# Unlocking Exploration in RLVR: Uncertainty-aware Advantage Shaping for Deeper Reasoning

**Can Xie, Ruotong Pan, Xiangyu Wu, Yunfei Zhang, Jiayi Fu, Tingting Gao, Guorui Zhou***

Kuaishou Technology

xiecan02@gmail.com

{panruotong03, wuxiangyu06, zhangyunfei05, fujiayi, lisize, zhouguorui}@kuaishou.com

## Abstract

Reinforcement Learning with Verifiable Rewards (RLVR) has shown significant promise for enhancing the reasoning capabilities of large language models (LLMs). However, prevailing algorithms like GRPO broadcast a uniform advantage signal across all tokens in a sequence. This coarse-grained approach overlooks the pivotal role of uncertain, high-stakes decisions during reasoning, leading to inefficient exploration and the well-documented problem of entropy collapse. To address this, we introduce **U**n**C**ertainty-aware **A**dvantage **S**haping (**UCAS**), a model-free method that refines credit assignment by leveraging the model's internal uncertainty signals. UCAS operates in two stages: it first modulates the response-level advantage using a logit-space self-confidence proxy, and then applies an asymmetric token-level penalty based on raw logit certainty. This dual mechanism encourages exploration of high-uncertainty paths that yield correct answers while penalizing overconfident yet erroneous reasoning, effectively balancing the exploration-exploitation trade-off. Extensive experiments on five mathematical reasoning benchmarks show that UCAS significantly outperforms strong RLVR baselines across multiple model scales, including 1.5B and 7B. Our analysis confirms that UCAS not only achieves higher rewards but also promotes greater reasoning diversity and successfully mitigates entropy collapse. Code is available at https://github.com/xvolcano02/UCAS.

## 1 Introduction

Reinforcement learning (RL) has recently become a cornerstone for enhancing the complex reasoning abilities of Large Language Models (LLMs), moving beyond simple pattern matching toward more robust problem-solving. Among the various RL approaches, Reinforcement Learning with Verifiable Rewards (RLVR) has proven particularly effective. In this paradigm, a policy model explores a vast solution space and receives feedback from verifiable signals, such as the correctness of a final answer in mathematical reasoning. This direct feedback loop has enabled policy optimization algorithms like Group Relative Policy Optimization (GRPO) (Shao et al., 2024) to achieve substantial performance gains, powering state-of-the-art systems such as DeepSeek-R1 (Guo et al., 2025).

However, the success of RLVR reveals a critical underlying tension: the trade-off between precision and diversity. While methods like GRPO excel at increasing the probability of generating correct answers, they often do so at the cost of exploration. Due to the absence of a critic model, the learning signal in GRPO, which applies a single uniform advantage across all tokens, provides an indiscriminate and overly coarse form of credit assignment. It rewards all steps of a correct path equally and penalizes all steps of an incorrect one, failing to distinguish crucial reasoning leaps from trivial ones. This coarse-grained feedback drives the policy to converge prematurely on a small set of "safe" high-reward trajectories. A common side effect is entropy collapse (Cui et al., 2025b), where the output distribution contracts, reducing solution diversity and impairing performance on complex problems that demand novel reasoning strategies.

Previous studies (Wang et al., 2023; Lightman et al., 2024; Chen et al., 2024; Zhang et al., 2024; Sun et al., 2025a) have attempted to employ process-level reward models to deliver more fine-grained signals. However, as DeepSeek (Guo et al., 2025) points out, training fine-grained reward models is costly, difficult to scale, limited in its ability to provide accurate signals, and vulnerable to reward hacking. Some recent efforts (Chen et al., 2025; Cheng et al., 2025; Wang et al., 2025a) have tried to incorporate entropy-based feedback to enhance advantages, such as integrating semantic

*Corresponding author



*Findings of the Association for Computational Linguistics: ACL 2026*, pages 19057–19076
July 2-7, 2026 ©2026 Association for Computational Linguistics

entropy or policy entropy related to the response into advantage calculations. Yet, most studies either pursue low entropy to improve accuracy or encourage high entropy to maintain exploration, lacking fine-grained modeling of the relationship between responses and their policy entropy.

To address the above challenges, we propose an **U**n**C**ertainty-aware **A**dvantage **S**haping (**UCAS**), a model-free method that refines credit assignment in RLVR by leveraging the model's intrinsic uncertainty. UCAS is designed to resolve the precision–diversity dilemma by reshaping the advantage signal at two complementary levels. At the response level, UCAS modulates the sequence-level advantage using a logit-space self-confidence proxy, amplifying rewards for correct-but-uncertain responses and penalties for incorrect-but-confident ones. At the token level, it further introduces an asymmetric certainty-based penalty derived directly from raw logits, suppressing overconfident errors while safeguarding positive reasoning trajectories from negative token-level updates. Collectively, these mechanisms promote exploration of uncertain but potentially fruitful reasoning paths, while efficiently suppressing confidently wrong solutions. Extensive experiments on five mathematical reasoning benchmarks demonstrate that UCAS consistently outperforms strong RLVR baselines at both the 1.5B and 7B model scales. Beyond reward improvements, UCAS fosters greater reasoning diversity and substantially mitigates entropy collapse, confirming the effectiveness of uncertainty as a fine-grained learning signal.

Our contributions can be summarized as follows:

- We propose UCAS, an extra-model-free fine-grained advantage shaping mechanism based on internal confidence signals, which performs uncertainty-aware advantage adjustment at both response and token levels.

- We provide a novel mechanism to adaptively calibrate advantages based on uncertainty, enabling steady reward gains, longer reasoning chains, and entropy recovery, thus preventing entropy collapse in RLVR and improving reasoning accuracy.

- Extensive experiments on multiple mathematical reasoning benchmarks demonstrate that UCAS significantly improves model reasoning performance, validating its effectiveness in enhancing exploration diversity and optimization outcomes.

