# Where Hindsight Credit Can Reside: A Signed-Capacity View of Token Updates in RLVR

Anonymous ACL submission

## Abstract

Reinforcement Learning with Verifiable Rewards (RLVR) improves the reasoning ability of Large Language Models (LLMs), but sparse outcome rewards make token-level credit assignment difficult. We study token-level credit as a reward-conditioned shift from the behavior policy to a hindsight posterior. In autoregressive RLVR, this shift can be expressed through Conditional Mutual Information (CMI), which shows that token entropy upper-bounds possible hindsight credit. Entropy, however, indicates capacity rather than update direction, so we introduce the Four Quadrant Decomposition to separate updates by reward polarity and token entropy. Controlled interventions show that these two factors jointly shape token updates. Sustained reasoning gains concentrate in signed high-entropy quadrants, whereas low-entropy updates saturate quickly. Based on this analysis, we propose Hindsight-Aware Policy Optimization (HAPO), a sign-preserving modification to GRPO that performs capacity-guided advantage reallocation. Experiments on mathematical reasoning benchmarks in two model settings show that HAPO achieves competitive performance among entropy-aware baselines.

## 1 Introduction

Reinforcement Learning with Verifiable Rewards (RLVR) improves reasoning in Large Language Models (LLMs) by optimizing against outcome rewards from self-generated responses (Guo et al., 2025; Lambert et al., 2024; Cobbe et al., 2021; Hendrycks et al., 2021; Jimenez et al., 2024), bypassing the need for dense step-level supervision (Chu et al., 2025a). In practice, methods such as GRPO (Shao et al., 2024) broadcast a single sequence-level reward to every token in a trajectory. This creates a credit assignment problem: a trajectory's outcome may hinge on a few pivotal decisions, yet most tokens are routine continuations that receive identical credit (Minsky, 1995; Lightman et al., 2024). A central question for RLVR is therefore how outcome feedback should be assigned at the token level.

Recent work studies two observable axes for this problem: **reward polarity**, which redistributes credit by positive and negative samples to decide whether sampled tokens should be reinforced or discouraged (Yuan et al., 2023; Zhu et al., 2025), and **token entropy**, which identifies uncertain tokens where the policy has more room to change (Wang et al., 2025c; Cheng et al., 2025). Each axis captures only part of the problem. Polarity specifies the update direction but lacks within-trajectory selectivity, whereas entropy is the reverse. In other words, high-entropy tokens in successful and failed trajectories require different corrections. This leads to a sharper question—*given the outcome, how should the probabilities of earlier tokens be adjusted relative to the behavior policy?*

We answer this question from a **hindsight** perspective, since the reward is observed only after generation. **Hindsight Credit Assignment** (HCA) formalizes the ideal learning signal as a reward-conditioned posterior-prior shift: increase the probability of a token in proportion to how much conditioning on the final outcome makes it more likely, and decrease it if vice versa (Harutyunyan et al., 2019; Andrychowicz et al., 2017). Direct estimation of this hindsight posterior is infeasible in autoregressive LLMs without counterfactual rollouts or an auxiliary posterior model. We therefore use HCA as an analytical target. Under this view, reward polarity provides a coarse correction sign, and entropy indicates where the behavior policy has enough distributional capacity for a non-trivial correction.

To make this target analyzable, we connect hindsight credit to **Conditional Mutual Information** (CMI). The CMI between the next token and the reward characterizes the expected posterior-prior shift. This quantity remains intractable, but token

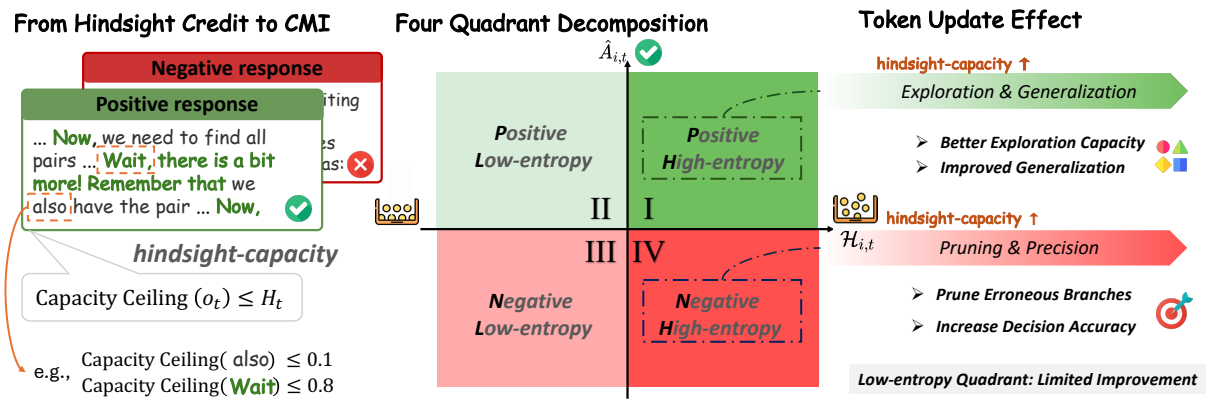



Figure 1: Tokens in **bold** exhibit high entropy. From an information-theoretic view, high-entropy and low-entropy tokens differ in how much hindsight information they can carry. We decompose token updates in RLVR into four quadrants to study token-level credit assignment. See Appendix I for full entropy examples.

entropy upper-bounds it. Entropy therefore acts as a computable ceiling on hindsight capacity: low-entropy tokens can support only small expected hindsight shifts, while high-entropy tokens flag candidate positions for larger distribution shifts. Because this bound is one-sided, we introduce the **Four Quadrant Decomposition** as a diagnostic framework that factorizes token updates along reward polarity and entropy (Figure 1). The diagnostic tests quadrant-isolated variants to locate the useful update effects predicted by this hindsight-capacity view. The interventions show that low-entropy updates can help early optimization but quickly saturate, whereas signed high-entropy updates produce more persistent gains by reinforcing successful uncertain branches and suppressing failed ones.

These diagnostics motivate **Hindsight-Aware Policy Optimization** (**HAPO**), a lightweight modification of GRPO for capacity-guided credit assignment. HAPO down-weights low-entropy updates and reallocates uniform outcome feedback to signed high-entropy positions, where hindsight credit is more likely to reside. On mathematical reasoning and out-of-domain benchmarks across two models, HAPO achieves strong aggregate performance and competitive per-benchmark results. Our contributions are summarized as follows:

- We formulate token-level credit in RLVR from a hindsight perspective and connect it to CMI, establishing entropy as a computable capacity ceiling for possible posterior-prior shifts.
- We introduce the **Four Quadrant Decomposition**, a diagnostic framework that factorizes token updates by reward polarity and token entropy to identify where sustained reasoning improvements concentrate.
- Based on this theoretical and empirical analysis, we propose **HAPO**, a capacity-guided method that achieves competitive performance.

## 2 Preliminaries

### 2.1 Reinforcement Learning with Verifiable Rewards

Formally, we consider an LLM parameterized by $\theta$. Given a dataset of query prompts $Q$, standard RLVR learns a policy $\pi_\theta$ to minimize the objective:

$$\mathcal{L}_{\text{RLVR}}(\theta) = -\mathbb{E}_{q\sim Q, o\sim\pi_\theta(\cdot|q)}[r(o,q)], \quad (1)$$

where $r(o,q)$ is a verifiable reward that evaluates the correctness of the response $o$ given the query $q$. To enhance training stability and efficiency, GRPO samples a group of responses $\{o_i\}_{i=1}^G$ for each query from the old behavior policy $\pi_{\theta_{\text{old}}}$ and optimizes the following surrogate objective loss (Shao et al., 2024; Yu et al., 2025):

$$\mathcal{L}_{\text{GRPO}}(\theta) = -\mathbb{E}_{q\sim Q,\{o_i\}_{i=1}^G\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_{i=1}^G|o_i|}\sum_{i=1}^G\sum_{t=1}^{|o_i|}\min\Big(\rho_{i,t}\cdot\hat{A}_{i,t}, \text{clip}(\rho_{i,t}, 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}})\cdot\hat{A}_{i,t}\Big)\right], \quad (2)$$

where $\hat{A}_{i,t} = \frac{r_i - \text{mean}(\{r_i\}_{i=1}^G)}{\text{std}(\{r_i\}_{i=1}^G)}$ represents the group advantage, normalizing the outcome reward $r_i$. $\rho_{i,t} = \frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})}$ denotes the importance sampling ratio at step $t$ of trajectory $o_i$. $\epsilon_{\text{high/low}}$ indicates the clipping range of importance ratios.

### 2.2 Hindsight Credit Assignment

Hindsight Credit Assignment (HCA) offers a backward view of token updates in RLVR (Harutyunyan et al., 2019). At decoding step $t$, the model observes the prefix-conditioned state $s_{i,t} := (q, o_{i,<t})$ and samples the token $o_{i,t} \sim \pi_\theta(\cdot|s_{i,t})$. HCA conceptualizes learning the reward-conditioned hindsight posterior over the next token via the posterior-prior ratio. This ratio $\rho^{\text{hs}}$ is the ideal HCA target for token-level credit:

$$\rho^{\text{hs}} = \frac{h_{\pi_\theta}(o_{i,t} \mid s_{i,t}, r)}{\pi_\theta(o_{i,t} \mid s_{i,t})}. \tag{3}$$

where $h_{\pi_\theta}$ is the hindsight posterior distribution over the next token. If $\rho^{\text{hs}} > 1$, it makes the sampled token more likely than under the behavior policy $\pi_\theta$, and vice versa. However, precise $h_{\pi_\theta}$ estimation is difficult in high-dimensional autoregressive spaces, as it would require counterfactual future rollouts for alternative next tokens from the same prefix or an additional model for the outcome-conditioned posterior (Poole et al., 2019; Foerster et al., 2018; Tan et al., 2026). We therefore use HCA as an analytical target, and next reinterpret $\rho^{\text{hs}}$ through CMI.

### 2.3 Policy Entropy

Policy entropy quantifies the uncertainty of the model's predictive distribution over the vocabulary $\mathcal{V}$ (Kuhn et al., 2023). Given a query $q$ and a partial trajectory $o_{i,<t}$, the token entropy is defined as the Shannon entropy (Shannon, 1948) of the output distribution at step $t$: $\mathcal{H}_{i,t} = -\sum_{v\in\mathcal{V}} \pi_\theta(v \mid q, o_{i,<t}) \log \pi_\theta(v \mid q, o_{i,<t})$. The policy entropy for a trajectory $o_i$ is approximated as the average token entropy. Prior RLVR methods use entropy to identify uncertain or forking positions and to shape reward signals (Wang et al., 2025c; Cheng et al., 2025; Chen et al., 2025a). While these methods reshape reward, the underlying mechanisms of credit assignment remain underexplored.

## 3 Entropy-Bounded Hindsight Credit

Because the outcome reward is observed only after generation, token-level credit in RLVR is inherently a hindsight question: how would the next-token distribution at a prefix change after conditioning on the final outcome? Here, the HCA posterior-prior shift serves as the analytical target, and we adapt the information-theoretic credit view of Arumugam et al. (2021) to the autoregressive LLM setting.

### 3.1 HCA to CMI

Recall from Section 2.2 that HCA defines the posterior-prior ratio $\rho^{\text{hs}}$, which serves as the ideal but intractable pointwise hindsight credit target. CMI makes this target analyzable by aggregating pointwise hindsight evidence into a distribution-level shift. It measures how much the next-token distribution would change once the trajectory outcome is revealed.

**Proposition 1** (HCA–CMI Connection)**.** *For any prefix state $s_{i,t}$, let $h_{\pi_\theta}(\cdot \mid s_{i,t}, r)$ denote the reward-conditioned hindsight distribution over the random next token $\boldsymbol{o}_{i,t}$. Then*

$$I(\boldsymbol{o}_{i,t};\, r \mid s_{i,t}) = \mathbb{E}\left[\mathrm{KL}(h_{\pi_\theta}(\cdot \mid s_{i,t}, r) \,\|\, \pi_\theta(\cdot \mid s_{i,t}))\right].$$

Proposition 1 instantiates CMI as the expected distribution-level counterpart of the HCA posterior-prior shift. Equivalently, CMI is the expected log HCA ratio over outcomes and hindsight-conditioned token choices, so it aggregates rather than replaces the pointwise target. Exact computation would require the hindsight distribution $h_{\pi_\theta}$, which is not directly available as discussed in Section 2.2. This connection leads to a tractable consequence:

**Proposition 2** (Entropy Capacity Bound)**.** *For any prefix state $s_{i,t}$, the information contribution of the next token to the outcome $r$ satisfies:*

$$I(\boldsymbol{o}_{i,t};\, r|s_{i,t}) \leq \mathcal{H}_{i,t}.$$

That is, $\mathcal{H}_{i,t}$ **upper-bounds the token's reward-relevant information in expectation**. In this sense, entropy is not an estimator of token credit itself, but a computable ceiling on where non-trivial hindsight credit can reside. Low-entropy tokens (syntax, routine algebra, common suffixes) have very low $\mathcal{H}_{i,t}$ and therefore cannot support a large expected hindsight shift. High-entropy positions, by contrast, merely mark candidate branching positions where such a shift is possible. The full derivation and interpretation are in Appendix D.1.

### 3.2 Signed-Capacity Predictions

In standard GRPO, the advantage $\hat{A}_{i,t}$ is constant for all tokens within a trajectory. This broadcasts the same outcome signal to tokens with very different hindsight capacity—routine tokens with little room for reward-conditioned shift and branching positions where such shifts are more plausible. Proposition 2 therefore motivates comparing this

uniform update with the token-level geometry of reasoning trajectories. In this view, the issue is not only that uniform assignment is coarse. It also ignores where a reward-conditioned posterior-prior shift can plausibly reside.

