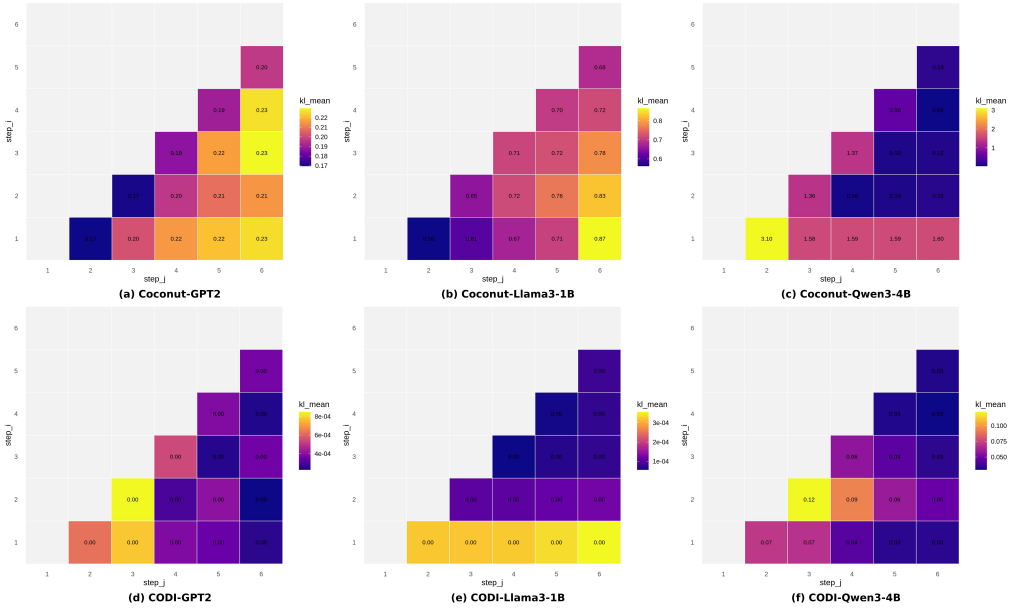



Figure 12: **Dense influence matrices for latent reasoning (CommonsenseQA; COCONUT/CODI).** Each cell $(t, s)$ shows $W_{t,s}$ from Eq. 11 under teacher-forced readouts.

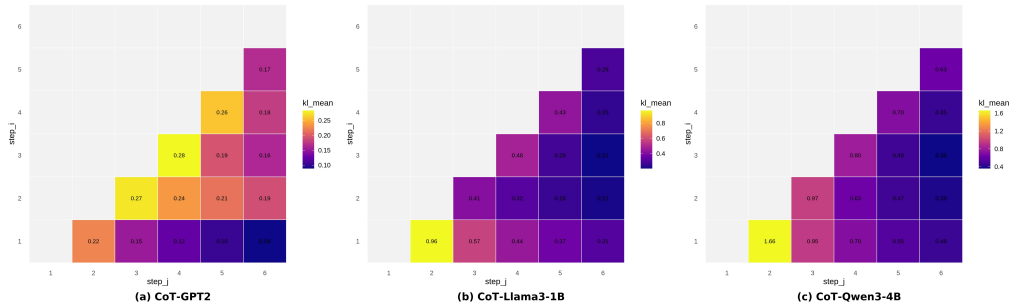



Figure 13: **Dense influence matrices for explicit CoT (CommonsenseQA; CoT-SFT).** Each cell $(t, s)$ shows $W_{t,s}$ from Eq. 11 computed on segmented CoT-step states.

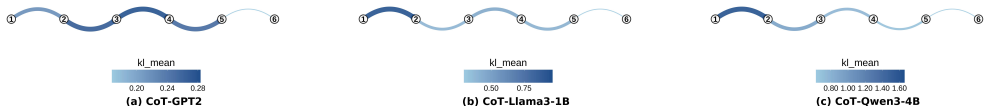



Figure 14: **Explicit CoT principal influence graphs (CommonsenseQA; CoT-SFT baselines).** Nodes denote the first $T$=6 segmented CoT steps. Edge $t \rightarrow s$ indicates propagation strength $W_{t,s}$ from Eq. 11. For readability we show only top-1 outgoing edges after thresholding at $\alpha$=0.1 $\cdot$ $\max(W)$.

