Token-Level Policy Optimization: Linking Group-Level Rewards to Token-Level Aggregation via Sequence-Level Likelihood

Xingyu Lin[1,2,3], **Yilin Wen**[3], **Du Su**[5], **Jinchang Hou**[3],
En Wang[1,2] *, **Wenbin Liu**[1,2] *, **Chenfu Bao**[3,4] *, **Zhonghou Lv**[3] *
[1]College of Computer Science and Technology, Jilin University
[2]Key Laboratory of Symbolic Computation and Knowledge Engineering of MOE,
Jilin University
[3]Baidu Inc. [4]Tsinghua University
[5]State Key Laboratory of AI Safety, Institute of Computing Technology,
Chinese Academy of Sciences

Abstract

Group Relative Policy Optimization (GRPO) has significantly advanced the reasoning ability of large language models (LLMs), particularly in their mathematical reasoning performance. However, GRPO and related entropy regularization methods still struggle with token-level sparse-rewards, which is an inherent challenge in chain-of-thought (CoT) reasoning. These approaches often rely on undifferentiated token-level entropy regularization, which easily leads to entropy collapse or model degradation under sparse token rewards. In this work, we propose `TEPO`, a novel token-level framework that (1) leverages sequence-level likelihood to link group-level rewards with individual tokens via token-level aggregation, and (2) introduces a token-level KL-Divergence mask constraint that targets tokens with positive advantages and decreasing entropy to mitigate abrupt policy updates. Experiments demonstrate that `TEPO` not only achieves state-of-the-art performance on mathematical reasoning benchmarks but also markedly enhances training stability, reducing convergence time by 50% compared with GRPO/DAPO.

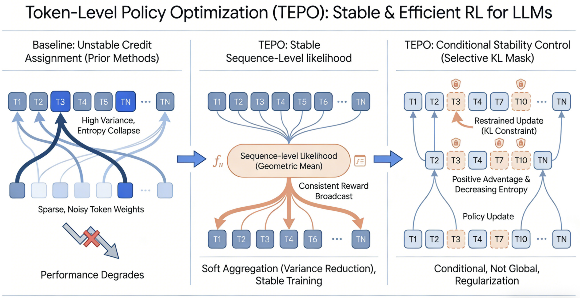


Figure 1: Overview of the `TEPO` Framework `TEPO` (1) replaces baselines' noisy, sparse token-level credit assignment with sequence-level likelihood, using soft aggregation to broadcast group rewards to tokens and stabilize training. (2) A selective KL mask curbs abrupt updates exclusively for tokens with positive advantage and decreasing entropy, balancing entropy reduction and stability.

1 Introduction

GRPO(Shao et al., 2024) has significantly advanced the reasoning ability of large language models (LLMs), particularly in mathematical reasoning. However, in CoT reasoning, learning is fundamentally challenged by sparse token-level rewards, under which GRPO and related entropy-regularized methods often struggle. Specifically, these approaches rely on undifferentiated token-level entropy regularization, which can lead to entropy collapse or policy degradation when rewards are sparse(Yu et al., 2025). For entropy regularization, methods either minimize entropy to ensure credible outputs (Agarwal et al., 2025) or maximize it to enhance exploration; however, such undifferentiated entropy adjustments yield only marginal improvements. For KL-Divergence, this regularization strategy is fragile and degrades final performance without extensive parameter tuning, manifesting as entropy collapse or model collapse (Zheng et al., 2025a).

Core Insight: These challenges are rooted in the inherent sparse-reward of CoT reasoning, which creates instability that GRPO's critic-free framework struggles to mitigate. The absence of a critic in GRPO (Shao et al., 2024), combined with the token-level sparse-reward nature of long-chain reasoning tasks, exacerbates its susceptibility to high-variance gradient estimates. Policies exploring novel CoT structures often diverge substantially from their initial distribution (Cheng et al., 2025), leading to cumulative noise across extended reasoning sequences (Zheng et al., 2025b). Furthermore, sole reliance on undifferentiated entropy regularization or KL-Divergence exacerbates this issue by triggering model collapse in sparse-reward CoT settings (Chu et al., 2025; Gao et al., 2025; Zheng

---

[*] These authors contributed equally to this work.
Corresponding authors: En Wang (enwang@jlu.edu.cn), Wenbin Liu (liuwenbin@jlu.edu.cn), Chenfu Bao (baochenfu@baidu.com, bcf25@mails.tsinghua.edu.cn), Zhonghou Lv (lvzhonghou@baidu.com)

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 32256–32269
July 2-7, 2026 ©2026 Association for Computational Linguistics

et al., 2025b), further undermining training stability.

To address this token-level sparse-reward, we propose `TEPO`, a token-level framework designed to align group-level rewards with token-level credit assignment. `TEPO` (1) leverages sequence-level likelihood to bridge group-level rewards with individual tokens via token-level aggregation, and (2) introduces a token-level KL-Divergence mask (applied to tokens with positive advantage and decreasing entropy) to mitigate abrupt policy updates. Notably, (Cui et al., 2025) establishes a well-documented relationship between model performance (R) and policy entropy ($\mathcal{H}$): $R = -a \cdot \exp(\mathcal{H}) + b$, revealing that performance improvements are fundamentally achieved through systematic entropy reduction. By resolving state distribution shift with sequence-level likelihood(Section 3.3.1), `TEPO` enables trading larger $\Delta\mathcal{H}$ for better downstream performance. Additionally, the token-level KL-Divergence mask mitigates rapid entropy decay. Our key contributions are summarized as follows:

- **Token-Level Policy Optimization**: We propose a highly efficient and adaptive token-level optimization strategy tailored to critic-free paradigms. Notably, our method achieves peak performance in only 72 optimization steps, while GRPO/DAPO require 132 steps to reach comparable performance, reducing convergence time by nearly 50%.
- **Analysis of KL-Divergence and Entropy Regularization in GRPO**: We provide both theoretical and empirical evidence demonstrating that undifferentiated KL-Divergence and entropy regularization struggle in sparse-reward settings.
- **Comprehensive Experimental Validation**: Our method achieves an approximately 2% improvement in average accuracy over the baseline GRPO. We further conduct ablation studies to verify that GRPO/DAPO and related entropy regularization methods exhibit performance bottlenecks under token-level sparse-reward conditions, and the results confirm the effectiveness of `TEPO`.

2 Preliminaries in LLMs

We list all core notations in Table 5.

2.1 Entropy Gradient Derivation

In LLMs, the state s corresponds to the prompt context, and an action a refers to a token from the vocabulary $\mathcal{A}$. For a state s and action a, the policy is defined as follows:

$$\pi_\theta(a \mid s) = \frac{\exp(\phi_\theta(s,a))}{\sum_{a' \in \mathcal{A}} \exp(\phi_\theta(s,a'))},$$

which is a softmax function. Here, $\phi_\theta(s,a) \in \mathbb{R}$ is the token logit. The entropy of policy distribution $\pi_\theta(\cdot \mid s)$ measures its uncertainty:

$$\mathcal{H}\left(\pi_\theta(\cdot \mid s)\right) = -\sum_{a} \pi_\theta(a \mid s) \log \pi_\theta(a \mid s).$$

Applying the chain rule yields $\frac{\partial \mathcal{H}}{\partial \phi_\theta(a_i|s)}$:

$$\pi_\theta(a_i \mid s)\left(\log \pi_\theta(a_i \mid s) + \mathcal{H}\left(\pi_\theta(\cdot \mid s)\right)\right). \quad (1)$$

The partial derivatives are given by:

$$\frac{\partial \pi_\theta(a_i \mid s)}{\partial \phi_\theta(s,a_i)} = \begin{cases} \pi_\theta(a \mid s)\left(1 - \pi_\theta(a \mid s)\right), & a = a_i \\ -\pi_\theta(a \mid s)\pi_\theta(a_i \mid s), & a \neq a_i \end{cases},$$

$$\frac{\partial \log \pi_\theta(a_i \mid s)}{\partial \phi_\theta(s,a_i)} = \begin{cases} 1 - \pi_\theta(a_i \mid s), & a = a_i \\ -\pi_\theta(a_i \mid s), & a \neq a_i \end{cases}.$$