Entropy alone is still unsigned: it identifies where a non-trivial shift is possible, but not whether the sampled token should be reinforced or suppressed. The sign of the group advantage supplies this coarse reward polarity, while entropy supplies the capacity axis. Because this condition is necessary rather than sufficient, we treat the implications below as empirical predictions to be probed by quadrant-restricted interventions before proposing a method. This gives a signed-capacity hypothesis for controlled quadrant-isolated training:

**Prediction 1** (Low-entropy saturation). *Training exclusively on low-entropy tokens delivers at most modest or short-lived gains, since they have little capacity for reward-relevant information* $I(\boldsymbol{o}_{i,t}; r|s_{i,t})$.

**Prediction 2** (High-entropy candidate effects). *Training on high-entropy positions is a plausible source of sustained improvements because it permits larger information* $I(\boldsymbol{o}_{i,t}; r|s_{i,t})$.

**Prediction 3** (Polarity–entropy interaction). *Because the hindsight distribution* $h_{\pi_\theta}$ *differs between successful and failed outcomes, high-entropy updates should exhibit asymmetric effects across reward polarity rather than a uniform response.*

# 4 Four Quadrant Decomposition

Section 3 explains why entropy constrains where hindsight credit can plausibly reside. Controlled interventions then ask where signed GRPO updates produce durable behavioral gains. Building on Zhu et al. (2025), who decompose RLVR into positive and negative paradigms based on reward polarity, we cross reward polarity with token entropy, yielding a four-quadrant experimental framework. By restricting gradient updates to specific quadrants, this framework yields diagnostic readouts of where signed update effects are concentrated without treating the observed metrics as direct measurements of latent token credit.

## 4.1 Quadrant Setup

Prior work links token entropy to exploratory behaviors (Wang et al., 2025c; Cheng et al., 2025), but how tokens of different entropy levels influence model optimization and generalization remains underexplored.

These observations motivate us to decompose token reinforcement in GRPO into four quadrants based on two criteria: (1) **Reward polarity (positive / negative)**: whether a token belongs to a positive or negative trajectory, determined by the outcome reward $r_i$; (2) **Token entropy (high / low)**: whether the token entropy $\mathcal{H}_{i,t}$ is relatively higher or lower than the average entropy of the group trajectories $\{\mathcal{H}_{i,t}\}_{i=1}^{G}$. In this diagnostic, entropy provides the capacity axis, while reward polarity supplies the signed correction direction. Using these criteria, we define four diagnostic quadrants for token reinforcement as shown in Figure 1:

- **PHR**: Positive High-entropy Reinforcement,
- **PLR**: Positive Low-entropy Reinforcement,
- **NHR**: Negative High-entropy Reinforcement,
- **NLR**: Negative Low-entropy Reinforcement,

To investigate the underlying learning dynamics of GRPO, we formulate four corresponding sub-objectives, each restricting gradient updates exclusively to tokens within one quadrant. The diagnostic is instantiated with `Qwen2.5-Math-7B` and evaluated on AIME24 using Avg@32 and Pass@32 as exploitation- and diversity-oriented behavioral readouts. Appendices B and C report the detailed training and evaluation setup.

## 4.2 Findings of Optimization Dynamics

Figure 2 tracks training dynamics across all four quadrants. The key diagnostic findings are:

**Finding 1: Low-entropy updates optimize early but have limited hindsight capacity.** PLR and NLR can accelerate early optimization because they act on segments the model is already confident about. However, their gains saturate quickly and can even become detrimental, as seen most clearly in PLR's entropy collapse and degraded diversity in Figures 2c and 2d. PLR reaches the same Avg@32 as PHR only after $5.5\times$ more steps and trails PHR by 12% on Pass@32. Test-time performance in Appendix E.2 further documents this phenomenon, where PLR underperforms other quadrants and even the base model in Pass@256.

**Finding 2: PHR consolidates successful high-entropy branches.** PHR and PLR both improve training-time solve rate in Figure 2a, but they diverge at evaluation in Figure 2c-2d. PHR continues to improve both Avg@32 and Pass@32, indicating

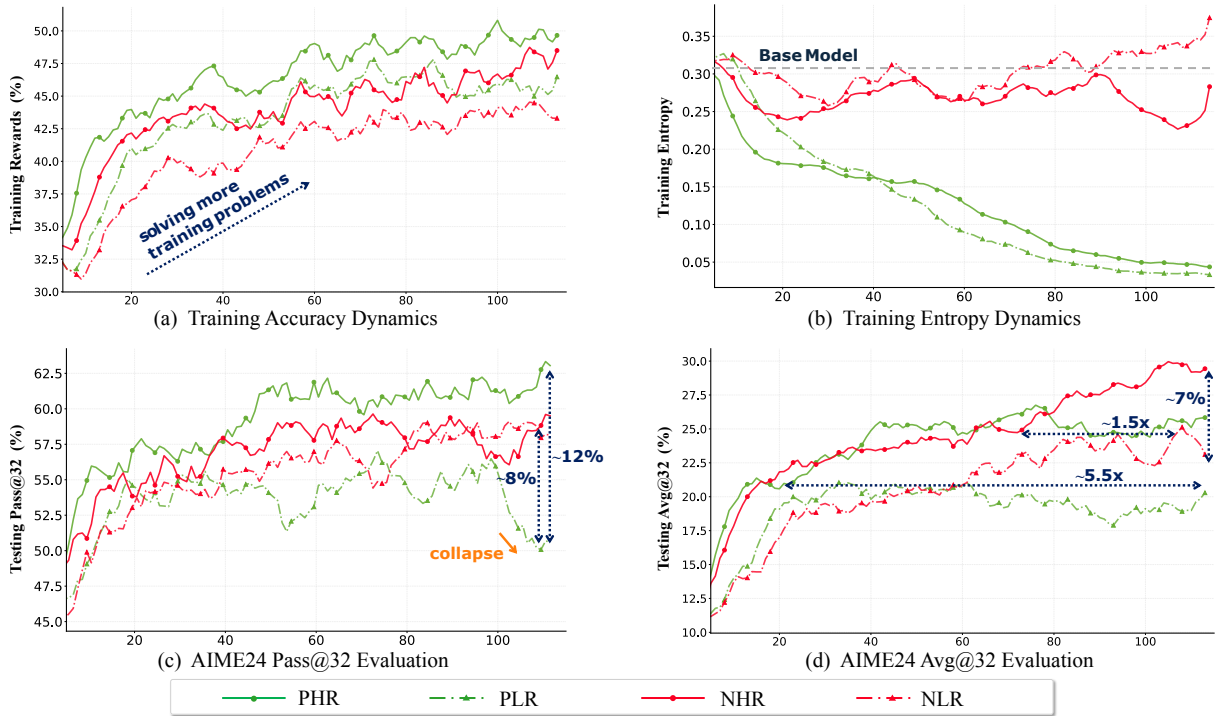



Figure 2: Training dynamics and evaluation for four-quadrant diagnostic interventions. (a) Training solve rate. (b) Training entropy. (c) Pass@32 on AIME24. (d) Avg@32 on AIME24. The x-axis denotes training steps.

that successful high-entropy branches offer reinforcing hindsight evidence that transfers beyond the sampled rollouts. Under the capacity view, these positions are not credited merely because they are uncertain; positive reward polarity supplies the direction, while high entropy leaves enough room for a non-trivial posterior-prior shift.

**Finding 3: NHR prunes failed high-entropy branches.** Both negative quadrants preserve more entropy than positive-quadrant training and improve evaluation metrics, consistent with prior observations that negative reinforcement can be effective in RLVR (Zhu et al., 2025) across Figures 2. The key difference is that NHR converges about $1.5\times$ faster than NLR and attains stronger Avg@32 performance in Figure 2d. Empirically, the benefit of negative feedback is concentrated in NHR rather than NLR, pointing to suppressive or redirecting effects at failed high-entropy branches rather than uniform penalties on routine low-entropy tokens.

**Hindsight-Capacity Readout.** Together, these interventions support the signed-capacity predictions from Section 3.2: low-entropy quadrants (PLR/NLR) behave as low-capacity regions with limited or short-lived gains, while signed high-entropy quadrants (PHR/NHR) show stronger sustained improvements after reward polarity resolves the update direction. Appendix E gives additional training and test-time evidence. We next turn to the gradient level to connect this pattern to GRPO's update rule.

### 4.3 Gradient-Level Analysis of Token Updates

For notational simplicity, we fix a trajectory and drop the trajectory index $i$. Consider the simplified GRPO loss for a sampled token $o_t$ at position $t$. Let $z_v$ denote the logit of token $v \in \mathcal{V}$ and $\pi_{o_t} = \pi_\theta(o_t|q, o_{<t})$. The standard softmax derivation yields the token-level gradient descent (full derivation in Appendix D.2):

$$-\nabla_{z_v}\mathcal{L}_t \approx \begin{cases} \hat{A}_t \cdot (1-\pi_{o_t}), & \text{if } v = o_t \\ -\hat{A}_t \cdot \pi_v, & \text{if } v \neq o_t \end{cases} \tag{4}$$

The critical factor is $(1-\pi_{o_t})$. In expectation, $\mathbb{E}_{o_t\sim\pi}[1-\pi_{o_t}] = 1-\sum_v \pi_v^2$, so less concentrated high-entropy regimes tend to be more gradient-actionable on average. The CMI bound adds a complementary filter: only positions with sufficient entropy can support a sizable reward-conditioned posterior-prior shift. Thus, high-entropy updates are plausible candidate locations for hindsight credit, while low-entropy updates have limited hindsight capacity. By contrast, GRPO still assigns the same advantage to both. This helps explain the

stronger PHR/NHR readouts and the more limited role of PLR/NLR in Section 4, motivating a practical objective that preserves reward polarity while applying bounded, capacity-guided modulation.

## 5 Hindsight-Aware Policy Optimization

The preceding analysis motivates a bounded proxy allocation rule for GRPO. The ideal hindsight correction depends on the reward-conditioned posterior-prior shift, but this shift is unavailable during training. HAPO does not estimate this posterior. Instead, it uses two observable quantities already present in GRPO: reward polarity as the correction sign and token entropy as the capacity proxy. Viewed from the Four Quadrant Decomposition, **Hindsight-Aware Policy Optimization (HAPO)** is a soft relaxation of the diagnostic intervention. Rather than hard-masking one quadrant, it continuously reallocates update magnitude toward signed high-entropy positions while preserving GRPO's optimization structure.

**Entropy-Normalized Capacity Proxy.** Proposition 2 gives an upper bound on reward-relevant token information, not the exact hindsight credit. Accordingly, HAPO uses token entropy as a stable allocation prior for how much of the trajectory-level advantage should be exposed at each position. Let $\mu_{\mathcal{H}}$ and $\sigma_{\mathcal{H}}$ denote the mean and standard deviation of the token entropies $\{\mathcal{H}_{i,t}\}$ within the sampled group. We define

$$C_{i,t} = \mathrm{clip}\left(\frac{\mathcal{H}_{i,t} - \mu_{\mathcal{H}}}{\sigma_{\mathcal{H}}}, -\frac{\left|\hat{A}_{i,t}\right|}{\varphi}, \frac{\left|\hat{A}_{i,t}\right|}{\varphi}\right), \quad (5)$$

where $\varphi > 1$ bounds the capacity proxy, and $C_{i,t}$ is positive for above-average-entropy tokens and negative for below-average ones.

**Sign-Preserving Advantage Shaping.** We then incorporate this proxy into GRPO through a shaped advantage:

$$\hat{A}^{\mathrm{h}}_{i,t} = \hat{A}_{i,t} + \alpha \cdot \mathrm{sign}(\hat{A}_{i,t}) \cdot \mathrm{detach}(C_{i,t}), \quad (6)$$

where $\alpha$ controls the modulation strength and $\mathrm{sign}(\cdot)$ follows the reward polarity. Equivalently, $\hat{A}^{\mathrm{h}}_{i,t} = \mathrm{sign}(\hat{A}_{i,t})(|\hat{A}_{i,t}| + \alpha\,\mathrm{detach}(C_{i,t}))$. Given $\alpha \in (0, 1]$ and $\varphi > 1$, clipping $C_{i,t}$ by $|\hat{A}_{i,t}|/\varphi$ lets the shaped advantage preserves the sign of $\hat{A}_{i,t}$.

The resulting effect mirrors the diagnostic pattern: signed high-entropy updates are strengthened, while routine low-entropy updates are dampened. The $\mathrm{detach}(\cdot)$ operator (i.e., stop-gradient) prevents entropy from becoming a separate optimization target. It only rescales the policy gradient. In practice, we replace $\hat{A}_{i,t}$ in Equation 2 with $\hat{A}^{\mathrm{h}}_{i,t}$ and keep the rest of GRPO unchanged. Hyperparameter sensitivity, including the effect of $\alpha$, is deferred to Appendix F.3.

## 6 Experiments and Analysis

### 6.1 Experiment Setup

**Training.** HAPO is evaluated under two LLM settings: `Qwen2.5-Math-7B` for standard-length 4k math reasoning and `DeepScaleR-1.5B-8k` for long-chain reasoning up to 24k tokens (Yang et al., 2024; Guo et al., 2025). Following prior work (Xu et al., 2025; Luo et al., 2025), both models are trained on the DeepScaleR dataset (Luo et al., 2025). During rollout, decoding uses a temperature of $1.0$ and top-p of $0.95$. For HAPO, the scaling coefficient is $\alpha = 0.2$ and the clipping range is $\varphi = 2$. Full implementation details appear in Appendix B.