## 2 Background: Reinforcement Learning with Verifiable Rewards

In the training of large language models, early mainstream reinforcement learning alignment methods primarily relied on PPO. By introducing a clipping ratio into the objective function, PPO stabilizes training by constraining the magnitude of policy updates. This method has been widely adopted in Reinforcement Learning from Human Feedback (RLHF), where reward models provide preference-based scores to gradually shape model behavior. However, PPO exhibits key limitations: it depends on critic-based value estimation and requires large-scale preference annotation, both of which are costly and prone to noise accumulation.

To overcome these limitations, recent research has introduced RLVR. RLVR converts open-ended outputs into programmatically checkable signals, such as numerical consistency in mathematics, unit-test pass rates in code generation, or formal constraint satisfaction (Su et al., 2025; Wang et al., 2025b; Lin et al., 2026; Li et al., 2026; Jiang et al., 2026), thereby avoiding the noise and cost of preference models. By forming a closed loop of model–environment–verifier, RLVR enables policies to be updated directly from binary or graded correctness signals, improving both sample efficiency and reproducibility in structured reasoning tasks.

In the concrete implementation of RLVR, GRPO (Shao et al., 2024) emerges as a representative algorithm. Unlike PPO, which relies on critic-based value estimation, GRPO computes advantages by normalizing group-level verifiable rewards and updates the policy directly.

Formally, the objective is given by:

$$\begin{aligned} \mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\, o\sim\pi_{\theta_{\text{old}}}}\Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \\ \min\Big(r_{i,t}(\theta)\hat{A}_{i,t}, \\ \text{clip}(r_{i,t}(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_{i,t}\Big) \\ -\beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})\Bigg] \end{aligned} \tag{1}$$

where

$$r_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}, \tag{2}$$

denotes the probability ratio between the new and old policies for token $o_{i,t}$, and the advantage $\hat{A}_{i,t}$ is estimated from group rewards as:

$$\hat{A}_{i,t} = \frac{R_i - \mu(R)}{\sigma(R) + \epsilon}, \tag{3}$$

with $R_i$ the cumulative verifiable reward of trajectory $o_i$, $\mu(R)$ and $\sigma(R)$ the mean and standard deviation across the sampled group, and $\epsilon$ a small constant for numerical stability.

By eliminating dependency on value models and instead exploiting group-normalized verifiable rewards, GRPO achieves stable and cost-efficient training.

Building on GRPO, Decouple Clip and Dynamic Sampling Policy Optimization (DAPO) (Yu et al., 2025) is proposed to further improve stability and exploration. DAPO integrates four key techniques: Clip-Higher, Dynamic Sampling, Token-Level Policy Gradient Loss, and Overlong Reward Shaping. Similar to GRPO, DAPO samples multiple responses per prompt and optimizes the following objective:

$$\begin{aligned} \mathcal{J}_{\text{DAPO}}(\theta) &= \mathbb{E}_{\substack{(q,a)\sim\mathcal{D} \\ \{o_i\}\sim\pi_{\theta_{\text{old}}}}} \Bigg[ \frac{1}{\sum_{j=1}^{G} |o_j|} \sum_{i=1}^{G} \sum_{t=1}^{|o_i|} \\ &\quad \min\Big( r_t^i(\theta)\hat{A}_t^i, \text{clip}(r_t^i(\theta), \text{clip\_range})\hat{A}_t^i \Big) \Bigg], \\ &\text{s.t. } 0 < |\{i \mid \text{is\_equiv}(o^i, a)\}| < G \end{aligned} \tag{4}$$

where $\epsilon_{\text{low}}$ and $\epsilon_{\text{high}}$ denote the lower and upper bounds of the clipping range. Compared to GRPO, DAPO explicitly decouples the clipping bounds, incorporates adaptive sampling strategies, thereby alleviating entropy collapse and improving the generalizability of RLVR-trained models.

# 3 Method

To address the coarse credit assignment problem in RLVR, we introduce **Uncertainty-aware Advantage Shaping (UCAS)**, a method designed to replace the blunt instrument of uniform advantage with a more nuanced, two-stage mechanism. The central idea is to reshape the learning signal by considering uncertainty at two distinct granularities: the entire reasoning path (response-level) and the individual generative steps within it (token-level). This hierarchical approach first sets a *strategic* learning objective by evaluating the value of the overall trajectory, and then *locally* refines the policy update to encourage robust exploration and prevent the premature convergence that leads to entropy collapse.

## 3.1 Uncertainty Signals: From Confidence to Logits

To perform this hierarchical shaping, UCAS requires signals that capture the model's epistemic state at both macro and micro levels. We extract these directly from the model's intrinsic generative process, avoiding the need for auxiliary networks.

**Response-Level Confidence.** For a high-level assessment of a full reasoning trajectory, we use a logit-space self-confidence score. Let $\ell_{i,t,v}$ denote the pre-softmax logit of vocabulary item $v \in \mathcal{V}$ at step $t$ for response $o_i$, and define the token-level confidence as

$$C_{i,t} := \text{LSE}(\ell_{i,t}) - \frac{1}{|\mathcal{V}|} \sum_{v\in\mathcal{V}} \ell_{i,t,v}, \tag{5}$$

where $\text{LSE}(\ell_{i,t}) = \log \sum_{v\in\mathcal{V}} \exp(\ell_{i,t,v})$. We then define sequence-level confidence as

$$\mathcal{C}(o_i|q) := \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} C_{i,t}. \tag{6}$$

This formulation avoids probability-space instability when token probabilities approach zero. As we show in Appendix C, $C_{i,t}$ is equivalent to $\text{KL}(U(\mathcal{V}) \parallel p_{\pi_\theta}(\cdot \mid q, o_{i,<t})) + \log |\mathcal{V}|$, so larger values of $\mathcal{C}(o_i|q)$ indicate higher overall confidence and lower uncertainty. In contrast, when we discuss entropy collapse later in the paper, we use entropy or $\text{KL}(p \parallel U)$ as a macro-level diagnostic of distributional concentration. The two directions therefore play complementary roles: $\text{KL}(U \parallel p)$ serves as the training-time modulation signal, whereas $\text{KL}(p \parallel U)$ characterizes the collapse outcome.

