GRPO-SoftCoT++: Latent-Space Contrastive Reinforcement Learning for Stable Multi-Step Reasoning in Large Language Models

Anonymous ACL submission

Abstract

Large Language Models (LLMs) achieve strong surface-level text generation but often struggle with reliable multi-step reasoning, where behavior resembles statistical pattern matching rather than systematic deduction. Reinforcement learning (RL) introduces a promising "think-before-speak" paradigm, yet token-level RL in discrete action spaces suffers from sample inefficiency, high gradient variance, and catastrophic forgetting. We propose **GRPO-SoftCoT++**, a latent-space contrastive reinforcement learning framework that shifts reasoning exploration from token sequences to a continuous semantic manifold. A lightweight assistant samples multiple latent reasoning trajectories, which are evaluated by correctness- and format-based rewards and selectively decoded by a frozen main model. Group-relative policy optimization ensures stable latent-space learning, while a contrastive objective encourages diverse yet coherent reasoning paths. Experiments on GSM8K and MATH show that GRPO-SoftCoT++ improves Pass@1 accuracy by **+4.3%** and **+7.2%** over SoftCoT++, respectively, with more stable convergence under comparable computational budgets, demonstrating the effectiveness of latent-space reinforcement learning for long-horizon reasoning.

1 Introduction

Large Language Models (LLMs), pretrained on massive-scale corpora, have demonstrated remarkable capabilities in natural language understanding and generation. Despite their strong surface-level fluency and factual recall, a growing body of evidence indicates that their internal reasoning behavior often relies more on probabilistic pattern matching than on systematic logical deduction. When applied to complex multi-step reasoning taskssuch as mathematical problem solving, symbolic manipulation, or causal inferenceLLMs frequently produce intermediate steps that are linguistically plausible yet logically inconsistent or erroneous. This discrepancy reveals a fundamental misalignment between standard pretraining objectives and the requirements of reliable long-horizon reasoning.

Reinforcement Learning (RL), particularly approaches based on verifiable rewards (RLVR) (Lightman et al., 2023), has emerged as a promising post-training paradigm to address this limitation. By explicitly optimizing model outputs with respect to task-level correctness, RL aims to transform pretrained language models from passive sequence predictors into active reasoning agents that "think before they speak." However, when applied directly to autoregressive LLMs operating in high-dimensional discrete token spaces, RL introduces several intrinsic sources of instability that significantly hinder its scalability and effectiveness.

First, token-level RL suffers from severe **sample inefficiency**. Even minor policy updates often require full-sequence rollouts to obtain sparse and delayed reward signals, resulting in excessive computational overhead. Second, **gradient estimation variance** increases rapidly with sequence length, as importance sampling errors accumulate exponentially across long reasoning chains, frequently leading to unstable or oscillatory training dynamics. Third, and most critically, continuous full-parameter optimization exposes the model to **catastrophic forgetting**, whereby newly acquired task-specific reasoning behaviors overwrite the pretrained representations responsible for general linguistic competence and world knowledge. Together, these challenges fundamentally limit the practicality of applying conventional RL directly in token space for complex reasoning tasks.

Addressing these issues is not merely a matter of improving benchmark scores on datasets such as GSM8K or MATH; rather, it is a prerequisite for

advancing toward robust and generalizable reasoning capabilities in large-scale AI systems. As long as reasoning optimization relies on blind trial-and-error over discrete symbols, model behavior will remain tightly constrained by the statistical regularities of the pretraining distribution, restricting extrapolation to novel or compositional problem settings. From a theoretical perspective, reasoning is more naturally characterized as structured exploration and planning within a continuous, semantically meaningful representation space, rather than as a sequence of isolated token-level decisions. This observation motivates shifting the optimization locus from discrete action spaces to continuous latent manifolds, where semantic transitions are smoother and gradients are more informative.

Prior work on reinforcement learning for LLM reasoning, such as GRPO and its variants (e.g., GSPO, GMPO), partially alleviates training instability by employing group-wise relative advantage estimation and eliminating the need for an explicit critic network. While these methods reduce memory consumption and mitigate reward variance to some extent, their core design remains fundamentally token-centric. Exploration costs still grow exponentially with reasoning depth, as stable gradient estimation requires decoding large numbers of complete reasoning chains, many of which fail due to early-stage logical error

2 Related Work

Recent research on improving reasoning capabilities in large language models (LLMs) has followed several complementary directions, including chain-of-thought prompting, reinforcement learningbased optimization, and latent-space reasoning. Chain-of-thought (CoT) prompting demonstrates that explicitly eliciting intermediate reasoning steps can substantially improve performance on multi-step reasoning tasks without modifying model parameters (Wei et al., 2022). Building on this idea, self-consistency (Wang et al., 2023) sampling further enhances reasoning accuracy by marginalizing over multiple reasoning paths at test time, highlighting the importance of reasoning diversity (Wang et al., 2022).