**Baselines.** The comparison includes strong baselines discussed in Related Work Appendix A: classic RLVR methods (PRIME (Cui et al., 2025a), W-REINF (Zhu et al., 2025), GRPO (Shao et al., 2024), and DAPO (Yu et al., 2025)) and entropy-aware interventions (EntroReg (Cheng et al., 2025), Forking (Wang et al., 2025c), KL-Cov (Cui et al., 2025b), and EntroAdv (Cheng et al., 2025)). Baseline implementation details appear in Appendix B.4, and Appendix A.5 gives the systematic method comparison.

**Evaluation.** Evaluation covers six widely used reasoning benchmarks: AIME24, AIME25, AMC, MATH (Hendrycks et al., 2021), Minerva (Lewkowycz et al., 2022), and Olympiad (He et al., 2024). Inference uses vLLM (Kwon et al., 2023) with a temperature $0.6$ and top-p $0.95$. To reduce variance in stochastic decoding, we report Avg@32 and Pass@32 for AIME24, AIME25, and AMC, and Avg@4 and Pass@4 for the remaining benchmarks. Appendix C reports additional benchmark statistics and evaluation details.

### 6.2 Main Results

Tables 1 and 2 detail the reasoning performance of LLMs across various benchmarks.

**Performance gains concentrate on hard tasks.** Section 3 frames entropy as a necessary-condition

| Method | AIME24 | | AIME25 | | AMC | | MATH | | Minerva | | Olympiad | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Avg@32 | Pass@32 | Avg@32 | Pass@32 | Avg@32 | Pass@32 | Avg@4 | Pass@4 | Avg@4 | Pass@4 | Avg@4 | Pass@4 | avg |
| ***Classic RLVR Method*** | | | | | | | | | | | | | |
| Base-7B | 10.0 | 39.3 | 5.1 | 23.5 | 32.7 | 85.2 | 46.9 | 66.3 | 14.3 | 25.9 | 15.5 | 27.1 | 20.8 |
| PRIME | 15.3 | 54.5 | 6.1 | 26.3 | 42.4 | 90.5 | 56.6 | 74.1 | 15.7 | 37.8 | 20.8 | 34.7 | 30.1 |
| W-REINF | 31.2 | 58.2 | 10.1 | 34.0 | 58.1 | 87.1 | 76.2 | 84.5 | 34.2 | 51.2 | 38.9 | 49.3 | 41.6 |
| GRPO | 34.0 | 50.8 | 9.3 | 25.2 | 58.6 | 80.3 | 79.9 | 85.1 | 38.1 | 43.4 | 42.8 | 51.1 | 43.8 |
| DAPO | 35.7 | 67.4 | 17.1 | **40.8** | 61.0 | 84.8 | 82.6 | 88.1 | 42.8 | 50.4 | 43.9 | 52.8 | 47.2 |
| ***Policy Entropy Method*** | | | | | | | | | | | | | |
| EntroReg | 34.6 | 52.9 | 12.6 | 28.5 | 61.1 | 84.9 | 82.7 | 87.7 | 43.1 | 50.8 | 42.8 | 51.4 | 46.2 |
| Forking | 35.8 | 58.1 | 15.1 | 36.1 | 63.7 | 86.6 | **84.1** | **89.4** | 43.0 | 50.1 | **45.5** | **54.4** | 47.9 |
| EntroAdv | 36.8 | 65.0 | 16.3 | 35.7 | **63.8** | **89.4** | 83.5 | 88.6 | 42.7 | 49.3 | 44.5 | 53.6 | 47.9 |
| KL-Cov | 38.9 | 61.6 | 13.8 | 33.6 | 59.2 | 85.9 | 81.5 | 87.2 | 40.9 | 54.1 | 44.2 | 51.8 | 46.4 |
| HAPO | **39.8** | **68.9** | **17.2** | 37.6 | 62.1 | 88.8 | 83.7 | 88.7 | **43.9** | **57.3** | 45.1 | 53.8 | **48.6** |

Table 1: Main results in percentage (%) on mathematics reasoning benchmarks with `Qwen2.5-Math-7B`. The best results are in **bold**, and the second-best results are underlined. All subsequent tables follow this format. The `avg` is computed by averaging the Avg@$k$ across all benchmarks.

| Method | AIME24 | | AIME25 | | AMC | | MATH | | Minerva | | Olympiad | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Avg@32 | Pass@32 | Avg@32 | Pass@32 | Avg@32 | Pass@32 | Avg@4 | Pass@4 | Avg@4 | Pass@4 | Avg@4 | Pass@4 | avg |
| ***Classic RLVR Method*** | | | | | | | | | | | | | |
| Base-1.5B | 32.5 | 71.6 | 24.3 | 51.9 | 69.4 | 89.9 | 85.7 | 91.9 | 37.5 | 46.7 | 55.2 | 63.8 | 50.7 |
| W-REINF | 35.0 | 66.1 | 25.0 | 49.3 | 69.8 | 91.0 | 87.5 | 92.6 | 37.8 | 37.8 | 55.9 | 65.2 | 51.8 |
| GRPO | 38.1 | 64.8 | 27.1 | 49.1 | 71.4 | 92.4 | 88.9 | 93.4 | 39.4 | 48.0 | 58.7 | 67.4 | 53.9 |
| DAPO | 40.9 | **75.6** | 28.6 | 53.6 | 73.6 | 90.8 | 89.9 | 94.0 | 39.1 | 48.0 | 58.4 | 67.1 | 55.1 |
| ***Policy Entropy Method*** | | | | | | | | | | | | | |
| EntroReg | 34.6 | 70.4 | 26.4 | 49.2 | 70.2 | 90.6 | 86.8 | 92.5 | 37.6 | 47.5 | 56.2 | 67.0 | 51.9 |
| Forking | 38.1 | 71.7 | 28.1 | **55.5** | 71.0 | 90.7 | 87.6 | 93.1 | 38.7 | 48.6 | 59.1 | 68.4 | 53.7 |
| EntroAdv | 44.0 | 73.9 | **30.4** | 54.5 | 73.9 | 92.9 | 90.2 | 94.5 | 40.4 | 46.2 | 59.1 | 68.9 | 56.3 |
| KL-Cov | 43.9 | 71.8 | 30.1 | 54.9 | 75.0 | 91.0 | 90.0 | 94.2 | **40.5** | **50.2** | 59.9 | 69.0 | 56.5 |
| HAPO | **45.1** | 74.6 | 30.1 | 55.1 | **75.5** | **95.5** | **91.1** | **95.2** | 39.7 | 49.1 | **60.8** | **69.5** | **57.0** |

Table 2: Main results in percentage (%) on mathematics reasoning benchmarks with `DeepScaleR-1.5B-8k`.

signal for where large hindsight shifts can reside, rather than an estimator of token credit itself. Harder reasoning tasks contain more uncertain branch points, giving the signed-capacity proxy more opportunities to correct the uniform reward broadcast in GRPO. Accordingly, HAPO achieves its most significant gains on demanding benchmarks. On `Qwen2.5-Math-7B`, it reaches 39.8 Avg@32 on AIME24 (vs. DAPO 35.7, EntroAdv 36.8) and posts the highest aggregate score of 48.6%. This trend persists on `DeepScaleR-1.5B-8k` with longer reasoning traces (up to 24k tokens), where HAPO secures a 57.0% aggregate despite increased optimization difficulty (Chang et al., 2025). It also maintains competitive performance on partially saturated benchmarks (AMC, MATH), where fewer uncertain branch points make credit assignment a less dominant bottleneck.

**Accuracy and diversity improve jointly.** Our four-quadrant analysis (Section 4) reveals that PHR and NHR interventions improve different parts of the Avg/Pass profile without directly observing latent token credit. Tables 1 and 2 show that HAPO improves the Avg/Pass profile relative to GRPO across the two model settings. In contrast, baselines like DAPO often sacrifice Avg@$k$ to boost Pass@$k$, showing how less targeted update rules can trade accuracy for diversity rather than enhancing both.

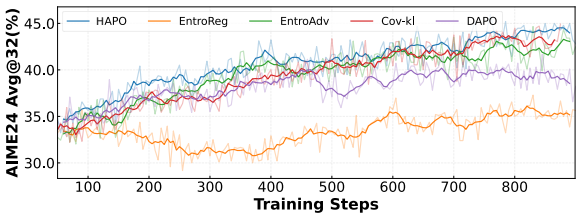
**Comparison with entropy-aware alternatives.** Forking uses a binary high-entropy mask, and EntroAdv adds an entropy bonus. Both surpass GRPO, but their single-dimensional interventions remain suboptimal. Visualizations in Ap-

pendix H illustrate that HAPO applies continuous four-quadrant modulation, using bounded proxy allocation instead of relying on hard cutoffs. Appendix F.1 further shows that HAPO maintains stable entropy throughout training, avoiding the entropy collapse typical of GRPO.

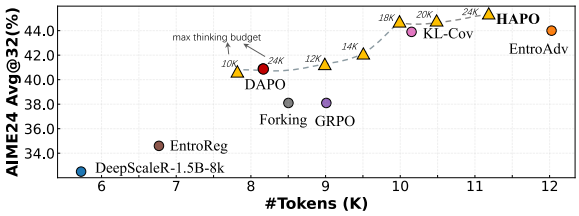
Out-of-domain evaluation on MMLU-Pro mirrors these main results. As detailed in Appendix G, HAPO achieves the best overall accuracy (36.6%), indicating that the bounded proxy allocation may transfer beyond mathematical reasoning.

### 6.3 Efficiency Analysis

Efficiency metrics offer complementary evidence. By concentrating update budget where entropy leaves room for signed hindsight correction, HAPO can improve training and inference efficiency.




(a) Comparison of training sample efficiency




(b) Reasoning token efficiency analysis

Figure 3: Performance–efficiency trade-off for 1.5B models on AIME24. We plot HAPO's Pareto frontier under different max response length budget.

Figure 3 exhibits the same trend on AIME24. During training, HAPO outpaces GRPO and DAPO, highlighting good sample efficiency. At test time, we set the max response length budget to 24K for all baselines. HAPO avoids EntroAdv's over-generation, yielding a better Pareto frontier between accuracy and token usage. Taken together, these improvements are better explained by bounded proxy allocation than by longer reasoning chains alone. Additional efficiency evidence on MATH and Olympiad appears in Appendix F.2.

### 6.4 Ablation Study

This ablation masks the HAPO proxy term for one quadrant (PHR, PLR, NHR, or NLR), reverting those tokens to vanilla GRPO updates, and asks whether the same high-entropy proxy pattern persists. The experiments are averaged over three trials.

| | AIME24/25 | AMC | MATH | Minerva | Olympiad | avg |
|---|---|---|---|---|---|---|
| HAPO | **39.8/17.2** | 62.1 | 83.7 | **43.9** | **45.1** | **48.6** |
| ***w/o* PHR** | 35.6/15.6 | **63.4** | **83.9** | 42.9 | 43.2 | 47.4 |
| ***w/o* PLR** | 39.2/15.8 | 63.1 | 83.2 | 43.1 | 44.5 | 48.1 |
| ***w/o* NLR** | 36.7/17.1 | 61.2 | 83.2 | **43.9** | 43.8 | 47.7 |
| ***w/o* NHR** | 35.7/16.3 | 61.8 | 82.8 | 44.0 | 42.9 | 47.2 |

Table 3: Ablation study of HAPO with 7B model. Avg@32 for AIME24, AIME25, AMC, and Avg@4 for other benchmarks are reported in percentage (%).

- **High-Entropy (PHR & NHR).** *w/o* NHR yields the lowest aggregate score (47.2%), and *w/o* PHR reduces the score to 47.4%. These drops indicate that signed high-entropy modulation contributes the most useful proxy signal in both failed and successful trajectories.
- **Low-Entropy Quadrants (PLR & NLR).** Removing low-entropy components produces only modest changes (e.g., *w/o* PLR causes a $\sim$0.5% drop). This pattern matches the entropy ceiling: low-entropy positions cannot support large $I(\mathbf{o}_{i,t}; r|s_{i,t})$, so removing their modulation changes the objective less.

These ablations mirror the diagnostic pattern in Section 4 and indicate that the useful proxy signal is predominantly concentrated in high-entropy regions. Appendix F.3 reports hyperparameter sensitivity.

## 7 Conclusion

This paper studies token-level credit assignment in RLVR from a hindsight perspective. This CMI adaptation shows that the expected posterior-prior shift induced by the final reward is upper-bounded by token entropy, making entropy a tractable ceiling rather than an exact credit estimator. The Four Quadrant Decomposition and gradient analysis provide diagnostic evidence that sustained gains are more pronounced in signed high-entropy updates, motivating HAPO as a simple, sign-preserving modification to GRPO. Across mathematical reasoning benchmarks and two model settings, HAPO improves accuracy, diversity, and efficiency while preserving the simplicity of GRPO. The polarity-entropy framework can also serve as a diagnostic tool for credit assignment in RLVR beyond the specific instantiation presented here.

## Limitations

We identify several main limitations of our work. First, entropy is a one-sided capacity signal rather than a direct estimator of token-level hindsight credit. HAPO therefore does not estimate the true posterior-prior ratio; it implements bounded, sign-preserving proxy allocation. Similarly, the quadrant decomposition is diagnostic and should be interpreted as intervention evidence rather than direct observation of latent token credit. Second, our method relies on outcome-based verifiable rewards, which are readily available in domains like mathematics. Preliminary results on MMLU-Pro (Appendix G) suggest broader applicability. Third, HAPO introduces additional hyperparameters, $\alpha$ and $\varphi$. Appendix F.3 reports sensitivity analysis for $\alpha$, and our experiments fix $\varphi = 2$.