2.2 Policy Gradient Derivation

Proximal Policy Optimization (PPO) (Schulman et al., 2017) and RLVR (Lambert et al., 2024) aim to maximize a rule-based reward $A_t = \frac{r(\boldsymbol{y}) - \mathrm{mean}\left(r(\boldsymbol{y}^{1:G})\right)}{\mathrm{std}(r(\boldsymbol{y}^{1:G}))}$ (Williams, 1992) :

$$\max_\theta J(\theta) := \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}, \boldsymbol{y} \sim \pi_\theta(\boldsymbol{x})}\left[A(\boldsymbol{y})\right], \quad (2)$$

where $\boldsymbol{x} \sim \mathcal{D}$ is the input prompt, and $\boldsymbol{y} = \{y_1, \ldots, y_T\}$ is the generated sequence of length T. Plugging in the partial derivatives into the policy gradient framework, we obtain:

$$\begin{aligned} \frac{\partial J}{\partial \phi_\theta(s,a_i)} &= \mathbb{E}_{\pi_\theta} \frac{\partial \log \pi_\theta(a_i \mid s)}{\partial \phi_\theta(s,a_i)} A(s,a) \\ &= \pi_\theta(a_i \mid s)\left(A(s,a_i) - \mathbb{E}_{\pi_\theta}\left[A(s,a)\right]\right), \end{aligned} \quad (3)$$

where $\mathbb{E}_{\pi_\theta}\left[A(s,a)\right] = \sum_a \pi_\theta(a \mid s) A(s,a)$.

3 Methodology

3.1 Framework of TEPO

In Fig.1, `TEPO` leverages sequence-level likelihood and a token-level selective KL-Divergence mask to bridge group-level rewards with individual tokens. Our method's loss function is formalized as Equation 4, where the blue component denotes token-level aggregation, the green component represents sequence-level likelihood, and the red component ($M_{i,t}$) corresponds to the KL-Divergence mask (applied to the t-th token in the i-th response).

$$J(\theta) = \frac{1}{\sum_{i=1}^{G} \|o_i\|} \sum_{i=1}^{G} \sum_{t=1}^{\|o_i\|} \left[\min\left(w_i(\theta), \text{clip}\left(w_i(\theta), 1-\epsilon, 1+\epsilon \right) \right) \cdot A_{i,t} - \beta \cdot \text{KL}\big(\pi_\theta \parallel \pi_{\theta_{\text{old}}}\big) \cdot M \right],$$
$$M = \begin{cases} 1, & \text{if } A_{i,t} > 0 \wedge \Delta\mathcal{H}_{i,t} < 0, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

3.2 Undifferentiated KL-Divergence unfits in GRPO

3.2.1 The Fragility of the KL-Divergence

Lemma 3.1. *For a softmax policy $\pi_\theta(a \mid s)$ at iteration $k+1$, the update rule is:*

$$\pi_{k+1}(a \mid s) \propto \pi_k(a \mid s) \exp\left(\beta^{-1} A(s,a)\right),$$

where β balances stability and performance.

Proof. At iteration k, we obtain the next policy π_{k+1}, subject to $\sum_a \pi_{k+1}(a \mid s) = 1:$

$$\max_p \ \mathbb{E}_{a\sim p}\left[A(s,a)\right] - \beta \cdot \text{KL}\left(\pi_{k+1} \| \pi_k(\cdot \mid s)\right)$$

Formulating the Lagrangian optimization problem yields an iterative policy update rule:

$$\pi_{k+1}(a \mid s) = \frac{\pi_k(a \mid s) \exp\left(\beta^{-1} A(s,a)\right)}{\mathbb{E}_{a' \sim \pi_k(\cdot|s)}\left[\exp\left(\beta^{-1} A(s,a')\right)\right]}. \tag{5}$$

Equivalently, the update can be written as:

$$\pi_{k+1}(a \mid s) \propto \pi_k(a \mid s) \exp\left(\beta^{-1} A(s,a)\right).$$

Remark: KL-Divergence constraint preserves stability while hurts performance. Supporting evidence is shown in Panel A of Table 3. □

3.2.2 Inner Product Between Entropy Gradient and Policy Gradient

Lemma 3.2. *For a softmax policy $\pi_\theta(a \mid s) \propto \exp(\phi_\theta(s,a))$, the alignment between entropy gradient and policy gradient exhibits as:*

- *For suboptimal actions ($A(s,a) < 0$):* $\langle \nabla_{\phi_\theta} \mathcal{H}, \nabla_{\phi_\theta} J \rangle > 0$
- *For optimal actions ($A(s,a) > 0$):* $\langle \nabla_{\phi_\theta} \mathcal{H}, \nabla_{\phi_\theta} J \rangle < 0$

Proof. We analyze the inner product between the entropy gradient and the policy gradient:

$$\langle \nabla_{\phi_\theta} \mathcal{H}, \nabla_{\phi_\theta} J \rangle = \sum_a \frac{\partial \mathcal{H}}{\partial \phi_\theta(s,a)} \cdot \frac{\partial J}{\partial \phi_\theta(s,a)}.$$

Substituting the entropy gradient from Eq. 1 and the policy gradient from Eq. 3 yields:

$$\langle \nabla_{\phi_\theta} \mathcal{H}, \nabla_{\phi_\theta} J \rangle = \sum_a \frac{\partial \mathcal{H}}{\partial \phi_\theta(s,a)} \cdot \frac{\partial J}{\partial \phi_\theta(s,a)}.$$
$$= \sum_{a_i} \pi_\theta(a_i|s)^2 \left(\log \pi_\theta(a_i|s) + \mathcal{H}(\pi_\theta(\cdot|s))\right) A(s,a_i).$$

Case 1: Suboptimal Actions

- For actions with small probability ($\pi_\theta(a \mid s) \to 0^+$), we have $\log \pi_\theta(a \mid s) \to -\infty$ and $\mathcal{H}(\pi_\theta(\cdot \mid s)) \to 0^+$, making $\log \pi_\theta(a \mid s) + \mathcal{H}(\pi_\theta(\cdot \mid s)) < 0$.
- With $\pi_\theta(a \mid s)^2 > 0$, $A(s,a) < 0$, therefore: $\langle \nabla_{\phi_\theta} \mathcal{H}, \nabla_{\phi_\theta} J \rangle > 0$.

Case 2: Optimal Actions

- For actions with large probability ($\pi_\theta(a \mid s) \to 1^-$), we have $\log \pi_\theta(a \mid s) \to 0^-$, making $\log \pi_\theta(a \mid s) + \mathcal{H}(\pi_\theta(\cdot \mid s)) = (1 - \pi_\theta(a \mid s)) \log \pi_\theta(a \mid s) < 0$
- With $\pi_\theta(a \mid s)^2 > 0$, $A(s,a) > 0$, therefore: $\langle \nabla_{\phi_\theta} \mathcal{H}, \nabla_{\phi_\theta} J \rangle < 0$.

□

3.2.3 Undifferentiated Entropy Regularization unfits

Theorem 3.1. *$\Delta\mathcal{H}$ (entropy change) characterizes the exploration tendency as follows:*

- *For $A(s,a) < 0$: $(\langle \nabla\mathcal{H}, \nabla J \rangle > 0) \to \Delta\mathcal{H} > 0$, **promoting** exploration.*
- *For $A(s,a) > 0$: $(\langle \nabla\mathcal{H}, \nabla J \rangle < 0) \to \Delta\mathcal{H} < 0$, **suppressing** exploration.*

Proof. Policy gradient update with $\alpha > 0$ is:

$$\phi_\theta(s,a) \leftarrow \phi_\theta(s,a) + \alpha \cdot \frac{\partial J}{\partial \phi_\theta(s,a)}.$$

$\Delta\mathcal{H}$ is approximated via a first-order Taylor expansion as:

$$\Delta\mathcal{H} \approx \frac{\partial \mathcal{H}}{\partial \phi_\theta(s,a)} \cdot \Delta\phi_\theta(s,a) = \alpha \cdot \langle \nabla_{\phi_\theta} \mathcal{H}, \nabla_{\phi_\theta} J \rangle,$$

where $\Delta\phi_\theta(s,a) = \alpha \cdot \frac{\partial J}{\partial \phi_\theta(s,a)}$. Therefore:

- When $A(s,a) < 0$, $(\langle \nabla\mathcal{H}, \nabla J \rangle > 0) \to \Delta\mathcal{H} > 0$, **promoting** exploration.