**Token-Level Certainty.** While self-confidence is effective at the sequence level, it is derived from post-softmax probabilities, which can suffer from poor calibration (Liu et al., 2025a; Ma et al., 2025). This can cause the model to appear equally confident in different choices, masking subtle but important variations in uncertainty. To capture a more direct and sensitive signal at the token level, we use the model's raw logit value for the chosen token $o_{i,t}$ as a proxy for certainty. Let $\ell_{i,t}$ be the logit corresponding to token $o_{i,t}$ at step $t$. A higher logit value indicates greater model certainty in its choice, prior to softmax normalization.

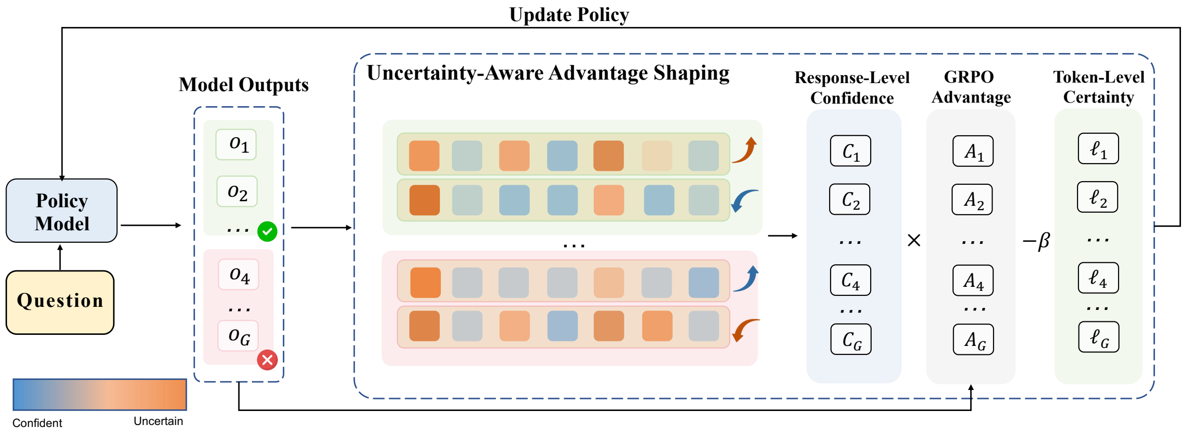


Figure 1: **Overview of the UCAS Advantage Shaping Mechanism.** UCAS refines the uniform GRPO advantage through a two-stage process. **Stage 1 (Macro-level)**: It applies Response-Level Modulation using the trajectory's overall self-confidence to determine its strategic value for exploration vs. exploitation. **Stage 2 (Micro-level)**: It introduces a Token-Level Certainty Penalty using raw logits to discourage local overconfidence. The final shaped advantage $\hat{A}_{i,t}^{\text{UCAS}}$ guides a more nuanced policy update.

### 3.2 UCAS: Two-Stage Advantage Shaping

Given a group of $G$ responses $\{o_1, \ldots, o_G\}$ to a prompt $q$, UCAS reshapes the original GRPO advantage $\hat{A}_i$ into a fine-grained, token-specific advantage $\hat{A}_{i,t}^{\text{UCAS}}$. This process unfolds in two complementary stages.

**Stage 1: Response-Level Advantage Modulation.** This stage adjusts the advantage for an entire response to encourage exploration of novel correct paths and suppress confident, well-trodden incorrect paths. First, we compute the self-confidence $\mathcal{C}(o_i|q)$ for each response $o_i$ in the group. To assess confidence relative to other responses in the same group, we apply z-score normalization:

$$\hat{\mathcal{C}}_i = \frac{\mathcal{C}(o_i|q) - \mu_{\mathcal{C}}}{\sigma_{\mathcal{C}} + \epsilon}, \tag{7}$$

where $\mu_{\mathcal{C}}$ and $\sigma_{\mathcal{C}}$ are the mean and standard deviation of confidence scores across the group. This standardization makes the modulation weights comparable within each prompt group and reduces dependence on model-specific raw logit scales.

We then compute a modulation weight $W(\hat{\mathcal{C}}_i)$ based on the sign of the original advantage $\hat{A}_i$, which directly encodes the correctness of the answer. Theoretically, we select an exponential form to act as a non-linear filter. This addresses the compressed variance often found in group-normalized scores, where linear rescaling fails to sufficiently distinguish "novel" exploration from "routine" exploitation. More theoretical explanation can be found in Appendix C.

$$W(\hat{\mathcal{C}}_i) = \begin{cases} \exp(-\alpha \cdot \hat{\mathcal{C}}_i) & \text{if } \hat{A}_i > 0 \quad \text{(Correct)} \\ \exp(\alpha \cdot \hat{\mathcal{C}}_i) & \text{if } \hat{A}_i < 0 \quad \text{(Incorrect)} \end{cases} \tag{8}$$

where $\alpha > 0$ is a hyperparameter controlling the shaping intensity. This formulation ensures that for correct responses, lower confidence (negative $\hat{\mathcal{C}}_i$) results in a larger weight, amplifying the reward. For incorrect responses, higher confidence (positive $\hat{\mathcal{C}}_i$) results in a larger weight, amplifying the penalty. The resulting modulated advantage is $\hat{A}_i^{\text{mod}} = W(\hat{\mathcal{C}}_i) \cdot \hat{A}_i$.