## Ethical Considerations

This work focuses on enhancing the mathematical reasoning capabilities of Large Language Models through improved reinforcement learning techniques. While our primary goal is to advance the state of automated reasoning, we acknowledge potential ethical implications, such as the consumption of computational resources and energy. We rely exclusively on publicly available datasets and benchmarks, which contain no private or sensitive information. We confirm that our use of these artifacts aligns with their intended use and access conditions. We also strictly adhere to the licensing terms of all utilized models and data. Our use of existing artifacts, including `Qwen2.5-Math-7B`, `DeepScaleR-1.5B-8k`, the mathematical reasoning datasets (AIME24/25, AMC, MATH, Minerva, Olympiad), and MMLU-Pro, falls within their intended use for research and development purposes. To promote transparency and reproducibility, we intend to release our code and derivative artifacts under a permissive open-source license.

## References


Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. 2025. The unreasonable effectiveness of entropy minimization in LLM reasoning. *CoRR*, abs/2505.15134.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. 2017. Hindsight experience replay. *CoRR*, abs/1707.01495.

Dilip Arumugam, Peter Henderson, and Pierre-Luc Bacon. 2021. An information-theoretic perspective on credit assignment in reinforcement learning. *Workshop on Biological and Artificial Reinforcement Learning (NeurIPS 2020)*, abs/2103.06224.

Edward Y. Chang, Yuxuan Tong, Morry Niu, Graham Neubig, and Xiang Yue. 2025. Demystifying long chain-of-thought reasoning in llms. *CoRR*, abs/2502.03373.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. INSIDE: llms' internal states retain the power of hallucination detection. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Xinzhu Chen, Xuesheng Li, Zhongxiang Sun, and Weijie Yu. 2025a. Beyond high-entropy exploration: Correctness-aware low-entropy segment-based advantage shaping for reasoning llms. *arXiv preprint arXiv:2512.00908*.

Zhipeng Chen, Xiaobo Qin, Youbin Wu, Yue Ling, Qinghao Ye, Wayne Xin Zhao, and Guang Shi. 2025b. Pass@k training for adaptively balancing exploration and exploitation of large reasoning models. *Preprint*, arXiv:2508.10751.

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

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4299–4307.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V. Le, Sergey Levine, and Yi Ma. 2025a. SFT memorizes, RL generalizes: A comparative study of foundation model post-training. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*. OpenReview.net.

Xiangxiang Chu, Hailang Huang, Xiao Zhang, Fei Wei, and Yong Wang. 2025b. GPG: A simple and strong reinforcement learning baseline for model reasoning. *CoRR*, abs/2504.02546.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Thomas M. Cover and Joy A. Thomas. 2006. *Elements of information theory (2. ed.)*. Wiley.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, Jiarui Yuan, Huayu Chen, Kaiyan Zhang, Xingtai Lv, Shuo Wang, Yuan Yao, Xu Han, Hao Peng, Yu Cheng, and 4 others. 2025a. Process reinforcement through implicit rewards. *CoRR*, abs/2502.01456.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, Zhiyuan Liu, Hao Peng, Lei Bai, Wanli Ouyang, Yu Cheng, Bowen Zhou, and Ning Ding. 2025b. The entropy mechanism of reinforcement learning for reasoning language models. *CoRR*, abs/2505.22617.

Wenlong Deng, Yi Ren, Muchen Li, Danica J. Sutherland, Xiaoxiao Li, and Christos Thrampoulidis. 2025a. On the effect of negative gradient in group relative deep reinforcement optimization. *CoRR*, abs/2505.18830.

Wenlong Deng, Yi Ren, Yushu Li, Boying Gong, Danica J. Sutherland, Xiaoxiao Li, and Christos Thrampoulidis. 2025b. Token hidden reward: Steering exploration-exploitation in group relative deep reinforcement learning. *CoRR*, abs/2510.03669.

Mehdi Fatemi, Banafsheh Rafiee, Mingjie Tang, and Kartik Talamadupula. 2025. Concise reasoning via reinforcement learning. *CoRR*, abs/2504.05185.

Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2018. Counterfactual multi-agent policy gradients. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 2974–2982. AAAI Press.

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*.

Anna Harutyunyan, Will Dabney, Thomas Mesnard, Mohammad Gheshlaghi Azar, Bilal Piot, Nicolas Heess, Hado van Hasselt, Greg Wayne, Satinder Singh, Doina Precup, and Rémi Munos. 2019. Hindsight credit assignment. In *Advances in Neural Information Processing Systems*, volume 32.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. 2024. Olympiadbench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 3828–3850.

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 *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Andreas Hochlehnert, Hardik Bhatnagar, Vishaal Udandarao, Samuel Albanie, Ameya Prabhu, and Matthias Bethge. 2025. A sober look at progress in language model reasoning: Pitfalls and paths to reproducibility. *CoRR*, abs/2504.07086.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R. Narasimhan. 2024. Swe-bench: Can language models resolve real-world github issues? In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron C. Courville, and Nicolas Le Roux. 2025. Vineppo: Refining credit assignment in RL training of llms. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles, SOSP 2023, Koblenz, Germany, October 23-26, 2023*, pages 611–626.

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*.

Jieh-Sheng Lee. 2025. Instructpatentgpt: training patent language models to follow instructions with human feedback. *Artif. Intell. Law*, 33(3):739–782.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur,

Guy Gur-Ari, and Vedant Misra. 2022. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Li Erran Li, Raluca Ada Popa, and Ion Stoica. 2025. Deepscaler: Surpassing o1-preview with a 1.5b model by scaling rl. *Notion Blog*.

Marvin Minsky. 1995. Steps toward artificial intelligence. *Proceedings of the IRE*, 49:8–30.

Youssef Mroueh. 2025. Reinforcement learning with verifiable rewards: Grpo's effective loss, dynamics, and success amplification. *CoRR*, abs/2503.06639.

Ben Poole, Sherjil Ozair, Aäron van den Oord, Alexander A. Alemi, and George Tucker. 2019. On variational bounds of mutual information. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 5171–5180.

Yi Ren and Danica J. Sutherland. 2025. Learning dynamics of LLM finetuning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*.

Claude E. Shannon. 1948. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27(4):623–656.

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

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2025. Hybridflow: A flexible and efficient RLHF framework. In *Proceedings of the Twentieth European Conference on Computer Systems, EuroSys 2025, Rotterdam, The Netherlands, 30 March 2025 - 3 April 2025*, pages 1279–1297.

Hui-Ze Tan, Xiao-Wen Yang, Hao Chen, Jie-Jing Shao, Yi Wen, Yuteng Shen, Weihong Luo, Xiku Du, Lan-Zhe Guo, and Yu-Feng Li. 2026. Hindsight credit assignment for long-horizon LLM agents. *CoRR*, abs/2603.08754.

Xinyu Tang, Yuliang Zhan, Zhixun Li, Wayne Xin Zhao, Zhenduo Zhang, Zujie Wen, Zhiqiang Zhang, and Jun Zhou. 2025. Rethinking sample polarity in reinforcement learning with verifiable rewards. *CoRR*, abs/2512.21625.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, Chuning Tang, Congcong Wang, Dehao Zhang, Enming Yuan, Enzhe Lu, Fengxiang Tang, Flood Sung, Guangda Wei, Guokun Lai, and 75 others. 2025. Kimi k1.5: Scaling reinforcement learning with llms. *CoRR*, abs/2501.12599.

Hieu Tran, Zonghai Yao, and Hong Yu. 2025. Exploiting tree structure for credit assignment in rl training of llms. *arXiv preprint arXiv:2509.18314*.

Jonathan Uesato, Nate Kushman, Ramana Kumar, H. Francis Song, Noah Y. Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. 2022. Solving math word problems with process- and outcome-based feedback. *CoRR*, abs/2211.14275.

Jiakang Wang, Runze Liu, Fuzheng Zhang, Xiu Li, and Guorui Zhou. 2025a. Stabilizing knowledge, promoting reasoning: Dual-token constraints for RLVR. *CoRR*, abs/2507.15778.

Jiawei Wang, Jiacai Liu, Yuqian Fu, Yingru Li, Xintao Wang, Yuan Lin, Yu Yue, Lin Zhang, Yang Wang, and Ke Wang. 2025b. Harnessing uncertainty: Entropy-modulated policy gradients for long-horizon LLM agents. *CoRR*, abs/2509.09265.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. 2024. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 9426–9439. Association for Computational Linguistics.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. 2025c. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for LLM reasoning. *CoRR*, abs/2506.01939.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2024. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Hongling Xu, Qi Zhu, Heyuan Deng, Jinpeng Li, Lu Hou, Yasheng Wang, Lifeng Shang, Ruifeng Xu, and Fei Mi. 2025. KDRL: post-training reasoning llms via unified knowledge distillation and reinforcement learning. *CoRR*, abs/2506.02208.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang

Ren, and Zhenru Zhang. 2024. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *CoRR*, abs/2409.12122.

Zhihe Yang, Xufang Luo, Zilong Wang, Dongqi Han, Zhiyuan He, Dongsheng Li, and Yunjian Xu. 2025. Do not let low-probability tokens over-dominate in RL for llms. *CoRR*, abs/2505.12929.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, and 16 others. 2025. DAPO: an open-source LLM reinforcement learning system at scale. *CoRR*, abs/2503.14476.

Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. RRHF: rank responses to align language models with human feedback. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. 2025. Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *CoRR*, abs/2503.18892.

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. 2025. The lessons of developing process reward models in mathematical reasoning. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 10495–10516.

Xinyu Zhu, Mengzhou Xia, Zhepei Wei, Wei-Lin Chen, Danqi Chen, and Yu Meng. 2025. The surprising effectiveness of negative reinforcement in llm reasoning. *ArXiv*, abs/2506.01347.

Brian D. Ziebart. 2010. *Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy*. Ph.D. thesis, Carnegie Mellon University, USA.

Brian D. Ziebart, Andrew L. Maas, J. Andrew Bagnell, and Anind K. Dey. 2008. Maximum entropy inverse reinforcement learning. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008*, pages 1433–1438.

# A Related Works

## A.1 Reinforcement Learning with Verifiable Rewards

Reinforcement Learning with Verifiable Rewards (RLVR) has emerged as a scalable and effective paradigm for enhancing the reasoning capabilities of Large Language Models (LLMs) (Fatemi et al., 2025; Zeng et al., 2025; Mroueh, 2025). Unlike Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017; Lee, 2025), which relies on preference models, RLVR utilizes rule-based outcome rewards, making it particularly suitable for domains with objective correctness such as mathematics and coding. Recent successes, including DeepSeek-R1 (Guo et al., 2025) and Kimi 1.5 (Team et al., 2025), demonstrate that RLVR can incentivize long-chain reasoning and improve generalization beyond supervised fine-tuning. However, most RLVR approaches, such as GRPO (Shao et al., 2024) and its variants (Wang et al., 2025c; Chu et al., 2025b; Cui et al., 2025b), typically assign sequence-level rewards uniformly across all tokens. This uniform assignment overlooks the distinct roles tokens play in reasoning chains, exacerbating the credit assignment problem (Minsky, 1995). Recent studies have analyzed token patterns in reasoning (Wang et al., 2025c,a), but the mechanism by which outcome rewards should be allocated across token positions remains underexplored. In this work, we focus on outcome-based RLVR and investigate how sequence-level rewards manifest as token-level update effects.

## A.2 Credit Assignment in Outcome-Based RLVR

The credit assignment problem—identifying which specific steps contribute to a final success or failure—remains a fundamental bottleneck in training long-horizon reasoning LLMs (Uesato et al., 2022; Lightman et al., 2024). Traditional approaches often employ process-based RL with step- or token-level supervision, such as Process Reward Models (PRMs) (Cui et al., 2025a; Wang et al., 2024) and value estimation (Kazemnejad et al., 2025; Tran et al., 2025). However, these methods suffer from high annotation and training costs, as well as susceptibility to reward hacking (Wang et al., 2024; Lightman et al., 2024). Consequently, recent research has shifted towards refining credit assignment within outcome-based RLVR. One line of work dissects the disparate impacts of positive and negative samples (Chen et al., 2025b; Zhu et al., 2025; Zhang et al., 2025), with some arguing that reinforcing negative samples is surprisingly more effective for generalization (Zhu et al., 2025). Another direction differentiates token update magnitude based on entropy (Wang et al., 2025c; Agarwal et al., 2025; Wang et al., 2025b). However, these works yield conflicting views on whether

high-entropy or low-entropy tokens should be prioritized for learning (Wang et al., 2025c; Agarwal et al., 2025). The Four Quadrant Decomposition helps reconcile these perspectives by separating reward polarity from token entropy and offering a diagnostic framework for studying signed token update effects.

### A.3 Information-Theoretic Perspectives

Beyond empirical analyses, a growing body of theoretical work seeks principled foundations for credit assignment in RL. Arumugam et al. (2021) formalized action credit through Conditional Mutual Information (CMI) between actions and returns. It implies that policy entropy upper-bounds the information an action can carry about the outcome. This result, originally developed for general MDPs, suggests a natural interpretation for RLVR: many routine tokens in a reasoning chain are near-deterministic, so the CMI bound rules out large expected reward-conditioned shifts at these positions. This work maps the CMI framework to the RLVR setting (Section 3), deriving signed-capacity predictions that our four-quadrant experiments subsequently probe. Concurrently, Deng et al. (2025b) introduced the Token Hidden Reward framework, which learns an implicit per-token reward signal and found that reward-carrying tokens overlap with high-entropy positions by over 90%—empirical evidence in line with the hindsight-capacity view. Ren and Sutherland (2025) analyzed learning dynamics during fine-tuning and identified a "squeezing effect" at low-entropy positions, matching our gradient analysis showing that the factor $(1 - \pi_{o_t})$ vanishes as confidence increases. These complementary perspectives converge on the same conclusion: entropy is a useful capacity signal for locating where non-trivial hindsight shifts may reside, even though it is not itself an exact credit estimator.