- When $A(s,a) > 0$, $(\langle \nabla\mathcal{H}, \nabla J\rangle < 0) \rightarrow \Delta\mathcal{H} < 0$, **suppressing** exploration.

Remark: GRPO grants a mechanism that suppresses high-advantage exploration and redirects it to low-advantage regions in critic-free GRPO, which counteracts with undifferentiated entropy regularization. Evidence that undifferentiated entropy regularization is unsuitable is provided in Panel B of Table 3. □

3.2.4 Why Token-Level Importance Sampling unfits

In GRPO (Shao et al., 2024), we decompose the policy entropy change $\mathcal{H}(\pi_{k+1}) - \mathcal{H}(\pi_k)$, under the state distributions d^{π_k} and $d^{\pi_{k+1}}$:

$$\underbrace{\mathbb{E}_{s\sim d^{\pi_{k+1}}}\mathcal{H}(\pi_{k+1}(\cdot \mid s)) - \mathbb{E}_{s\sim d^{\pi_k}}\mathcal{H}(\pi_{k+1}(\cdot \mid s))}_{\text{State Distribution Shift: } \Delta\mathcal{H} \text{ with Importance Sampling}}$$

$$+ \underbrace{\mathbb{E}_{s\sim d^{\pi_k}}\mathcal{H}(\pi_{k+1}(\cdot \mid s)) - \mathbb{E}_{s\sim d^{\pi_k}}\mathcal{H}(\pi_k(\cdot \mid s))}_{\Delta\mathcal{H} \text{ During Sampling}}$$

The discrepancy between its sampling strategy and model update strategy gives rise to State Distribution Shift: $\Delta\mathcal{H}$ with Importance Sampling. Consequently, critic-free GRPO struggles to perform token-level importance sampling (IS) $\frac{\pi_\theta(y_{i,t}|x,y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t}|x,y_{i,<t})}$, as this token-level metric fails to capture the global state distribution shift ($d^{\pi_{k+1}} \neq d^{\pi_k}$). We further identify that critic-free GRPO inherently suffers from sparse token-level rewards, a finding validated by ablation studies in Panel C of Table 3 and formal mathematical derivations in Section B.

3.3 Components in `TEPO`

This section explains the rationale behind each component of our method.

3.3.1 Sequence-level Likelihood

Leveraging the Markov factorization (where each token's probability depends on prior tokens (Chung, 1967)), we define the sequence-level weight $w_i(\theta)$ as the geometric mean of token-level importance ratios $\left(\frac{\pi_\theta(y_i|x)}{\pi_{\theta_{\text{old}}}(y_i|x)}\right)^{\frac{1}{|y_i|}}$:

$$w_i(\theta) = \exp\left(\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\log\frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})}\right),$$

which represents a geometric mean. This approach uses sequence-level likelihood to connect group-level rewards and token-level aggregation, balancing exploration and exploitation with these key advantages:

- Reduces gradient bias (Figure 2a);
- Lowers reasoning time (338 vs. 357 **seconds per step** for DAPO/GRPO) (Figure 2b);
- Maintains training stability and exploration capability (Figure 3).

3.3.2 Analysis of Token-Level KL Regularization

As shown in Section 3.2.2, for the t-th token in response i, the misalignment condition

$$(A_{i,t} > 0 \wedge \Delta\mathcal{H}_{i,t} < 0) \vee (A_{i,t} < 0 \wedge \Delta\mathcal{H}_{i,t} > 0)$$

holds. This discrepancy can lead to excessive policy updates; we therefore hypothesize that token-level KL-Divergence regularization can effectively mitigate this issue.

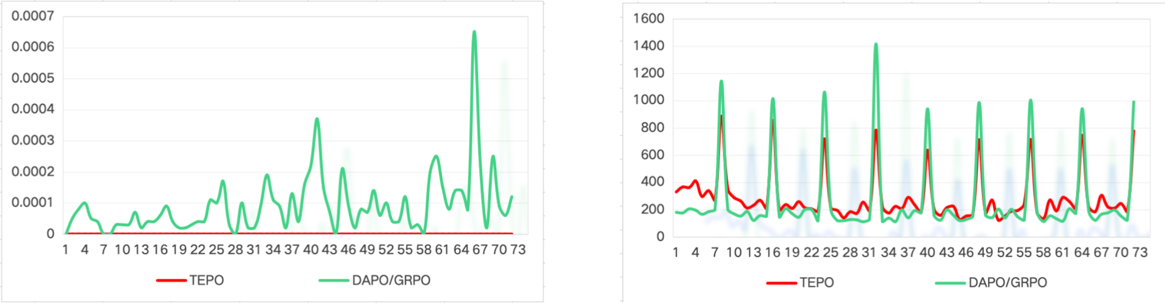
To verify this hypothesis, we conduct ablation experiments (Panel E of Table 3) comparing three variants: `TEPO` without KL regularization, `TEPO` with KL applied to $(A_{i,t} < 0 \wedge \Delta\mathcal{H}_{i,t} > 0)$,$(A_{i,t} > 0 \wedge \Delta\mathcal{H}_{i,t} < 0) \vee (A_{i,t} < 0 \wedge \Delta\mathcal{H}_{i,t} > 0)$ and `TEPO` with KL applied only to tokens satisfying $(A_{i,t} > 0 \wedge \Delta\mathcal{H}_{i,t} < 0)$. The results confirm the effectiveness of `TEPO` with KL applied only to tokens satisfying $(A_{i,t} > 0 \wedge \Delta\mathcal{H}_{i,t} < 0)$. The performance drop observed under the condition $A < 0 \wedge \Delta\mathcal{H} > 0$ suggests that misleading responses in this regime fail to provide reliable gradient directions for policy optimization.

4 Experiment

4.1 Experimental Setup

Implementation Details. For each rollout step, we processed 64 prompts per batch and sampled 8 responses per prompt with temperature 1.0. The policy was updated 8 times using these responses. To keep training effective, we removed prompts whose sampled responses were all correct or all wrong, following (Yu et al., 2025). Key hyperparameters were: learning rate $lr = 5 \times 10^{-7}$, maximum prompt length $max\_prompt\_length = 2024$, maximum response length $max\_response\_length = 8192$, and training prompt mini-batch size $train\_prompt\_mini\_bsz = 16$.

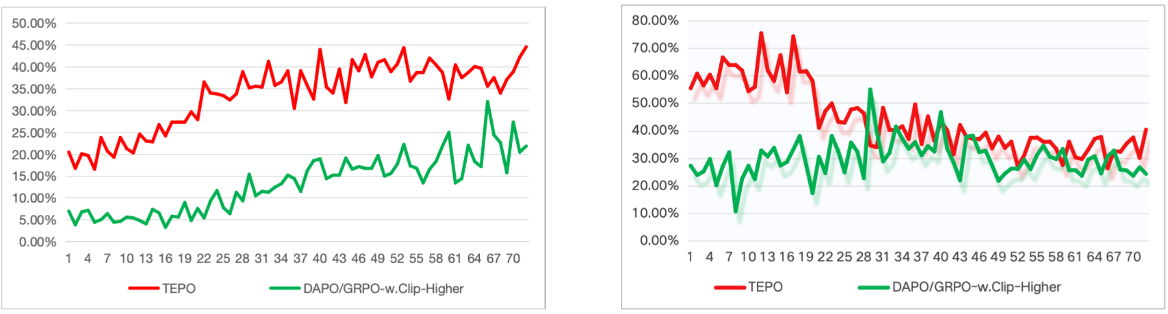
Datasets. We trained different models on DAPO-MATH (Yu et al., 2025) (Approximately 80*64 = 5120 pieces of data.) and evaluated on seven math benchmarks: MATH-500, AIME24/25 (Li et al., 2024), AMC, OMNI-MATH, OlympiadBench, and



(a) Clip Ratio Over Training Steps

(b) Reasoning Time Evolution Over Training Steps

Figure 2: Lower Gradient Bias and Faster Reasoning Efficiency with Markov Likelihood: The left panel shows that our method with a lower clip ratio effectively mitigates gradient bias, while the right panel indicates that it reduces the generation of redundant reasoning steps. Specifically, the average reasoning time of TEPO is 338 **seconds per step**, which is lower than that of DAPO/GRPO (357).