**Stage 2: Token-Level Certainty Penalty.** Response-level modulation sets a global learning objective for each trajectory, but this modulated advantage, $\hat{A}_i^{\text{mod}}$, is still a uniform signal broadcast to all tokens within that sequence. This alone is insufficient to prevent the model from developing localized overconfidence—a key driver of entropy collapse. The second stage therefore introduces a token-specific penalty to directly address this. The penalty is applied asymmetrically: it strongly suppresses locally overconfident tokens on unfavorable trajectories, while positive trajectories are protected from being flipped into negative token-level updates.

We use the raw logit $\ell_{i,t}$ as our certainty measure and apply Min-Max normalization within each sequence to create a standardized penalty score

$\hat{\ell}_{i,t} \in [0, 1]$:

$$\hat{\ell}_{i,t} = \frac{\ell_{i,t} - \min_k(\ell_{i,k})}{\max_k(\ell_{i,k}) - \min_k(\ell_{i,k}) + \epsilon} \tag{9}$$

A value of $\hat{\ell}_{i,t}$ close to 1 indicates high relative certainty for that token choice. This penalty acts as a regularizer, complementing the directional guidance from Stage 1.

**Final Advantage Shaping Formula.** By combining these two stages, UCAS creates a composite advantage signal that is both globally informed and locally sensitive. The final shaped advantage for each token is:

$$\hat{A}^{\text{UCAS}}_{i,t} = \begin{cases} \max(0, \hat{A}^{\text{mod}}_i - \beta\hat{\ell}_{i,t}), & \hat{A}^{\text{mod}}_i \geq 0 \\ \hat{A}^{\text{mod}}_i - \beta\hat{\ell}_{i,t}, & \hat{A}^{\text{mod}}_i < 0 \end{cases} \tag{10}$$

where $\beta > 0$ is a hyperparameter controlling the penalty strength. When $\hat{A}^{\text{mod}}_i \geq 0$, the clamp prevents correct trajectories from receiving negative token-level advantages. When $\hat{A}^{\text{mod}}_i < 0$, the penalty further suppresses high-certainty errors, assigning stronger negative updates to confidently wrong transitions. This asymmetric structure steers the model toward novel correct solutions while discouraging localized overconfidence, thereby mitigating entropy collapse and fostering more robust problem-solving abilities. This final advantage term then replaces the original advantage in the RL objective:

$$\begin{aligned} \mathcal{J}_{\text{UCAS}}(\theta) = \mathbb{E}_{\substack{(q,a)\sim\mathcal{D} \\ \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}}} \Bigg[ \frac{1}{\sum_{j=1}^G |o_j|} \\ \sum_{i=1}^G \sum_{t=1}^{|o_i|} \min \Big( r^i_t(\theta) \hat{A}^{\text{UCAS}}_{i,t}, \\ \text{clip}\big(r^i_t(\theta), 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}}\big) \hat{A}^{\text{UCAS}}_{i,t} \Big) \Bigg] \\ \text{s.t. } 0 < |\{i \mid \text{is\_equiv}(o^i, a)\}| < G \end{aligned} \tag{11}$$

The complete implementation process of UCAS is shown in Algorithm 1.

**Algorithm 1** Uncertainty-aware Advantage Shaping (UCAS)

**Input:**
A group of $G$ responses $\{o_i\}_{i=1}^G \sim \pi_\theta$;
Rule-based rewards $\{R_i\}_{i=1}^G$;
Hyperparameters $\alpha, \beta$.
Compute group advantages $\{\hat{A}_i\}_{i=1}^G$.
Compute self-confidence $\{\mathcal{C}(o_i|q)\}_{i=1}^G$.
Normalize confidences to get $\{\hat{\mathcal{C}}_i\}_{i=1}^G$.
**for** $i = 1$ to $G$ **do**
    ***Stage 1: Response Modulation***
    Calc $W(\hat{\mathcal{C}}_i)$ via Eq. 8.
    $\hat{A}^{\text{mod}}_i \leftarrow W(\hat{\mathcal{C}}_i) \cdot \hat{A}_i$.
    ***Stage 2: Token Certainty Penalty***
    Get logits $\{\ell_{i,t}\}$ for tokens in $o_i$.
    Normalize logits to get $\{\hat{\ell}_{i,t}\}$.
    **for** $t = 1$ to $|o_i|$ **do**
        **if** $\hat{A}^{\text{mod}}_i \geq 0$ **then**
            $\hat{A}^{\text{UCAS}}_{i,t} \leftarrow \max(0, \hat{A}^{\text{mod}}_i - \beta \cdot \hat{\ell}_{i,t})$.
        **else**
            $\hat{A}^{\text{UCAS}}_{i,t} \leftarrow \hat{A}^{\text{mod}}_i - \beta \cdot \hat{\ell}_{i,t}$.
        **end if**
    **end for**
**end for**
**Output:** Token-level advantages $\{\hat{A}^{\text{UCAS}}_{i,t}\}$.

## 4 Experiments

### 4.1 Experimental Setup

**Models and Baselines.** We employ two variants of the Qwen2.5-Math (Yang et al., 2024) series as our foundation models: Qwen2.5-Math-1.5B and Qwen2.5-Math-7B. To quantify the performance improvement introduced by our method, we compare against two widely used RLVR baselines, GRPO and DAPO. In addition, we benchmark against several representative recent methods in math reasoning and RLVR, including SimpleRL-Zoo (Zeng et al., 2025), PRIME-Zero (Cui et al., 2025a), OpenReasonerZero (Hu et al., 2025), Oat-Zero (Liu et al., 2025b), GRPO with Entropy Adv. (Cheng et al., 2025), and KTAE (Sun et al., 2025b). Detailed descriptions of the baselines are provided in Appendix A.1.