### A.4 The Entropy Interventions Approach in RLVR

Policy entropy has long served as a regularizer to promote exploration in reinforcement learning (Ziebart, 2010; Ziebart et al., 2008). In RLVR, entropy is increasingly used to shape token updates, improving training efficiency and stability. Prior work largely falls into two camps: (1) **Exploration-centric methods**, which amplify gradients at high-entropy branching points (e.g., Forking (Wang et al., 2025c), EntroAdv (Cheng et al., 2025)); and (2) **Stability-centric methods**, which prioritize low-entropy tokens to ground knowledge and prevent model collapse (Wang et al., 2025b; Agarwal et al., 2025; Yang et al., 2025). Our work bridges these entropy interventions. Wang et al. (2025c) reported the "80/20 rule" (a minority of tokens drive reasoning), and our Four Quadrant Decomposition offers one capacity-based interpretation of why this is plausible: the CMI bound (Proposition 2) rules out large expected hindsight shifts at low-entropy positions, so the "important 20%" are candidates whose entropy permits non-trivial reward-conditioned shifts. HAPO does not reject this minority-token phenomenon; instead, it refines the hard cutoff into continuous, bounded proxy allocation, strengthening signed updates at high-entropy forks while conservatively down-weighting low-entropy segments. This yields a capacity-guided proxy allocation mechanism that remains compatible with the standard GRPO objective.

### A.5 Detailed Method Comparison

Table 4 summarizes HAPO alongside concurrent and related entropy-based update allocation methods.

Among these methods, A3PO (Tang et al., 2025) is the most related concurrent work, as it is also polarity-aware. However, the two approaches differ in three key respects. First, A3PO differentiates tokens by their absolute probability (high- vs. low-probability), but HAPO uses group-normalized entropy, which captures relative distributional uncertainty within the sampled group rather than pointwise confidence. Second, A3PO applies adaptive time-decaying modulation that gradually reduces the effect of its token weighting during training. In contrast, HAPO maintains bounded sign-preserving modulation throughout, preserving the entropy-aware signal at all stages. Third, HAPO is motivated by the CMI upper bound (Proposition 2), which supports using entropy as a capacity proxy for reward-conditioned correction.

## B Training Setup

### B.1 Four Quadrant Training Objective

The Four Quadrant Decomposition consists of four components: *Positive High-entropy Reinforcement* (PHR), *Positive Low-entropy Reinforcement* (PLR), *Negative High-entropy Reinforcement* (NHR), and *Negative Low-entropy Reinforcement* (NLR). This decomposition allows us to diagnostically investigate how signed high- and low-entropy token up-

| Method | Token Criterion | Polarity-Aware | CMI-Bound Grounding | Modulation Type | Auxiliary Model |
|---|---|---|---|---|---|
| GRPO | None (uniform) | ✗ | ✗ | Uniform advantage | ✗ |
| Forking | Entropy (top-20%) | ✗ | ✗ | Binary mask | ✗ |
| EntroAdv | Entropy (additive) | ✗ | ✗ | Additive entropy bonus | ✗ |
| KL-Cov | KL divergence | ✗ | ✗ | Continuous | ✗ |
| PRIME | Learned reward | ✗ | ✗ | Token-level reward | ✓ |
| A3PO | Probability (high/low) | ✓ | ✗ | Adaptive time-decaying | ✗ |
| HAPO | Group-normalized entropy | ✓ | CMI bound | Bounded sign-preserving | ✗ |

Table 4: Comparison of token-level update allocation methods. "Polarity-Aware" indicates whether the method applies different modulations to positive vs. negative samples. "CMI-Bound Grounding" denotes whether the design is motivated by the entropy capacity bound used in this work.

dates shape model behavior. Formally, we decompose the GRPO objective into quadrant-restricted subobjectives based on Equation 2 as follows:

$$\begin{aligned}\mathcal{L}_{\text{GRPO}}(\theta) = &-\mathbb{E}_{q\sim Q,\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \\ &\Bigg[\frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\sum_{m\in\{\text{I,II,III,IV}\}}\mathbb{I}_m \min\Big(\rho_{i,t}\cdot\hat{A}_{i,t}, \\ &\text{clip}(\rho_{i,t},1-\epsilon_{\text{low}},1+\epsilon_{\text{high}})\hat{A}_{i,t}\Big)\Bigg] \\ &= \mathcal{L}_{\text{PHR}}(\theta,\mathbb{I}_{\text{I}}) + \mathcal{L}_{\text{PLR}}(\theta,\mathbb{I}_{\text{II}}) + \\ &\mathcal{L}_{\text{NHR}}(\theta,\mathbb{I}_{\text{III}}) + \mathcal{L}_{\text{NLR}}(\theta,\mathbb{I}_{\text{IV}})\end{aligned}$$

where each $\mathbb{I}_m$ is an indicator function that evaluates to 1 if the token belongs to quadrant $m$ and 0 otherwise. The four masks are defined as follows:

$$\mathbb{I}_{\text{I}} = \begin{cases}1, & \text{if } \mathcal{H}_{i,t} \geq \mu_{\mathcal{H}}, r_i \geq 0 \\ 0, & \text{otherwise}\end{cases}$$

$$\mathbb{I}_{\text{II}} = \begin{cases}1, & \text{if } \mathcal{H}_{i,t} < \mu_{\mathcal{H}}, r_i \geq 0 \\ 0, & \text{otherwise}\end{cases}$$

$$\mathbb{I}_{\text{III}} = \begin{cases}1, & \text{if } \mathcal{H}_{i,t} \geq \mu_{\mathcal{H}}, r_i < 0 \\ 0, & \text{otherwise}\end{cases}$$

$$\mathbb{I}_{\text{IV}} = \begin{cases}1, & \text{if } \mathcal{H}_{i,t} < \mu_{\mathcal{H}}, r_i < 0 \\ 0, & \text{otherwise}\end{cases}$$

where $\mu_{\mathcal{H}}$ is the average token entropy over all tokens in the sampled group. In our diagnostic experiments, only the selected quadrant contributes gradients; the decomposition is therefore used to probe where useful signed update effects concentrate, rather than to claim direct observation of true token credit.

### B.2 Quadrant Decomposition Experiments

All experiments use Verl (Sheng et al., 2025) as the RL training framework and run on a single node with 8 NVIDIA A800 GPUs. Negative samples use $r_i = -1$, and positive samples use $r_i = +1$. Thus, the diagnostic masks are defined by the binary outcome reward, while HAPO uses the sign of the group-normalized advantage in its sign-preserving shaping. The training dataset is DeepScaleR, which consists of 40,315 unique math reasoning problem-answer pairs. Additionally, we adopt dynamic sampling and Clip-Higher from DAPO (Yu et al., 2025), as they are commonly used in modern RLVR for efficiency and stability. Although the KL constraint is commonly used to regularize policy updates, the core learning signal still comes from the verifiable reward, so these experiments use GRPO without the KL constraint to isolate the effect of reward learning. For PHR, PLR, NHR, and NLR, configuration details are listed in Table 5. Each diagnostic intervention is run twice, and the reported results are averaged over these runs.

| Config | `Qwen2.5-Math-7B` |
|---|---|
| lr | 1e-6 |
| kl_coef | 0.0 |
| max_prompt_length | 1024 |
| max_response_length | 3072 |
| train_batch_size | 256 |
| ppo_mini_batch_size | 32 |
| clip_ratio_low $\epsilon_{low}$ | 0.20 |
| clip_ratio_high $\epsilon_{high}$ | 0.28 |
| temperature | 1.0 |
| rollout.n $G$ | 8 |
| total_training_steps | 120 |

Table 5: Training hyperparameters for the four-quadrant experiments.

### B.3 Main Experiments

The main experiments use two open-source LLMs, `Qwen2.5-Math-7B`[1] and `DeepScaleR-1.5B-8k`[2].

[1] https://huggingface.co/Qwen/Qwen2.5-Math-7B
[2] https://huggingface.co/cmu-deepscaler-colab/DeepScaleR-8K

`Qwen2.5-Math-7B` is a base model that generates answers directly, without prefacing them with "<think>". For comparison, we also consider `DeepScaleR-1.5B-8k`, a representative model that outputs long chain-of-thought reasoning wrapped between "<think>" and "</think>". It is a length-compressed, fine-tuned variant from the DeepSeek family, offering better training efficiency. For the experiments of HAPO, configuration details are listed in Table 6. The training data comes from DeepScaleR. For `Qwen2.5-Math-7B`, the original prompt template is used. For `DeepScaleR-1.5B-8k`, the suffix "`\nPlease reason step by step, and put your final answer within \boxed{}.`" is appended to the original prompt as the template.

| Config | `Qwen2.5-Math-7B` | `DeepScaleR-1.5B-8k` |
|---|---|---|
| lr | 1e-6 | 1e-6 |
| kl_coef | 0.0 | 0.0 |
| max_prompt_length | 1024 | 2048 |
| max_response_length | 3072 | 24576 |
| train_batch_size | 512 | 64 |
| ppo_mini_batch_size | 32 | 32 |
| clip_ratio_low $\epsilon_{low}$ | 0.20 | 0.20 |
| clip_ratio_high $\epsilon_{high}$ | 0.28 | 0.28 |
| temperature | 1.0 | 1.0 |
| rollout.n $G$ | 8 | 16 |
| total_training_steps | 220 | 900 |

Table 6: Training hyperparameters for HAPO.

### B.4 Experiment Setup for Other Baselines

For reimplemented entropy baselines, we use the same training infrastructure and DAPO-style stability techniques, including Clip-Higher, dynamic sampling, token-level policy gradient loss, and overlong penalty. When a baseline releases model weights (i.e., W-REINF, GRPO, and PRIME), we directly re-evaluate the released weights under our evaluation pipeline. For baselines that require local reproduction, we use the corresponding open-source code and match the reported implementation choices when available. We do not report PRIME on the 1.5B model because it requires training a policy-matched implicit process reward model.

**GRPO** (Shao et al., 2024) eliminates the memory-intensive critic model by optimizing the policy using group-based relative advantages derived from a group of rollouts. For the locally reproduced GRPO baseline, we use the group-based relative advantage objective and set the clipping parameter $\epsilon_{\text{high}}/\epsilon_{\text{low}}$ for the policy objective to 0.2.

**DAPO** (Yu et al., 2025) builds upon and refines the GRPO framework. It addresses entropy collapse in long-chain-of-thought reasoning by introducing techniques like Clip-Higher and dynamically filtering out zero-variance sample groups.

**W-REINF** (Zhu et al., 2025) explicitly up-weights negative samples in the objective function to preserve generation diversity. We set the positive advantage weight to 0.1, effectively prioritizing the gradient signal from incorrect reasoning paths (negative samples) to prevent mode collapse.

**PRIME** (Cui et al., 2025a) enables dense, token-level reward optimization without an explicit reward model by deriving implicit process rewards from the log-likelihood ratio between the policy and a reference model. We directly report the results of the released model weights.

**EntroReg** (Cheng et al., 2025) augments the policy loss with an entropy term $\mathcal{H}(\pi)$ to encourage uniform exploration across the vocabulary, serving as a baseline for exploration effectiveness.

**EntroAdv** (Cheng et al., 2025) modifies the advantages with a clipped, gradient-detached entropy term to incentivize exploration at high-uncertainty positions. The shaped advantage is computed as $A_t^{\text{shaped}} = A_t + \psi(\mathcal{H}_t)$, where $\psi(\mathcal{H}_t) > 0$. The authors' recommended hyperparameters are used.

**Forking** (Wang et al., 2025c) improves training efficiency by restricting policy gradient updates to the top 20% of tokens with the highest entropy ("forking tokens"), which are treated as candidate high-uncertainty positions for useful updates.

**KL-Cov** (Cui et al., 2025b) integrates a coverage-aware penalty based on the Kullback-Leibler divergence, formulated as $\mathcal{L}_{\text{cov}} = \mathbb{E}_{a\sim\pi}[\min(\mathbb{D}_{KL}(\pi_\theta \| \pi_{\text{ref}}), \delta)]$. This keeps exploration within a trust region $\delta$ that covers diverse reasoning paths without diverging into incoherent token sequences.

## C Evaluation Setup

### C.1 Metrics

Accuracy based on greedy decoding can be unreliable (Hochlehnert et al., 2025). To comprehensively evaluate the exploitation and exploration performance of LLMs on mathematical reasoning tasks, we employ two primary metrics: Avg@$k$ and Pass@$k$. During evaluation, the model generates responses at a temperature of 0.6 and a top-p of 0.95. In particular, Pass@$k$ is defined as the fraction of problems for which at least one correct answer is produced in $k$ independent trials. Avg@$k$ estimates the expected correctness of a

sampled response and serves as a robust proxy for single-sample accuracy. They are calculated as:

$$\text{Pass@}k = \mathbb{E}_{q\sim Q}\left[1 - \frac{\binom{n-c}{k}}{\binom{n}{k}}\right] \tag{7}$$

$$\text{Avg@}k = \mathbb{E}_{q\sim Q}\left[\frac{c}{n}\right] \tag{8}$$

where $n$ is the total number of sampled responses used for estimation and $c$ is the number of correct rollouts. The answer verification functions are implemented from Qwen[3] and Math-Verify[4] for accurate evaluation.

### C.2 Information for Test Benchmarks

HAPO and baselines are evaluated on six widely used mathematical reasoning benchmarks: AIME24/25, AMC23[5], MATH (Hendrycks et al., 2021), Minerva (Lewkowycz et al., 2022), and Olympiad (He et al., 2024). The statistics of these benchmarks are summarized in Table 7.

| Benchmark | #Questions | Level |
|---|---|---|
| AIME24 | 30 | Olympiad |
| AIME25 | 30 | Olympiad |
| AMC | 83 | Intermediate |
| MATH | 500 | Advanced |
| Minerva | 272 | Graduate |
| Olympiad | 675 | Olympiad |

Table 7: Dataset statistics.

