Invariant Reasoning Directions in Latent Trajectories of Language Models

Arun Vignesh Malarkkan [1] **Manan Roy Choudhury** [*1] **Utkarsh Byahut** [*1] **Yash Ravindra Charde** [1]
Vivek Gupta [1] **Yanjie Fu** [1]

Abstract

Latent reasoning models perform multi-step inference directly in hidden-state space, yet the structure of these latent reasoning trajectories remains poorly understood. We show that contrastive refinement signals between stronger and weaker reasoning trajectories exhibit a highly concentrated low-rank structure, while unconstrained latent updates remain sensitive to paraphrases, checkpoint choice, and trajectory perturbations. These observations suggest that latent reasoning trajectories contain stable invariant directions mixed with unstable instance-specific variation. We introduce **Trajectory-Invariant Latent Refinement (TILR)**, a training-free intervention framework for identifying and manipulating stable reasoning directions in latent space. TILR first learns a low-rank invariant subspace from contrastive trajectory differences across inputs, then constrains latent interventions to this subspace while suppressing poorly aligned updates through an adaptive alignment gate. Across six reasoning benchmarks, we find that a small number of latent directions explain most variation between strong and weak reasoning trajectories. Interventions on these directions causally improve reasoning consistency and reduce trajectory instability under paraphrases and perturbations. TILR improves answer consistency under paraphrase by $\sim 10\%$ and reduces latent trajectory variance by up to 50% while preserving reasoning accuracy. These results support a geometric view of latent reasoning in which transferable reasoning behavior emerges from stable low-dimensional structure within hidden-state trajectories.

[*]Equal contribution [1]Arizona State University, Tempe, Arizona. Correspondence to: Arun Vignesh Malarkkan <arun.malarkkan@asu.edu>.

Mechanistic Interpretability Workshop at the 43rd International Conference on Machine Learning, Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

1. Introduction

Latent reasoning models perform multi-step inference directly in hidden-state space, iteratively updating internal representations without generating explicit intermediate tokens. This paradigm enables reasoning without the token overhead of Chain-of-Thought prompting (Wei et al., 2022), while retaining competitive performance across mathematical and multi-hop reasoning tasks. However, despite growing interest in latent reasoning (Hao et al., 2024), little is known about the internal structure of the latent reasoning trajectories these models produce. In practice, semantically equivalent inputs often induce divergent latent trajectories, even when they produce the same final answer. Conversely, some latent directions remain stable across paraphrases, perturbations, and reference checkpoints. These observations suggest that latent reasoning trajectories may contain both stable reasoning-associated structure and unstable instance-specific variation.

We hypothesize that reasoning behavior that remains stable across paraphrases, perturbations, and reference checkpoints is concentrated in a low-dimensional set of invariant latent directions, while unstable trajectory components correspond to paraphrase-sensitive, checkpoint-specific, or input-specific variation. Under this view, reasoning failures arise not because latent reasoning lacks useful signal, but because existing methods fail to distinguish stable reasoning-associated directions from unstable trajectory components.

Recent latent refinement methods (Wang et al., 2026) steer trajectories using contrastive signals derived from stronger and weaker reference checkpoints. These methods improve reasoning performance without retraining, but the resulting latent updates are often unstable: semantically equivalent inputs can produce divergent trajectories, and refinement behavior can vary substantially across perturbations and reference choices. This suggests that existing methods operate on unconstrained latent directions that mix stable reasoning-associated structure with unstable variation.

Our Perspective. We argue that latent reasoning trajectories contain a structured decomposition between stable reasoning-associated directions and unstable instance-specific components. Existing refinement methods treat contrastive updates as homogeneous vectors in the full

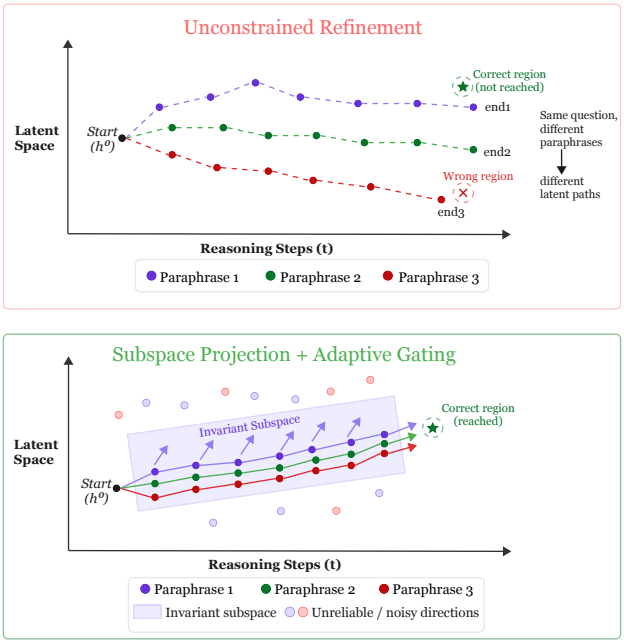

Figure 1. **Why invariant component?**

embedding space, propagating both components equally. Empirically, we find that differences between strong and weak reasoning trajectories are highly concentrated in low-dimensional subspaces that remain stable across paraphrases and reference choices. This suggests that reasoning-quality information occupies a restricted set of latent directions distinct from generic trajectories.

