LATENT REASONING DOES NOT LACK SIGNAL, IT LACKS INVARIANCE: SUBSPACE-CONSTRAINED REFINEMENT FOR ROBUST REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Latent reasoning models such as Coconut Hao et al. (2024) perform multi-step inference in embedding space, avoiding the token overhead of Chain-of-Thought while preserving reasoning capability. Recent training-free refinement methods on latent reasoning models improve accuracy by steering latent trajectories using contrastive signals from stronger and weaker reference checkpoints. We show that these gains are structurally fragile along three axes: (i) **hyperparameter sensitivity**, where fixed step sizes and memory rates exhibit task-dependent optima; (ii) **reference-pair dependence**, where contrastive directions reflect pair-specific artifacts rather than stable reasoning signals; and (iii) **input sensitivity**, where updates in the full embedding space allow surface-level variations to induce divergent trajectories. We argue that reliable latent refinement requires operating on trajectory-level invariances rather than unconstrained embedding directions. We propose **Trajectory-Invariant Latent Refinement (TILR)**, a training-free framework with two components. First, we identify a spectrally-concentrated refinement subspace in which reference models consistently differ across a calibration set, and project contrastive updates onto this subspace to filter unstable variation. Second, we apply an adaptive gating mechanism that scales update magnitude according to the alignment between the current signal and the learned subspace, suppressing unreliable corrections and recovering the unmodified backbone when alignment is low. Across six reasoning benchmarks, TILR consistently improves over prior refinement methods while substantially reducing sensitivity to input perturbations and reference selections. In particular, TILR improves answer agreement under semantically equivalent paraphrasing by 11.4% and reduces trajectory variance by $\sim 30\%$ relative to the existing refinement baselines on Coconut backbone, while maintaining comparable accuracy. These results demonstrate that constraining refinement to invariant subspaces yields more reliable latent reasoning without requiring retraining or backpropagation.

1 INTRODUCTION

Chain-of-Thought (CoT) prompting (Wei et al., 2022) improves reasoning in large language models by explicitly generating intermediate steps, but incurs substantial token overhead and produces trajectories that cannot be revised once generated. Latent reasoning methods instead perform multi-step inference directly in embedding space, iteratively updating a hidden state without emitting intermediate tokens. This approach reduces generation cost while retaining competitive reasoning performance across mathematical and multi-hop tasks. This shift to embedding-level reasoning raises a question: *which directions in embedding space should a post-hoc correction act on, and when should it act at all?*

We observed that once a latent trajectory deviates from a correct reasoning path, the model has no explicit means to recover (Hao et al., 2024). Recent work (Wang et al., 2026) addresses this by introducing training-free refinement methods that steer latent trajectories using contrastive signals derived from stronger and weaker reference checkpoints. These methods improve accuracy without modifying model parameters, making them attractive for deployment. However, we show that these gains are structurally fragile. We identify three recurring failure modes. First, *hyperparameter*

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

sensitivity: the refinement depends on fixed step sizes and memory rates whose optimal values vary across tasks and inputs. Second, *reference-pair dependence*: the contrastive direction derived from a single checkpoint pair can reflect pair-specific artifacts rather than general reasoning quality. Third, *input sensitivity*: because updates are applied in the full embedding space, semantically equivalent inputs (e.g., paraphrases or reordered options) can produce divergent latent trajectories.

Our Perspective. We argue that these failure modes can be largely attributed to a common structural issue. The existing refinement methods treat the contrastive signal as a monolithic vector in the full embedding space, without distinguishing between components that are stable across inputs and reference choices and those that are instance-specific or spurious. As a result, unstable components are propagated into every refinement step, which leads to sensitivity along all three axes. The fix needs two components, organized around one geometric principle: (i) identify the subspace where refinement should operate only along directions that consistently encode reasoning-quality differences; (ii) apply corrections when the contrastive signal aligns with these stable directions, and suppressed otherwise.

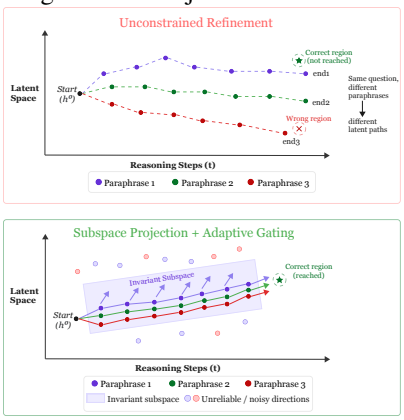

Figure 1: **Refinement vs TILR**

Proposed Framework. We introduce **Trajectory-Invariant Latent Refinement (TILR)**, a training-free framework with two components. First, *invariant subspace identification* extracts a spectrally concentrated refinement subspace in which strong and weak reference models consistently differ across a calibration set. Projecting contrastive updates onto this subspace filters out components that vary across inputs and reference pairs. Second, an *adaptive stability gate* scales the update magnitude based on the alignment between the current contrastive signal and the learned subspace. When the alignment is high, the gate permits the full projected correction and when alignment is low, the update is suppressed and the method recovers the unmodified backbone, providing a principled fallback. Both components require only forward computation and add negligible inference overhead.

To evaluate refinement methods beyond average accuracy, we introduce a robustness-focused protocol that measures worst-case performance over hyperparameter neighborhoods, answer agreement under semantically equivalent input perturbations, and sensitivity to reference checkpoint choice. Across six reasoning benchmarks, TILR preserves the accuracy gains of prior refinement methods while improving robustness to input perturbations and reference selection on most datasets.

Contributions.

- We identify three systematic fragility sources in latent reasoning refinement: hyperparameter sensitivity, reference dependence, and input sensitivity, and argue that they arise from a common failure to distinguish stable and unstable components of the contrastive signal.
- We propose TILR, a training-free refinement framework that constrains updates to an invariant subspace and modulates their magnitude based on signal reliability via invariant subspace projection and adaptive stability gating.
- We introduce a robustness evaluation protocol for latent reasoning that complements standard accuracy and empirically demonstrate that TILR improves robustness across multiple benchmarks while maintaining competitive accuracy and negligible computational overhead.

2 Related Work

Latent reasoning. Chain-of-Thought prompting (Wei et al., 2022) proved the value of intermediate reasoning steps for LLMs, but each step consumes output tokens. Coconut (Hao et al., 2024) reformulates reasoning as a recursive update in embedding space, eliminating the per-step token cost while matching CoT accuracy. This includes methods that learn to generate discrete reasoning paths in latent space (Deng et al., 2023) and methods to internalize reasoning within hidden states during training (Deng et al., 2024). TILR operates on the latent reasoning paradigm and is agnostic to how the backbone was trained; it requires only that the backbone produce a sequence of latent states that can be intervened on at inference time.

Post-training inference-time refinement. Recent work modifies LLM behavior at inference time without parameter updates. Contrastive decoding (Li et al., 2023) and its derivatives steer token-level generation by combining logits from stronger and weaker model variants. Activation steering (Turner et al., 2023; Rimsky et al., 2024) edits intermediate representations along directions identified from contrastive examples, for alignment rather than reasoning quality. The recent work on post-refinement (Wang et al., 2026) adapts the contrastive principle to latent reasoning, using the difference between stronger and weaker checkpoints to steer Coconut's continuous thoughts. AdaAnchor (Sheshanarayana et al., 2026) introduces adaptive halting for latent reasoning, terminating refinement when anchor representations stabilize. Where prior methods accept the contrastive signal as given, TILR decomposes it into stable and unstable components and applies only the stable part.