**Training Data and Benchmarks.** During the training phase, we utilize the widely-used MATH dataset as our training set. To maintain consistency with prior research, we only use the more challenging subset of this dataset for training, specifically problems from levels 3 to 5. To comprehensively evaluate the reasoning capabilities of the model trained with our method, we select five widely recognized benchmarks in the mathematical reasoning domain for testing: AIME24 (LI et al., 2024), MATH-500 (Hendrycks et al., 2021), AMC (LI et al., 2024), Minerva (Lewkowycz et al., 2022), and OlympiadBench (Huang et al., 2024), which collectively contain 1,560 problems.

### 4.2 Main Results

The greedy pass@1 performance comparison between 1.5B and 7B models across five mathematical reasoning benchmarks is presented in Table 1. We can clearly find that the UCAS model achieved the highest performance across all five math reasoning benchmarks on both the 1.5B and 7B parameter scales. Compared with the DAPO baseline, UCAS improves the average accuracy from 41.2 to 47.3 (+6.1) on Qwen2.5-Math-1.5B and from 50.5 to 56.7 (+6.2) on Qwen2.5-Math-7B. Beyond DAPO, UCAS also surpasses strong baselines such as KTAE and Oat-Zero, with pronounced gains on challenging benchmarks including AIME24, AMC, and OlympiadBench. These results highlight the robustness and scalability of uncertainty-aware advantage shaping, demonstrating consistent benefits across model sizes and diverse reasoning tasks. Additional four-seed results in Appendix B.4 show that these gains are stable and not driven by a single favorable run.

### 4.3 Analysis

**Ablation Study.** The ablation comparison between response-level and token-level uncertainty modeling is presented in Table 2. We can clearly observe that both response-level and token-level uncertainty bring consistent gains over the DAPO baseline. Compared with the model trained with DAPO, incorporating response-level confidence increases the average score on Qwen2.5-Math-1.5B from 41.2 to 44.7 (+3.5%), while token-level uncertainty further raises it to 45.1 (+3.9%). A similar trend holds on the 7B model, where both variants surpass the DAPO baseline. Their integration in UCAS achieves the best performance, confirming that both signals are individually useful and jointly necessary.

**Hyperparameter Sensitivity.** We evaluate the robustness of UCAS by varying the response-level modulation $\alpha$ and token-level penalty $\beta$ on the Math-500 benchmark (Table 3). This stability is supported by strict normalization throughout the method: group advantages and response-level confidence are z-score normalized, while token logits are min–max normalized to $[0, 1]$. A moderate penalty ($\beta = 0.01$) yields optimal performance. Lower values (0.005) fail to sufficiently counteract entropy collapse, while excessive penalization ($\beta \geq 0.05$) over-regularizes the policy. This suggests that high $\beta$ artificially flattens the distribution even for necessary, high-certainty steps, hindering coherent reasoning chains. The method exhibits stability within $\alpha \in [0.1, 0.2]$. However, aggressive modulation ($\alpha = 0.4$) causes a performance drop. This implies a "signal dominance" issue: overly strong confidence scaling overshadows the fundamental correctness signal, introducing variance that distracts from the primary objective of mathematical accuracy. Notably, the same $(\alpha, \beta)$ setting also transfers effectively across the 1.5B and 7B models in Table 1, indicating limited sensitivity to moderate model-scale variation.

**Training Dynamics.** The training process highlights several key performance trends, as shown in Figure 2. Compared to vanilla GRPO, UCAS demonstrates a consistent increase in the inference reward on the MATH500 benchmark. Regarding the average response length, the inclusion of UCAS enables the model to generate longer reasoning chains, reflecting more comprehensive problem-solving (Guo et al., 2025; Cheng et al., 2025), while simultaneously improving accuracy. For generation entropy, UCAS shows an early decline but later recovers and stabilizes at a higher level, effectively avoiding the entropy collapse reported in prior work (Cui et al., 2025b). Notably, the model's reward continues to rise even as the entropy increases, which indicates a stable and effective training dynamic where exploration and optimization are well-balanced.

**Pass@k Evaluation.** Prior studies (Wang et al., 2022; Wu et al., 2024) have shown that with a limited number of rollouts, models often struggle to solve certain tasks. In contrast, when the rollout budget is sufficiently large, the probability of sampling effective solutions increases considerably. This observation suggests that pass@k accuracy with a large k provides a more reliable estimate of a model's potential performance (Yue et al., 2025). Under this evaluation protocol, a problem is considered solved if any of the k sampled reasoning trajectories yield the correct answer. Figure 3 reports pass@k results on the AIME24 benchmark. The results indicate that UCAS achieves more consistent improvements as k grows. In contrast, Vanilla-GRPO and its enhanced variants show slower growth, consistent with findings from Yue et al. (2025). The stronger performance of UCAS under the pass@k metric highlights its effectiveness, which can be attributed to differences in exploration strategies. Unlike Vanilla-GRPO, which