We introduce **Trajectory-Invariant Latent Refinement (TILR)**, a training-free intervention framework for identifying and manipulating stable reasoning-associated directions in latent space. TILR first identifies a low-rank invariant subspace in which stronger and weaker reasoning trajectories consistently differ across inputs. Contrastive updates are then constrained to this subspace, isolating stable contrastive structure from unstable trajectory variation. An adaptive alignment gate further modulates intervention strength according to how strongly the current trajectory aligns with the learned invariant structure, suppressing poorly aligned interventions while preserving stable latent directions.

Our goal is not to identify semantic neurons or explicit reasoning circuits, but to study the geometric structure of latent reasoning trajectories through subspace-level interventions. To analyze latent reasoning beyond standard accuracy metrics, we introduce trajectory-level analyses that measure paraphrase consistency, latent trajectory variance, and sensitivity to reference checkpoint choice. Across six benchmarks, we find that a small number of latent directions explain most variation between strong and weak reasoning trajectories. Subspace interventions consistently affect reasoning consistency and trajectory stability under paraphrases and perturbations while preserving accuracy.

Contributions.

- We provide evidence that latent reasoning trajectories exhibit highly concentrated low-rank structure, with stable reasoning-quality differences occupying a restricted set of latent directions.
- We introduce TILR, a training-free intervention framework that isolates and manipulates invariant reasoning-associated directions through subspace-constrained latent interventions.
- We show through subspace interventions that these invariant directions play a functional role in reasoning consistency and trajectory stability.
- We introduce trajectory-level analyses for studying latent reasoning invariance, including paraphrase consistency, trajectory variance, and checkpoint-pair sensitivity.

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.

Design implication. Since the contrastive direction d_t is fixed by the reference pair and input, robustness must be achieved by controlling how d_t is applied. In particular, instability arises when components of d_t that are specific to individual inputs or reference pairs are treated on equal footing with components that reflect consistent reasoning-quality differences.

4. Invariant Structure in Latent Reasoning Trajectories

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: Low-Rank Structure Let $\delta_t = h_t^{\text{good}} - h_t^{\text{bad}}$ denote the contrastive difference. Empirically, we observe that these differences are highly concentrated in a low-dimensional subspace 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.

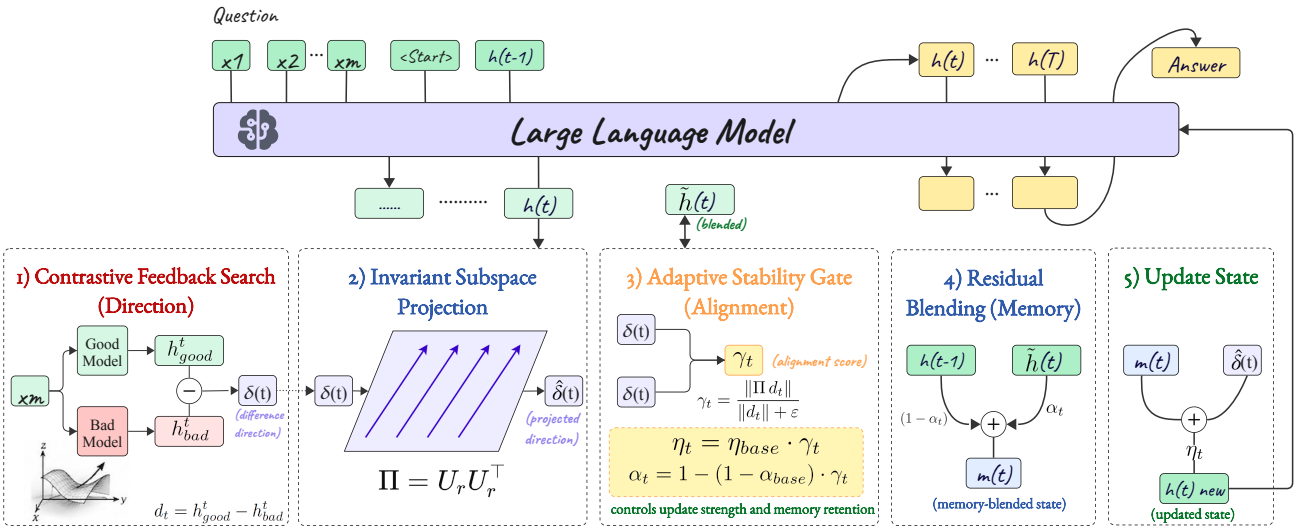

Figure 2. **The Trajectory-Invariant Latent Refinement (TILR) Framework Overview.** TILR enhances latent reasoning by constraining contrastive updates to a stable reasoning subspace. (1) **Contrastive Feedback Search** identifies a direction d_t pointing from weak to strong reference models. (2) **Invariant 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.

4.1. Invariant Latent Directions

4.1.1. Identifying Stable Latent Structure

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

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

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 low rank structure hypothesis; we ablate against per-step subspaces in the appendix.