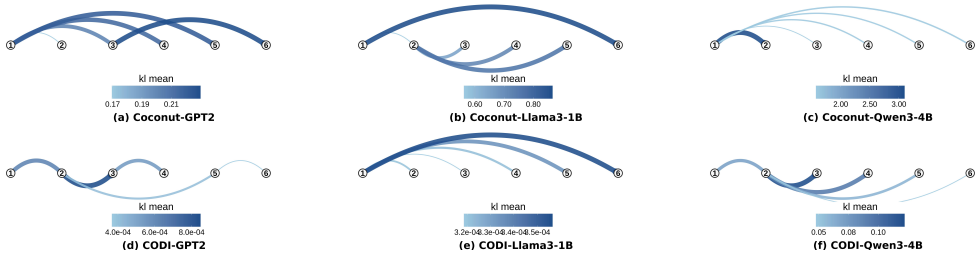



Figure 15: **Latent principal influence graphs (CommonsenseQA; COCONUT/CODI).** Nodes are latent steps $t \in \{1, \ldots, 6\}$. Edge weights follow Eq. 11 under single-step interventions, rendered with the same sparsification protocol as Figure 14.

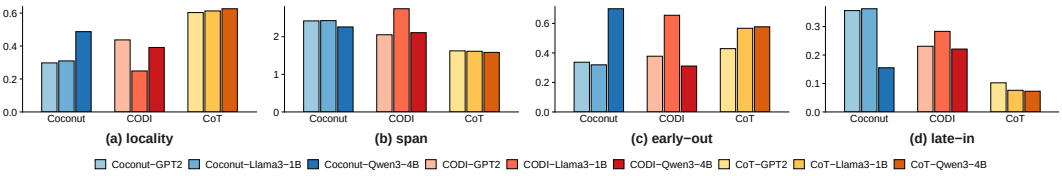



Figure 16: **Structure metrics on influence graphs (CommonsenseQA).** Metrics are computed on the normalized matrix $\bar{W}$ (Eq. 15) and use the same hyperparameters as the main text.

### C.3 DEFINITIONS OF STRUCTURE METRICS

We compute structure metrics on a normalized influence matrix $\bar{W}$ obtained by re-scaling $W$ over valid entries $(t < s)$:

$$\bar{W}_{t,s} = \frac{W_{t,s}}{\sum_{a<b} W_{a,b} + \epsilon}. \tag{15}$$

Let $\mathcal{E} = \{(t, s) : 1 \leq t < s \leq T\}$. We define:

$$\text{Locality}(k) = \sum_{(t,s)\in\mathcal{E}} \mathbf{1}\{s - t \leq k\}\, \bar{W}_{t,s}, \tag{16}$$

$$\text{Span} = \sum_{(t,s)\in\mathcal{E}} (s - t)\, \bar{W}_{t,s}, \tag{17}$$

$$\text{EarlyOut}(m) = \sum_{t\leq m} \sum_{s>t} \bar{W}_{t,s}, \tag{18}$$

$$\text{LateIn}(m) = \sum_{s\geq m} \sum_{t<s} \bar{W}_{t,s}. \tag{19}$$

In all experiments we use $k$=1 and $m$=2 (early) / $m$=5 (late) for $T$=6.

## D ADDITIONAL RQ3 DETAILS

### D.1 TRAJECTORY SAMPLING AND LATENT-STATE COLLECTION

RQ3 studies multi-mode latent dynamics by analyzing multiple trajectories per input. For each example $x$, we generate $N$ independent rollouts under the same decoding configuration as in the main experiments (temperature/top-$p$/top-$k$ when sampling; greedy decoding when deterministic rollouts are required). For each rollout, we record (i) the final decoded answer $\hat{y}$ and (ii) the realized latent states $h_{1:T} = (h_1, \ldots, h_T)$ at the model-defined latent steps, where each $h_t \in \mathbb{R}^d$ is the last-layer hidden state associated with step $t$.