Subspace projection and invariance. Identifying low-rank structure in neural representations in domain adaptation (Ben-David et al., 2010) and continual learning (Saha et al., 2021; Lin et al., 2022) are well-established, where projecting gradient updates onto subspaces preserves prior knowledge or filters distribution-specific variation. Invariant Risk Minimization (Arjovsky et al., 2019) and the invariant prediction framework (Peters et al., 2016) formalize the principle that features whose predictive relationship is stable across environments generalize better than features merely correlated with the target. TILR draws on this principle without inheriting the optimization overhead: because the contrastive difference is directly observed via forward passes through reference models, the invariant directions can be extracted in closed form from a single SVD rather than learned through adversarial or bilevel optimization.

3 Background and Problem Setup

Latent Reasoning - Coconut backbone. Latent reasoning models perform multi-step inference by iteratively updating a hidden state in embedding space rather than generating explicit intermediate tokens (Hao et al., 2024). Let f denote a frozen decoder block and $h_t \in \mathbb{R}^d$ the latent reasoning state at step t. The trajectory evolves as

$$h_t = f(h_{t-1}), \quad t = 1, \ldots, T.$$

which provides no mechanism for error correction once the trajectory diverges from a correct reasoning path.

Post-training latent refinement. Recent work introduces training-free refinement mechanisms that modify the latent trajectory at inference time. A common approach combines residual blending with contrastive steering using a pair of reference models $(f_{\text{good}}, f_{\text{bad}})$:

$$\tilde{h}_t = \alpha h_{t-1} + (1-\alpha) f(h_{t-1}),$$

$$h_t = \tilde{h}_t + \eta\, d_t, \quad \text{where} \quad d_t = h_t^{\text{good}} - h_t^{\text{bad}}.$$

Here $h_t^{\text{good}} = f_{\text{good}}(\tilde{h}_t)$ and $h_t^{\text{bad}} = f_{\text{bad}}(\tilde{h}_t)$. Under the stopped-gradient approximation, d_t depends only on the reference pair and the current input. This refinement improves accuracy without parameter updates, but introduces a key structural limitation: the same contrastive direction is applied uniformly across all inputs and steps.

Observed fragility. Empirically, this leads to three recurring failure modes: (i) sensitivity to hyperparameters (α, η), (ii) dependence on the specific reference checkpoint pair, and (iii) divergence under semantically equivalent input reformulations.

Problem formulation. Given a frozen coconut backbone f and a reference pair $(f_{\text{good}}, f_{\text{bad}})$, we seek a refinement operator

$$h_t = \mathcal{R}(\tilde{h}_t, h_t^{\text{good}}, h_t^{\text{bad}})$$

that improves reasoning performance while being robust to: (i) hyperparameter variation, (ii) reference checkpoint choice, and (iii) input reformulation. The key constraint is that $\mathcal{R}$ must operate without parameter updates or backpropagation through the reference models.

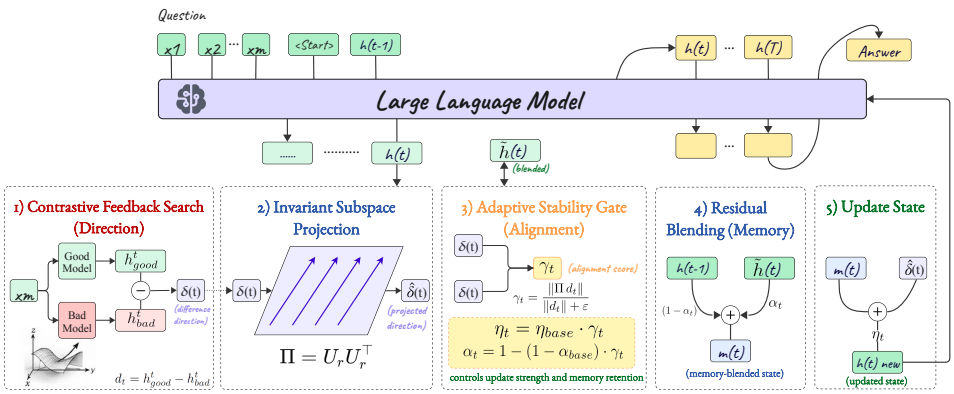

Figure 2: **The Trajectory-Invariant Latent Refinement (TILR) Framework Overview.** TILR enhances latent reasoning by constraining contrastive updates to a stable reasoning subspace. (1) **Contrastive Feedback Search** identifies a direction d_t pointing from weak to strong reference models. (2) **Invariant Subspace Projection** filters out instance-specific noise by projecting d_t onto a calibrated subspace U_r. (3) **Adaptive Stability Gating** computes an alignment score γ^t to modulate the update strength. (4) **Residual Blending** maintains trajectory stability, and (5) **State Updating** produces the refined latent state h^t_{new} for the next reasoning step.

4 Trajectory-Invariant Latent Refinement (TILR)

TILR is motivated by a single observation: the contrastive direction d_t that drives latent-reasoning refinement is not a homogeneous signal, but a mixture of components that are stable across inputs and reference choices, and components that are instance-specific and noisy. Existing methods apply d_t uniformly in the full embedding space, allowing unstable components to propagate into every update. TILR addresses this gap by separating these components and applying only the stable part of the signal, with magnitude determined by its reliability. Concretely, TILR adds two components to the refinement contrastive signal: an invariant subspace that constrains where the correction can act, and an adaptive gate that controls how strongly it acts. The subspace operationalizes the decomposition; the gate provides the fallback.

Empirical Hypothesis: Spectral Concentration Let $\delta_t = h_t^{\text{good}} - h_t^{\text{bad}}$ denote the contrastive difference. Empirically, we observe substantial concentration of contrastive energy in the leading spectral directions across inputs and reasoning steps.

We therefore posit that there exists a subspace $S \subset \mathbb{R}^d$ such that

$$\mathbb{E}[\|\Pi_S^{\perp}\delta_t\|^2] \ll \mathbb{E}[\|\delta_t\|^2],$$

where $\Pi_S^{\perp}$ is the orthogonal complement projection. This is validated empirically in Section 5.

4.1 Invariant Subspace-Projected Contrastive Refinement

4.1.1 Invariant Subspace Identification

Given a calibration set $\mathcal{C} = \{x_1, \ldots, x_N\}$ of modest size N, we construct the subspace via forward passes only. We draw $\mathcal{C}$ uniformly at random from the training split of each benchmark with N=200 and a fixed seed, disjoint from the test set. The procedure uses only the contrastive differences δ_i^t from forward passes through the reference models and requires no labels; full construction details are in Appendix E. For each x_i, we run the Coconut backbone with residual blending (Section 3) to obtain $\{\tilde{h}_i^t\}_{t=1}^T$, we pass each $\tilde{h}_i^t$ through both reference models, and form the contrastive differences

$$\delta_i^t = h^t_{\text{good},i} - h^t_{\text{bad},i} \in \mathbb{R}^d, \quad i = 1, \ldots, N, \quad t = 1, \ldots, T. \tag{1}$$

Stacking all NT differences column-wise gives

$$\Delta = \left[\delta_1^1, \ \ldots, \ \delta_1^T, \ \delta_2^1, \ \ldots, \ \delta_N^T\right] \in \mathbb{R}^{d \times NT}. \tag{2}$$

We compute the truncated SVD

$$\Delta \approx U_r \Sigma_r V_r^\top, \quad U_r \in \mathbb{R}^{d \times r}, \quad r = \min\left\{k : \textstyle\sum_{j=1}^{k} \sigma_j^2 \,/\, \sum_{j=1}^{d} \sigma_j^2 \geq \tau\right\}, \tag{3}$$

where r is chosen to retain a fraction τ of the variance. The invariant subspace $\mathcal{S}_r$ is defined by the projection

$$\Pi = U_r U_r^\top \in \mathbb{R}^{d \times d}. \tag{4}$$

The columns of U_r are the principal components of the *difference* between strong and weak reasoning outputs, not of the embedding space itself. This distinction matters: PCA on raw embeddings would capture directions of maximal input variance (dominated by lexical and stylistic variation), while our procedure captures directions of maximal reasoning-quality variance. We measure the geometric overlap between these two bases by computing the mean principal angle; a large angle confirms that the subspace captures structure distinct from generic input variation. Pooling δ_i^t across steps implements the spectral structure hypothesis; we ablate against per-step subspaces in the appendix.