4.1.2. Subspace-Constrained Latent Interventions

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

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

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:

$$\eta_t = \eta_{\text{base}} \cdot \gamma_t. \tag{8}$$

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

$$h^t \longrightarrow \begin{cases} \tilde{h}^t + \eta_{\text{base}} \Pi d_t, & \gamma_t \to 1, \\ \alpha_{\text{base}} h^{t-1} + (1 - \alpha_{\text{base}}) f(h^{t-1}), & \gamma_t \to 0. \end{cases} \tag{9}$$

Here, $\gamma_t \to 1$ corresponds to the full projected correction, while $\gamma_t \to 0$ recovers the uncorrected backbone. When γ_t is high, the full projected correction is applied on top of the residual blend. When γ_t is low, the correction vanishes and TILR recovers the uncorrected system exactly, providing a strict no-harm guarantee for low-confidence inputs.

Sensitivity flattening on low-confidence inputs. A direct consequence of the multiplicative gate $\eta_t = \eta_{\text{base}} \cdot \gamma_t$ 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 Eq. (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.

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**
  $\tilde{h}^t \leftarrow \alpha_{\text{base}} h^{t-1} + (1 - \alpha_{\text{base}}) f(h^{t-1})$
  $d_t \leftarrow f_{\text{good}}(\tilde{h}^t) - f_{\text{bad}}(\tilde{h}^t)$
  $\gamma_t \leftarrow \|\Pi d_t\| / (\|d_t\| + \varepsilon)$; $\eta_t \leftarrow \eta_{\text{base}} \gamma_t$
  $h^t \leftarrow \tilde{h}^t + \eta_t \Pi d_t$
end for
$y \leftarrow \text{Decode}(h^T)$
return y

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.

Inference procedure. Algorithm 1 summarizes the full TILR update. The computation is sequential: α_{base} is fixed, γ_t is computed once per step from the projected contrastive direction, and the correction is applied additively. In the low-alignment limit $\gamma_t \to 0$, the correction vanishes and TILR recovers the uncorrected residual-blended backbone. In the high-alignment limit $\gamma_t \to 1$, TILR applies the full projected correction.

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:** Do contrastive differences exhibit low-rank structure distinct from generic representation variance? **RQ2:** Is the learned subspace functionally necessary for refinement performance? **RQ3:** Does TILR reduce sensitivity to semantically equivalent input reformulations? **RQ4:** Does TILR improve or preserve reasoning accuracy relative to existing refinement methods? **RQ5:** Is TILR robust to the choice of contrastive reference checkpoint pair? **RQ6:** How much do subspace projection and adaptive gating individually contribute to TILR's performance? Additional analyses of hyperparameter sensitivity and cross-architecture generalization are reported in the Appendix.

Benchmarks. We evaluate on six reasoning benchmarks spanning spanning both standard and robustness-focused settings. For standard math reasoning: **GSM8K** (Cobbe et al., 2021), **MathQA** (Amini et al., 2019), and **AQUA-RAT** (Ling et al., 2017). For robustness-stressed math reasoning: **SVAMP** (Patel et al., 2021) (lexical perturbations of math word problems) and **GSM-Plus** (Li et al., 2024) (eight-way numeric and structural perturbations of GSM8K). For commonsense multi-hop reasoning: **StrategyQA** (Geva et al., 2021). This combination allows us to isolate both accuracy and robustness properties under controlled perturbations.

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.

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.

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: Geometric Structure of Latent Reasoning Directions. We study whether the contrastive refinement direction is concentrated in a low-dimensional latent subspace, and whether that subspace 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. Low $r_{0.90}$ values show that the refinement signal is highly concentrated; high retained energy after input control indicates that the learned direction is not merely ordinary input variance. Table 1 shows that the learned contrastive signal is consistently low-rank: the average $r_{0.90}$ is 12, and three datasets require only a single direction to explain 90% of the contrastive energy. GSM8K, MathQA, and AQuA-RAT show richer but still compact structure, with $r_{0.90} \in \{8, 27, 34\}$ and top-10 explained variance between 0.801 and 0.913. The input-controlled analysis separates two regimes. On MathQA and AQuA-RAT, a large fraction of contrastive energy is intact after removing the input-PCA component, which supports a non-trivial refinement-specific subspace. On SVAMP, GSM-Plus, and StrategyQA, the signal is dominated by a single direction, which explains the rank-one or near rank-one spectra.

Table 1. Spectral structure of the learned refinement subspace. Low $r_{0.90}$ and high EV indicate low-rank structure.

Dataset	$r_{0.90}$	Top-1 EV	Top-10 EV	Align input	Ctrl. energy
GSM8K	8	0.432	0.913	0.779	0.265
MathQA	27	0.336	0.836	0.586	0.756
AQuA-RAT	34	0.416	0.801	0.621	0.789
SVAMP	1	0.931	1.000	0.939	0.012
GSM-Plus	1	0.924	0.981	0.914	0.038
StrategyQA	1	0.952	1.000	0.922	0.002

RQ2: Functional Role of Invariant Latent Directions. 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 causal intervention results 3 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 all datasets. This supports the interpretation that the learned directions carry task-relevant refinement information, not just low-rank structure.