(a) Reward Progression Over Steps

(b) Gradient Norm Over Steps

Figure 3: Markov Likelihood enhanced performance (We transform the raw data into percentage-based values.): The left panel shows that our method achieves steadier and higher rewards across training steps, demonstrating more efficient learning dynamics. The right panel indicates that our method exhibits consistently higher gradient norms, reflecting more active and effective parameter reasoning.

Minerva (Lewkowycz et al., 2022). For AIME and AMC we used temperature 0.6; for the others we used greedy decoding. The maximum generation length was 8192 tokens. AIME, AIME25, and AMC results were reported with 32 samples per problem (@32) following prior work (Guo et al., 2025; DeepSeek-AI et al., 2025).

Baseline Methods. GRPO/DAPO (Shao et al., 2024; Yu et al., 2025) adopt the Clip-Higher bound in the PPO loss, with $\epsilon_{\text{low}} = 0.2$ and $\epsilon_{\text{high}} = 0.28$. Clip-Cov (Cui et al., 2025) clips tokens with high covariance using a clip ratio $r = 2 \times 10^{-4}$. For KL-Cov, the parameter k is set to 2×10^{-3} and the KL coefficient $\beta = 1$. An entropy-based method (Cheng et al., 2025) incorporates entropy regularization into advantage estimation, with scale $\alpha = 4 \times 10^{-4}$ and clipping parameter $\kappa = 2$. Our proposed `TEPO` is configured with $\beta = 0.001$ for Qwen2.5-7b and $\beta = 1$ for Qwen3-14B. All baseline hyperparameters are adopted from their original papers or recommended settings, and we did not perform additional tuning to favor TEPO.

Ablation Evaluations. We conduct ablation experiments to verify the effectiveness of key components in `TEPO`:

- **Entropy and KL Regularization**: We incorporate KL-Divergence regularization and entropy regularization (maximizing or minimizing entropy) in the first two parts. STEEL (Hao et al., 2025) proposes an adaptive token-reweighting method that mitigates entropy collapse and achieves state-of-the-art performance on mathematical and coding benchmarks.
- **Importance Sampling Strategies** (All details are listed in Table 5.): We compare three importance sampling variants: **a.** GPG (Chu et al., 2025), which removes the reference model and abandons importance sampling; **b.** CISPO (Chen et al., 2025), which performs single-step policy updates by employing $sg[\beth_i(\theta)]\cdot A_i\cdot\log\pi$, where sg means stop policy gradient; **c.** Prefix importance sampling, $w_{i,j\leq t}$ denotes the prefix of the i-th sentence up to the t-th token.
- **Aggregation Methods**: Evaluations for aggregation:**a.** 'sequence-mean token-mean aggrega-

Table 1: Performance Comparison of 7B and 14B Models on Mathematical Reasoning Benchmarks. Notations: 1-(w/o. GRPO/DAPO) indicates models not applying GRPO/DAPO, while (w. GRPO/DAPO) indicates models applying GRPO/DAPO; 2-higher values indicate better performance; 3-best results are marked in **bold**; 4-Δ denotes the performance difference between models *without* and *with* our method; .

Method	AIME24	AIME25	AMC	MATH-500	OMNI-MATH	OlympiadBench	Minerva	Avg.
Qwen2.5-7B	0.94	0.94	14.34	43.20	13.73	16.74	21.69	13.30
w. GRPO/DAPO	11.25	5.00	42.80	76.60	25.00	37.92	33.08	30.85
w. ClIP-Cov	10.83	7.40	42.84	75.60	26.11	39.40	**37.86**	31.64
w. KL-Cov	12.29	8.23	42.77	75.60	25.64	39.11	34.92	31.60
w. Entropy-based Term	11.35	6.15	43.18	74.80	26.14	40.14	36.02	31.62
w. GPG	**13.54**	6.04	43.29	75.00	26.28	**40.44**	34.55	31.91
w. GSPO	11.77	6.04	42.77	75.80	25.72	38.37	35.66	31.33
`TEPO`	12.60	**8.75**	**43.48**	**77.40**	**27.17**	**40.44**	34.92	**32.59**
`TEPO` (Δ vs GRPO/DAPO, %)	1.35	3.75	5.68	0.80	2.17	2.52	1.84	1.74
Qwen3-14B	19.16	16.77	51.01	82.80	30.27	43.40	45.95	38.34
w. GRPO/DAPO	22.08	19.06	55.94	85.60	33.01	44.00	47.79	41.51
w. CLIP-Cov	22.70	20.10	54.96	86.00	32.94	43.55	45.58	41.29
w. KL-Cov	23.85	19.06	56.47	**86.40**	32.87	43.85	47.79	41.85
w. Entropy-based Term	24.27	18.54	55.87	85.20	33.04	42.37	48.52	41.56
w. GPG	23.85	19.89	57.34	84.40	32.22	43.55	47.05	42.16
w. GSPO	23.85	20.41	56.58	86.00	33.54	44.44	48.52	42.28
`TEPO`	**24.37**	**23.75**	**58.96**	**86.40**	**35.11**	**46.37**	**49.26**	**44.02**
`TEPO` (Δ vs GRPO/DAPO, %)	5.21	4.69	3.02	0.80	2.10	2.37	1.47	2.51

Table 2: Performance Comparison of Various Models on Mathematical Reasoning Benchmarks

	AIME24	AIME25	AMC	MATH-500	OMNI-MATH	OlympiadBench	Minerva	Avg.
DeepSeek-R1-Distill-Llama-8B	12.70	13.33	46.49	70.40	25.50	36.88	23.52	32.46
w. GRPO/DAPO	**27.50**	18.43	**61.89**	78.40	32.97	41.62	26.10	42.23
`TEPO`	25.72	**18.85**	60.84	**79.60**	**33.11**	**43.55**	**29.77**	**42.76**
Mistral-7B-Instruct-v0.2	-	-	2.18	9.80	3.98	1.19	5.88	2.77
w.GRPO/DAPO	-	-	3.13	10.80	4.27	**2.37**	**8.09**	3.37
`TEPO`	-	-	**3.58**	**12.80**	**4.59**	1.93	**8.09**	**3.65**
DeepSeek-R1-Distill-Qwen-7B	29.89	20.93	59.48	83.40	33.86	42.66	34.19	43.23
w. GRPO/DAPO	32.08	23.33	66.10	**87.80**	38.42	47.70	40.44	45.99
`TEPO`	**32.70**	**24.58**	**66.41**	**87.80**	**39.55**	**49.92**	**40.80**	**48.60**

tion'/GSPO (Zheng et al., 2025a); **b.** 'sequence-mean token-sum aggregation'.

- **Token-Level KL-Divergence:** Ablation experiments on the token-wise scope of KL-Divergence masking are performed to validate `TEPO`'s effectiveness.

4.2 Main Results

In Table 1 and Table 2, `TEPO` achieves the highest average accuracy across all seven mathematical reasoning benchmarks.

- **Consistently State-of-the-Art Overall Performance:** Notably, `TEPO` outperforms the baseline method GRPO/DAPO as well as all other comparative variant methods, including CLIP-Cov, KL-Cov, Entropy-based Term, GPG, and GSPO:
  - Qwen2.5-7B: `TEPO` attains an average accuracy of 32.59%. Compared with the baseline GRPO/DAPO, which achieves an average accuracy of 30.85%, this represents a 1.74 percentage points (pp) improvement. Additionally, `TEPO` outperforms GPG (the second-best variant method with an average accuracy of 31.91%) by 0.68 pp;
  - Qwen3-14B: `TEPO` reaches an average accuracy of 44.02%. It surpasses the GRPO/DAPO baseline (with an average accuracy of 41.51%) by 2.51 pp. Furthermore, `TEPO` outperforms GSPO, which is identified as the second-best variant method with an average accuracy of 42.28%, by 1.74 pp;

Table 3: Ablation Studies of Key Components. All panels are compared with`TEPO`-w/o. KL: Panel A demonstrates the fragility of undifferentiated KL-Divergence and its performance degradation effect; Panel B shows that undifferentiated entropy regularization yields marginal improvements to `TEPO`-w/o-KL; Panel C validates `TEPO`'s effectiveness across different importance sampling strategies and corroborates that GRPO exhibits sparse token-level rewards; Panel D confirms the superiority of token-mean aggregation over alternative strategies; Panel E verifies that `TEPO` achieves optimal control of token-level KL-Divergence.