**Categorize answers and defining modes.** We categorize final answers into a normalized form (e.g., extracting the final option letter for multiple-choice tasks, or applying the same numeric/-boolean normalization used for evaluation). Given the set of normalized answers from $N$ rollouts for the same input, we define the two dominant modes $A$ and $B$ as the two most frequent terminal answers. Rollouts whose terminal answers fall outside $\{A, B\}$ are treated as residual modes and are excluded from the binary-mode analysis for that input. This procedure yields a set of labeled latent trajectories $\{(h^{(i)}_{1:T}, m^{(i)})\}_{i=1}^{M}$ with $m^{(i)} \in \{A, B\}$, where $M \leq N$ after filtering.

### D.2 INTERMEDIATE-STEP READOUT DETAIL IMPLEMENTATION

**Probes for step-wise mode tracking.** To measure how mode information evolves across latent steps, we train lightweight probes on frozen latent states. For each step $t$, we fit a classifier $\pi_{\phi,t}$ that maps the realized latent state $h_t$ to a distribution over modes:

$$\pi_{\phi,t} : \mathbb{R}^d \to \Delta^2, \qquad (\hat{s}_A(t), \hat{s}_B(t)) = \pi_{\phi,t}(h_t), \tag{20}$$

where $\hat{s}_A(t)$ and $\hat{s}_B(t)$ denote the probe-predicted probabilities of modes $A$ and $B$ at step $t$. Unless otherwise stated, we use linear probes (logistic regression) trained with cross-entropy loss and standard $\ell_2$ regularization, which keeps probe capacity minimal and reduces the risk of overfitting artifacts unrelated to the model's latent dynamics. When class imbalance arises after mode filtering, we balance the probe training set by subsampling to equalize the number of trajectories per mode.

**Teacher-forced readouts used in RQ3.** In addition to probe-based tracking, we use teacher-forced readouts to deterministically score candidate answers and reduce sampling noise when needed. We follow the same teacher-forcing protocol described in Appendix A.3 (Teacher-forced readouts). Importantly, the canonical answer template is method-dependent rather than dataset-dependent: for COCONUT we use "[prefix] `### {answer}`", while for CODI we use "[prefix]

`The answer is {answer}`". Given a readout step $t$, teacher forcing provides a deterministic score (e.g., token-aggregated log-probability) for each candidate answer string, which we use as a complementary signal to the probe outputs in robustness checks.

## E Additional Discussion

**Principal influence structure reveals functional routes of latent computation.** Step-wise necessity profiles alone do not identify *how* a perturbation travels through the remaining steps. The principal influence graphs from RQ2 add this missing structure by highlighting the dominant directed routes along which an intervention at step $t$ manifests at later readouts. A recurring implication is that latent computation can be organized around a small number of effective long-range paths, where early steps shape later representations without requiring strong adjacent mediation at every intermediate step. This structural view helps reconcile cases where a step exhibits modest direct necessity but nonetheless participates in a high-influence route: its role may be to shape downstream states that only become consequential when combined with later consolidation. More generally, the contrast to explicit CoT graphs suggests that linguistic step adjacency is not a reliable proxy for computational adjacency in latent reasoning.

## F Limitations

Our conclusions are tied to specific methodological choices, including the step-level causal interface, hidden-state overwrite interventions, and teacher-forced readout. These operations—particularly strong interventions like zeroing—may induce off-manifold distribution shifts. Furthermore, this study is limited to single-step edits, a fixed latent budget ($T$=6), and a confined set of paradigms (Coconut/CODI), backbones, and CoT-supervised benchmarks. Broader evaluation across more paradigms, longer horizons, and varied intervention types is needed to assess the generalizability of the observed patterns.