RQ3. Trajectory Invariance Under Paraphrases We evaluate input invariance by measuring answer agreement (M3, ↑) and latent trajectory variance (M4, ↓) across $M{=}5$ semantically equivalent reformulations per test input gener-

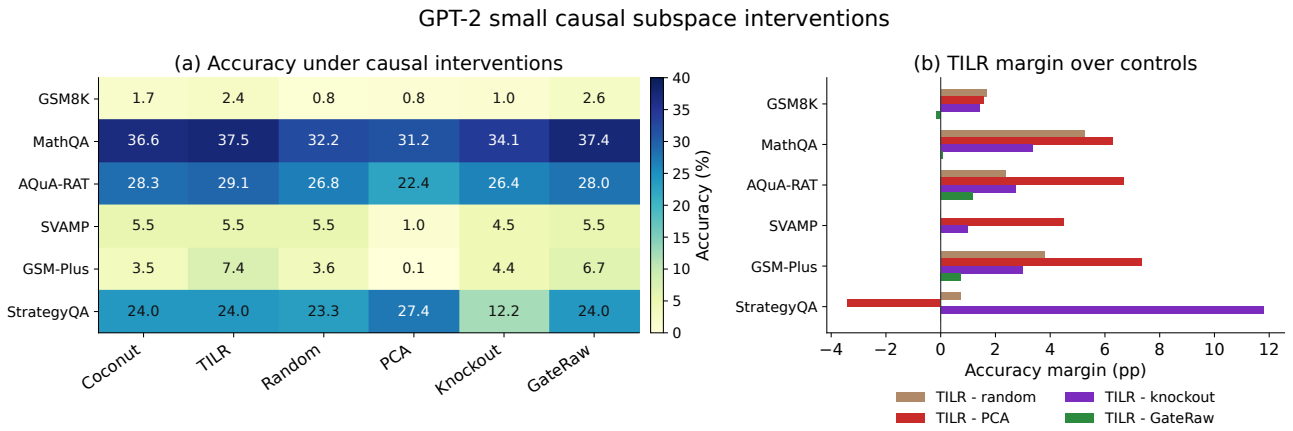

Figure 3. **Causal subspace interventions.** Panel (a) reports accuracy under different subspaces. Panel (b) shows TILR margins over the strongest causal controls.

ated using Llama-3-Instruct (Grattafiori et al., 2024) (Figure 4). We observe that TILR reduces trajectory variance by $25 - 53\%$ relative to Refinement on every benchmark, with a mean reduction of 39%, while the unconstrained refinement (Wang et al., 2026) leaves trajectory variance essentially unchanged from the Coconut baseline. This supports the claim that subspace projection suppresses paraphrase-specific components of the contrastive direction before they affect the trajectory (Section B). Unconstrained refinement, on the other hand uses the unprojected d_t, has no such filter and inherits the surface-level variation directly into the trajectory. Second, the trajectory-level invariance translates into answer-level consistency: TILR improves M3 over Refinement by $+0.083$ on average. On StrategyQA, the M3 improvement is small ($+0.02$) despite the largest M4 reduction (-43%), reflecting the binary-task answer space rather than a failure of the mechanism: trajectory invariance has limited room to translate into answer changes when there are only two possible answers. We note that absolute agreement remains far from 1.0 even under TILR (mean 0.60 across benchmarks), indicating that complete input invariance in latent reasoning at this scale remains an open problem.

RQ4: Effects of Invariant Latent Interventions on Reasoning. We compare TILR on GPT2 with the baseline methods across six reasoning benchmarks to test whether subspace-projected, gate-modulated refinement improves accuracy without training. Table 2 shows that TILR outperforms the baselines on all six benchmarks. Specifically, TILR improves over unconstrained Refinement by an average of 2.1%, while preserving the token efficiency of latent reasoning. The largest gains over Coconut appear on StrategyQA ($+5.5\%$), SVAMP ($+3.9\%$), and MathQA ($+3.6\%$), with clear improvements on the robustness-oriented benchmarks SVAMP and GSM-Plus. CoT remains strongest on GSM8K, which is consistent with the known advantage of explicit symbolic traces on open-ended multi-step arithmetic reasoning. 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 backbone (Appendix H), where TILR improves over unconstrained refinement on most benchmarks, with larger gains on robustness-oriented datasets.

RQ5: Stability of Latent Reasoning Directions Across Checkpoints. 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 5. TILR substantially reduces checkpoint-pair variability on GSM8K, AQuA-RAT, and GSM-Plus, with relative standard-deviation reductions of 72.7%, 74.2%, and 53.0%, respectively. One exception is MathQA: TILR is more sensitive to the choice of reference pair, which is consistent with the lower subspace overlap observed on that dataset. These results suggest that the learned invariant subspace is stable on most tasks, but not universally so.

RQ6: Contributions of Subspace Projection and Adaptive Gating. We evaluate the contribution of the two components of TILR: subspace projection and adaptive gating on SVAMP and StrategyQA (Table 4). Both components independently improve over the baseline refinement method. Applying projection alone yields gains of $+1.2$ and $+2.0$ points over Refinement on SVAMP and StrategyQA, respectively, while gating alone provides smaller but consistent improvements. Combining both components produces the strongest performance, with TILR outperforming Refinement by $+2.4$ and $+3.1$ points. These results indicate that the two components play complementary roles. Subspace projection contributes the majority of the improvement by constraining updates to directions that are consistent across inputs, while adaptive gating further refines the update by modulating its magnitude based on signal reliability.