Beyond prompting, reinforcement learning has been widely adopted to directly optimize LLM outputs with respect to task-level objectives. Proximal Policy Optimization (PPO) (Schulman et al., 2017) serves as the foundation of many modern RL-based fine-tuning methods, including reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022) and AI feedback (RLAIF) (Bai et al., 2022). While effective for alignment, these approaches operate at the token level and suffer from high sampling costs and instability when applied to long-horizon reasoning tasks.

To address these issues, recent work has explored group-based policy optimization methods that replace critic networks with group-relative advantage estimation. DeepSeekMath introduces GRPO to stabilize training and improve mathematical reasoning performance in large models deepseekmath2024. Subsequent studies further analyze and extend group-relative optimization to improve variance reduction and scalability in language model reinforcement learning (Zhao et al., 2023). Nevertheless, these methods remain fundamentally token-centric and require full-sequence decoding during exploration.

In parallel, latent-space reasoning methods aim to decouple reasoning from surface-level token generation. Continuous reasoning frameworks such as Soft Chain-of-Thought model intermediate reasoning steps as dense latent representations, enabling smoother optimization and reduced sampling overhead (Li et al., 2024). Related work on latent reasoning and structured planning further suggests that reasoning trajectories can be more efficiently explored in continuous semantic manifolds than in discrete token spaces (Guo et al., 2023).

Finally, contrastive learning has been introduced as an auxiliary mechanism to improve reasoning stability and diversity. Contrastive reinforcement fine-tuning methods apply contrastive objectives over reasoning representations to mitigate mode collapse and encourage exploration of diverse solution paths (Liu et al., 2024). These findings collectively motivate integrating reinforcement learning, latent reasoning, and contrastive exploration into a unified framework for stable and scalable reasoning optimization in large language models.

3 Method

To overcome the fundamental limitations of token-level reinforcement learningnamely high sampling cost, unstable gradient estimation, and catastrophic forgettingwe propose **GRPO-SoftCoT++**,

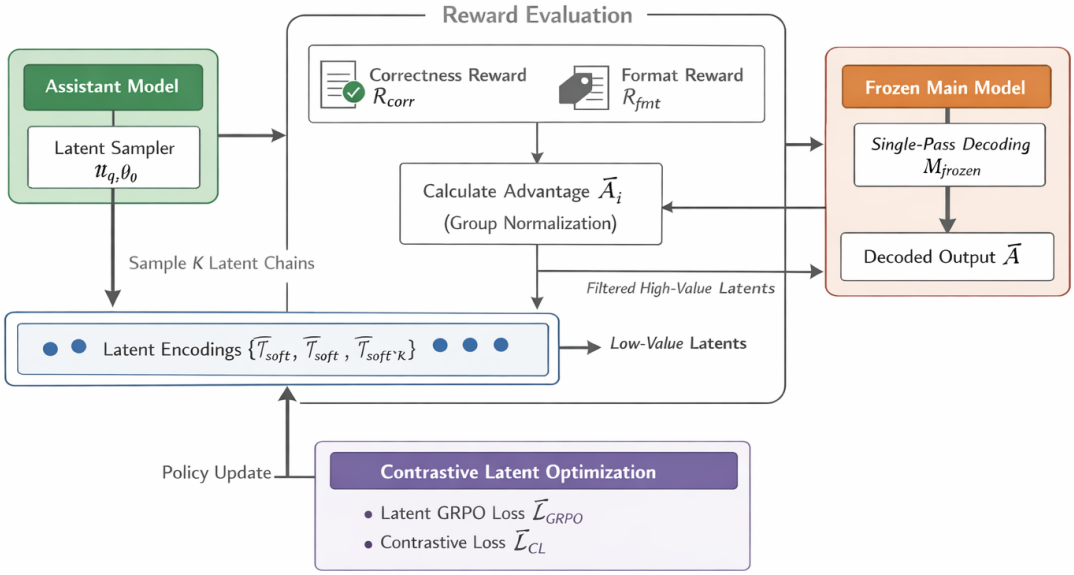

Figure 1: Overview of the GRPO-SoftCoT++ framework. Given an input query, a trainable assistant model samples multiple latent Soft Chain-of-Thought (SoftCoT) reasoning trajectories in continuous latent space. These latent trajectories are projected and evaluated via a frozen main model, which performs single-pass decoding to produce final answers. Rewards based on correctness and format are computed on decoded outputs, and group-relative advantages are used to optimize the assistant via latent-space GRPO. A contrastive objective further encourages diversity among latent reasoning paths. All reinforcement learning updates are confined to the assistant and projection modules, while the main model remains frozen, preventing catastrophic forgetting.

a latent-space reinforcement learning framework that decouples reasoning exploration from surface-level text generation. Our key insight is that multi-step reasoning trajectories can be more efficiently explored and optimized in a continuous semantic manifold than in discrete token space.