Method	**AIME24**	**AIME25**	**AMC**	**MATH-500**	**OMNI-MATH**	**OlympiadBench**	**Minerva**	**Avg.**
`TEPO`-w/o. KL	12.70	6.35	**43.56**	75.40	27.00	39.55	**37.50**	32.21
Panel A: `TEPO` *w. Undifferentiated KL-Divergence*								
w. $\beta = 1$(model collapse in 24 steps)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
w. $\beta = 0.1$ (leads to Avg drop of 4.6%.)	5.52	3.96	42.84	75.60	22.41	32.74	30.88	26.25
w. $\beta = 0.01$	11.45	7.40	42.24	76.20	26.32	40.59	33.82	31.64
w. $\beta = 0.001$	12.29	8.96	42.24	77.20	25.89	40.11	36.39	31.81
w. $\beta = 0.0001$	11.45	5.63	43.18	76.20	26.53	37.33	37.50	31.61
Panel B: `TEPO` *w. Undifferentiated Entropy Regularization*								
w. Max-Entropy	13.02	5.94	42.80	76.60	26.85	38.37	37.13	31.65
w. Min-Entropy	11.97	5.63	41.94	72.40	25.75	36.29	35.29	30.68
w. STEEL	12.70	5.63	41.64	76.00	26.82	37.03	36.39	31.30
w. ClLIP-Cov	10.83	7.40	42.84	75.60	26.11	39.40	**37.86**	31.64
w. KL-Cov	12.29	8.23	42.77	75.60	25.64	39.11	34.92	31.60
w. Entropy-based Term	11.35	6.15	43.18	74.80	26.14	40.14	36.02	31.62
Panel C: GRPO/DAPO w. Different Importance sampling(IS)								
w. GRPO/DAPO (token-level)	11.25	5.00	42.80	76.60	25.00	37.92	33.08	30.85
w. GPG (remove IS)	**13.54**	6.04	43.29	75.00	26.28	**40.44**	34.55	31.91
w. CISPO/Reinforce (token-level)	11.04	6.77	42.80	74.80	26.96	38.51	32.72	31.58
w. Sentence Prefix IS	11.08	6.67	42.44	73.40	25.11	39.11	35.66	30.95
Panel D: Sentence Likelihood w. Different Aggregation								
w. sequence-mean token-mean/GSPO aggregation	11.77	6.04	42.77	75.80	25.72	38.37	35.66	31.33
w. sequence-mean token-sum aggregation	12.50	5.52	43.18	76.00	26.57	38.96	36.39	31.80
Panel E: `TEPO` *w. Token-Level KL-Divergence*								
`TEPO`-w/o. KL	12.70	6.35	**43.56**	75.40	27.00	39.55	37.50	32.21
`TEPO`-w. KL($A_{i,t} < 0 \wedge \Delta\mathcal{H}_{i,t} > 0$)	11.35	4.37	42.62	76.40	26.11	39.25	35.66	31.47
`TEPO`-w. KL($A_{i,t} > 0 \wedge \Delta\mathcal{H}_{i,t} < 0) \vee (A_{i,t} < 0 \wedge \Delta\mathcal{H}_{i,t} > 0$)	13.43	5.73	44.05	75.60	26.25	38.81	35.66	32.03
`TEPO`	12.60	**8.75**	43.48	**77.40**	**27.17**	**40.44**	34.92	**32.59**

  - Non-Qwen models: Detailed results reported in Table 2 show that `TEPO` delivers state-of-the-art average accuracies across three distinct non-Qwen model architectures. Specifically: On DeepSeek-R1-Distill-Llama-8B: `TEPO` achieves 42.76% average accuracy, which is 0.53 pp higher than GRPO/DAPO's 42.23%; On Mistral-7B-Instruct-v0.2: `TEPO` attains 3.65% average accuracy, representing a 0.28 pp gain over GRPO/DAPO's 3.37%; On DeepSeek-R1-Distill-Qwen-7B: `TEPO` reaches 48.60% average accuracy, surpassing GRPO/DAPO's 45.99% by 2.61 pp.
- **Stable Performance Across All Sub-Benchmarks:** Unlike some variant methods that exhibit performance bottlenecks on specific sub-benchmarks, `TEPO` maintains consistent top-tier performance across all seven mathematical reasoning sub-benchmarks:
  - **Qwen2.5-7B Model**: `TEPO` achieves rank-1 performance on four core sub-benchmarks with the following accuracies: MATH-500 (77.40%), OMNI-MATH (27.17%), AMC (43.48%), AIME25 (8.75%), and OlympiadBench(40.44%). For the remaining two sub-benchmarks (AIME24, Minerva), `TEPO` secures a top-3 position with accuracies of 12.60% (AIME24), and 34.92% (Minerva), outperforming most variant methods;
  - **Qwen3-14B Model**: `TEPO` achieves rank-1 performance across all seven sub-benchmarks, with the following leading accuracies: AIME24 (24.37%), AIME25 (23.75%), AMC (58.96%), MATH-500 (86.40%), OMNI-MATH (35.11%), OlympiadBench (46.37%), and Minerva (49.26%). This represents a comprehensive performance advantage over GRPO/DAPO (+5.21 pp on AIME24, +4.69 pp on AIME25) and the others;
  - **Non-Qwen Models**: DeepSeek-R1-Distill-Llama-8B: `TEPO` ranks first on five sub-benchmarks with the following accuracies: AIME25 (18.85%), MATH-500 (79.60%), OMNI-MATH (33.11%), OlympiadBench (43.55%), and Minerva (29.77%); Mistral-7B-Instruct-v0.2: `TEPO` leads on four

Table 4: Ablation Studies of Random Seeds. We report performance on seven mathematical reasoning benchmarks using two different random seeds for each method. “Mean” denotes the average performance over two seeds. `TEPO` consistently outperforms baseline methods across all settings.

Method	**AIME24**	**AIME25**	**AMC**	**MATH-500**	**OMNI-MATH**	**OlympiadBench**	**Minerva**	**Avg.**
GRPO/DAPO (72 steps, seed1)	11.56	6.25	40.47	72.40	26.00	37.77	36.76	30.30
GRPO/DAPO (72 steps, seed2)	11.25	5.00	42.80	76.60	25.00	37.92	33.08	30.85
GRPO/DAPO (132 steps, seed1)	12.08	7.08	42.65	76.20	26.07	39.85	37.86	31.73
GRPO/DAPO (132 steps, seed2)	12.08	6.98	41.67	74.20	27.42	39.85	36.76	31.69
TEPO (72 steps, seed1)	12.50	7.60	45.18	77.80	26.78	41.03	38.97	32.62
TEPO (72 steps, seed2)	12.60	8.75	43.48	77.40	27.17	40.44	34.92	32.59
Mean over Two Seeds								
GRPO/DAPO (72 steps, mean)	11.41	5.63	41.64	74.50	25.50	37.85	34.92	30.58
GRPO/DAPO (132 steps, mean)	12.08	7.03	42.16	75.20	26.75	39.85	**37.31**	31.71
`TEPO` (72 steps, mean)	**12.55**	**8.18**	**44.33**	**77.60**	**26.98**	**40.74**	36.95	**32.61**

sub-benchmarks with accuracies of AMC (3.58%), MATH-500 (12.80%), OMNI-MATH (4.59%), and Minerva (8.09%); DeepSeek-R1-Distill-Qwen-7B: `TEPO` achieves rank-1 performance across all seven sub-benchmarks, with key accuracies including AIME24 (32.70%), AIME25 (24.58%), AMC (66.41%), MATH-500 (87.80%), OMNI-MATH (39.55%), OlympiadBench (49.92%), and Minerva (40.80%), outperforming GRPO/DAPO by 2.61 pp on average.