4.1.2 Subspace-Projected Contrastive Refinement

At inference, we replace the unconstrained update with its subspace-projected counterpart. The per-instance contrastive direction is

$$d_t \triangleq h_{\text{good}}^t - h_{\text{bad}}^t \in \mathbb{R}^d, \tag{5}$$

and the projected refinement is

$$h^t = \tilde{h}^t + \eta_t \cdot \Pi\, d_t, \tag{6}$$

where η_t is the adaptive step size defined next. The projection Πd_t retains only the component of d_t aligned with the calibrated reasoning-quality subspace, attenuating instance-specific and pair-specific variation to the extent the hypothesis holds. This addresses reference invariance and input invariance: the components of d_t responsible for these two fragility sources lie outside $\mathcal{S}_r$ by the assumption and are filtered out by Π.

4.2 Adaptive Stability Gate

The projection controls where the update acts; the gate controls how strongly. We define the **subspace alignment score**

$$\gamma_t = \frac{\|\Pi\, d_t\|}{\|d_t\| + \varepsilon}, \tag{7}$$

with ε controlling numerical stability. $\gamma_t \in [0, 1]$ measures the fraction of the current correction signal that lies within $\mathcal{S}_r$: $\gamma_t \approx 1$ indicates the signal is well-aligned with historically consistent quality-difference directions; $\gamma_t \approx 0$ indicates the signal is dominated by instance-specific components. We use γ_t to scale the step size:

$$\eta_t = min(\eta_{\text{base}} \cdot \gamma_t^2, 1). \tag{8}$$

The base mixing rate α_{base} is *not* gated. The gate interpolates smoothly between two regimes:

$$h^t \;\longrightarrow\; \begin{cases} \tilde{h}^t + \eta_{\text{base}} \cdot \Pi d_t & \text{as } \gamma_t \to 1 \quad \text{(full projected correction)}, \\ \alpha_{\text{base}} h^{t-1} + (1 - \alpha_{\text{base}}) f(h^{t-1}) & \text{as } \gamma_t \to 0 \quad \text{(uncorrected backbone)}. \end{cases} \tag{9}$$

When γ_t is high, the full projected correction is applied on top of the residual blend. As $\gamma_t \to 0$, the correction vanishes and TILR recovers the uncorrected system.

Sensitivity flattening on low-confidence inputs. A direct consequence of the multiplicative gate $\eta_t = min(\eta_{\text{base}} \cdot \gamma_t^2, 1)$ is that inputs with small γ_t become approximately insensitive to the choice of η_{base}: scaling η_{base} by any factor scales a near-zero quantity by that factor, leaving the effective correction negligible. This provides a plausible mechanism by which the gate addresses fragility source (i): low-γ inputs contribute disproportionately to the worst-case sensitivity region in the (α, η) grid, and TILR suppresses them. A local sensitivity analysis of the adaptive gating mechanism is provided in Appendix D, showing that low-alignment inputs induce suppressed update sensitivity with respect to perturbations in the nominal step size.

Why α is not gated. TILR is a post-training add-on, and the appropriate "do no harm" baseline is the uncorrected system at $\alpha = \alpha_{\text{base}}$. With a fixed α, the limit $\gamma_t \to 0$ recovers this baseline exactly: the correction vanishes and equation 6 collapses to the residual blend. Modulating α_t with γ_t would instead push the limit to $h^t = h^{t-1}$ (state freeze), which is not the original system's behavior and may be strictly worse. More fundamentally, α_{base} controls the backbone's memory-vs-computation balance, a property of the model dynamics; γ_t measures the reliability of the correction direction and carries no information about whether $f(h^{t-1})$ itself is trustworthy. Modulating α by γ_t conflates these two independent uncertainties.

Overall, TILR decomposes the contrastive signal as

$$d_t = \underbrace{\Pi d_t}_{\text{stable}} + \underbrace{(I - \Pi) d_t}_{\text{unstable}},$$

and applies only the stable component, scaled by its reliability. This directly addresses input and reference sensitivity, while the gating mechanism limits the impact of unreliable corrections.

Computational Overhead. At each reasoning step, TILR requires two matrix-vector products to compute $U_r^\top d_t$ and $U_r(U_r^\top d_t)$, with total complexity $\mathcal{O}(dr)$. For the GPT-2 backbone ($d = 768$), the learned subspace ranks range from $r = 38$ to 171 across the evaluated benchmarks. In practice, these operations add less than 3% wall-clock inference overhead relative to unconstrained refinement. Calibration is performed once per benchmark and reference pair and is dominated by the reference forward passes and a truncated SVD of the $d \times NT$ contrastive-difference matrix.

5 Experimental Setup

We design experiments to evaluate whether TILR improves the reliability of latent reasoning refinement while preserving accuracy. In particular, we test five questions: **RQ1 (Accuracy):** Does TILR improve reasoning accuracy relative to existing latent refinement methods? **RQ2 (Reference Robustness):** Is TILR robust to the choice of contrastive reference checkpoint pair? **RQ3 (Input Invariance):** Does TILR produce more consistent predictions and latent trajectories to semantically equivalent input reformulations? **RQ4 (Subspace Geometry):** Do contrastive refinement signals exhibit a spectrally concentrated subspace distinct from generic representation variance? **RQ5 (Causal Role of Subspace):** Is the learned subspace functionally necessary for refinement performance? Additional ablations, hyperparameters and cross architecture generalization analysis are reported in the Appendix. **Benchmarks.** We evaluate on six reasoning benchmarks covering arithmetic, robustness-oriented mathematical reasoning, and symbolic deduction. **GSM8K-Aug** (Deng et al., 2023) serves as the standard arithmetic reasoning benchmark. **SVAMP** (Patel et al., 2021) evaluates robustness to lexical and structural variation in arithmetic word problems, while **GSM-Plus** (Li et al., 2024) introduces controlled numerical and structural perturbations of GSM8K problems. We additionally evaluate on **GSM-Symbolic** (Mirzadeh et al., 2025), which generates symbolic variants of GSM8K-style problems, and on the deductive reasoning benchmarks **ProsQA** (Hao et al., 2024) and **ProntoQA** (Saparov & He, 2022). This combination of benchmarks allows us to evaluate both standard reasoning performance and robustness under controlled changes to the problem representations.

Models and checkpoints. The primary backbone is **GPT-2 base (117M)** trained with the Coconut latent reasoning framework (Hao et al., 2024). A separate backbone is trained per benchmark to avoid confounding refinement quality with cross-task transfer. Reference models $(f_{\text{good}}, f_{\text{bad}})$ are selected from different training epochs of the same backbone, with f_{good} achieving higher validation accuracy. Unless otherwise specified, we use later-stage and early-stage checkpoints to construct the contrastive pair. To test whether the findings hold when scaled, we additionally evaluate **Llama-3.2-1B** (Grattafiori et al., 2024) and **Qwen2.5-Math-1.5B** (Yang et al., 2024) in the Appendix. No checkpoint, hyperparameter, or subspace is selected using any test split.

Baselines. We compare TILR against five methods under identical conditions: (i) **No-CoT** (Radford et al., 2019): Direct prediction without reasoning steps; (ii) **CoT**: Token-based chain-of-thought reasoning with a separately trained model; (iii) **Coconut** (Hao et al., 2024): Latent backbone without refinement; (iv) **Coconut + Refinement** (Wang et al., 2026): Unconstrained contrastive latent refinement; and (v) **AdaAnchor** (Sheshanarayana et al., 2026): Adaptive anchor refinement with step-level adaptivity. Except CoT, all baselines operate on the same per-benchmark Coconut backbone, so observed differences trace to the refinement mechanism alone.