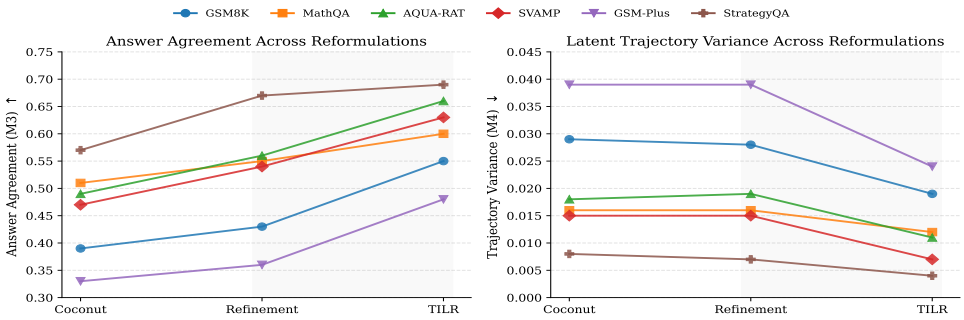

Figure 4. **Input invariance under perturbations. Left:** Answer agreement (M3, ↑) across semantically equivalent reformulations per input. **Right:** Latent trajectory variance (M4, ↓).

Table 2. 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 over 3 seeds. **Bold** indicates best overall; indicates best among latent methods; underline indicates second-best latent method. Token counts are averaged across seeds.

Method	GSM8K Acc.	GSM8K Tok.	MathQA Acc.	MathQA Tok.	AQUA-RAT Acc.	AQUA-RAT Tok.
No-CoT (Radford et al., 2019)	16.5±0.5	2.5	24.5±0.4	3	22.1±0.5	3
CoT (Wei et al., 2022)	42.9±0.2	28	34.2±0.3	46	27.1±0.4	45
Coconut (Hao et al., 2024)	34.1±1.5	8	38.5±0.5	9	26.4±0.5	9
Refinement (Wang et al., 2026)	34.5±1.2	8	40.1±0.9	9	30.4±1.5	9
AdaAnchor (Sheshanarayana et al., 2026)	23.5±0.8	6	37.5±1.1	9	29.6±0.3	9
TILR	35.8±0.6	8	42.1±0.4	9	32.3±0.8	9

Method	SVAMP Acc.	SVAMP Tok.	GSM-Plus Acc.	GSM-Plus Tok.	StrategyQA Acc.	StrategyQA Tok.
No-CoT (Radford et al., 2019)	14.1±0.6	2	12.3±0.6	3	52.3±0.2	1
CoT (Wei et al., 2022)	28.2±0.5	24	24.7±0.8	30	55.2±0.5	22
Coconut (Hao et al., 2024)	30.0±0.9	4	20.9±1.2	8	60.0±0.9	2
Refinement (Wang et al., 2026)	31.5±0.4	4	21.5±0.2	8	62.4±1.2	2
AdaAnchor (Sheshanarayana et al., 2026)	18.4±0.9	4	19.2±0.5	8	56.1±0.4	2
TILR	33.9±0.4	4	23.2±0.8	8	65.5±0.6	2

Table 3. **Robustness to checkpoint-pair selection.** Lower values indicate stronger robustness.

Dataset	Sensitivity across pairs			Subspace	
	Ref. std	TILR std	Std. red.	Range red.	Mean overlap
GSM8K	0.160	**0.044**	**+72.7**	**+66.7**	0.837
MathQA	**0.128**	0.294	-129.3	-154.5	0.326
AQuA-RAT	0.719	**0.186**	**+74.2**	**+80.0**	0.434
GSM-Plus	0.350	**0.165**	**+53.0**	**+43.5**	0.988

Table 4. **Ablation study on SVAMP and StrategyQA.** Mean accuracy over 3 seeds. We evaluate the contribution of subspace projection and adaptive gating.

Method	SVAMP	StrategyQA
Coconut	30.0	60.0
Refinement	31.5	62.4
TILR (full)	33.9	65.5
Projection only	32.7	64.4
Gating only	31.9	63.3

Overall, the results show that TILR preserves or improves accuracy, reduces sensitivity to semantically equivalent inputs, and makes the refinement signal more structured and more useful.

7. Conclusion

We studied the geometric structure of latent reasoning trajectories and introduced Trajectory Invariant Latent Refinement (TILR), a training-free intervention framework for isolating stable reasoning-associated directions in latent space. Our results show that differences between strong and weak reasoning trajectories are highly concentrated in low-dimensional invariant subspaces, and that interventions on these directions consistently affect reasoning consistency and trajectory stability. Across six reasoning benchmarks, TILR improves paraphrase consistency and reduces latent trajectory variance while preserving reasoning accuracy. Empirically, we observed that the latent reasoning behavior is not distributed uniformly across the embedding space. Instead, a small number of latent directions capture much of the transferable structure associated with stable reasoning behavior across paraphrases, perturbations, and reference choices. These findings support a geometric view of latent reasoning in which stable reasoning behavior emerges from restricted low-dimensional structure within hidden-state trajectories. Under this view, failures in latent reasoning arise not because useful signal is absent, but because a stable reasoning-associated structure is mixed with unstable trajectory variation. TILR improves latent interventions by separating these components and constraining updates to invariant latent directions. Moreover, while TILR substantially reduces trajectory divergence, it does not eliminate latent instability entirely. Our goal is therefore not to identify explicit reasoning circuits or semantic neurons, but to study latent reasoning through trajectory geometry and subspace-level interventions. Extending this perspective toward larger models and finer-grained mechanistic analyses remains an important direction for future work.