| Models | AIME24 | MATH-500 | AMC | Minerva | Olympiad | Avg |
|---|---|---|---|---|---|---|
| | | | *Qwen2.5-Math-1.5B* | | | |
| Base Model | 7.3 | 61.8 | 43.4 | 15.1 | 28.4 | 31.2 |
| GRPO | 15.6 | 76.0 | 51.8 | 22.1 | 36.3 | 40.4 |
| DAPO | 16.7 | 77.6 | 47.0 | 25.7 | 39.0 | 41.2 |
| Oat-Zero(Liu et al., 2025b) | 20.0 | 74.4 | 50.6 | 23.9 | 37.0 | 41.2 |
| KTAE(Sun et al., 2025b) | 20.0 | 77.6 | 50.6 | 29.0 | 40.0 | 43.4 |
| SEED-GRPO(Chen et al., 2025) | 23.3 | 75.4 | 50.6 | 26.8 | 41.3 | 43.5 |
| UCAS | **23.3** | **80.6** | **59.0** | **31.6** | **42.1** | **47.3** |
| | | | *Qwen2.5-Math-7B* | | | |
| Base Model | 11.0 | 69.0 | 45.8 | 21.3 | 28.4 | 35.1 |
| GRPO | 30.0 | 81.0 | 57.8 | 32.7 | 43.2 | 48.9 |
| DAPO | 30.5 | 81.8 | 60.2 | 34.5 | 45.3 | 50.5 |
| PRIME-Zero (Cui et al., 2025a) | 23.3 | 82.2 | 57.8 | 36.0 | 39.9 | 47.8 |
| OpenReasonerZero (Hu et al., 2025) | 17.9 | 78.4 | 45.8 | 27.9 | 45.0 | 43.0 |
| Oat-Zero(Liu et al., 2025b) | 32.1 | 79.8 | 61.4 | 30.5 | 41.8 | 49.1 |
| Simple RL-Zero(Zeng et al., 2025) | 26.7 | 78.6 | 59.0 | 33.8 | 43.4 | 48.3 |
| GRPO with Entropy Adv. (Cheng et al., 2025)† | 33.7 | 83.1 | **69.8** | - | - | - |
| KTAE(Sun et al., 2025b) | 36.7 | 83.2 | 63.9 | 35.3 | 43.7 | 52.6 |
| SEED-GRPO(Chen et al., 2025) | 43.3 | 82.2 | 64.7 | 35.0 | 45.2 | 54.7 |
| UCAS | **43.3** | **85.6** | 68.7 | **37.6** | **48.0** | **56.7** |

Table 1: The greedy pass@1 performance of 1.5B and 7B models across five math reasoning benchmarks. †: results from Cheng et al. (2025). Our method UCAS consistently surpasses all baselines in both parameter scales.

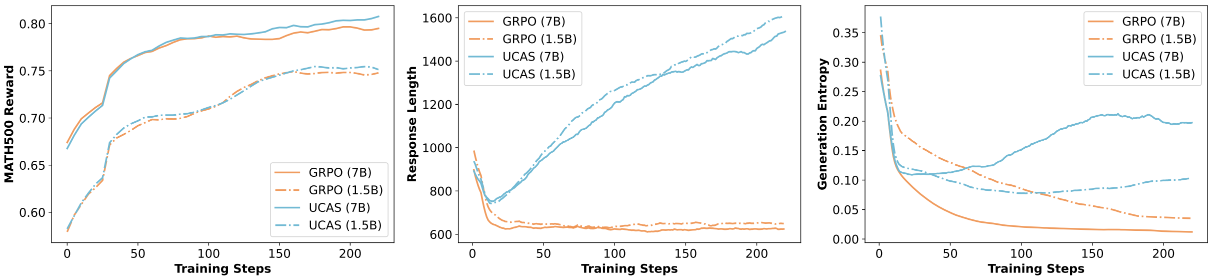


Figure 2: Training dynamics of UCAS compared with GRPO across both 7B and 1.5B models. **Left:** Reward; **Middle:** Response Length; **Right:** Generation Entropy.

often suffers from exploration stagnation, where the model repeatedly samples low-diversity rollouts, UCAS leverages uncertainty-aware advantage shaping to sustain diverse exploration and escape local optima. Additional pass@k results on AMC and MATH-500, reported in Appendix B.3, show the same trend on broader benchmarks.

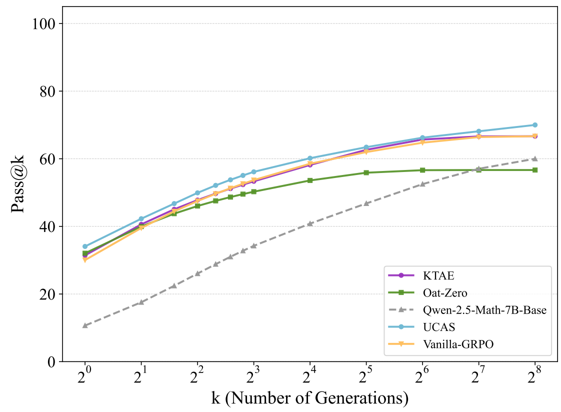


Figure 3: Comparison of pass@k results on the AIME24 Benchmark.

## 5 Related Work

### 5.1 RL for LLM Reasoning

Recent advances in reinforcement learning have transformed the training of large language models for reasoning tasks. Process reward models (PRMs) (Lightman et al., 2023; Nie et al., 2026; Wu et al., 2026) have emerged as a key innovation, providing step-level supervision that improves both

| Models | AIME24 | MATH-500 | AMC | Minerva | Olympiad | Avg |
|---|---|---|---|---|---|---|
| *Qwen2.5-Math-1.5B* | | | | | | |
| Base Model | 7.3 | 61.8 | 43.4 | 15.1 | 28.4 | 31.2 |
| w/ DAPO | 16.7 | 77.6 | 47.0 | 25.7 | 39.0 | 41.2 |
| w/ DAPO + Response-Level Confidence | **23.3** | 79.6 | 51.8 | 27.6 | 41.0 | 44.7 |
| w/ DAPO + Token-Level Certainty | 20.0 | 80.2 | 55.4 | 29.7 | 40.1 | 45.1 |
| w/ DAPO + UCAS (Ours) | **23.3** | **80.6** | **59.0** | **31.6** | **42.1** | **47.3** |
| *Qwen2.5-Math-7B* | | | | | | |
| Base Model | 11.0 | 69.0 | 45.8 | 21.3 | 28.4 | 35.2 |
| w/ DAPO | 30.5 | 81.8 | 60.2 | 34.5 | 45.3 | 50.5 |
| w/ DAPO + Response-Level Confidence | 40.0 | 85.0 | 63.9 | 36.7 | 47.4 | 54.6 |
| w/ DAPO + Token-Level Certainty | 36.7 | 84.6 | 65.0 | 29.7 | 47.7 | 52.7 |
| w/ DAPO + UCAS (Ours) | **43.3** | **85.6** | **68.7** | **37.6** | **48.0** | **56.7** |