Table 1: Exact-match accuracy (%) and average generation tokens per inference on six reasoning benchmarks under GPT-2 backbones trained per-benchmark. Values are mean ± standard error. **Bold** indicates best overall; † best among latent methods; underline second-best method. Token counts are averaged across seeds.

Method	GSM8K Acc.	GSM8K Tok.	SVAMP Acc.	SVAMP Tok.	GSM-Plus Acc.	GSM-Plus Tok.
No-CoT	6.87 ± 0.3	2.5	23.20 ± 1.33	2	2.71 ± 0.2	3
CoT	$\mathbf{32.83 \pm 1.3}$	28	28.20 ± 1.5	24	$\mathbf{16.20 \pm 0.4}$	30
Coconut	26.61 ± 1.2	8	30.00 ± 0.8	4	12.93 ± 0.3	8
Refinement	26.72 ± 1.1	8	$\underline{31.50 \pm 1.4}$	4	13.46 ± 0.4	8
AdaAnchor	26.54 ± 0.8	8	27.10 ± 1.5	4	12.98 ± 0.3	8
TILR	$\underline{27.67 \pm 1.2}^{\dagger}$	8	$\mathbf{33.70 \pm 1.4}^{\dagger}$	4	$\underline{13.70 \pm 0.3}^{\dagger}$	8

Method	GSM-Symbolic Acc.	GSM-Symbolic Tok.	ProsQA Acc.	ProsQA Tok.	ProntoQA Acc.	ProntoQA Tok.
No-CoT	4.32 ± 1.1	3	76.41 ± 1.5	9	82.22 ± 0.88	3
CoT	10.83 ± 0.8	22	72.34 ± 1.1	51	92.20 ± 0.6	85
Coconut	10.86 ± 0.6	9	88.50 ± 1.1	15	92.93 ± 1.2	9
Refinement	$\underline{10.87 \pm 0.5}$	8	$\underline{88.53 \pm 0.4}$	16	93.46 ± 1.1	9
AdaAnchor	10.09 ± 1.2	8	87.10 ± 0.1	16	$\underline{92.98 \pm 0.3}$	9
TILR	$\mathbf{11.72 \pm 0.6}^{\dagger}$	8	$\mathbf{90.4 \pm 0.1}^{\dagger}$	16	$\mathbf{94.70 \pm 0.1}^{\dagger}$	9

Metrics. Average accuracy alone does not capture the reliability of latent refinement. We therefore evaluate using four complementary metrics: **M1: Accuracy** is exact-match on the test set, averaged over 3 seeds. **M2: Reference Sensitivity** is the variation in accuracy across different reference checkpoint pairs, measured via standard deviation and range. **M3: Answer agreement** is the fraction of consistent predictions across multiple semantically equivalent reformulations of each input. **M4: Trajectory variance** is the mean pairwise ℓ_2 distance between latent trajectories induced by input reformulations. These metrics correspond directly to the fragility sources identified in Section 3.

6 Experimental Results

RQ1: Overall Reasoning Performance. Across six reasoning benchmarks, we evaluate whether constraining latent refinement to the learned subspace improves reasoning accuracy without sacrificing the efficiency advantage of latent reasoning. Table 1 shows that TILR outperforms latent refinement baselines on all six benchmarks, while preserving the token efficiency of latent reasoning. CoT remains strongest on GSM8K-Aug and GSM-Plus, which is consistent with the known advantage of explicit symbolic traces on open-ended multi-step arithmetic reasoning. The comparison with explicit CoT further illustrates the accuracy–efficiency trade-off. Overall, TILR improves the reliability of latent refinement within the same efficiency regime rather than closing the token-level CoT gap. AdaAnchor is below Coconut on most benchmarks in this setup, so we treat it as a secondary reference point rather than a strong competing baseline. We observe similar trends on a stronger Qwen2.5-Math-1.5B and Llama3.2-1B-Instruct backbones (Appendix H), where TILR improves over unconstrained refinement on all benchmarks. Notably, these results use a single hyperparameter configuration selected on GSM8K-Aug and transferred unchanged to the remaining benchmarks, rather than tuning TILR separately for each test domain.

RQ2: Checkpoint-pair sensitivity. To isolate the effect of the contrastive pair, we evaluate six good/bad checkpoint pairs per dataset using Coconut backbones trained without stepwise internalization. For each pair, we report the standard deviation, and range across pairs. As shown in Table 2, TILR substantially reduces checkpoint-pair variability on GSM8K-Aug, ProsQA, ProntoQA, and GSM-Plus, with relative standard-deviation reductions of 72.7%, 60%, 64.2%, and 53.0%, respectively. The reduction in both dispersion measures indicates that TILR's performance depends less strongly on the particular reference pair used at inference time. Importantly, the robustness improvement is observed across datasets with substantially different mean subspace overlap. Overall, these results provide empirical evidence that constraining contrastive refinement to the learned subspace makes inference less sensitive to reference-checkpoint selection.

Table 2: **Robustness to checkpoint-pair selection.** Lower values indicate stronger robustness.

Dataset	Sensitivity across pairs (pp, lower is better) Ref. std	TILR std	Std. red.	Subspace Range red.	Mean overlap
GSM8K	0.160	**0.044**	**+72.7**	**+66.7**	0.837
ProsQA	0.525	**0.210**	**+60.0**	**+68.5**	0.564
ProntoQA	0.605	**0.216**	**+64.2**	**+60.2**	0.370
GSM-Plus	0.350	**0.165**	**+53.0**	**+43.5**	0.988

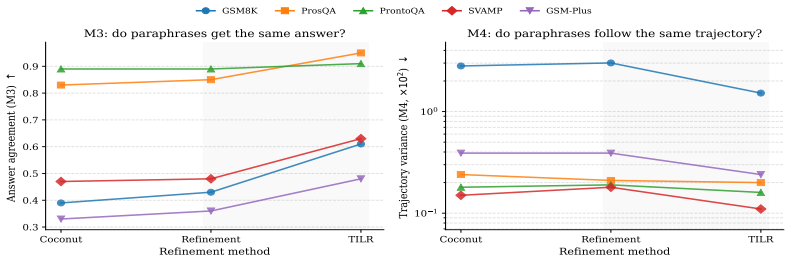

Figure 3: **Input invariance under perturbations. Left:** Answer agreement (M3, ↑) across semantically equivalent reformulations per input. **Right:** Latent trajectory variance (M4, ↓).

RQ3. Input Invariance We next examine whether TILR produces more stable reasoning behavior under semantically equivalent reformulations of the same input at two levels: Measuring answer agreement (M3, ↑), which captures whether reformulations produce the same final prediction and latent trajectory variance (M4, ↓), which measures divergence between the corresponding latent reasoning trajectories across $M{=}5$ semantically equivalent reformulations per test input generated using Llama-3-Instruct (Grattafiori et al., 2024) (Figure 3). We observe that TILR reduces trajectory variance relative to Refinement on every benchmark, with reductions of 49.5% on GSM8K, 4.8% on ProsQA, 15.8% on ProntoQA, 38.9% on SVAMP, and 38.5% on GSM-Plus, respectively. This supports the claim that subspace projection suppresses paraphrase-specific components of the contrastive direction before they affect the trajectory (Section B). The same pattern is observed in answer agreement. TILR improves answer agreement over unconstrained Refinement on all five evaluated benchmarks by an average of $\sim 11.4\%$. These results show that TILR enables semantically equivalent inputs to be more likely to produce the same answer while following similar latent trajectories.