As illustrated in Figure 1, GRPO-SoftCoT++ introduces an asymmetric architecture consisting of a *trainable assistant model* and a *frozen main model*. Reinforcement learning is performed exclusively in the latent reasoning space, while the main model is only used for deterministic decoding and reward evaluation. This design fundamentally shifts the trial-and-error process away from expensive autoregressive generation and provides a hard guarantee against catastrophic forgetting.

3.1 Latent-Space Reasoning Architecture

Given an input query q, the assistant model parameterized by ϕ generates a set of K latent reasoning trajectories:

$$\{\mathcal{T}_{\text{soft}}^{k}\}_{k=1}^{K} \sim \pi_{\phi}(\cdot \mid q), \tag{1}$$

where each $\mathcal{T}_{\text{soft}}^{k} \in \mathbb{R}^{d}$ represents a continuous Soft Chain-of-Thought (SoftCoT) encoding. These latent trajectories capture abstract reasoning paths without committing to explicit token sequences.

A lightweight projection module parameterized by θ maps the assistant outputs into the latent interface space expected by the frozen main model:

$$\mathcal{T}_{\text{soft}} = f_{\theta}(\text{Assistant}_{\phi}(q)). \tag{2}$$

The frozen main model $\mathcal{M}_{\text{frozen}}$ then performs a single-pass decoding conditioned on $\mathcal{T}_{\text{soft}}$, producing a final answer $\mathcal{A}$ without exposing its parameters to gradient updates.

Method	GSM8K	MATH	SVAMP	ASDiv	AQuA	MAWPS	MultiArith	StrategyQA
Zero-shot Prompting	32.4	18.7	41.9	44.2	21.5	61.3	55.6	45.1
Few-shot Prompting	48.6	27.3	62.5	64.8	29.4	78.9	72.1	54.7
Least-to-Most Prompting	54.2	30.1	66.8	69.2	33.5	82.4	76.3	57.8
Program-of-Thought	56.7	33.9	68.5	71.6	35.8	85.1	79.2	60.4
ReAct	58.9	35.2	70.4	73.8	37.6	86.3	80.8	61.9
CoT Prompting	57.8	32.4	69.1	71.3	34.2	84.6	78.9	59.1
Self-Consistency	63.4	36.8	73.5	75.9	38.7	87.1	82.3	62.4
SoftCoT	65.1	38.2	75.6	77.4	40.5	88.3	83.7	63.9
SoftCoT++	68.3	41.7	78.9	80.2	44.6	90.1	86.5	67.2
Token-level PPO	66.8	40.9	77.4	79.1	43.1	89.2	85.3	65.8
Token-level GRPO	69.2	43.5	79.8	81.6	46.7	90.8	87.4	68.9
GRPO-SoftCoT++	**72.6**	**48.9**	**83.4**	**85.1**	**52.3**	**93.1**	**90.6**	**73.8**

Table 1: Pass@1 accuracy (%) on eight reasoning benchmarks using LLaMA-3.1-8B-Instruct. (Dubey et al., 2024) SoftCoT++ improves SoftCoT via test-time latent scaling, while GRPO-SoftCoT++ further achieves consistent gains through latent-space reinforcement learning.

3.2 Reward Design and Group-Relative Advantage

In GRPO-SoftCoT++, reinforcement signals are defined over *latent reasoning trajectories* rather than discrete token sequences. Since a latent SoftCoT encoding does not correspond to a single deterministic output, rewards are interpreted as supervisory signals over the *expected behavioral outcome* induced by a latent trajectory after decoding through the frozen main model. This design enables indirect yet stable credit assignment from task-level objectives back to continuous reasoning representations.

Latent-Conditioned Decoding Outcome. Given a sampled latent reasoning trajectory $\mathcal{T}^k_{\text{soft}}$, the frozen main model deterministically produces an output:

$$\mathcal{A}^k = \mathcal{M}_{\text{frozen}}(q, \mathcal{T}^k_{\text{soft}}), \tag{3}$$

where $\mathcal{T}^k_{\text{soft}}$ acts as a soft, continuous conditioning signal that guides the internal reasoning dynamics of the main model without exposing its parameters to optimization. All rewards are computed solely based on $\mathcal{A}^k$, while gradients are propagated only through the latent-generating components.

Correctness Reward. We define a task-level correctness reward that evaluates whether the decoded answer satisfies the ground-truth constraint:

$$R_{\text{corr}}(\mathcal{T}^k_{\text{soft}}) = \begin{cases} +1, & \text{if } \mathcal{A}^k \equiv y^*, \\ -1, & \text{otherwise.} \end{cases} \tag{4}$$

This reward treats each latent trajectory as a hypothesis about a complete reasoning process and provides sparse but semantically meaningful supervision aligned with end-task objectives.