Table 2: Ablation study of uncertainty modeling. Both sentence-level and token-level uncertainty bring consistent gains over the DAPO baseline.

| Response-Level $\alpha$ | Token-Level $\beta$ | Math-500 |
|---|---|---|
| *Varying Token-Level Penalty ($\beta$), Fixed $\alpha = 0.2$* | | |
| 0.2 | 0.005 | 79.6 |
| 0.2 | 0.01 | **80.4** |
| 0.2 | 0.05 | 78.8 |
| 0.2 | 0.1 | 78.4 |
| *Varying Response Modulation ($\alpha$), Fixed $\beta = 0.01$* | | |
| 0.1 | 0.01 | 80.0 |
| 0.2 | 0.01 | **80.4** |
| 0.4 | 0.01 | 78.8 |

Table 3: Sensitivity analysis on Math-500 when varying the token-level penalty $\beta$ (upper block) and the response-level modulation coefficient $\alpha$ (lower block).

efficiency and accuracy compared to outcome-only rewards. Approaches such as PRIME (Cui et al., 2025a) eliminate costly human annotation by deriving implicit process feedback, while OmegaPRM (Luo et al., 2024) leverages Monte Carlo Tree Search (MCTS) to automatically identify reasoning errors. Alongside this, DeepSeek-R1 (Guo et al., 2025) demonstrates that sophisticated reasoning can emerge purely from RL without supervised fine-tuning, enabled by GRPO, which replaces value functions with group-based baselines. These advances redefine alignment and reasoning in LLMs, positioning reinforcement learning with verifiable or process-level rewards as a scalable and principled alternative to preference-model-based RLHF.

### 5.2 Reinforcement Learning from Verifiable Rewards

RLVR has emerged as a scalable alternative to preference-based alignment by converting open-ended outputs into checkable signals such as mathematical correctness or unit-test pass rates (Guo et al., 2025; Yue et al., 2025; An et al., 2025; Zhou et al., 2026). While early implementations demonstrated strong gains in pass@1 accuracy, subsequent studies revealed a consistent policy entropy collapse: models rapidly concentrate probability mass on a narrow set of high-reward trajectories, diminishing output diversity and limiting exploration (Cui et al., 2025b; Hao et al., 2025). Empirical analyses show that RLVR-trained models often underperform base models on pass@$k$ (Shao et al., 2024; Yue et al., 2025), highlighting a precision–diversity trade-off (Wu et al., 2025; Dong et al., 2025; Zhao et al., 2026).

Algorithmic responses to entropy collapse vary. Standard entropy or KL penalties provide partial remedies, though their effectiveness often depends heavily on the divergence form (Li et al., 2025). More recent uncertainty-aware approaches have sought to refine the learning signal, though with differing philosophies. For instance, SEED-GRPO (Chen et al., 2025) leverages semantic entropy to downscale updates for uncertain queries, adopting a conservative risk-mitigation strategy. In stark contrast, UCAS adopts an exploratory philosophy: we explicitly amplify rewards for correct-but-uncertain trajectories to incentivize venturing into novel reasoning domains, rather than inhibiting learning from uncertainty.

Similarly, while entropy-based shaping methods (Cheng et al., 2025) introduce indiscriminate entropy bonuses to encourage diversity, UCAS implements a *conditional*, two-stage mechanism. By combining a logit-space response-level confidence proxy with token-level raw logits, which we find to be a more sensitive signal of local overconfidence than post-softmax entropy, UCAS distinguishes between productive exploration and blind guessing. Unlike pure entropy-based frameworks, UCAS introduces correctness-contingent modulation, amplifying penalties for confident errors while guiding exploration through uncertainty, offering a more fine-grained solution to the entropy collapse problem than global regularization or token-level covariance control (Cui et al., 2025b).

## 6 Conclusion

In this work, we introduced UnCertainty-aware Advantage Shaping (UCAS), a fine-grained advantage estimation framework that leverages internal confidence signals without requiring additional reward models. By jointly modeling uncertainty at both the response and token levels, UCAS reshapes advantages to highlight critical uncertain reasoning steps, suppress overconfident errors, and preserve non-negative token-level updates on positive trajectories. Experimental results on major mathematical reasoning benchmarks show that UCAS achieves substantial performance improvements over GRPO and its enhanced variants. Analysis of the training dynamics further reveals that, as training progresses, UCAS demonstrates steadily increasing rewards, longer reasoning chains, and an entropy trajectory that first declines and then rises, reflecting stronger exploratory capability. These findings indicate that uncertainty-aware advantage shaping offers an effective pathway toward more robust reinforcement learning for large language models.

## Limitations

Although UCAS demonstrates significant improvements in reasoning capabilities and exploration efficiency, we acknowledge several limitations that identify clear directions for future research. First, our experiments are exclusively focused on 1.5B and 7B parameter models, and the performance of UCAS on largerscale models has not yet been fully verified due to limitations of computing resources. Second, our method relies on self-confidence and raw logits as proxies for model uncertainty. While these internal signals are computationally efficient and effective, future work could explore alternative or complementary uncertainty metrics. Techniques such as Monte Carlo dropout, model ensembles, or semantic entropy could potentially capture different facets of model uncertainty and lead to even more refined and robust advantage shaping. Investigating these areas will be essential for understanding the broader generalizability of our approach.

## References


Shengnan An, Xunliang Cai, Xuezhi Cao, Xiaoyu Li, Yehao Lin, Junlin Liu, Xinxuan Lv, Dan Ma, Xuanlin Wang, Ziwen Wang, and Shuang Zhou. 2025. Amo-bench: Large language models still struggle in high school math competitions. *Preprint*, arXiv:2510.26768.

Guoxin Chen, Minpeng Liao, Chengxi Li, and Kai Fan. 2024. Step-level value preference optimization for mathematical reasoning. *arXiv preprint arXiv:2406.10858*.