## D Theoretical Analysis

This appendix gives additional theoretical details for the information-theoretic hindsight analysis introduced in Section 3. It first presents the formal CMI derivation and its implications (Appendix D.1), then analyzes how this manifests in gradient dynamics (Appendix D.2), and finally discusses connections to related theoretical frameworks (Appendix D.3).

### D.1 HCA–CMI Derivation

We formalize the HCA target in RLVR through the lens of CMI, building upon the information-theoretic framework of Arumugam et al. (2021).

[3]https://github.com/QwenLM/Qwen2.5-Math
[4]https://github.com/huggingface/Math-Verify
[5]https://huggingface.co/datasets/AI-MO/aimo-validation-amc

**Notation and Setup.** We map the RLVR token generation process to the standard RL formalism as follows. At each decoding step $t$, the "state" is the context $s_{i,t} := (q, o_{i,<t})$, the "action" is the sampled choice $o_{i,t} \sim \pi_\theta(\cdot|s_{i,t})$, and the "return" is the outcome reward $r$. We use bold $\mathbf{o}_{i,t}$ to denote the random next token in comparison with the observed token $o_{i,t}$. The policy entropy at step $t$, $\mathcal{H}_{i,t} = H(\mathbf{o}_{i,t}|s_{i,t}) = -\sum_{v\in\mathcal{V}} \pi_\theta(v|s_{i,t}) \log \pi_\theta(v|s_{i,t})$, quantifies the model's uncertainty over the next token. For HCA analysis, $\pi_\theta$ denotes the behavior policy at rollout time; in GRPO this corresponds to $\pi_{\theta_{\text{old}}}$.

**Proposition 1 (Hindsight-CMI Connection).** *For any prefix state $s_{i,t} = (q, o_{i,<t})$, let $h_{\pi_\theta}(\cdot \mid s_{i,t}, r)$ denote the reward-conditioned hindsight distribution over the next token. Then*

$$\begin{aligned} &I(\mathbf{o}_{i,t}; r|s_{i,t}) \\ &= \mathbb{E}_{r|s_{i,t}}\left[\mathrm{KL}(h_{\pi_\theta}(\cdot|s_{i,t}, r) \,\|\, \pi_\theta(\cdot|s_{i,t}))\right]. \end{aligned}$$

*Proof.* Following HCA's future-conditional distribution definition (Harutyunyan et al., 2019), $h_{\pi_\theta}$ is the posterior over the earlier token after observing the future outcome. By Bayes' rule, its posterior-prior ratio satisfies

$$\begin{aligned} \rho^{\text{hs}}(v, s_{i,t}, r) &:= \frac{h_{\pi_\theta}(v \mid s_{i,t}, r)}{\pi_\theta(v \mid s_{i,t})} \\ &= \frac{p(r \mid s_{i,t}, \mathbf{o}_{i,t} = v)}{p(r \mid s_{i,t})}. \end{aligned}$$

The KL form of CMI gives (Arumugam et al., 2021)

$$\begin{aligned} &I(\mathbf{o}_{i,t}; r|s_{i,t}) \\ &= \mathbb{E}_{r|s_{i,t}}\left[\mathrm{KL}(p(\mathbf{o}_{i,t} \mid s_{i,t}, r) \,\|\, p(\mathbf{o}_{i,t} \mid s_{i,t}))\right]. \end{aligned}$$

Under the HCA notation, this future-conditioned posterior is $h_{\pi_\theta}(\cdot \mid s_{i,t}, r)$, while the prior over the next token is the behavior policy $\pi_\theta(\cdot \mid s_{i,t})$; this gives the proposition. Expanding the KL term then gives the log-ratio form:

$$\begin{aligned} I(\mathbf{o}_{i,t}; r|s_{i,t}) &= \mathbb{E}_{r|s_{i,t}} \mathbb{E}_{v\sim h_{\pi_\theta}(\cdot|s_{i,t},r)}\left[\log \frac{h_{\pi_\theta}(v \mid s_{i,t}, r)}{\pi_\theta(v \mid s_{i,t})}\right] \\ &= \mathbb{E}_{r|s_{i,t}} \mathbb{E}_{v\sim h_{\pi_\theta}(\cdot|s_{i,t},r)}\left[\log \rho^{\text{hs}}(v, s_{i,t}, r)\right]. \end{aligned}$$

Therefore, CMI is the expectation of the log HCA ratio over outcomes and hindsight-conditioned token choices, establishing it as a distribution-level measure of the expected posterior-prior shift. □

**Interpretation.** Proposition 1 shows that CMI is the outcome-weighted KL shift from the behavior policy to the hindsight distribution. For a sampled token $o_{i,t}$ with observed outcome $r$, the corresponding pointwise hindsight evidence is the log posterior-prior ratio

$$c^{\text{hs}} = \log \frac{h_{\pi_\theta}(o_{i,t} \mid s_{i,t}, r)}{\pi_\theta(o_{i,t}|s_{i,t})} = \log \frac{p(r|s_{i,t}, o_{i,t})}{p(r|s_{i,t})}.$$

This pointwise evidence is useful for interpretation, but Proposition 1 and Proposition 2 bound only its expectation through the CMI, not every realized log-ratio. GRPO does not estimate this hindsight distribution or its KL shift. Instead, it broadcasts a coarse outcome-conditioned scalar $\hat{A}_{i,t}$ to all tokens in the trajectory.

**Proposition 2 (Entropy Capacity Bound).** *For any prefix state $s_{i,t} = (q, o_{i,<t})$, the information contribution of the next-token choice to the outcome $r$ satisfies:*

$$I(\mathbf{o}_{i,t};\, r|s_{i,t}) \leq \mathcal{H}_{i,t}.$$

*Proof.* Starting from Proposition 1, write $h_r(v) = h_{\pi_\theta}(v \mid s_{i,t}, r)$ and $\pi(v) = \pi_\theta(v \mid s_{i,t})$. Expanding the hindsight-KL form gives

$$\begin{aligned} I(\mathbf{o}_{i,t};\, r|s_{i,t}) &= \mathbb{E}_{r|s_{i,t}}\left[\mathrm{KL}(h_r \,\|\, \pi)\right] \\ &= \mathbb{E}_{r|s_{i,t}} \sum_{v\in\mathcal{V}} h_r(v) \log \frac{h_r(v)}{\pi(v)}. \end{aligned}$$

Let $m(v) = \sum_r p(r \mid s_{i,t}) h_r(v)$. Separating the two log terms gives

$$\begin{aligned} I(\mathbf{o}_{i,t};\, r|s_{i,t}) = &-H(\mathbf{o}_{i,t} \mid r, s_{i,t}) \\ &- \sum_{v\in\mathcal{V}} m(v) \log \pi(v). \end{aligned}$$

By marginalization,

$$\begin{aligned} m(v) &= p(\mathbf{o}_{i,t} = v \mid s_{i,t}) \\ &= \pi_\theta(v \mid s_{i,t}). \end{aligned}$$

Therefore,

$$\begin{aligned} I(\mathbf{o}_{i,t};\, r|s_{i,t}) &= H(\mathbf{o}_{i,t} \mid s_{i,t}) - H(\mathbf{o}_{i,t} \mid r, s_{i,t}) \\ &= \mathcal{H}_{i,t} - \underbrace{H(\mathbf{o}_{i,t} \mid r, s_{i,t})}_{\geq 0} \leq \mathcal{H}_{i,t}. \end{aligned}$$

The inequality follows from the non-negativity of conditional entropy (Cover and Thomas, 2006). Equality holds if and only if the random next token $\mathbf{o}_{i,t}$ is a deterministic function of $(r, s_{i,t})$—that is, knowing both the context and the final outcome fully determines the token choice. □

**Interpretation for GRPO.** Proposition 2 establishes that policy entropy $\mathcal{H}_{i,t}$ upper-bounds the CMI, the expected reward-conditioned distributional shift at a token position. Low-entropy tokens therefore leave little room for such a shift, whereas high-entropy tokens provide a larger capacity ceiling. This bound does not imply that high-entropy tokens necessarily carry credit; it only marks where a meaningful hindsight correction is possible. HAPO uses entropy as this training-free capacity proxy.

**Diagnostic Readout: Entropy–Reward Association.** Exact computation of $I(\mathbf{o}_{i,t}; r|s_{i,t})$ requires the hindsight distribution $h_{\pi_\theta}$, but a diagnostic statistic can be computed from the training data. Specifically, we measure the mutual information between the token entropy value $\mathcal{H}_{i,t}$ and the binary reward $r_i$:

$$\widehat{I}(\mathcal{H}_{i,t};\, r_i) = H(\mathcal{H}_{i,t}) - H(\mathcal{H}_{i,t}|r_i).$$

Note that $\mathcal{H}_{i,t} = H(\mathbf{o}_{i,t}|s_{i,t})$ is a deterministic function of the state $s_{i,t}$ alone (it depends on the policy distribution, not on the sampled token). Therefore, $\widehat{I}(\mathcal{H}_{i,t}; r_i)$ is not an estimator of the action-level CMI $I(\mathbf{o}_{i,t}; r|s_{i,t})$; it is a dataset-level diagnostic of whether entropy regimes are associated with different outcomes. Since $I(\mathbf{o}_{i,t}; r|s_{i,t}) \leq \mathcal{H}_{i,t}$ (Proposition 2), low-entropy positions rule out large CMI regardless of the observed reward. Conversely, large $\widehat{I}$ signals that entropy differs across reward outcomes, indicating regimes where the entropy ceiling is high enough to permit non-trivial correction. The diagnostic therefore serves as a *necessary-condition detector*: it identifies positions where large hindsight shifts *cannot* reside (low entropy), while flagging high-entropy positions as *candidates* for non-trivial correction. This statistic can be directly computed from the entropy distributions of positive and negative samples (Figure 4).

The entropy distribution in Figure 4 shows that in the *low-entropy regime*, the distributions for positive and negative tokens are nearly identical, implying $H(\mathcal{H}_{i,t}|r_i) \approx H(\mathcal{H}_{i,t})$ and thus $\widehat{I} \approx 0$. In the *high-entropy tail*, a significant divergence emerges—negative samples exhibit higher density at high entropy—yielding $\widehat{I} \gg 0$. This empirically supports the theoretical prediction: low-entropy tokens have limited capacity for reward-conditioned information, while high-entropy tokens are candidate locations where non-trivial correction may

reside.

### D.2 From Information Theory to Gradient Dynamics

*Note on concurrent work.* The gradient derivation below follows a standard analysis of the softmax-parameterized policy gradient. A similar one-dimensional (reward-polarity) gradient analysis appears concurrently in Zhu et al. (2025) and Deng et al. (2025a). Our contribution is the two-dimensional extension along both reward polarity *and* token entropy, which yields the four-quadrant interpretation, explains how GRPO's group-normalized advantage treats low-capacity and candidate high-capacity positions uniformly, and connects the resulting behavior to the CMI bound (Proposition 2). A summary of the key result is presented in Section 4.3.

The CMI bound (Proposition 2) explains *why* entropy bounds possible expected hindsight shifts. The following analysis shows *how* this manifests in the gradient dynamics of GRPO: the gradient magnitude for the sampled token is modulated by the confidence term $(1 - \pi_{o_t})$, which is aligned in expectation with the entropy-based capacity view.

To investigate the token-level gradient dynamics, let $z_v$ denote the logit of a token $v$ in the vocabulary $\mathcal{V}$. The probability of generating token $v$ at step $t$, denoted as $\pi_v := \pi_\theta(v|q, o_{<t})$, is given by the softmax function:

$$\pi_v = \frac{e^{z_v}}{\sum_{k\in\mathcal{V}} e^{z_k}}. \tag{9}$$

Consider the simplified GRPO loss for a sampled token $o_t$ at step $t$. Omitting the clipping operation and the normalization constant $\frac{1}{\sum_{i=1}^{G}|o_i|}$ focuses on the core gradient dynamics. The simplified objective is:

$$\mathcal{L}_t(\theta) = -\mathbb{E}\left[\frac{\pi_\theta(o_t|q, o_{<t})}{\pi_{\text{old}}(o_t|q, o_{<t})}\hat{A}_t\right]. \tag{10}$$

The gradient of this loss with respect to the logit $z_v$ can be derived using the chain rule:

$$\frac{\partial \mathcal{L}_t}{\partial z_v} = \frac{\partial \mathcal{L}_t}{\partial \pi_{o_t}} \cdot \frac{\partial \pi_{o_t}}{\partial z_v}. \tag{11}$$

First, the derivative of the linear importance sampling ratio with respect to the probability $\pi_{o_t}$ is:

$$\frac{\partial \mathcal{L}_t}{\partial \pi_{o_t}} = -\frac{\hat{A}_t}{\pi_{\text{old}}(o_t)}. \tag{12}$$

Second, the derivative of the softmax function $\pi_{o_t}$ with respect to the logit $z_v$ is given by:

$$\begin{aligned}\frac{\partial \pi_{o_t}}{\partial z_v} &= \frac{\mathbb{I}(v = o_t)e^{z_{o_t}} \sum_{k\in\mathcal{V}} e^{z_k} - e^{z_{o_t}} e^{z_v}}{\left(\sum_{k\in\mathcal{V}} e^{z_k}\right)^2} \\ &= \frac{e^{z_{o_t}} \cdot \left(\mathbb{I}(v = o_t)\sum_{k\in\mathcal{V}} e^{z_k} - e^{z_v}\right)}{\left(\sum_{k\in\mathcal{V}} e^{z_k}\right)\cdot\left(\sum_{k\in\mathcal{V}} e^{z_k}\right)} \\ &= \pi_{o_t}(\mathbb{I}(v = o_t) - \pi_v)\end{aligned} \tag{13}$$

where $\mathbb{I}(v = o_t)$ is the indicator function.