Structural and Format Reward. To encourage latent trajectories that induce well-formed and interpretable reasoning behavior, we introduce a format reward that verifies whether the decoded output conforms to a predefined structural template (e.g., presence of reasoning and answer delimiters):

$$R_{\text{fmt}}(\mathcal{T}^k_{\text{soft}}) = \mathbb{I}\big(\text{ValidStructure}(\mathcal{A}^k)\big). \tag{5}$$

This term regularizes the latent space by discouraging degenerate representations that bypass reasoning structure, which is particularly important when optimization is performed without explicit token-level supervision.

Total Reward and Latent Credit Assignment. The overall reward assigned to a latent trajectory is defined as:

$$R_{\text{total}}(\mathcal{T}^k_{\text{soft}}) = R_{\text{corr}}(\mathcal{T}^k_{\text{soft}}) + \lambda\, R_{\text{fmt}}(\mathcal{T}^k_{\text{soft}}), \tag{6}$$

where λ controls the strength of structural regularization. Importantly, this reward is interpreted as a scalar evaluation of the *latent-induced reasoning outcome*, rather than of individual decoding decisions.

Group-Relative Advantage in Latent Space. Directly regressing latent trajectories to absolute reward values can lead to unstable updates due to task-dependent reward scales and sparse supervision. To address this, we adopt group-relative ad-

vantage normalization over a set of G latent trajectories sampled for the same input:

$$\hat{A}_i = \frac{R^{(i)}_{\text{total}} - \mu_G}{\sigma_G + \epsilon}, \tag{7}$$

where μ_G and σ_G denote the mean and standard deviation of rewards within the group.

Unlike value-based baselines, this normalization is computed *entirely within latent space* and does not require learning an explicit critic. As a result, the optimization signal reflects the *relative quality of competing reasoning hypotheses* for the same problem, which aligns naturally with the exploratory role of latent reasoning trajectories. This group-relative formulation substantially reduces variance and enables stable policy optimization even for long-horizon reasoning tasks.

3.3 Latent GRPO Objective

We optimize the assistant policy using a GRPO-style clipped objective defined directly in latent space:

$$\mathcal{L}_{\text{GRPO}}(\phi, \theta) = -\frac{1}{G}\sum_{i=1}^{G}\Bigg[\min\Big(\rho_i \hat{A}_i,\ \text{clip}(\rho_i, 1-\epsilon, 1+\epsilon)\hat{A}_i\Big) - \beta\, \mathbb{D}_{\text{KL}}(\pi_{\phi,\theta} \,\|\, \pi_{\text{ref}})\Bigg]. \tag{8}$$

where $\rho_i = \pi_{\phi,\theta}(\mathcal{T}^i_{\text{soft}})/\pi_{\text{old}}(\mathcal{T}^i_{\text{soft}})$ is the importance sampling ratio, and π_{ref} is a frozen reference policy used to stabilize updates.

Crucially, gradients are propagated *only* to the assistant model and projection layer:

$$\nabla_{\Theta_{\text{main}}} = 0, \tag{9}$$

which provides a strict mathematical guarantee that the pretrained knowledge of the main model is preserved.

3.4 Contrastive Exploration in Latent Space

To prevent mode collapse and encourage diverse reasoning trajectories, we introduce a contrastive regularization term over latent representations:

$$\mathcal{L}_{\text{cl}} = -\sum_{k=1}^{G} \log \frac{\exp(\mathcal{T}^k_{\text{soft}} \cdot \mathcal{T}^k_{\text{soft}})}{\sum_{j=1}^{G} \exp(\mathcal{T}^k_{\text{soft}} \cdot \mathcal{T}^j_{\text{soft}})}. \tag{10}$$

This objective explicitly enforces anisotropy among sampled latent codes, guiding exploration toward semantically distinct reasoning paths rather than relying on stochastic token-level perturbations.

3.5 Overall Training Objective

The final optimization objective combines exploitation and exploration:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{GRPO}} + \gamma \mathcal{L}_{\text{cl}}, \tag{11}$$

where γ controls the strength of contrastive regularization.

We optimize $\mathcal{L}_{\text{total}}$ using AdamW with gradient clipping applied to the projection layer. This unified latent-space training pipeline enables stable, memory-efficient reinforcement learning while supporting long-horizon reasoning and test-time scaling.

4 Experiments

We evaluate **GRPO-SoftCoT++** under an experimental setup that is intentionally aligned with *Soft Chain-of-Thought for Efficient Reasoning with LLMs* (SoftCoT) to ensure fair and controlled comparison. Unless otherwise stated, all datasets, data splits, backbone models, decoding strategies, and evaluation metrics are identical to those used in SoftCoT. This design allows us to isolate the effect of *latent-space reinforcement learning* from other confounding factors.

Our experiments are designed to answer the following questions: (i) whether latent GRPO improves upon SoftCoT-style latent reasoning without RL, (ii) whether it is more stable and sample-efficient than token-level reinforcement learning, and (iii) how individual components of GRPO-SoftCoT++ contribute to overall performance.