RQ4: Spectral Geometry of the learned refinement subspace. Here, we study whether the contrastive refinement direction is concentrated in a low-dimensional spectral structure, and whether that structure is distinct from ordinary input-hidden-state variance. We measure the effective rank r_τ needed to explain τ fraction of the contrastive covariance spectrum, the variance captured by the leading directions, and the alignment between the learned contrastive basis and PCA bases computed from input hidden states or latent refinement trajectories. The input-controlled columns remove the input-PCA component before measuring the residual contrastive signal. Table 3, the signal is strongly concentrated in its leading directions: the first component explains 32.0–52.0% of the variance, while the top 10 components capture 76.0–83.6%. However, the rank required to explain 90% of the variance varies substantially across datasets, from 38 on SVAMP to 171 on GSM8K, indicating a concentrated leading spectrum followed by a task-dependent long tail. The learned contrastive subspace also exhibits varying alignment with input-PCA directions, ranging from 0.340 to 0.804, suggesting that the refinement signal contains substantial structured concentration. These results support a reduced-dimensional representation of the dominant refinement signal rather than a uniformly low-rank structure.

RQ5: Functional Role of the Learned Subspace. We test whether the learned TILR subspace is causally useful rather than merely correlated with generic hidden-state variance. We use the same no-internalization Coconut backbones as in RQ2, we compare TILR against the base Coconut refinement trajectory, random subspaces, input/trajectory PCA subspaces, an orthogonal knockout intervention, and a raw-update gate baseline. The controlled intervention results 4 show that the learned subspace is consistently more useful than random or knocked-out directions, and the orthogonal knockout baseline is worse than TILR on five out of six datasets. This supports the interpretation that the learned directions carry task-relevant refinement information, not just low-rank structure.

Table 3: **Spectral structure of the learned refinement subspace.** Low $r_{0.90}$ and high explained variance indicate strong spectrally concentrated structure.

Dataset	$r_{0.90}$	Top-1 EV	Top-10 EV	Align(input)
GSM8K	171	0.320	0.760	0.779
ProsQA	66	0.432	0.836	0.620
ProntoQA	84	0.520	0.810	0.691
SVAMP	38	0.425	0.765	0.340
GSM-Plus	156	0.402	0.781	0.804

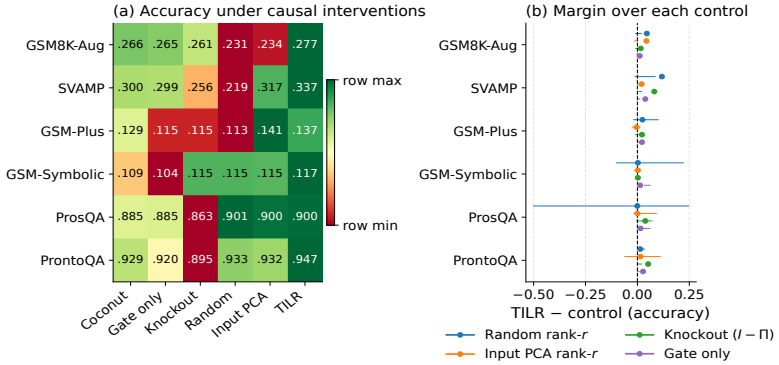

Figure 4: **Causal subspace interventions.** Panel (a) reports accuracy under different subspaces. Panel (b) shows TILR margins over the strongest causal controls.

Overall, TILR improves reasoning accuracy while making latent refinement more robust to reference selection and input reformulation. The learned subspace captures structured, functionally useful refinement directions, with additional sensitivity and ablation analyses provided in the Appendix.

7 Conclusion

We introduced TILR, a training-free refinement method for latent reasoning that separates the contrastive signal into subspace-aligned and residual components, applies updates only within a learned invariant refinement subspace, and modulates their magnitude based on alignment with that subspace. This formulation reframes latent refinement as a problem of selecting and trusting directions in embedding space, rather than uniformly applying contrastive signals. Across six evaluated benchmarks, TILR improves accuracy over existing refinement methods while consistently reducing trajectory variance and improving answer consistency under input reformulations. Our analysis shows that contrastive refinement signals are highly structured, with substantial variance concentrated in the leading directions between strong and weak reasoning trajectories. Controlled subspace intervention experiments further indicate that these directions are functionally useful for refinement, rather than incidental low-dimensional spectral structure. These results suggest that the key limitation of existing latent refinement methods is not the absence of useful signals, but the inability to distinguish stable structure from instance-specific variation. TILR addresses this by filtering and gating the contrastive signal, yielding more reliable behavior without retraining. However, we also observed some limitations. The learned subspace is empirical rather than theoretically grounded, and its stability varies across datasets. In addition, while TILR improves input invariance, it does not fully eliminate trajectory divergence under reformulation. Understanding how invariant structure evolves across model scale and training regimes remains an open problem. Our results point to a geometric perspective on latent reasoning, where the effectiveness of refinement depends on identifying and exploiting spectrally concentrated structure in representation space. Latent reasoning does not lack signal; it lacks invariance, and TILR fills the gap by selectively preserving the dominant refinement structure. Extending this perspective to other forms of inference-time control on larger models is a promising direction for future work.

AI USE STATEMENT

In this work, we used generative AI tools for debugging, standard code implementation, generation of evaluation data, and text polishing. Specifically, Llama-3-Instruct was used to generate the $M = 5$ semantically equivalent input reformulations (paraphrases and lexical variations) used in the robustness evaluation protocol described in Sections 5 and 6. Commercial LLM-based assistants were also used to support debugging and routine code implementation, as well as during the post-drafting phase to improve sentence structure and grammatical consistency. Generative AI tools were not used to develop the core research ideas, experimental design, or mathematical formulations. All AI-assisted code, evaluation data, and text were reviewed by the authors, including verification of generated reformulations for semantic equivalence to the original evaluation instances. The authors take responsibility for the final content, claims, experimental results, and artifacts presented in this work.

ETHICS STATEMENT

While Trajectory-Invariant Latent Refinement (TILR) is an algorithmic framework designed to improve the internal consistency of latent reasoning models, its downstream deployment carries considerations regarding reliability. TILR improves trajectory invariance under semantic perturbations rather than guaranteeing factual correctness. Consequently, systematic biases or errors encoded by the underlying reference checkpoints may persist under refinement, and improved consistency should not be interpreted as evidence of correctness or safety. Furthermore, because TILR improves inference-time robustness without requiring parameter retraining, it may facilitate the deployment of latent reasoning systems without addressing limitations inherited from the underlying models. We therefore view TILR as a robustness enhancement rather than a standalone verification or safety alignment solution. In safety-critical applications, it should be combined with task-specific verification and appropriate human oversight.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our results, we have provided the code repository and comprehensive details of our framework across both the main text and supplementary materials. The complete step-by-step inference procedure for TILR is formalized in Algorithm 1 (Section 4). Detailed specifications of our calibration set construction, including sample sizes ($N = 200$) and conditions for the truncated Singular Value Decomposition (SVD), are explicitly provided in Appendix E. Full theoretical derivations regarding the full Jacobian formulation are detailed in Appendix A. All baseline configurations, training epochs, and datasets (including GSM8K-Aug, SVAMP, GSM-Plus, GSM-Symbolic, ProsQA, and ProntoQA) are defined in Section 5. All experiments were executed on a hardware cluster comprising 4$\times$ NVIDIA H100 GPUs, requiring under 30 seconds for the one-time calibration phase as described in Appendix B. Code to replicate the calibration and inference steps will be made publicly available upon publication.

REFERENCES

Martin Arjovsky, Léon Bottou, Ishaan Gulchandani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Yuntian Deng, Kiran Prasad, Roland Fernandez, Paul Smolensky, Vishrav Chaudhary, and Stuart Shieber. Implicit chain of thought reasoning via knowledge distillation. *arXiv preprint arXiv:2311.01460*, 2023.