4.3 Ablation Studies

As presented in Table 3, comprehensive ablation studies on key components further validate the components of `TEPO`:

- Panel A demonstrates the fragility of undifferentiated KL-Divergence and its performance degradation effect (vs. `TEPO`-w/o-KL’s 32.21%): High β values ($\beta = 1$) cause severe performance collapse (0% accuracy across all benchmarks) within 24 training steps; Moderate $\beta = 0.1$ results in a 4.6% average accuracy drop; Even low β values (0.01, 0.001, 0.0001) fail to exceed `TEPO`-w/o-KL (32.21%), with average accuracies of 31.64%, 31.81%, and 31.61% respectively.
- Panel B shows that undifferentiated entropy regularization yields marginal improvements to `TEPO`-w/o-KL:
  - Max-Entropy achieves 31.65% average accuracy (0.56% lower than `TEPO`-w/o-KL);
  - Min-Entropy leads to a 1.53% average drop (30.68% vs. `TEPO`-w/o-KL’s 32.21%).
- Panel C validates the effectiveness of `TEPO` across different IS strategies and corroborates that GRPO inherently suffers from sparse token-level rewards. Specifically, GRPO/DAPO adopt a token-level IS scheme; GPG abandons IS entirely, treating all tokens uniformly; CISPO reverts to the original REINFORCE framework to eliminate the impact of IS; and Sentence Prefix IS attempts to leverage clause-level IS to guide the reasoning process. Quantitative results further substantiate these design differences: 1) The baseline GRPO/DAPO achieves only 30.85% average accuracy, a 1.74 percentage point (pp) drop compared with `TEPO`; 2) GPG, the second-best variant, reaches 31.91% average accuracy, still 0.68 pp lower than `TEPO`; 3) CISPO/Reinforce (31.58%) and Sentence Prefix IS (30.95%) lag further behind.
- Panel D confirms the superiority of token-mean aggregation over alternative strategies: Sequence-mean-token-mean (GSPO) and sequence-mean-token-sum aggregations reach 31.33% and 31.80% average accuracy respectively; Both are outperformed by `TEPO`’s aggregation design (32.21% average accuracy), which balances token-level sparsity and sequence-level consistency.
- Panel E verifies that `TEPO` achieves optimal control of token-level KL-divergence: We conduct ablation experiments (Panel E of Table 3) comparing three `TEPO` variants: (1) no KL regularization (Avg. 32.21%), (2) KL applied to $(A_{i,t} > 0 \wedge \Delta\mathcal{H}_{i,t} < 0) \vee (A_{i,t} < 0 \wedge \Delta\mathcal{H}_{i,t} > 0)$ (Avg. 32.03%), and (3) KL restricted to $A_{i,t} > 0 \wedge \Delta\mathcal{H}_{i,t} < 0$ (Avg. 32.59%). Results confirm variant (3)’s superiority.

Table 4 presents the results of ablation studies on random seeds, aiming to verify reducing convergence time by nearly 50%. Three training strategies are compared in the experiment: GRPO/DAPO (72 steps), GRPO/DAPO (132 steps), and TEPO (72 steps). Each method is independently run with two different random seeds (seed1 and seed2), and the average performance over the two seeds is shown in the lower part of the table. It is clearly observed from the results that with the same batch size and throughput, TEPO at 72 steps outperforms GRPO/DAPO at both 72 steps and 132 steps, cutting training steps by nearly half. Specifically, TEPO (72 steps) achieves an average performance of 32.61% over two seeds, which is significantly higher than GRPO/DAPO (72 steps, 30.58%) and GRPO/DAPO (132 steps, 31.71%). Meanwhile, all methods show slight performance fluctuations across different random seeds, but the performance trend remains consistent, indicating good stability. All results confirm TEPO achieves significantly faster convergence under strictly matched computational budgets, and its superiority over baseline methods is further verified by its robust performance across different random initializations.

5 Related Works

Balancing the exploration-exploitation (E-E) tradeoff is a core challenge in reinforcement learning (RL) (Sutton and Barto, 1998). Proximal Policy Optimization (PPO) uses an entropy bonus to sustain exploration (Schulman et al., 2017), while Soft Actor-Critic (SAC) directly optimizes a maximum-entropy objective (Haarnoja et al., 2017, 2018). However, the role of entropy in RL for large language models (LLMs) remains unclear.

Reinforcement Learning from Human Feedback (RLHF) typically employs a KL penalty relative to a reference policy (Ouyang et al., 2022; Hu et al., 2024). Notably, GRPO and recent studies find minimal or ambiguous benefits from standard entropy bonuses (Shao et al., 2024; Chu et al., 2025; Zheng et al., 2025a), leaving entropy's impact on generation quality and training stability an open question.

Existing LLM-RL methods adopt diverse frameworks: GPG uses REINFORCE for straightforward training (Chu et al., 2025); CISPO improves efficiency via clipped, detached importance weights (Chen et al., 2025); GSPO shifts to sequence-level learning with whole-sequence likelihood ratios (Zheng et al., 2025b). Additionally, (Cui et al., 2025) derived a performance-entropy relationship $R = -a \exp(\mathcal{H}) + b$, indicating lower entropy generally correlates with better performance.

6 Conclusion

GRPO advances LLMs' mathematical reasoning but suffers from intractable token-level sparse-reward issues in CoT reasoning. To address these drawbacks, we introduce `TEPO`, which (1) leverages sequence-level likelihood to bridge group-level rewards and individual tokens via token-level aggregation, and (2) deploys a token-level KL-divergence mask to mitigate abrupt policy updates. Empirical results show `TEPO` achieves state-of-the-art performance on mathematical reasoning benchmarks and significantly enhances training stability, cutting convergence time by 50% relative to GRPO/DAPO.

Limitations

Despite proposing a novel token-level framework that (1) uses sequence-level likelihood to connect group-level rewards and individual tokens via token-level aggregation, and (2) employs a token-level KL-divergence mask to alleviate abrupt policy updates, our work has two key limitations. We neither clarify the mechanism behind the effectiveness of the token constraint (targeting positive-advantage, entropy-decreasing tokens) nor distinguish how different token types uniquely impact model performance. Future work should explore the distinct roles of tokens in CoT reasoning and design a more universal framework for bridging token-level operations with group-level rewards.

7 Acknowledgement

This work is supported in part by National Natural Science Foundation of China under Grant Nos. 92567204, 62272193, and 62472194, and the Fundamental Research Funds for the Central Universities, and Jilin Science and Technology Research Project 20260101016JJ.

References

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. 2025. The unreasonable effectiveness of entropy minimization in llm reasoning. *arXiv preprint arXiv:2505.15134*.

Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, and 1 others. 2025. Minimax-m1: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*.

Daixuan Cheng, Shaohan Huang, Xuekai Zhu, Bo Dai, Wayne Xin Zhao, Zhenliang Zhang, and Furu Wei. 2025. Reasoning with exploration: An entropy perspective. *arXiv preprint arXiv:2506.14758*.

Xiangxiang Chu, Hailang Huang, Xiao Zhang, Fei Wei, and Yong Wang. 2025. Gpg: A simple and strong reinforcement learning baseline for model reasoning. *arXiv preprint arXiv:2504.02546*.

Kai Lai Chung. 1967. Markov chains. *Springer-Verlag, New York*.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, and 1 others. 2025. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*.

DeepSeek-AI, Daya Guo, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Zitian Gao, Lynx Chen, Joey Zhou, and Bryan Dai. 2025. One-shot entropy minimization. *arXiv preprint arXiv:2505.20282*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. 2017. Reinforcement learning with deep energy-based policies. In *International conference on machine learning*, pages 1352–1361. PMLR.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. Pmlr.

Zhezheng Hao, Hong Wang, Haoyang Liu, Jian Luo, Jiarui Yu, Hande Dong, Qiang Lin, Can Wang, and Jiawei Chen. 2025. Rethinking entropy interventions in rlvr: An entropy change perspective. *arXiv preprint arXiv:2510.10150*.

Jian Hu, Xibin Wu, Weixun Wang, Xianyu, Dehao Zhang, and Yu Cao. 2024. Openrlhf: An easy-to-use, scalable and high-performance rlhf framework. *arXiv preprint arXiv:2405.11143*.

Sham M Kakade. 2001. A natural policy gradient. *Advances in neural information processing systems*, 14.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and 1 others. 2024. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, and 1 others. 2022. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, and 1 others. 2024. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*, 13(9):9.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y.K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

R.S. Sutton and A.G. Barto. 1998. Reinforcement learning: An introduction. *IEEE Transactions on Neural Networks*, 9(5):1054–1054.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, and 1 others. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, and 1 others. 2025a. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, and 1 others. 2025b. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*.

A Target Policy Construction and KL-Divergence Minimization

Our current policy $\pi_{k+1}(a|s)$ aims to emulate an ideal "target" policy $\pi^*(a|s)$:

$$\pi^*(a|s) = \frac{\pi_{\mathrm{k}}(a|s)}{Z(s)} \exp\left(A(a,s)/\beta\right), \tag{6}$$

where $Z(s) = \sum_a \pi_{\mathrm{k}}(a|s) \exp\left(A(a,s)/\beta\right)$ is the partition function for normalization.