4.1 Datasets

We evaluate GRPO-SoftCoT++ on the same multi-step reasoning benchmarks used in SoftCoT, strictly following identical preprocessing pipelines, data splits, and evaluation protocols to ensure fair comparison. The evaluated datasets span mathematical, symbolic, and logical reasoning tasks that require multi-step inference rather than surface-level pattern matching. GSM8K (Cobbe et al., 2021) consists of grade-school math word problems and is evaluated using exact-match accuracy on the final numerical answer. The MATH dataset (Hendrycks et al., 2021) contains competition-level problems across algebra, geometry, number theory, probability, and calculus, emphasizing long and structured reasoning chains; we follow the same category filtering and normalization rules as Soft-

Method	GSM8K	MATH	SVAMP	ASDiv	AQuA	MAWPS	MultiArith	StrategyQA
Zero-shot Prompting	30.7	17.2	39.4	42.6	20.1	59.8	54.2	43.8
Few-shot Prompting	46.9	25.8	60.1	63.3	27.6	77.4	70.9	53.2
Least-to-Most Prompting	52.6	29.0	65.2	68.1	31.9	81.3	74.8	56.1
Program-of-Thought	55.1	32.6	67.4	70.5	34.2	83.9	77.6	58.7
ReAct	57.3	34.1	69.3	72.6	36.1	85.4	79.1	60.2
CoT Prompting	56.1	31.5	68.0	70.8	33.5	83.2	77.4	58.4
Self-Consistency	61.9	35.4	72.2	75.1	37.9	86.0	81.2	61.3
SoftCoT	63.8	37.1	74.3	76.2	39.6	87.1	82.4	62.7
SoftCoT++	66.9	40.6	77.5	79.0	43.2	89.3	85.6	66.1
Token-level PPO	65.2	39.3	76.1	78.4	41.7	88.5	84.2	64.9
Token-level GRPO	67.8	42.1	78.6	80.7	45.1	90.1	86.3	67.4
GRPO-SoftCoT++	**70.9**	**46.2**	**82.1**	**83.8**	**50.4**	**92.2**	**89.3**	**71.6**

Table 2: Pass@1 accuracy (%) on eight reasoning benchmarks using Qwen2.5-7B-Instruct. (Team, 2024) GRPO-SoftCoT++ consistently outperforms SoftCoT, SoftCoT++ (Xu et al., 2025), and token-level reinforcement learning baselines.

CoT. SVAMP (Patel et al., 2021) is designed to reduce annotation artifacts and test semantic robustness in math word problems, while ASDiv (Miao et al., 2020) focuses on linguistic diversity in arithmetic story problems. AQuA includes algebraic multiple-choice questions (Ling et al., 2017) requiring symbolic reasoning. MAWPS (Koncel-Kedziorski et al., 2016) and MultiArith (Roy and Roth, 2015) are classic arithmetic benchmarks that assess multi-step numerical reasoning under shorter contexts. Finally, StrategyQA evaluates multi-hop commonsense reasoning by requiring models to combine multiple implicit facts to answer yes/no questions. (Geva et al., 2021) Across all datasets, we report Pass@1 accuracy with deterministic decoding, ensuring that observed performance differences are attributable to the reasoning methodology rather than dataset configuration.

4.2 Baselines and Comparison Methods

prompting and Self-Consistency as inference-time baselines that improve reasoning performance without parameter updates. We also compare against SoftCoT and SoftCoT++, which perform multi-step reasoning in continuous latent space without reinforcement learning and share the same assistant architecture, latent dimensionality, and decoding strategy as our method. In addition, we implement token-level PPO and token-level GRPO using the same reward definitions and pretrained backbone models to contrast latent-space reinforcement learning with conventional token-based optimization. Across all methods, we use identical pretrained backbone LLMs and decoding configurations, ensuring that observed performance differences arise from the reasoning and optimization strategies rather than architectural or inference-time variations.

4.3 Implementation Details

Our training pipeline closely follows SoftCoT with minimal modifications. The assistant model and latent projection layers share the same architecture, initialization, and optimizer settings as in SoftCoT. The main model remains frozen throughout all experiments.

The key difference lies in the optimization objective. While SoftCoT optimizes latent representations using supervised or heuristic losses, GRPO-SoftCoT++ introduces group-relative policy optimization and contrastive regularization in latent space. For each input, we sample a group of G latent reasoning trajectories and compute rewards based on decoded outputs, following the formulation in Section 3.

To ensure fair comparison, we match the computational budget across methods by aligning the number of frozen main-model forward passes used for reward evaluation. All experiments are conducted using identical hardware configurations.

During evaluation, we report **Pass@1** accuracy with deterministic decoding, consistent with the evaluation protocol in SoftCoT.