Acknowledgement

This research was partially supported by the National Science Foundation (NSF) under Grant Nos. 2426340, 2416727, 2421864, 2421865, and 2421803, and by the National Academy of Engineering Grainger Foundation Frontiers of Engineering Grants. Any opinions, findings, conclusions, or recommendations expressed in this work are those of the authors and do not necessarily reflect the views of the funding agencies.

References

Amini, A., Gabriel, S., Lin, S., Koncel-Kedziorski, R., Choi, Y., and Hajishirzi, H. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2357–2367, 2019.

Arjovsky, M., Bottou, L., Gulchandani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Deng, Y., Prasad, K., Fernandez, R., Smolensky, P., Chaudhary, V., and Shieber, S. Implicit chain of thought reasoning via knowledge distillation. *arXiv preprint arXiv:2311.01460*, 2023.

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

Geva, M., Khashabi, D., Segal, E., Khot, T., Roth, D., and Berant, J. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9: 346–361, 2021.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., and et al., A. A.-D. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

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

Li, Q., Cui, L., Zhao, X., Kong, L., and Bi, W. 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.

Li, X. L., Holtzman, A., Fried, D., Liang, P., Eisner, J., Hashimoto, T. B., Zettlemoyer, L., and Lewis, M. 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.

Lin, S., Yang, L., Fan, D., and Zhang, J. Trgp: Trust region gradient projection for continual learning. *arXiv preprint arXiv:2202.02931*, 2022.

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

Patel, A., Bhattamishra, S., and Goyal, N. 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.

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

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Rimsky, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., and Turner, A. 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.

Saha, G., Garg, I., and Roy, K. Gradient projection memory for continual learning. *arXiv preprint arXiv:2103.09762*, 2021.

Sheshanarayana, D., Pal, R. S., Sinha, M., and Dasgupta, T. Thinking in latents: Adaptive anchor refinement for implicit reasoning in llms. *arXiv preprint arXiv:2603.15051*, 2026.

Turner, A. M., Thiergart, L., Leech, G., Udell, D., Vazquez, J. J., Mini, U., and MacDiarmid, M. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023.

Wang, X., Wang, D., Ying, W., Bai, H., Gong, N., Dong, S., Liu, K., and Fu, Y. 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.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

Yang, A., Zhang, B., Hui, B., Gao, B., Yu, B., Li, C., Liu, D., Tu, J., Zhou, J., Lin, J., Lu, K., Xue, M., Lin, R., Liu, T., Ren, X., and Zhang, Z. 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:

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

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. The correction can therefore only help or be neutral; it cannot degrade performance below the uncorrected baseline on low-confidence inputs.

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.

Computational Overhead. TILR adds one matrix-vector product ($\mathcal{O}(dr)$ via $U_r^\top d_t$ then $U_r(\cdot)$), one norm computation ($\mathcal{O}(r)$), and one scalar multiplication per reasoning step, compared to case of gpt2 at $d = 768$ and $r \leq 20$, this is negligible relative to the $\mathcal{O}(d^2)$ cost of a single transformer forward pass. The one-time calibration is dominated by NT reference forward passes plus a truncated SVD on a $d \times NT$ matrix.

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: $|\operatorname{Acc}(\alpha', \eta') - \operatorname{Acc}(\alpha, \eta)| \leq \epsilon$ 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.

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

$$\eta_t = \eta_{\text{base}} \cdot \gamma_t, \qquad \gamma_t = \frac{\|\Pi d_t\|}{\|d_t\| + \epsilon}.$$

Substituting the gated step size gives

$$h_t = \tilde{h}_t + \eta_{\text{base}} \frac{\|\Pi d_t\|}{\|d_t\| + \epsilon} \Pi d_t.$$

We compare the sensitivity of this update to perturbations in the nominal step size

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

Unconstrained refinement. For the standard refinement update

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

the induced perturbation is

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

with magnitude

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

Thus the sensitivity scales linearly with the full contrastive signal magnitude.

TILR update sensitivity. Under TILR, the same perturbation yields

$$\Delta h_t^{\mathrm{TILR}} = \Delta\eta \frac{\|\Pi d_t\|}{\|d_t\| + \epsilon} \Pi d_t,$$

and therefore

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

Since

$$\|\Pi d_t\| \leq \|d_t\|,$$

we obtain the bound

$$\|\Delta h_t^{\mathrm{TILR}}\| \leq |\Delta\eta| \, \|d_t\|.$$

More importantly, when the projected component is small relative to the full signal,

$$\|\Pi d_t\| \ll \|d_t\|,$$

the perturbation scales quadratically in the projected magnitude:

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

Thus, low-alignment inputs exhibit strongly suppressed sensitivity to perturbations in the nominal step size. The projection operator removes components of the contrastive signal that lie outside the learned invariant subspace, while the multiplicative gate further suppresses updates whose remaining projected component is weak. Consequently, inputs dominated by unstable or instance-specific directions contribute little to the effective refinement update, even when the nominal step size is large. This analysis is local and mechanistic: it characterizes how the gated update responds to perturbations in the nominal step size, rather than establishing global optimization or generalization guarantees.