The emulation is equivalent to minimizing the KL-Divergence between π_{k+1} and π^*:

$$\min_\theta \mathrm{KL}(\pi_{k+1} \| \pi^*) \tag{7}$$

Substituting the expression for $\log \pi^*(a|s)$ from (6) into (7), and omitting the $\log Z(s)$ term (which is independent of π_θ), we obtain:

$$\mathrm{KL}(\pi_{k+1}\|\pi^*) = \frac{1}{\beta}\mathbb{E}_{a\sim\pi_\theta}\left[\beta \cdot \mathrm{KL}(\pi_{k+1}\|\pi_k) - A(a,s)\right] \tag{8}$$

This expression is structurally equivalent to the PPO-KL objective, where policy updates are constrained by a KL-Divergence regularizer scaled by the inverse temperature parameter $1/\beta$.

B Why Sentence Likelihood Derivation Fits GRPO

To illustrate why sentence likelihood derivation is suitable for GRPO, we first formalize the reinforcement learning (RL) objective function and conduct a step-by-step gradient derivation. This process reveals the core connection between sentence likelihood modeling and the GRPO framework.

We define the standard RL objective function for policy optimization as follows:

$$J(\theta) = \mathbb{E}_{\tau\sim p_\theta}[A(\tau)] = \int p_\theta(\tau) A(\tau) d\tau,$$

where θ denotes the policy parameters, τ represents a trajectory generated by the policy p_θ, and $A(\tau)$ is the advantage function that quantifies the quality of trajectory τ.

We then derive the gradient of the objective function with respect to θ step by step, which is essential for policy update:

$$\begin{aligned}
\nabla J(\theta) &= \int \nabla p_\theta(\tau) A(\tau) d\tau \\
&= \mathbb{E}_{\tau\sim p_\theta}\left[\nabla \log p_\theta(\tau) A(\tau)\right] \\
&= \mathbb{E}_{\tau\sim p_\theta}\left[\nabla \log p_\theta(\tau) A_{\theta_{\text{old}}}(\tau)\right] \\
&= \mathbb{E}_{\tau\sim p_{\theta_{\text{old}}}}\left[\frac{p_\theta(\tau)}{p_{\theta_{\text{old}}}(\tau)} \nabla \log p_\theta(\tau) A_{\theta_{\text{old}}}(\tau)\right] \\
&= \mathbb{E}_{\tau\sim p_{\theta_{\text{old}}}}\left[\frac{\nabla p_\theta(\tau)}{p_{\theta_{\text{old}}}(\tau)} A_{\theta_{\text{old}}}(\tau)\right].
\end{aligned}$$

In the derivation above, the third equality replaces the advantage $A(\tau)$ with $A_{\theta_{\text{old}}}(\tau)$ to stabilize the training process. The fourth equality employs the importance sampling technique, which allows us to estimate the expectation under the target policy p_θ using samples drawn from the old policy $p_{\theta_{\text{old}}}(\tau)$.

Notably, $p_\theta(\tau)$ represents the probability of the entire trajectory τ. For sequential decision-making processes, the trajectory probability can be decomposed into the product of token-level transition probabilities:

$$\begin{aligned}
\nabla p_\theta(\tau) &= \nabla\left(\prod_i p_\theta(x_i|s_{i-1})\right) \\
&= \sum_i \left(\left(\prod_{j\neq i} p_\theta(x_j|s_{j-1})\right) \nabla p_\theta(x_i|s_{i-1})\right)
\end{aligned}$$

$$p_{\theta_{\text{old}}}(\tau) = \prod_i p_{\theta_{\text{old}}}(x_i|s_{i-1}).$$

Substituting the above decompositions into the gradient expression, we obtain:

$$\begin{aligned}
\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim p_{\theta_{\text{old}}}}\Bigg[&A_{\theta_{\text{old}}}(\tau) \cdot \sum_{t=0}^{T-1} \\
&\nabla_\theta \log \pi_\theta(a_t \mid s_t) \cdot \prod_{j=0}^{T-1} \frac{\pi_\theta(a_j \mid s_j)}{\pi_{\theta_{\text{old}}}(a_j \mid s_j)}\Bigg].
\end{aligned}$$

A critical observation here is that the objective function of GRPO ($J_{\text{GRPO}}(\theta)$) omits the cross-token product term $\prod_{j\neq i} \frac{p_\theta(x_j|s_{j-1})}{p_{\theta_{\text{old}}}(x_j|s_{j-1})}$ in the above gradient. This omission simplifies the computation but introduces a discrepancy in GRPO objective.

C Derivation of Policy Entropy Change During Parameter Update

C.1 Notation Definitions and Preconditions

We sample G responses per prompt and compute the advantage with group-level normalization as follows:

$$A_t = \frac{r(\boldsymbol{y}) - \mathrm{mean}\left(r\left(\boldsymbol{y}^{1:G}\right)\right)}{\mathrm{std}\left(r\left(\boldsymbol{y}^{1:G}\right)\right)}. \tag{9}$$

C.2 Step 1: Taylor Expansion for Entropy Change

For $\Delta\theta$, the entropy change $\Delta H = H(\pi_{\theta+\Delta\theta}) - H(\pi_\theta)$ is approximated by:

$$\Delta H \approx \nabla_\theta H(\pi_\theta) \cdot \Delta\theta \tag{10}$$

Table 5: Core Notations for Entropy Derivation

Notation	Definition
$\{(x_i, y_i/o_i)\}_{i=1}^{G}$	Batch of prompt-response pairs,x denotes the prompt, and y/o_i denotes the response generated by LLMs
$y_{i,j\leq t}$	Prefix of the i-th sentence up to the t-th token
G	Number of responses generated by LLMs per prompt
$\lVert o_i \rVert$	Length of y_i/o_i
$\theta(a \mid s)$	Score assigned by large language models (LLMs)
τ	A complete solution trajectory for a single math problem.
$p_\theta(\tau)$	the probability of the entire trajectory
$\pi_\theta(a \mid s)$	Softmax score: $\pi_\theta(a \mid s) = \frac{e^{\phi_\theta(a \mid s)}}{\sum_{a'} e^{\phi_\theta(a' \mid s)}}$
$H(\pi_\theta)$	Policy entropy: $-\mathbb{E}_{a\sim\pi_\theta}\left[\log \pi_\theta(a \mid s)\right]$
$\Delta\theta$	Update of the score function $\phi_\theta(a \mid s)$: $\theta_{\text{new}} = \theta + \Delta\theta$
∇_θ	Gradient with respect to parameter θ
$A(s,a)$	Advantage function: $A_t = \frac{r(\boldsymbol{y})-\text{mean}\left(r\left(\boldsymbol{y}^{1:K}\right)\right)}{\text{std}(r(\boldsymbol{y}^{1:K}))}$
$\mathcal{F}$	Fisher information matrix: $\mathcal{F} = \mathbb{E}_{a\sim\pi_\theta}\left[\nabla_\theta \log \pi_\theta(a \mid s) \nabla_\theta \log \pi_\theta(a \mid s)^\top\right]$
$\frac{1}{\sum_{i=1}^{G} \lVert o_i \rVert} \sum_{i=1}^{G} \sum_{t=1}^{\lVert o_i \rVert}$	Token-level aggregation/token-mean aggregation
$\frac{1}{G} \sum_{i=1}^{G} \sum_{t=1}^{\lVert o_i \rVert}$	Sequence-mean token-sum aggregation
$\frac{1}{G} \sum_{i=1}^{G} \frac{1}{\lVert o_i \rVert} \sum_{t=1}^{\lVert o_i \rVert}$	Sequence-mean token-mean aggregation
$\frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})}$	Token-Level Importance sampling
$\left(\frac{\pi_\theta(y_i \mid x)}{\pi_{\theta_{\text{old}}}(y_i \mid x)}\right)^{\frac{1}{\lvert y_i \rvert}}$	Sequence-Level Importance Sampling
$\sum_{t=1}^{o_{i,j\leq t}} \left(\frac{\pi_\theta(y_{i,j\leq t} \mid x)}{\pi_{\theta_{\text{old}}}(o_{i,j\leq t} \mid x)}\right)^{\frac{1}{\lvert y_{i,j\leq t} \rvert}}$	Prefix Importance Sampling