Combining these terms, we obtain the gradient with respect to the logit $z_v$:

$$\frac{\partial \mathcal{L}_t}{\partial z_v} = -\frac{\hat{A}_t}{\pi_{\text{old}}(o_t)} \cdot \pi_{o_t}(\mathbb{I}(v = o_t) - \pi_v). \tag{14}$$

Under the standard assumption that the policy has not diverged significantly from the old policy (i.e., $\pi_\theta \approx \pi_{old}$), the ratio $\frac{\pi_{o_t}}{\pi_{old}(o_t)} \approx 1$. For example, $\frac{\pi_{o_t}}{\pi_{old}(o_t)}$ is between $[0.8, 1.28]$ in this paper. This yields the final gradient descent form:

$$\begin{aligned}-\nabla_{z_v}\mathcal{L}_t &\approx \hat{A}_t(\mathbb{I}(v = o_t) - \pi_v) \\ &= \begin{cases}\hat{A}_t \cdot (1 - \pi_{o_t}), & \text{if } v = o_t \\ -\hat{A}_t \cdot \pi_v, & \text{if } v \neq o_t\end{cases}\end{aligned} \tag{15}$$

where $\mathbb{I}(v = o_t)$ is an indicator function that equals 1 if token $v$ is the sampled token $o_t$, and 0 otherwise. Suppose a learning rate of $\eta$, the logit update for token $v$ is:

$$z_v \leftarrow z_v - \eta \cdot \nabla_{z_v}\mathcal{L}_t \tag{16}$$

This derivation reveals how sampled-token probability, which is correlated with entropy in expectation, modulates the gradient update:

**High-Entropy Quadrants (PHR & NHR).** In expectation, high-entropy regimes tend to place lower probability on sampled tokens than near-deterministic regimes. Consequently, the scaling factor $(1 - \pi_{o_t})$ tends to be larger on average, increasing the expected gradient magnitude.

- **PHR ($\hat{A}_t > 0$):** The model receives a positive-polarity update to increase the probability of the sampled token $o_t$ while decreasing the logits of all other tokens following Equation 16. This update can consolidate a correct branch that was previously uncertain, sharpening the local output distribution. This helps explain why PHR improves both accuracy (Avg@$k$) and diversity (Pass@$k$), as shown in Figure 2 and 5.

- **NHR ($\hat{A}_t < 0$):** The model receives a negative-polarity update. For the sampled token $o_t$, the logit is decreased. Crucially, for unsampled tokens $v \neq o_t$, the gradient update is positive because $-\nabla_{z_v}\mathcal{L}_t \propto -\hat{A}_t\pi_v$ (where $\hat{A}_t$ is negative). This increases the logits of unsampled tokens $z_v$ in Equation 16, redistributing probability mass to other plausible candidates and pruning erroneous branches. Figure 2 shows that NHR not only encourages diversity (Pass@32) but also improves accuracy (Avg@32), as the model becomes less likely to diverge into incorrect reasoning paths.

**Low-Entropy Quadrants (PLR & NLR).** Low entropy implies a peaked distribution where the model is confident in its choice ($\pi_{o_t} \approx 1$). Thus, the scaling gradient factor $(1 - \pi_{o_t})$ is relatively small.

- **PLR ($\hat{A}_t > 0$):** Since the model is already confident and correct, the output distribution is sharp. Further reinforcement has limited room to induce reward-conditioned redistribution and can instead encourage overconfidence or entropy collapse. This explains the stagnation of PLR performance observed in our experiments.

- **NLR ($\hat{A}_t < 0$):** Although the gradient magnitude is small, any non-zero update penalizes a token the model is confident in. In negative trajectories, low-entropy tokens often represent syntactically correct structures or robust logical transitions (e.g., "The answer is") rather than the root cause of the error. Penalizing these tokens with a negative trajectory-level advantage introduces noise, potentially degrading the model's linguistic capabilities without correcting the actual reasoning flaw. This accounts for why NLR is less effective than NHR in Figure 2.

### D.3 Connections to Related Theoretical Frameworks

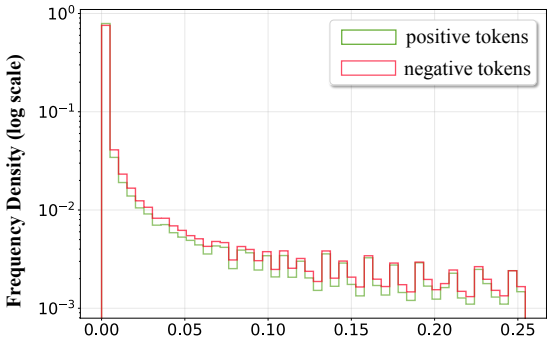
The CMI framework developed above offers a common capacity-based lens for interpreting several recent empirical findings.

**Negative Reinforcement Effectiveness.** Zhu et al. (2025) demonstrated the "surprising effectiveness" of negative reinforcement in LLM reasoning, showing that high-entropy negative-polarity updates can deliver strong learning effects. The CMI framework offers a capacity-based interpretation: at high-entropy positions where $\mathcal{H}_{i,t}$ is large, the bound $I(\mathbf{o}_{i,t}; r|s_{i,t}) \leq \mathcal{H}_{i,t}$ permits non-trivial reward-conditioned correction. Negative feedback at these positions (NHR) can prune erroneous branches by redistributing probability mass away from incorrect tokens in the observed interventions, as shown in the gradient analysis (Equation 15). Conversely, negative feedback at low-entropy positions (NLR) targets tokens with limited capacity under the CMI bound, explaining why it can introduce noise rather than useful updates. Recent analysis of negative gradient dynamics (Deng et al., 2025a) further corroborates this view, showing that the gradient contribution of negative samples is concentrated on tokens where the policy is most uncertain—matching the high-capacity regime identified by Proposition 2.

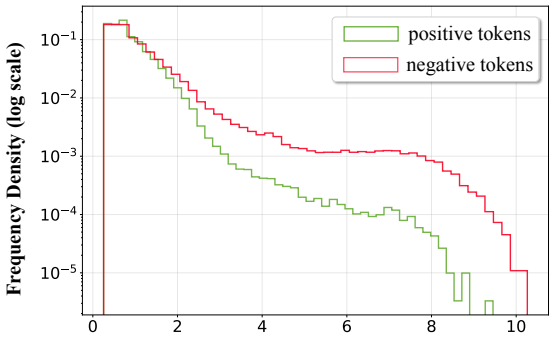
**Token Hidden Reward and Implicit Token Scoring.** The Token Hidden Reward (THR) framework (Deng et al., 2025b) identifies "reward-carrying" tokens through an importance-based scoring mechanism. Remarkably, the set of tokens identified as high-THR overlaps with high-entropy tokens by over 90% (Deng et al., 2025b). Our CMI analysis contextualizes this empirical coincidence: because the CMI bound $I(\mathbf{o}_{i,t}; r|s_{i,t}) \leq \mathcal{H}_{i,t}$ rules out large expected shifts at low-entropy positions, high-entropy positions remain candidates *capable* of supporting substantial reward-conditioned information. THR's learned scoring function implicitly discovers positions that often overlap with high-entropy regions; HAPO uses a training-free entropy-based proxy to target the same candidate regime without the overhead of learning an auxiliary reward model.

**Learning Dynamics and Entropy Collapse.** Studies on the learning dynamics of LLM fine-tuning (Ren and Sutherland, 2025) have identified a "squeezing effect" whereby PLR training progressively collapses the entropy of already-confident tokens, ultimately reducing the model's exploration capacity. This phenomenon is consistent with the gradient structure in the low-entropy regime: as $\pi_{o_t} \to 1$, the gradient factor $(1 - \pi_{o_t}) \to 0$, creating a self-reinforcing pattern where confident tokens receive diminishing updates but remain in a peaked distribution. From the CMI perspective, PLR operates in a regime where $\mathcal{H}_{i,t} \approx 0$, so the CMI upper bound implies near-zero capacity for reward-conditioned shifts regardless of the applied advantage magnitude. This helps explain why PLR exhibits stagnating performance in our

four-quadrant experiments (Figure 2): optimization effort is spent on tokens that have limited capacity for hindsight correction.




(a) Entropy Value in Low-entropy Quadrants (75%)




(b) Entropy Value in High-entropy Quadrants (25%)

Figure 4: Token entropy distribution. Low-entropy distributions remain aligned, but a sharp divergence emerges between positive and negative samples in the high-entropy quadrant.

# E Decomposition Training Results

## E.1 Training Entropy Distribution

Figure 4 visualizes the entropy distribution of tokens in positive and negative training rollouts. Only a minority of tokens (approximately 25%) exhibit high entropy. In the low-entropy regime, the distributions for positive and negative tokens are nearly identical. This overlap indicates that low-entropy tokens typically correspond to deterministic reasoning steps (e.g., syntax or factual recall) that remain stable regardless of the final outcome. However, a significant divergence emerges in the high-entropy tail, where the density of negative tokens is consistently higher than that of positive tokens. This pattern points to high-uncertainty states being more often associated with divergent or erroneous reasoning paths. These empirical findings support the targeted, signed proxy strategy used by HAPO: rather than assigning the same update to all tokens in a trajectory, the objective consolidates successful high-entropy branches and redirects failed high-entropy branches.

## E.2 Test-time Compute Performance

Evaluation uses the final optimization checkpoints for PHR, NHR, PLR, and NLR across varying Pass@$k$ values with decoding temperature set 1.0. In summary, PHR and NHR achieve robust performance across the full range of $k$. The test-time compute results on AMC, MATH, and AIME24 (Figure 5) support the following observations:

- Across the Pass@$k$ spectrum ($k = 1$ to 256), PHR and NHR generally outperform both the Base model and the low-entropy variants. This supports the view that signed updates at high-uncertainty positions are the main observed locus of sustained reasoning improvements in these diagnostics.

- PLR shows initial gains over the Base model at low $k$ values (e.g., $k = 1, 2$), but its performance scaling is significantly flatter. In AIME24, the Base model eventually surpasses PLR as $k$ increases. This trend is even more pronounced in less challenging datasets such as AMC and MATH, indicating that reinforcing deterministic, low-entropy tokens leads to over-exploitation and reduces output diversity.

- In line with Zhu et al. (2025), NHR alone is remarkably effective: it substantially outperforms NLR across the full Pass@$k$ spectrum and approaches the performance of the complete GRPO baseline at high $k$. This pattern indicates that penalizing incorrect high-entropy tokens can prune erroneous reasoning branches and redistribute probability mass toward plausible alternatives, providing a more targeted update than penalizing low-entropy tokens.

- NLR-only training yields lower scores at low Pass@$k$ (e.g., $k = 1, 2$) compared to NHR. This indicates that penalizing low-entropy tokens in incorrect responses provides a noisy or redundant update. These tokens (e.g., mathematical syntax such as `\frac` or common suffixes) are typically deterministic and correct even in failed trajectories. Penalizing them risks destabilizing the model's linguistic capabilities without correcting the underlying logical errors.

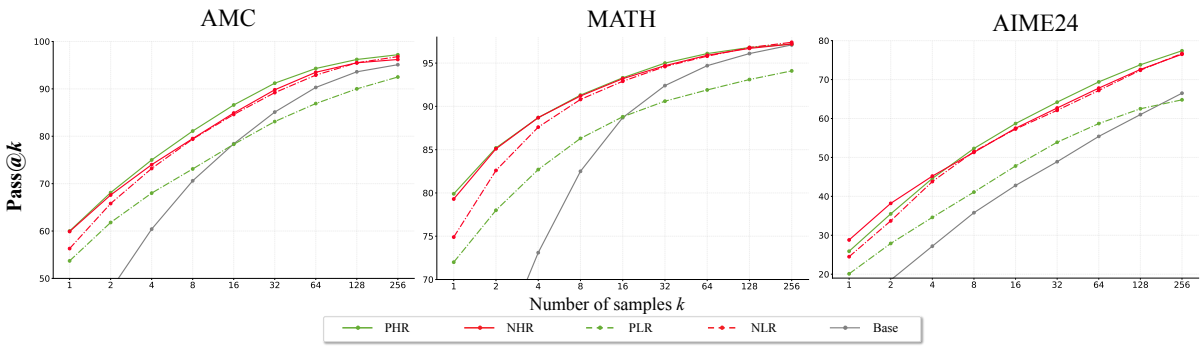
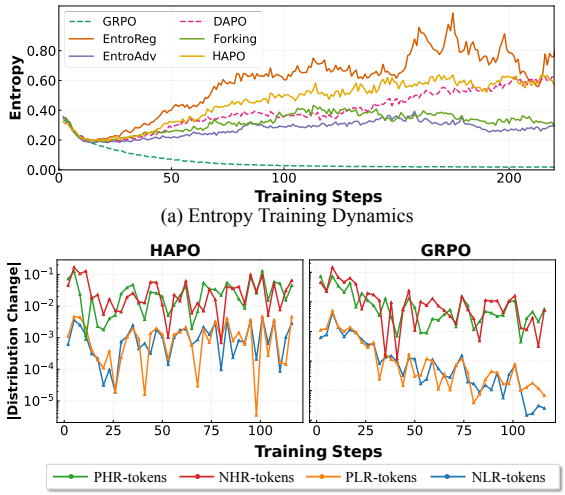



Figure 5: Test time compute performance of four-quadrant training objectives. The x-axis represents varying $k \in \{1, 2, 4, 8, 16, 32, 64, 128, 256\}$ values, covering the full spectrum of Pass@$k$.




(a) Entropy Training Dynamics

(b) Distribution change of HAPO and GRPO in four-quadrant tokens

Figure 6: Entropy dynamics analysis of 7B models.