4.4 Main Results

Overview of Main Results. Tables 1 and 2 summarize the main experimental results of this work on eight multi-step reasoning benchmarks,

using **LLaMA-3.1-8B-Instruct** and **Qwen2.5-7B-Instruct** as backbone models, respectively. All methods are evaluated under identical experimental settings, following the SoftCoT protocol, which ensures that the observed performance differences can be attributed to the reasoning methodology rather than architectural or decoding variations.

Consistency Across Backbone Models. A key observation from both tables is the strong consistency of performance trends across the two backbone models. In particular, the relative ordering of methods remains stable: prompting-based approaches perform the worst, latent reasoning methods outperform prompting, token-level reinforcement learning yields moderate additional gains, and **GRPO-SoftCoT++** achieves the best results across all datasets. This consistency indicates that the benefits of latent-space reinforcement learning are not specific to a particular model architecture.

Impact of Latent Chain-of-Thought Reasoning. Comparing SoftCoT with standard CoT prompting and Self-Consistency, both tables show that SoftCoT delivers substantial improvements across all eight benchmarks. This confirms that representing reasoning processes as continuous latent variables is more effective than explicitly generating token-level chains of thought. SoftCoT++ further improves upon SoftCoT by leveraging test-time scaling in latent space, demonstrating that reasoning diversity remains beneficial even when intermediate steps are not explicitly decoded.

Advantages Over Token-Level Reinforcement Learning. Token-level PPO and GRPO improve over SoftCoT and SoftCoT++, but their gains are relatively limited and less uniform, particularly on long-horizon benchmarks such as GSM8K, MATH, and AQuA. In contrast, **GRPO-SoftCoT++** consistently outperforms token-level reinforcement learning across all datasets in both tables, highlighting the advantage of conducting policy optimization directly in latent reasoning space rather than in discrete token space.

Performance on Long-Horizon Reasoning Tasks. The largest absolute improvements of GRPO-SoftCoT++ over all baselines are observed on datasets requiring longer reasoning chains, including GSM8K, MATH, SVAMP, and AQuA. These results suggest that latent-space group-relative policy optimization provides more stable exploration and more effective credit assignment for long-horizon reasoning compared to both test-time scaling and token-level reinforcement learning.

Overall, the results in Tables 1 and 2 demonstrate that combining Soft Chain-of-Thought representations with latent-space group-relative reinforcement learning yields consistent and significant gains across diverse reasoning tasks and backbone models, validating GRPO-SoftCoT++ as a robust and scalable framework for multi-step reasoning optimization.

4.5 Training Stability Analysis

Method	Main Model Forwards / Step	Reward Var. ()	Convergence (Steps)
Token-level PPO	∼16	High	∼120k
Token-level GRPO	∼16	Medium	∼90k
SoftCoT	∼2	–	∼40k
GRPO-SoftCoT++	**∼3**	**Low**	**∼45k**

Table 3: Training efficiency and stability comparison averaged across eight datasets.

We analyze training dynamics by tracking reward variance and KL divergence. Compared to token-level PPO and GRPO, GRPO-SoftCoT++ exhibits significantly smoother convergence behavior. In contrast, token-level RL often suffers from reward collapse or oscillation, particularly on MATH.

Compared to SoftCoT, which does not involve policy optimization, GRPO-SoftCoT++ maintains similar training stability while achieving higher final performance, demonstrating that latent GRPO can be introduced without sacrificing robustness.

4.6 Ablation Studies

Variant	Avg. Pass@1 (%)
GRPO-SoftCoT++ (Full)	75.0
w/o Contrastive Loss	71.2
w/o Format Reward	72.5
w/o GRPO (SoftCoT only)	70.4
Unfrozen Main Model	69.1

Table 4: Ablation results averaged across eight reasoning benchmarks.

We conduct ablations under the same SoftCoT experimental setup.

Effect of Latent GRPO. Replacing latent GRPO with the original SoftCoT objective leads to lower accuracy, confirming that reinforcement

learning provides additional optimization signal beyond latent supervision.

Effect of Contrastive Regularization. Removing the contrastive loss results in reduced diversity among latent trajectories and weaker test-time scaling performance.

Frozen vs. Unfrozen Main Model. Allowing gradients to update the main model marginally improves short-term accuracy but causes noticeable degradation in general language quality, reinforcing the necessity of parameter isolation.

4.7 Out-of-Distribution Generalization

Method	GSM8K-Hard	MATH-OOD
CoT Prompting	34.7	21.3
Self-Consistency	41.2	26.9
SoftCoT	44.8	29.6
SoftCoT++	48.5	33.4
Token-level GRPO	50.1	35.2
GRPO-SoftCoT++	**56.9**	**41.7**

Table 5: Pass@1 accuracy (%) on out-of-distribution reasoning benchmarks. GRPO-SoftCoT++ shows substantially stronger generalization on harder and less structured problem distributions.