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

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, and Ahmad Al-Dahle et al. The llama 3 herd of models, 2024. URL `https://arxiv.org/abs/2407.21783`.

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. *arXiv preprint arXiv:2412.06769*, 2024.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. Gsm-plus: A comprehensive benchmark for evaluating the robustness of llms as mathematical problem solvers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2961–2984, 2024.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori B Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 12286–12312, 2023.

Sen Lin, Li Yang, Deliang Fan, and Junshan Zhang. Trgp: Trust region gradient projection for continual learning. *arXiv preprint arXiv:2202.02931*, 2022.

Iman Mirzadeh, Keivan Alizadeh-Vahid, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. In *International Conference on Learning Representations*, volume 2025, pp. 94743–94765, 2025.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are nlp models really able to solve simple math word problems? In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies*, pp. 2080–2094, 2021.

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B*, 78(5):947–1012, 2016.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, 2024.

Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. *arXiv preprint arXiv:2103.09762*, 2021.

Abulhair Saparov and He He. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. *arXiv preprint arXiv:2210.01240*, 2022.

Disha Sheshanarayana, Rajat Subhra Pal, Manjira Sinha, and Tirthankar Dasgupta. Thinking in latents: Adaptive anchor refinement for implicit reasoning in llms. *arXiv preprint arXiv:2603.15051*, 2026.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023.

Xinyuan Wang, Dongjie Wang, Wangyang Ying, Haoyue Bai, Nanxu Gong, Sixun Dong, Kunpeng Liu, and Yanjie Fu. Efficient post-training refinement of latent reasoning in large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 33692–33700, 2026.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement, 2024. URL `https://arxiv.org/abs/2409.12122`.

Appendix

A Stopped-Gradient and the Full Jacobian

The simplification used in Section 3 treats $f_{\text{good}}(\tilde{h}^t)$ and $f_{\text{bad}}(\tilde{h}^t)$ as constants with respect to $\tilde{h}^t$. The full gradient of the contrastive objective includes Jacobian terms from both reference models:

$$\nabla_{\tilde{h}^t}\Big[\mathrm{MSE}(h^t_{\text{good}}, \tilde{h}^t) - \mathrm{MSE}(h^t_{\text{bad}}, \tilde{h}^t)\Big] = \tfrac{2}{d}(J_{\text{good}} - I)^\top(h^t_{\text{good}} - \tilde{h}^t) - \tfrac{2}{d}(J_{\text{bad}} - I)^\top(h^t_{\text{bad}} - \tilde{h}^t), \tag{10}$$

where J_{good} and J_{bad} are the Jacobians at $\tilde{h}^t$.

We isolate this derivation in the appendix because the main paper only needs the stopped-gradient intuition. The full expression makes clear that the refinement update is not simply a static pull toward one reference and push away from another; it also depends on how both reference trajectories locally deform the hidden-state manifold. This is precisely why the projection step is useful. It suppresses directions that are unstable across reference choices before those Jacobian-induced effects can dominate the update.

B Algorithm and Properties

Combining the projection and gate, the full TILR update at each step is summarized in Algorithm 1, where h^0 denotes the initial latent state produced by the backbone's embedding of the input tokens. The computation proceeds sequentially with no circular dependency: α_{base} is fixed, γ_t is computed once per step from the subspace projection, and the correction is applied additively.

Two design guarantees follow directly from the algorithm's structure, corresponding to the two failure modes the method was designed to handle.

Graceful degradation. As $\gamma_t \to 0$, the correction magnitude $\eta_t \cdot \Pi d_t$ vanishes and Step 4 reduces to

$$h^t \;\longrightarrow\; \alpha_{\text{base}} \cdot h^{t-1} + (1 - \alpha_{\text{base}}) \cdot f(h^{t-1}), \tag{11}$$

which is exactly the uncorrected residual-blended backbone.

Subspace-mediated input invariance. By construction, two inputs x, x' whose contrastive directions differ only outside $\mathcal{S}_r$ satisfy $\Pi d_t(x) = \Pi d_t(x')$. If additionally $\|d_t(x)\| = \|d_t(x')\|$, then $\gamma_t(x) = \gamma_t(x')$ and the TILR-induced divergence reduces to the backbone-induced divergence:

$$h^t(x) - h^t(x') \;=\; \tilde{h}^t(x) - \tilde{h}^t(x'). \tag{12}$$

The correction contributes no additional divergence beyond that produced by the backbone forward pass. Though TILR is conceptually inspired by invariant prediction (Peters et al., 2016; Arjovsky et al., 2019), we make no formal claim that $\mathcal{S}_r$ corresponds to causally invariant features (Peters et al., 2016), only that empirically it captures directions along which reasoning-quality differences are consistent across the calibration population.

Compute and reproducibility. All experiments are run on 4× NVIDIA H100 GPUs. Calibration takes < 30 seconds for GPT-2; TILR inference adds $< 3\%$ wall-clock overhead over unconstrained refinement.

C Formalisms of Robustness Desiderata

We formalize the three robustness properties referenced in Section 3 here.

1: Hyperparameter stability. For any (α', η') in a neighborhood $\mathcal{N}(\alpha, \eta)$ of the nominal configuration, the accuracy degradation is bounded: $|\,\mathrm{Acc}(\alpha', \eta') - \mathrm{Acc}(\alpha, \eta)| \le \epsilon$ for a small $\epsilon > 0$.

2: Reference invariance. The refinement direction is robust to the choice of reference pair: substituting a different pair of comparable CoT validation accuracy does not change the sign of the accuracy improvement over the unrefined backbone.

Algorithm 1 TILR: Inference Procedure

Require: Input x, backbone f, references $f_{\text{good}}, f_{\text{bad}}$, projection Π, hyperparameters $\eta_{\text{base}}, \alpha_{\text{base}}$, steps T
Ensure: Answer y
$h^0 \leftarrow$ initial latent state for input x
for $t = 1$ to T **do**
  $\tilde{h}^t \leftarrow \alpha_{\text{base}} h^{t-1} + (1 - \alpha_{\text{base}}) f(h^{t-1})$ ▷ Residual blending
  $d_t \leftarrow f_{\text{good}}(\tilde{h}^t) - f_{\text{bad}}(\tilde{h}^t)$ ▷ Contrastive direction
  $\gamma_t \leftarrow \|\Pi d_t\| / (\|d_t\| + \varepsilon); \quad \eta_t \leftarrow min(\eta_{\text{base}} \cdot \gamma_t^2, 1)$ ▷ Gated step size
  $h^t \leftarrow \tilde{h}^t + \eta_t \cdot \Pi d_t$ ▷ Projected correction
end for
$y \leftarrow \text{Decode}(h^T)$
return y

3: Input invariance. For any two semantically equivalent inputs x and x' (paraphrases or option reorderings), the refined latent trajectories satisfy $\frac{1}{T}\sum_{t=1}^{T} \|h^t(x) - h^t(x')\| \leq \delta$ for a small $\delta > 0$.

The unconstrained refinement Wang et al. (2026) satisfies none of these properties by construction: α and η are fixed scalars chosen per task (violating D1); the contrastive direction d_t is computed from a single reference pair with no mechanism to average over comparable pairs (violating D2); and d_t operates in the full embedding $\mathbb{R}^d$ without any constraint that surface-level input variations be projected away (violating D3).

TILR (Section 4) addresses D2 and D3 directly via the subspace projection: components of d_t that vary across reference pairs or across input reformulations are removed from the update by construction. D1 is addressed empirically rather than by construction. The gating mechanism scales the effective step size by the alignment score γ_t, which means that on low-confidence inputs (small γ_t) the effective η is suppressed regardless of the nominal η_{base}. This provides a plausible reason for flatter sensitivity surfaces, but it is still an empirical claim rather than a theorem. For that reason, the appendix reports the complete RQ2 surfaces rather than only the tuned operating points.