# F Additional Analyses

## F.1 Entropy Dynamics of HAPO

Figure 6 shows that HAPO maintains a stable entropy trajectory throughout training. In contrast, GRPO suffers from entropy collapse, which constrains exploratory capacity, while EntroReg tends to over-flatten the output distribution. The analysis also tracks absolute entropy shifts across the four quadrants. The largest observed entropy shifts occur in the high-entropy quadrants (PHR and NHR), with low-entropy quadrants remaining comparatively stable. This pattern accords with HAPO concentrating larger update effects on uncertain segments while preserving confidence in established predictions.

## F.2 Efficiency Analysis

If HAPO indeed concentrates updates on consequential decision points, it should reach a better accuracy–cost frontier than methods that rely on uniform advantage assignment or longer generations. Figure 3 in the main text gives the AIME24 view: HAPO converges more smoothly than GRPO and DAPO and avoids the over-generation pattern of EntroAdv, yielding a better accuracy–cost trade-off. Table 8 extends the same comparison to MATH and Olympiad. Across both benchmarks, HAPO achieves the highest accuracy among all methods. Compared to GRPO, DAPO, and Forking, it attains meaningfully higher Avg@4 at a moderate increase in generation length; compared to EntroAdv, it reaches comparable or better accuracy with fewer tokens, indicating a more favorable accuracy–cost trade-off.

## F.3 Impact of Coefficient $\alpha$

Table 9 examines the sensitivity of HAPO to the hyperparameter $\alpha$. Figure 7 shows that training remains robust for $\alpha \leq 0.4$; excessively large values slow convergence and degrade benchmark performance. Across both the trajectory plots and the benchmark summary, $\alpha = 0.2$ gives the strongest overall trade-off among the tested values, so it is the default setting in the main experiments. The clipping range remains fixed at $\varphi = 2$ throughout these experiments, and a full $\varphi$ sweep is left to future work.

# G MMLU-Pro Results

Table 10 presents the per-domain results on MMLU-Pro. Among the compared models, HAPO achieves the highest overall accuracy (36.6%), with particularly strong gains in reasoning-intensive domains such as economics (+7.7% over GRPO), biology (+11.4%), and psychology (+5.1%). Domains where performance is comparable to GRPO (e.g., health, chemistry) may rely more on factual retrieval than multi-step reasoning, in line with the

| Method | MATH Avg@4 | MATH Tokens | Olympiad Avg@4 | Olympiad Tokens |
|---|---|---|---|---|
| GRPO | 88.9 | 3172 | 58.7 | 5721 |
| DAPO | 89.9 | 2690 | 58.4 | 5113 |
| Forking | 87.6 | 2769 | 59.1 | 5193 |
| KL-Cov | 90.0 | 3396 | 59.9 | 6629 |
| EntroAdv | 90.2 | 3963 | 59.1 | 7752 |
| HAPO | **91.1** | 3569 | **60.8** | 7353 |

Table 8: Extended efficiency evidence on MATH and Olympiad for `DeepScaleR-1.5B-8k`. Higher Avg@4 is better; lower token counts indicate lower inference cost. The table is derived from the additional rebuttal measurements used to corroborate the AIME24 Pareto analysis.

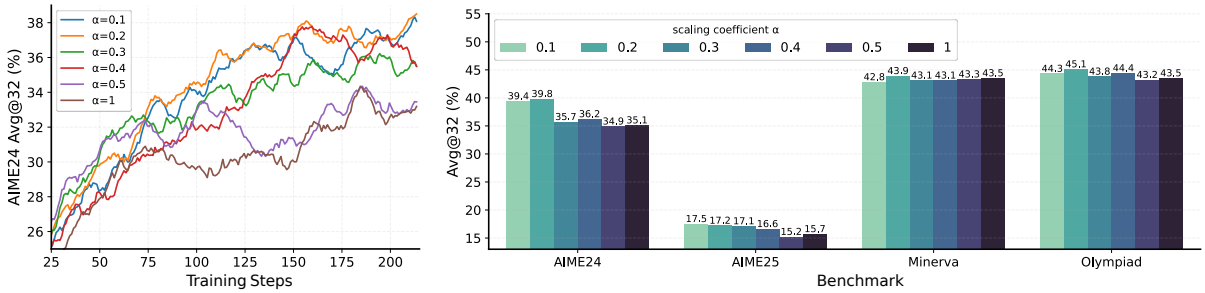



Figure 7: Hyperparameter analysis for $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0.5, 1\}$ on 7B models. *Left*: training efficiency comparison. *Right*: evaluation on various benchmarks.

| $\alpha$ | AIME24 | AIME25 | AMC | MATH | Minerva | Olympiad | avg |
|---|---|---|---|---|---|---|---|
| 1 | 35.1 | 15.7 | 63.5 | 84.3 | 43.5 | 43.5 | 47.6 |
| 0.5 | 34.9 | 15.2 | 62.9 | 83.5 | 43.3 | 43.2 | 47.1 |
| 0.4 | 36.2 | 16.6 | 62.7 | 83.8 | 43.1 | 44.4 | 47.8 |
| 0.3 | 35.7 | 17.1 | 61.8 | 82.8 | 43.1 | 43.8 | 47.3 |
| 0.2 | 39.8 | 17.2 | 62.1 | 83.7 | 43.9 | 45.1 | **48.6** |
| 0.1 | 39.4 | 17.5 | 61.2 | 83.1 | 42.8 | 44.3 | <u>48.0</u> |

Table 9: Avg@$k$ results on different benchmarks with 7B model. **Bold** and <u>underline</u> numbers denote the best and second-best results.

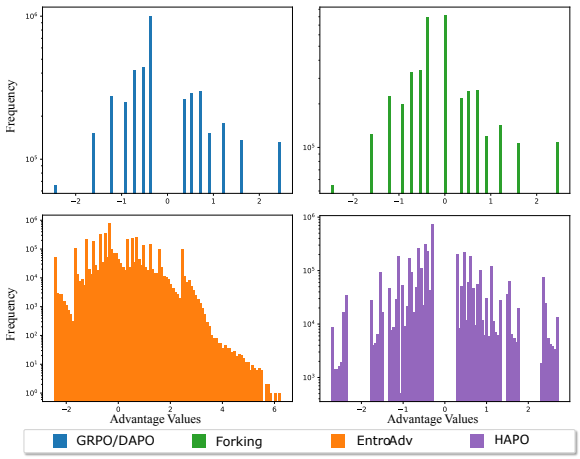



Figure 8: Advantage value distributions. Comparison of token-level advantage allocation among different methods.

possibility that the capacity-guided mechanism primarily benefits high-entropy decision points.

## H Advantage Allocation Analysis

This section compares our advantage allocation strategy with three strong baselines: DAPO (Yu et al., 2025), EntroAdv (Cheng et al., 2025), and Forking (Wang et al., 2025c). Figure 8 visualizes the token-level advantage distributions assigned by each method. Briefly, DAPO (and GRPO) adopts a group-based reward assignment that treats all tokens within a trajectory equally, disregarding their local uncertainty and capacity differences. EntroAdv modifies the advantage function by adding a gradient-detached entropy term, explicitly encouraging exploration at high-uncertainty positions. Forking implements a hard selection mechanism, restricting policy gradient updates to the top 20% of tokens with the highest entropy, under the assumption that these forking tokens are useful high-uncertainty candidates.

Figure 8 shows that GRPO/DAPO assigns uniform advantages, failing to distinguish candidate high-capacity branching points from routine tokens. Forking improves upon this by filtering out low-entropy tokens, yet its binary masking remains a coarse-grained approximation. EntroAdv

| Domain | Base | GRPO | HAPO |
|---|---|---|---|
| Overall (Mean) | 18.2 | 33.1 | **36.6** |
| Computer Science | 31.0 | 41.0 | **43.2** |
| Math | 32.9 | 34.1 | **37.9** |
| Chemistry | 28.9 | **30.4** | 29.2 |
| Engineering | 16.4 | **29.8** | **29.8** |
| Law | 7.1 | 12.8 | **16.2** |
| Biology | 21.6 | 44.1 | **55.5** |
| Health | 10.6 | **25.2** | 24.7 |
| Physics | 21.7 | 36.7 | **41.4** |
| Business | 24.0 | 47.9 | **50.1** |
| Philosophy | 13.0 | 25.7 | **31.3** |
| Economics | 7.4 | 48.2 | **55.9** |
| Other | 11.8 | 29.8 | **32.6** |
| Psychology | 8.4 | 40.5 | **45.6** |
| History | 10.8 | 16.8 | **24.2** |

Table 10: Per-domain accuracy (%) on MMLU-Pro with `Qwen2.5-Math-7B`. Best and tied-best results are in **bold**.

Figure 9: High-entropy token word cloud.

Figure 10: Low-entropy token word cloud.

introduces continuous entropy-based shaping, but its additive entropy bonus does not distinguish reward polarity and lacks a distinct mechanism for handling negative feedback. In contrast, our method implements a continuous, four-quadrant modulation. It modulates candidate high-capacity tokens by increasing positive feedback at successful branching points (PHR) and increasing negative feedback at failed branching points (NHR), while dampening the signal for low-entropy tokens to prevent over-exploitation and noise. This targeted approach concentrates update magnitude on branching points rather than routine continuations.

# I Empirical Observation

This section examines the responses of `Qwen2.5-Math-7B` by visualizing the training outputs with token entropy highlighted.

The visualization samples over 4k responses in the training set and collects over $10^6$ tokens. The statistical analysis of the entropy values of tokens is presented in word clouds in Figure 9 and 10, where a larger font size indicates a higher average token entropy. The visualization aligns with prior works (Wang et al., 2025c; Cheng et al., 2025): high-entropy tokens often coincide with candidate forks in reasoning trajectories, while low-entropy tokens often facilitate the stable execution of logic along the selected path. Specifically, high-entropy tokens frequently correspond to logical connectors such as "since," "therefore," and "thus," which bridge reasoning steps. Similarly, in mathematical contexts, tokens like "need," "given," and "let" serve to introduce assumptions or definitions. By contrast, low-entropy tokens typically include word suffixes, code snippets, or components of mathematical formulas, all of which are characterized by high structural determinism. However, existing works primarily target a binary classification of tokens into high and low entropy categories. A further question is whether entropy levels correspond to consistent model behaviors, regardless of correctness. For example, do low-entropy tokens usually indicate overconfident mistakes or robust reasoning steps in incorrect responses?

To this end, the analysis visualizes two representative examples in Figure 11, where token entropy is highlighted separately in incorrect and correct reasoning. The visualization reveals similar patterns. Despite the final correctness, high-entropy tokens often coincide with candidate decision points along the reasoning trajectory, while low-entropy tokens often execute deterministic steps along the established path. Although some low-entropy tokens exhibit overconfident hallucinations, the overall trend persists (Chen et al., 2024; Xiong et al., 2024).

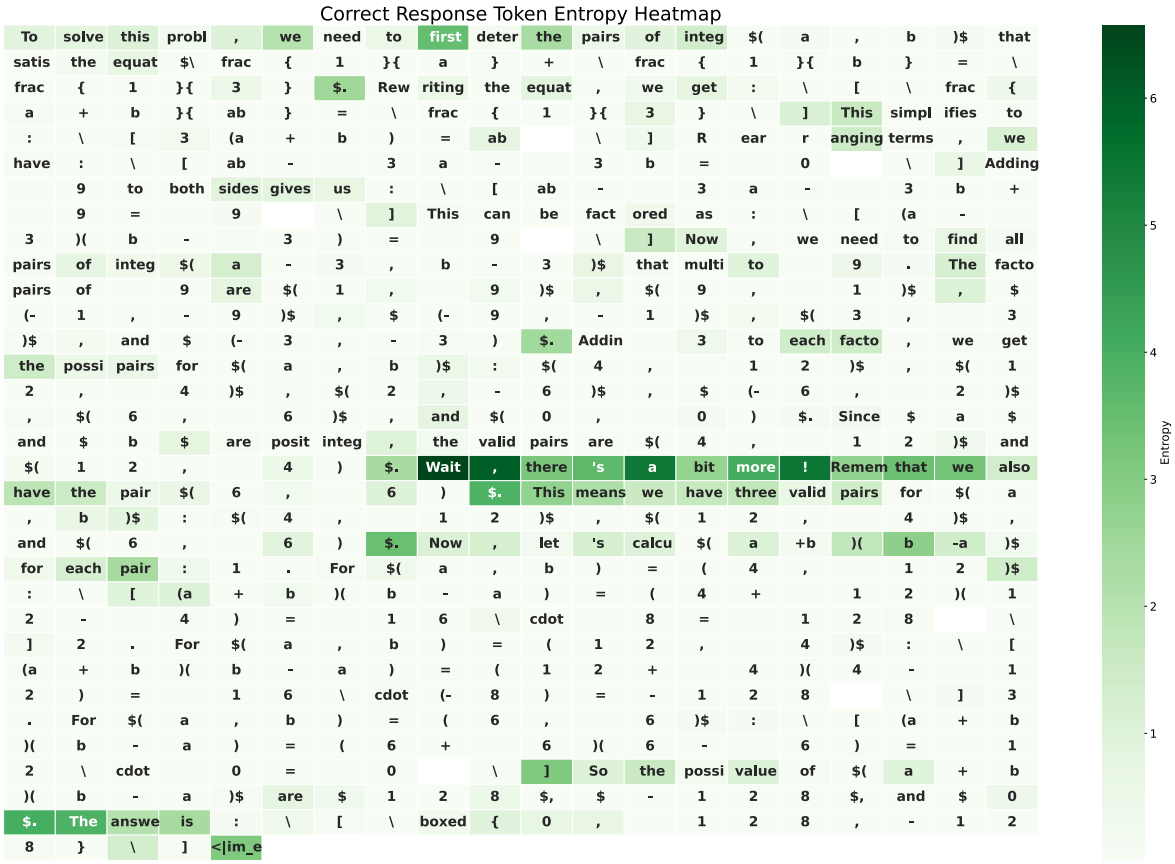



Figure 11: Visualization of token entropy in incorrect and correct responses.