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 $\delta_i^t = h_{\text{good},i}^t - h_{\text{bad},i}^t$. 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 per dataset. The truncated SVD is well-conditioned when $NT \gg r$, which holds across all benchmarks (NT=600, with $r_{0.90} \leq 34$ per RQ5), so the leading singular directions are stable to small perturbations of the calibration draw. Total calibration cost is NT reference forward passes plus the SVD, completing in under 30 seconds on a single H100. The cost is one-time and amortized across all subsequent inference.

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 compared to unconstrained refinement. For each dataset, we evaluate accuracy over a grid of (α, η) values. We report: (i) the best accuracy achieved within the grid, (ii) the worst-case accuracy in the local neighborhood of the best point, and (iii) the resulting sensitivity gap between best and worst performance.

Table 5 summarizes the sensitivity behavior on GSM8K and MathQA. On both datasets, TILR improves the best achievable accuracy while also increasing worst-case performance within the neighborhood of the optimum. This leads to a consistent reduction in the sensitivity gap relative to unconstrained refinement. On GSM8K, TILR improves the best accuracy from 0.345 to 0.358 and reduces the gap from 0.024 to 0.018. On MathQA, TILR improves the best accuracy from 0.401 to 0.421 while reducing the gap from 0.033 to 0.024. In both cases, the worst-case accuracy also increases, indicating that the improvement is not limited to a narrow region of the hyperparameter space. These results suggest that constraining the

Table 5. **RQ2: Hyperparameter sensitivity on completed benchmarks.** 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	0.345	0.321	0.024	**0.358**	**0.34**	**0.018**
MathQA	0.401	0.368	0.033	**0.421**	**0.397**	**0.024**

Table 6. **RQ3 on GPT-2 Medium: checkpoint-pair sensitivity.** Lower is better for std and range; higher is better for reductions.

Dataset	Ref. std	TILR std	Std. red.	Ref. range	TILR range	Range red.	Mean overlap
SVAMP	0.80	**0.61**	**23.8%**	2.00	**1.50**	**25.0%**	0.9968
StrategyQA	4.04	**0.33**	**91.8%**	10.00	**0.80**	**92.0%**	0.9947

contrastive update to an invariant subspace, combined with adaptive gating, produces a more stable refinement behavior with respect to hyperparameter variation. By suppressing components of the contrastive signal that are inconsistent across inputs, TILR reduces sensitivity to the choice of (α, η) while maintaining strong operating points.

G. GPT-2 Medium Results

We further evaluate TILR on GPT-2 Medium using the three evaluated checkpoint families for this backbone: GSM8K, SVAMP, and StrategyQA. All GPT-2 Medium runs use the same April-style checkpoint protocol as the GPT-2 experiments, but with hidden dimension $d = 1024$ and checkpoint directories `GPT2-Med/gpt2-medium_{dataset}`. These results test whether the stability and subspace-structure conclusions persist when moving from GPT-2 to a larger backbone.

G.1. RQ3: Checkpoint-Pair Sensitivity

The GPT-2 Medium RQ3 results are strongly favorable for TILR from the standpoint of robustness. GSM8K is already insensitive to checkpoint-pair choice under both methods, and TILR preserves this zero-variance behavior. On SVAMP, TILR reduces the standard deviation from 0.80 pp to 0.61 pp and the range from 2.00 pp to 1.50 pp. The most substantial effect appears on StrategyQA: standard refinement varies by 10.00 pp across checkpoint pairs, whereas TILR reduces the range to only 0.80 pp. This corresponds to a 92.0% reduction in range and a 91.8% reduction in standard deviation. The high subspace overlaps on SVAMP and StrategyQA further indicate that, for GPT-2 Medium, the learned refinement directions are highly stable across checkpoint-pair choices.

G.2. RQ5: Geometry of the Learned Subspace

The RQ5 results are good evidence for the core geometric claim: the contrastive refinement signal in GPT-2 Medium is concentrated in a very small subspace. GSM8K has $r_{0.90} = 3$, while SVAMP and StrategyQA require only one direction to explain 90%, 95%, and 99% of the contrastive energy. The GSM8K subspace is the most geometrically nontrivial: its alignment with input PCA is moderate rather than degenerate (0.7188), and it retains 23.39% of contrastive energy after input-PCA control. In contrast, SVAMP and StrategyQA are nearly rank-one and highly aligned with trajectory and good-only directions; after input control, little residual contrastive energy remains. Thus, RQ5 supports a strong low-rank story for GPT-2 Medium, while also showing that the semantic distinctness of the learned direction is dataset-dependent.

G.3. RQ7: Causal Subspace Intervention

RQ7 provides the most direct causal test for GPT-2 Medium. On StrategyQA, TILR is clearly favorable: it improves over Coconut by 0.20 pp, beats the random subspace by 8.00 pp, beats PCA by 2.60 pp, and outperforms the orthogonal knockout by 4.60 pp. This is the clearest GPT-2 Medium causal success case and supports the interpretation that the learned rank-one direction carries useful task information. On GSM8K, TILR ties Coconut, random, and GateRaw at 4.02%, while still outperforming PCA and knockout controls, so the learned direction is at least non-harmful and competitive with the base path in this setting. SVAMP is the main outlier, where TILR underperforms Coconut, PCA, and knockout. The appendix-supported framing is therefore selective rather than universal: GPT-2 Medium causal gains are strongest on StrategyQA, non-degrading on GSM8K, and dataset-dependent overall.