C.3 Step 2: Gradient of Policy Entropy

C.3.1 Expand Entropy Definition

Discrete policy entropy (sum over all actions):

$$H(\pi_\theta) = -\sum_a \pi_\theta(a|s) \log \pi_\theta(a|s) \tag{11}$$

Gradient of Eq. (11) as:

$$\nabla_\theta H(\pi_\theta) = -\sum_a \left[\nabla_\theta \pi_\theta(a|s) \log \pi_\theta(a|s) + \pi_\theta(a|s) \nabla_\theta \log \pi_\theta(a|s)\ \right] \tag{12}$$

C.3.2 Simplify with Probability Normalization

Since $\sum_a \pi_\theta(a|s) = 1$, we have $\sum_a \nabla_\theta \pi_\theta(a|s) = 0$. Using $\nabla_\theta \pi_\theta(a|s) = \pi_\theta(a|s) \nabla_\theta \log \pi_\theta(a|s)$, the first term in Eq. (12) vanishes. Thus:

$$\nabla_\theta H(\pi_\theta) = -\sum_a \pi_\theta(a|s) \nabla_\theta \log \pi_\theta(a|s) \tag{13}$$

Log-probability derivative for Softmax:

$$\nabla_\theta \log \pi_\theta(a|s) = \nabla_\theta \phi_\theta(a|s) - \mathbb{E}_{a'\sim\pi_\theta}[\nabla_\theta \phi_\theta(a'|s)] \tag{14}$$

Substitute into Eq. (13):

$$\nabla_\theta H(\pi_\theta) = -E_{\pi_\theta}\left[\nabla_\theta \phi_\theta(a|s) - \mathbb{E}_{a'\sim\pi_\theta}[\nabla_\theta \phi_\theta(a'|s)]\right] \tag{15}$$

Algorithm 1 Token-Level Policy Gradient Computation for `TEPO`

```
Require: $\pi_\theta$: Current policy network (LLM);
$\pi_{\theta_{\text{old}}}$: Pre-update (reference) policy;
$\{(x_i, y_i)\}_{i=1}^{G}$: Batch of prompt-response pairs ($x_i$ = prompt, $y_i$ = LLM-generated response);
$A_{i,t}$: Token-level advantage for the $t$-th token in response $i$;
$\text{Mask}_{i,t} \in \{0, 1\}$: Valid token mask (1 = valid token, 0 = padding token);
$\alpha$: Learning rate for gradient ascent;
$M = \sum_{i=1}^{G} \sum_{t=1}^{|y_i|} \text{Mask}_{i,t}$: Total number of valid tokens (normalization factor)
Ensure: Updated policy parameters $\theta$
for all response sequence $i \in \{1, \ldots, G\}$ do
    Step 1: Compute token-level log probability ratio
        $\log r_{i,t}(\theta) = \log \pi_\theta(y_{i,t} \mid x_i, y_{i,<t}) - \log \pi_{\theta_{\text{old}}}(y_{i,t} \mid x_i, y_{i,<t})$
    Step 2: Aggregate to sequence-level log weight (geometric mean)
        $\log w_i(\theta) = \frac{1}{|y_i|} \sum_{t=1}^{|y_i|} \log r_{i,t}(\theta) \cdot \text{Mask}_{i,t}$    ▷ Normalize by sequence length
    Step 3: Exponentiate to get sequence-level IS weight
        $w_i(\theta) = \exp(\log w_i(\theta))$
    Step 4: Compute unclipped token-level loss term
        $L_{i,t}(\theta) = w_i(\theta) \cdot A_{i,t} \cdot \text{Mask}_{i,t}$    ▷ Weight advantage by sequence-level IS ratio
end for
Step 5: Calculate normalized policy objective
    $J(\theta) = \frac{1}{M} \sum_{i=1}^{G} \sum_{t=1}^{|y_i|} L_{i,t}(\theta)$    ▷ Normalize by total valid tokens to avoid batch bias
Step 6: Compute gradient of the policy objective
    $\nabla_\theta J(\theta) = \frac{1}{M} \sum_{i=1}^{G} \sum_{t=1}^{|y_i|} \nabla_\theta L_{i,t}(\theta)$
    $\nabla_\theta L_{i,t}(\theta) = A_{i,t} \cdot \text{Mask}_{i,t} \cdot \frac{w_i(\theta)}{|y_i| \cdot \pi_\theta(y_{i,t}|x_i, y_{i,<t})} \cdot \nabla_\theta \pi_\theta(y_{i,t} \mid x_i, y_{i,<t})$    ▷ Chain rule for gradient
Step 7: Update policy parameters (gradient ascent)
    $\theta \leftarrow \theta + \alpha \cdot \nabla_\theta J(\theta)$
    return Updated policy parameters $\theta$
```

C.4 Step 3: Entropy Change with NPG Update

C.4.1 NPG Update Rule

NPG update (stabilized by Fisher matrix):

$$\Delta\theta = \beta^{-1} \cdot \nabla_\theta J(\pi_\theta), \tag{16}$$

$J(\pi_\theta)$ = expected cumulative reward:

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{s,a\sim\pi_\theta}[\nabla_\theta \log \pi_\theta(a|s) \cdot A(s,a)] \tag{17}$$

C.4.2 Final Entropy Change

Substitute Eqs. (13) and (16) into Eq. (10):

$$\Delta H \approx -\beta^{-1} \cdot \text{Cov}_{\pi_\theta}\left[\log \pi_\theta(a|s), A(s,a)\right] \tag{18}$$

where $\text{Cov}[X,Y] = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$. Natural Policy Gradient (NPG) (Kakade, 2001) define a formula measures **how the current decision will lead to changes in entropy(See Details in the Appendix C)**:

$$\begin{aligned} &\mathbb{E}_{s\sim d^k}\mathcal{H}(\pi_{k+1}(\cdot \mid s)) - \mathbb{E}_{s\sim d^k}\mathcal{H}(\pi_k(\cdot \mid s)) \\ &\approx -\frac{1}{\beta} \cdot \text{Cov}_{a\sim\pi_\theta^k(\cdot|s)}\left(\log \pi_k(a \mid s), r(s,a)\right), \end{aligned} \tag{19}$$

where the covariance term Cov tracks the entropy change. **This relationship highlights the core principle of E-E trade-off: balancing the two terms in Eq. 19.**

C.5 Key Conclusions

1. $\text{Cov}[\log \pi_\theta, A] > 0$: $\Delta H < 0$ (entropy ↓, policy more deterministic).
2. $\text{Cov}[\log \pi_\theta, A] < 0$: $\Delta H > 0$ (entropy ↑, policy more exploratory).
3. $\text{Cov}[\log \pi_\theta, A] = 0$: $\Delta H = 0$ (entropy unchanged).

This derivation underpins entropy regularization for balanced E-E trade-off in RL.

D Computation Graph for the Token-Level

We design a carefully structured backward pass to ensure training stability and theoretical consistency, primarily by handling Importance Sampling (IS) ratios at both the sequence-level and token-level.

The computation graph above stabilizes token-level policy updates by aggregating sequence-level Importance Sampling weights, which addresses the uneven impact of entropy and KL-Divergence regularization in CoT reasoning. This design aligns with the established performance-entropy relationship $R = -a \exp(\mathcal{H}) + b$ (Cui et al., 2025): optimizing downstream mathematical reasoning tasks tends to reduce policy entropy (increasing determinism), while artificially pushing entropy higher rarely improves performance, and adding a global KL-Divergence term often degrades stability. The core reason is that in CoT reasoning, token distributions shift dynamically across reasoning steps, so uniform entropy/KL regularization affects tokens unevenly and can lead to training collapse.

E Use of LLMs

Large language models (LLMs), specifically DeepSeek-R1 and GPT-4 Turbo (GPT-5 was not used, as it remains unreleased as of 2026), were employed solely as a writing assistance tool during the preparation of this manuscript. These LLMs were used only to refine the clarity, readability, and presentation of the text—they were not used for any research-critical tasks, including but not limited to: conceiving the research design, developing algorithms, conducting experiments, analyzing data, or interpreting results. The authors bear sole responsibility for the entire research conception, technical direction, scientific content, and interpretation of all experimental results. No LLMs were used to generate or modify experimental data, and all conclusions presented in this work are the authors' independent scientific judgments.