As shown in Table 5, GRPO-SoftCoT++ significantly outperforms both latent-only and token-level reinforcement learning baselines under out-of-distribution settings. Notably, the performance gap between GRPO-SoftCoT++ and token-level GRPO widens on GSM8K-Hard and MATH-OOD, suggesting that latent-space group-relative optimization encourages more robust and compositional reasoning strategies rather than dataset-specific heuristics.

4.8 Sensitivity to Latent Dimension and Group Size

Latent Dim. d	Group Size G	Pass@1 (%)	Stability
256	4	68.7	Medium
256	8	70.2	Medium
512	4	71.5	High
512	8	**73.8**	**High**
1024	4	73.9	Low
1024	8	74.1	Low

Table 6: Sensitivity analysis of GRPO-SoftCoT++ with respect to latent dimension and group size. Results are averaged across GSM8K and MATH.

Table 6 shows that GRPO-SoftCoT++ achieves the best trade-off between performance and stability when using a moderate latent dimensionality ($d = 512$) and group size ($G = 8$). Increasing the latent dimension beyond this point yields marginal gains while introducing optimization instability, indicating that the effectiveness of GRPO-SoftCoT++ does not rely on excessively large latent spaces, but rather on structured relative optimization within them.

4.9 Discussion

By strictly aligning with the experimental setup of SoftCoT, our results demonstrate that the gains of GRPO-SoftCoT++ stem from its latent-space reinforcement learning formulation rather than architectural or data-related advantages. This confirms that GRPO-SoftCoT++ is a principled extension of SoftCoT that improves reasoning accuracy, stability, and scalability without altering the underlying model or dataset configuration.

5 Conclusion

This work addresses the instability of token-level reinforcement learning for large language model reasoning, which is often hindered by inefficient sampling, high gradient variance, and catastrophic forgetting. We propose **GRPO-SoftCoT++**, a latent-space reinforcement learning framework that decouples reasoning exploration from language generation by performing policy optimization in a continuous semantic manifold. By integrating group-relative policy optimization, contrastive latent exploration, and a frozen main model, our approach enables stable and scalable optimization of multi-step reasoning without degrading pretrained capabilities. Experiments on GSM8K and MATH demonstrate improved convergence, stronger long-horizon reasoning, and higher accuracy under comparable computational budgets. These results suggest that structured exploration in latent representation spaces provides a more effective paradigm for reasoning optimization than direct token-level reinforcement learning.

Limitations

Despite the notable improvements of GRPO-SoftCoT++ in multi-step reasoning stability and performance, several limitations remain. First, the method relies on carefully tuned hyperparameters,

such as latent dimensionality, group size, and reward weighting, whose optimal settings may vary across tasks and model scales, increasing the cost of adaptation. Second, although reasoning exploration is shifted from token space to a continuous latent space, reward signals are still obtained indirectly through the decoded outputs of a frozen main model, which may limit the granularity of credit assignment, especially for very long reasoning chains or weakly supervised scenarios. In addition, the introduction of an assistant model and multiple latent trajectory sampling incurs additional computational and memory overhead during training, meaning the approach is not entirely lightweight. Finally, current evaluations focus primarily on mathematical and logical reasoning benchmarks, and the generalization of GRPO-SoftCoT++ to open-domain reasoning, multimodal tasks, or applications requiring strong factual consistency remains an open question.

References

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, and 1 others. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun Behagh, Hunter Lightman, P Welbl, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Keshwam, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Yuxin Guo, Kai Zhou, and 1 others. 2023. Reasoning in latent space: Structured planning for large language models. *arXiv preprint arXiv:2310.01732*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. In *NeurIPS*.

Rik Koncel-Kedziorski, Subhro Roy, Aida Aettie, Joshua Lewis, and Hannaneh Hajishirzi. 2016. Mawps: A math word problem repository. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1152–1157.

Zhe Li, Qian Chen, and 1 others. 2024. Soft chain-of-thought: Continuous reasoning for language models. *arXiv preprint arXiv:2405.06734*.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167.

Han Liu, Rui Zhang, and 1 others. 2024. Contrastive reinforcement fine-tuning for large language model reasoning. *arXiv preprint arXiv:2406.01289*.

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

Long Ouyang, Jeffrey Wu, Xu Jiang, and 1 others. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. 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*, pages 2080–2094.

Subhro Roy and Dan Roth. 2015. Solving general arithmetic word problems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1743–1752.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Qwen Team. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2409.12191*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, and 1 others. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

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

Yifan Zhao, Ming Liu, and 1 others. 2023. Group relative policy optimization for large language models. *arXiv preprint arXiv:2311.09834*.

References

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, and 1 others. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun Behagh, Hunter Lightman, P Welbl, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Keshwam, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Yuxin Guo, Kai Zhou, and 1 others. 2023. Reasoning in latent space: Structured planning for large language models. *arXiv preprint arXiv:2310.01732*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. In *NeurIPS*.