Table 7. **RQ5 on GPT-2 Medium: spectral structure and input-controlled geometry.**

Dataset	d	$r_{0.90}$	$r_{0.95}$	$r_{0.99}$	Align(input)	Align(traj.)	Align(good)	Controlled energy
GSM8K	1024	3	15	107	0.7188	0.7043	0.7151	0.2339
SVAMP	1024	1	1	1	0.9820	0.9828	0.9900	0.0068
StrategyQA	1024	1	1	1	0.8661	0.9937	0.9909	0.0030

Table 8. **RQ7 on GPT-2 Medium: causal subspace intervention and controls.** Higher is better.

Dataset	N	r	Coconut	TILR	Random	PCA	Knockout	GateRaw	TILR-Rand.	TILR-PCA
GSM8K	1319	3	**4.02**	**4.02**	**4.02**	1.52	2.20	**4.02**	+0.00	**+2.50**
StrategyQA	500	1	9.80	**10.00**	2.00	7.40	5.40	**10.00**	**+8.00**	**+2.60**

H. Cross-Architecture Generalization (Qwen2.5-Math-1.5B)

We evaluate TILR on a stronger backbone, Qwen2.5-Math-1.5B (Yang et al., 2024), to test whether the proposed refinement mechanism generalizes beyond GPT-2-based models. The backbone is trained using the same latent reasoning framework with LoRA adaptation (rank 16) under identical conditions. Table 9 reports exact-match accuracy and average generation tokens across six reasoning benchmarks. The Qwen results support two conclusions. First, the refinement mechanism underlying TILR is not specific to a particular backbone or training regime: the same subspace projection and gating procedure yields consistent improvements across architectures. Second, the gains are larger on datasets with greater surface variation (SVAMP, GSM-Plus), reinforcing the interpretation that TILR primarily improves robustness to input perturbations rather than uniformly increasing accuracy. However, MathQA shows only marginal gains, suggesting that when the contrastive signal is already stable, subspace filtering provides limited additional benefit (Qwen-2.5 Math is expected to perform on Math reasoning benchmarks). This is consistent with the subspace geometry analysis in Section 6, which shows dataset-dependent variation in the structure of the contrastive signal. Overall, these results indicate that TILR generalizes to stronger latent reasoning backbones while preserving its core properties: improved consistency under perturbations, modest accuracy gains, and no additional inference cost.

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 invariant 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 one-time SVD computation, 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 graceful degradation property described in Sec. 4 limits worst-case performance drops relative to the unmodified backbone, 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 invariant subspace is empirical, not 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. In some datasets, the learned subspace exhibits weaker geometric

Table 9. Exact-match accuracy (%) and average generation tokens per inference on six benchmarks for **Qwen2.5-Math-1.5B** with LoRA (rank 16) and stepwise internalization curriculum.

	GSM8K		MathQA		AQUA-RAT	
Method	**Acc.**	**Tok.**	**Acc.**	**Tok.**	**Acc.**	**Tok.**
No-CoT	35.4±0.5	3	42.2±0.5	5	37.1±0.6	4
CoT	**63.5±0.4**	105	52.4±0.5	68	44.8±0.5	82
Coconut	56.2±0.8	9	57.4±0.7	9	42.3±0.8	9
Refinement	58.1±0.6	9	59.6±0.6	9	45.1±1.0	9
AdaAnchor	48.5±1.0	7	53.1±1.1	9	40.5±0.9	9
TILR (ours)	60.5±0.5	9	60.1±1.2	9	47.6±0.6	9

	SVAMP		GSM-Plus		StrategyQA	
Method	**Acc.**	**Tok.**	**Acc.**	**Tok.**	**Acc.**	**Tok.**
No-CoT	32.6±0.6	3	27.5±0.7	3	57.9±0.5	1
CoT	47.4±0.4	42	**43.7±0.5**	108	63.1±0.5	32
Coconut	51.0±0.9	4	38.4±1.1	9	65.4±0.8	2
Refinement	52.6±0.7	4	40.2±0.7	9	67.3±0.9	2
AdaAnchor	43.8±1.1	4	34.7±0.9	7	61.6±0.8	2
TILR (ours)	55.9±0.5	4	43.0±0.7	9	70.3±0.5	2

consistency across reference checkpoints, which reduces the robustness gains obtained from subspace projection. TILR improves invariance, not correctness. If reference checkpoints encode systematic biases, the projected refinement preserves those errors more uniformly across inputs, and stable trajectories should not be read as evidence of factual accuracy. The evaluation is bounded in scale (GPT-2-class backbones in the main paper, Qwen-2.5 1.5B in the appendix) and in domain (mathematical and multi-hop reasoning); whether the same low-rank geometry persists in larger models or domains such as tool use, planning, or interactive agents is open. Finally, while TILR requires no training, it does require calibration forward passes and a truncated SVD per reference family, so maintaining stable refinement directions under distribution shift without repeated recalibration remains an open problem.