D Local Sensitivity Analysis of Adaptive Gating

We provide a formal local analysis of how the adaptive gate suppresses sensitivity to perturbations in the nominal refinement step size on low-alignment inputs.

We know that the TILR update is

$$h_t = \tilde{h}_t + \eta_t \Pi d_t,$$

where

$$\eta_t = \min\left(\eta_{\text{base}} \gamma_t^2, 1\right), \qquad \gamma_t = \frac{\|\Pi d_t\|}{\|d_t\| + \epsilon}.$$

We first consider the uncapped regime, $\eta_{\text{base}} \gamma_t^2 < 1$. Substituting the gated step size gives

$$h_t = \tilde{h}_t + \eta_{\text{base}} \left(\frac{\|\Pi d_t\|}{\|d_t\| + \epsilon}\right)^2 \Pi d_t.$$

We study the local, one-step sensitivity to a perturbation

$$\eta_{\text{base}} \to \eta_{\text{base}} + \Delta\eta,$$

holding $\tilde{h}_t$, d_t, and Π fixed.

Unconstrained refinement. For the standard refinement update

$$h_t^{\text{ref}} = \tilde{h}_t + \eta_{\text{base}} d_t,$$

the induced perturbation is

$$\Delta h_t^{\mathrm{ref}} = \Delta\eta\, d_t,$$

with magnitude

$$\|\Delta h_t^{\mathrm{ref}}\| = |\Delta\eta|\, \|d_t\|.$$

Thus, the local sensitivity scales linearly with the magnitude of the full contrastive signal.

TILR update sensitivity. For TILR in the uncapped regime, the same perturbation yields

$$\Delta h_t^{\mathrm{TILR}} = \Delta\eta \left(\frac{\|\Pi d_t\|}{\|d_t\| + \epsilon} \right)^2 \Pi d_t,$$

and therefore

$$\|\Delta h_t^{\mathrm{TILR}}\| = |\Delta\eta| \frac{\|\Pi d_t\|^3}{(\|d_t\| + \epsilon)^2}.$$

Relative to unconstrained refinement,

$$\frac{\|\Delta h_t^{\mathrm{TILR}}\|}{\|\Delta h_t^{\mathrm{ref}}\|} = \gamma_t^2 \frac{\|\Pi d_t\|}{\|d_t\|}.$$

Since Π is an orthogonal projection, $\|\Pi d_t\| \le \|d_t\|$, and hence

$$\frac{\|\Delta h_t^{\mathrm{TILR}}\|}{\|\Delta h_t^{\mathrm{ref}}\|} \le \gamma_t^2.$$

Equivalently,

$$\|\Delta h_t^{\mathrm{TILR}}\| \le \gamma_t^2 \|\Delta h_t^{\mathrm{ref}}\| \le \|\Delta h_t^{\mathrm{ref}}\|.$$

Thus, low-alignment inputs exhibit strongly suppressed local sensitivity to the nominal step size. In particular, for fixed $\|d_t\|$ and small ϵ, the sensitivity scales cubically with the magnitude of the projected component:

$$\|\Delta h_t^{\mathrm{TILR}}\| = O\left(\frac{\|\Pi d_t\|^3}{\|d_t\|^2} \right).$$

Projection diminishes components outside the learned refinement subspace, while the squared alignment gate further suppresses corrections whose projected component is weak. Whereas in the capped regime, $\eta_{\mathrm{base}}\gamma_t^2 > 1$, the effective step size is locally constant at $\eta_t = 1$. Therefore, sufficiently small perturbations in η_{base} that do not cross the cap boundary produce no change in the update. This analysis is local and mechanistic: it holds the current state and contrastive direction fixed and characterizes the one-step response to perturbations in η_{base}. This proof does not establish a global trajectory-level sensitivity or generalization guarantee.

E Calibration Set Construction

For each benchmark, we draw a calibration set $\mathcal{C} = \{x_1, \ldots, x_N\}$ uniformly at random from the training split, with N=200 and a fixed random seed. None of our benchmarks provides an official validation split, so calibration is drawn from training and held disjoint from the test set reported in Section 6. The procedure requires only forward passes and no ground-truth labels: for each x_i, we run the Coconut backbone with residual blending, pass each intermediate state $\tilde{h}_i^t$ through both reference models, and form the contrastive differences $\delta_i^t = h_{\mathrm{good},i}^t - h_{\mathrm{bad},i}^t$. Stacking all $NT = 600$ differences and computing a truncated SVD yields the invariant subspace. We use N=200 as a fixed default across all benchmarks rather than tuning the calibration size per dataset. As shown in Section 6, the resulting contrastive spectra are concentrated in their leading directions, although the rank required to explain 90% of the variance varies across datasets ($r_{0.90} = 38$–171). Total calibration cost is NT reference forward passes plus the SVD, completing in under 30 seconds on a single H100. The cost is incurred once per benchmark and reference pair and is amortized across subsequent inference.

Table 4: **Hyperparameter sensitivity** Each entry reports best accuracy, worst-case accuracy, and sensitivity gap over the non-step hyperparameter grid. Higher is better for Best and Worst; lower is better for Gap.

Dataset	Refinement			TILR		
	Best ↑	Worst ↑	Gap ↓	Best ↑	Worst ↑	Gap ↓
GSM8K	0.274	0.219	0.055	**0.279**	**0.275**	**0.003**
GSM-PLUS	0.136	0.095	0.041	**0.137**	**0.133**	**0.004**
SVAMP	0.315	0.218	0.097	**0.337**	**0.331**	**0.006**

Table 5: Ablation Study on GSM8K-Aug.

Variant	**Accuracy**
Residual blend only ($\eta = 0$, = backbone)	0.2561 ± 0.014
Correction only (no Π, no gate)	0.2691 ± 0.012
+ Projection only	0.2760 ± 0.010
+ Gating only	0.2737 ± 0.012
TILR(Projection + Gating)	$\mathbf{0.2767 \pm 0.012}$
Step and damping	
Linear damping ($\eta\gamma$)	0.2752 ± 0.013
Uncapped step (no $\min(\cdot, 1)$)	0.2721 ± 0.012
Relative step ($\eta\|\tilde{h}\|$)	0.2663 ± 0.011
Substituted projectors (same rank, scale and gate references)	
Complement projection ($I - \Pi$)	0.2667 ± 0.012
Rank-matched input PCA	0.2335 ± 0.012
Random rank-171 basis	0.2314 ± 0.012

F Hyperparameter Sensitivity

We evaluate the sensitivity of refinement methods to the hyperparameters (α, η) controlling residual blending and contrastive update strength. The goal is to assess whether TILR provides a more stable operating region than unconstrained refinement. For each dataset, we evaluate accuracy over a grid of (α, η) values and report: (i) the best accuracy within the grid, (ii) the worst-case accuracy in the local neighborhood of the best point, and (iii) the sensitivity gap between the best and worst performance. Table 4 shows that TILR substantially reduces hyperparameter sensitivity across all three evaluated benchmarks. On GSM8K, the sensitivity gap decreases from 0.055 to 0.003, while the worst-case accuracy increases from 0.219 to 0.275. Similar improvements are observed on GSM-Plus, where the gap decreases from 0.041 to 0.004, and on SVAMP, where it decreases from 0.097 to 0.006. TILR also improves the best accuracy on all three benchmarks, from 0.274 to 0.279 on GSM8K, 0.136 to 0.137 on GSM-Plus, and 0.315 to 0.337 on SVAMP. These results show that TILR maintains strong performance across a broader local hyperparameter region rather than relying on a narrow operating point. This behavior is consistent with the adaptive gate suppressing the effective update magnitude for low-alignment corrections, thereby reducing sensitivity to the nominal refinement step size.