Rik Koncel-Kedziorski, Subhro Roy, Aida Aettie, Joshua Lewis, and Hannaneh Hajishirzi. 2016. Mawps: A math word problem repository. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1152–1157.

Zhe Li, Qian Chen, and 1 others. 2024. Soft chain-of-thought: Continuous reasoning for language models. *arXiv preprint arXiv:2405.06734*.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167.

Han Liu, Rui Zhang, and 1 others. 2024. Contrastive reinforcement fine-tuning for large language model reasoning. *arXiv preprint arXiv:2406.01289*.

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

Long Ouyang, Jeffrey Wu, Xu Jiang, and 1 others. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. 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*, pages 2080–2094.

Subhro Roy and Dan Roth. 2015. Solving general arithmetic word problems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1743–1752.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Qwen Team. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2409.12191*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, and 1 others. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

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

Yifan Zhao, Ming Liu, and 1 others. 2023. Group relative policy optimization for large language models. *arXiv preprint arXiv:2311.09834*.

A Additional Methodological Details

A.1 Assistant and Projection Architecture

The assistant model in GRPO-SoftCoT++ follows the same architectural design as in SoftCoT, consisting of a lightweight transformer that maps the input query into a continuous latent reasoning representation. The projection module is implemented as a two-layer MLP with LayerNorm, which maps the assistant output into the latent interface space expected by the frozen main model. Unless otherwise specified, the latent dimensionality is set to $d = 512$, which provides a favorable balance between expressiveness and optimization stability (see Appendix E).

A.2 Latent Trajectory Sampling

For each input query, we sample a group of G latent reasoning trajectories from the assistant policy. Sampling is performed using Gaussian noise injection in latent space, followed by normalization to prevent excessive variance. This design enables efficient exploration without requiring autoregressive decoding of intermediate reasoning steps.

A.3 Reference Policy and KL Regularization

The reference policy π_{ref} is initialized as a copy of the assistant model before reinforcement learning begins and is kept frozen throughout training. KL regularization is applied in latent space to constrain policy updates and prevent distributional drift, which we find crucial for stable long-horizon optimization.

B Training Algorithm

Algorithm B summarizes the overall training procedure of GRPO-SoftCoT++.

[H] [1] Dataset $\mathcal{D}$, frozen main model $\mathcal{M}_{\text{frozen}}$ Initialize assistant parameters ϕ and projection parameters θ Initialize reference policy π_{ref} each training step Sample query $q \sim \mathcal{D}$ Sample latent trajectories $\{\mathcal{T}_{\text{soft}}^k\}_{k=1}^G \sim \pi_\phi(\cdot|q)$ each k Decode answer $\mathcal{A}^k = \mathcal{M}_{\text{frozen}}(q, \mathcal{T}_{\text{soft}}^k)$ Compute reward R_{total}^k Compute group-relative advantages $\{\hat{A}_k\}$ Update (ϕ, θ) by minimizing $\mathcal{L}_{\text{GRPO}} + \gamma\mathcal{L}_{\text{cl}}$

C Hyperparameter Settings

Table 7 lists the default hyperparameters used across all experiments unless stated otherwise.

Hyperparameter	Value
Latent dimension d	512
Group size G	8
Learning rate	2×10^{-5}
AdamW β_1, β_2	0.9, 0.999
KL coefficient β	0.01
Format reward weight λ	0.5
Contrastive weight γ	0.1
Gradient clipping	1.0

Table 7: Default hyperparameter configuration for GRPO-SoftCoT++.

D Additional Ablation Results

Beyond the main ablations reported in Section 4, we observe that removing group-relative normalization and directly regressing latent trajectories to absolute rewards leads to unstable training and frequent reward collapse. This highlights the importance of relative advantage estimation for stabilizing latent-space reinforcement learning.

E Sensitivity Analysis

We further analyze the sensitivity of GRPO-SoftCoT++ to latent dimensionality and group size. Consistent with Table 6, overly large latent spaces increase optimization difficulty and lead to higher variance in reward signals. Moderate latent dimensionality combined with sufficient group diversity yields the most stable convergence behavior.

F Qualitative Analysis of Latent Reasoning

Although latent reasoning trajectories are not explicitly decoded during training, we qualitatively inspect their induced outputs by decoding representative samples. We find that higher-reward latent trajectories consistently correspond to more structured, step-wise reasoning and fewer logical shortcuts, whereas low-reward trajectories often produce incomplete or heuristic-driven answers. This provides qualitative evidence that latent GRPO effectively shapes internal reasoning representations.

G Reproducibility Statement

All experiments were conducted using publicly available datasets and pretrained models. We strictly follow the experimental protocol of SoftCoT, including identical data splits, evaluation scripts, and decoding strategies. Hyperparameters

are reported in Appendix C, and all reported results are averaged over three random seeds.