G Ablation Study

We ablate the components of TILR on GSM8K-Aug (Table 5). Starting from the residual-blend backbone (0.2561), unconstrained correction improves accuracy to 0.2691. Adding subspace projection increases accuracy to 0.2760, while adaptive gating alone achieves 0.2737. Combining both components yields the strongest performance at 0.2767, indicating that projection provides the larger individual gain while gating provides an additional improvement. We further examine the update design and choice of projection subspace. Linear damping achieves 0.2752, while removing the step-size cap reduces accuracy to 0.2721. Replacing the learned subspace with its orthogonal complement, input-PCA directions, or a rank-matched random basis yields 0.2667, 0.2335, and 0.2314, respectively, all below TILR. These ablations show that performance depends not only on restricting the update, but on using the learned refinement directions together with adaptive gating.

Table 6: Exact-match accuracy (%) and average generation tokens per inference on three benchmarks for **Qwen2.5-Math-1.5B** with LoRA (rank 16) and stepwise internalization curriculum.

	GSM8K		SVAMP		GSM-Plus	
Method	**Acc.**	**Tok.**	**Acc.**	**Tok.**	**Acc.**	**Tok.**
No-CoT	35.4±0.5	3	32.6±0.6	3	27.5±0.7	3
CoT	**63.5±0.4**	105	47.4±0.4	42	**43.7±0.5**	108
Coconut	56.2±0.8	9	51.0±0.9	4	38.4±1.1	9
Refinement	58.1±0.6	9	52.6±0.7	4	40.2±0.7	9
AdaAnchor	48.5±1.0	7	43.8±1.1	4	34.7±0.9	7
TILR (ours)	60.5±0.5	9	55.9±0.5	4	43.0±0.7	9

Table 7: Exact-match accuracy (%) and average generation tokens per inference on three benchmarks for **Llama3.2-1B-Instruct**.

	GSM8K		SVAMP		GSM-Plus	
Method	**Acc.**	**Tok.**	**Acc.**	**Tok.**	**Acc.**	**Tok.**
No-CoT	30.1±0.3	3	27.1±0.3	3	26.1±0.4	3
CoT	**60.5±0.4**	105	41.2±0.5	42	**41.5±0.5**	100
Coconut	50.1±0.5	9	41.2±0.6	5	39.1±0.8	7
Refinement	52.5±0.2	8	45.6±1.1	4	40.1±0.5	9
AdaAnchor	44.5±1.2	7	46.4±0.4	4	28.2±0.3	7
TILR (ours)	53.5±0.3	9	47.9±0.5	5	41.1±0.7	9

H Cross-Architecture Generalization

We evaluate TILR on two additional backbones, Qwen2.5-Math-1.5B (Yang et al., 2024) and Llama3.2-1B-Instruct (Grattafiori et al., 2024), to test whether the proposed refinement mechanism generalizes beyond the GPT-2-based models used in the main experiments. Tables 6 and 7 show that TILR consistently improves over unconstrained Refinement across both architectures and all three evaluated benchmarks. On Qwen2.5-Math-1.5B, TILR improves accuracy by $+2.4$, $+3.3$, and $+2.8$ percentage points on GSM8K, SVAMP, and GSM-Plus, respectively. On Llama3.2-1B-Instruct, the corresponding gains are $+1.0$, $+2.3$, and $+1.0$ points, with comparable generation lengths to the other latent reasoning methods. We further evaluate whether the input-invariance improvements generalize across architectures. As shown in Table 8, TILR improves answer agreement (M3) and reduces trajectory variance (M4) relative to unconstrained Refinement in all six backbone–benchmark combinations. On Qwen2.5-Math-1.5B, mean answer agreement improves by 0.073, with trajectory variance reduced by 3.2–27.3% across datasets. On Llama3.2-1B-Instruct, mean answer agreement improves by 0.030, with trajectory-variance reductions of 10.7–14.3%. Together with the GPT-2 results, these experiments indicate that both the accuracy and input-invariance benefits of TILR extend across backbone architectures and model scales, although the magnitude of the improvement varies across models and datasets.

I Broader Impact

Trajectory Invariant Latent Refinement (TILR) is a training-free inference-time approach designed to improve the stability of latent reasoning in language models. By restricting refinement to an empirically identified refinement subspace and selectively applying updates based on signal alignment, TILR reduces sensitivity to paraphrasing and reference-checkpoint variations without requiring retraining. In practice, this leads to more predictable reasoning behavior under semantically equivalent inputs, which can improve the reliability and auditability of LLM systems in settings where consistency matters, such as educational applications, scientific assistance, and decision-support workflows. Because the method relies only on forward passes and a calibration SVD for each reference configuration,

Table 8: Cross-architecture input invariance under semantically equivalent reformulations. M3 measures answer agreement (↑), and M4 measures latent trajectory variance (↓). Best results within each backbone and benchmark are shown in bold.

		M3: Answer Agreement ↑			M4: Trajectory Variance ↓		
Backbone	**Dataset**	**Coconut**	**Refinement**	**TILR**	**Coconut**	**Refinement**	**TILR**
Qwen2.5-1.5B	GSM8K	0.61	0.64	**0.68**	0.26	0.26	**0.21**
	SVAMP	0.51	0.55	**0.67**	0.33	0.31	**0.30**
	GSM-Plus	0.58	0.55	**0.61**	0.34	0.33	**0.24**
Llama3.2-1B-Instruct	GSM8K	0.55	0.55	**0.57**	0.29	0.28	**0.25**
	SVAMP	0.44	0.47	**0.51**	0.28	0.28	**0.24**
	GSM-Plus	0.51	0.52	**0.55**	0.22	0.22	**0.19**

it also offers a relatively compute-efficient path toward robustness improvements. At the same time, TILR does not address the fundamental problem of model correctness. The method improves invariance and consistency, not factual accuracy or calibration. If a model exhibits systematic errors or flawed reasoning, TILR may make those failures more stable and repeatable rather than correcting them. Although the adaptive gate suppresses unreliable corrections and recovers the uncorrected backbone as the alignment score approaches zero, it does not eliminate biases or inaccuracies inherited from the underlying checkpoints. More broadly, reducing the cost of reliable inference may accelerate deployment of LLM systems in high-stakes environments before adequate safeguards, evaluation standards, or verification mechanisms are in place. For these reasons, we view TILR as a robustness enhancement rather than a standalone trustworthiness solution. Downstream deployments should combine it with task-specific verification, uncertainty estimation, and human oversight where appropriate, and should avoid treating improved input invariance as evidence of correctness or safety.

J LIMITATIONS

TILR improves the stability of latent reasoning, but several limitations remain. The learned refinement subspace is empirical rather than theoretically grounded: we do not establish that its directions correspond to causally invariant reasoning features, and its quality depends on the calibration data and reference pair. The geometry of the learned subspace also varies across datasets, and the relationship between this variation and downstream robustness remains incompletely understood. TILR improves invariance, not correctness. If reference checkpoints encode systematic biases or errors, constraining refinement to stable directions does not necessarily correct them, and consistent trajectories should not be interpreted as evidence of factual accuracy. The evaluation is also limited in scale and domain: the main experiments use GPT-2-class backbones, with Qwen2.5-Math-1.5B and Llama3.2-1B-Instruct evaluated in the appendix, and the benchmarks focus on mathematical and deductive reasoning. Whether similar spectrally concentrated refinement structure persists in substantially larger models or domains such as tool use, planning, and interactive agents remains open. Finally, although TILR requires no additional training, it requires calibration forward passes and a truncated SVD for each reference configuration. How to maintain stable refinement directions under distribution shift without repeated recalibration remains an open problem.