Minghan Chen, Guikun Chen, Wenguan Wang, and Yi Yang. 2025. Seed-grpo: Semantic entropy enhanced grpo for uncertainty-aware policy optimization. *arXiv preprint arXiv:2505.12346*.

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

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, and 1 others. 2025a. Process reinforcement through implicit rewards. *arXiv preprint arXiv:2502.01456*.

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

Yihong Dong, Xue Jiang, Yongding Tao, Huanyu Liu, Kechi Zhang, Lili Mou, Rongyu Cao, Yingwei Ma, Jue Chen, Binhua Li, and 1 others. 2025. Rl-plus: Countering capability boundary collapse of llms in reinforcement learning with hybrid-policy optimization. *arXiv preprint arXiv:2508.00222*.

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*.

Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Yu Wu, YK Li, and 1 others. 2024. Deepseek-coder: When the large language model meets programming–the rise of code intelligence. *arXiv preprint arXiv:2401.14196*.

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

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

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. *arXiv preprint arXiv:2103.03874*.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. 2025. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *arXiv preprint arXiv:2503.24290*.

Zhen Huang, Zengzhi Wang, Shijie Xia, Xuefeng Li, Haoyang Zou, Ruijie Xu, Run-Ze Fan, Lyumanshan Ye, Ethan Chern, Yixin Ye, and 1 others. 2024. Olympicarena: Benchmarking multi-discipline cognitive reasoning for superintelligent ai. *Advances in Neural Information Processing Systems*, 37:19209–19253.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2024. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*.

Kehan Jiang, Haonan Dong, Zhaolu Kang, Zhengzhou Zhu, and Guojie Song. 2026. Foe: Forest of errors makes the first solution the best in large reasoning models. *Preprint*, arXiv:2604.02967.

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

Guocong Li, Jinjian Zhang, Ping Wang, Dongnan Liu, Tian Liang, Qiuyi Qi, Hao Huang, Siyan Guo, Mutian Bao, Wei Zhou, Linjian Mo, Hongxia Xu, and Jian Wu. 2026. Mol: Adaptive mixture-of-length reasoning for efficient question answering with context. In *The Fourteenth International Conference on Learning Representations*.

Jia LI, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, and Stanislas Polu. 2024. Numinamath. [https://huggingface.co/AI-MO/NuminaMath-CoT](https://github.com/project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf).

Long Li, Jiaran Hao, Jason Klein Liu, Zhijian Zhou, Xiaoyu Tan, Wei Chu, Zhe Wang, Shirui Pan, Chao Qu, and Yuan Qi. 2025. The choice of divergence: A neglected key to mitigating diversity collapse in reinforcement learning with verifiable reward. *arXiv preprint arXiv:2509.07430*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

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*.

Zhiming Lin, Kai Zhao, Sophie Zhang, Peilai Yu, and Canran Xiao. 2026. Cec-zero: Zero-supervision character error correction with self-generated rewards. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 23612–23620.

Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. 2025a. Uncertainty quantification and confidence calibration in large language models: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 6107–6117.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025b. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Meiqi Guo, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, and 1 others. 2024. Improve mathematical reasoning in language models by automated process supervision. *arXiv preprint arXiv:2406.06592*.

Huan Ma, Jingdong Chen, Guangyu Wang, and Changqing Zhang. 2025. Estimating llm uncertainty with logits. *arXiv e-prints*, pages arXiv–2502.

Shuaiyi Nie, Siyu Ding, Wenyuan Zhang, Linhao Yu, Tianmeng Yang, Yao Chen, Tingwen Liu, Weichong Yin, Yu Sun, and Hua Wu. 2026. Attnpo: Attention-guided process supervision for efficient reasoning. *arXiv preprint arXiv:2602.09953*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Yi Su, Dian Yu, Linfeng Song, Juntao Li, Haitao Mi, Zhaopeng Tu, Min Zhang, and Dong Yu. 2025. Crossing the reward bridge: Expanding rl with verifiable rewards across diverse domains. *arXiv preprint arXiv:2503.23829*.

Wei Sun, Qianlong Du, Fuwei Cui, and Jiajun Zhang. 2025a. An efficient and precise training data construction framework for process-supervised reward model in mathematical reasoning. *arXiv preprint arXiv:2503.02382*.

Wei Sun, Wen Yang, Pu Jian, Qianlong Du, Fuwei Cui, Shuo Ren, and Jiajun Zhang. 2025b. Ktae: A model-free algorithm to key-tokens advantage estimation in mathematical reasoning. *arXiv preprint arXiv:2505.16826*.

Peiyi Wang, Lei Li, Zhihong Shao, RX Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. 2023. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. *arXiv preprint arXiv:2312.08935*.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, and 1 others. 2025a. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.01939*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Liyuan Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, and 1 others. 2025b. Reinforcement learning for reasoning in large language models with one training example. *arXiv preprint arXiv:2504.20571*.

Fang Wu, Weihao Xuan, Ximing Lu, Zaid Harchaoui, and Yejin Choi. 2025. The invisible leash: Why rlvr may not escape its origin. *arXiv preprint arXiv:2507.14843*.

Fei Wu, Zhenrong Zhang, Qikai Chang, Jianshu Zhang, Quan Liu, and Jun Du. 2026. Step potential advantage estimation: Harnessing intermediate confidence and correctness for efficient mathematical reasoning. *Preprint*, arXiv:2601.03823.

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. 2024. Inference scaling laws: An empirical analysis of compute-optimal inference for problem-solving with language models. *arXiv preprint arXiv:2408.00724*.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, and 1 others. 2024. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.

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

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. 2025. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *arXiv preprint arXiv:2504.13837*.

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. *arXiv preprint arXiv:2503.18892*.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. 2024. Generative verifiers: Reward modeling as next-token prediction. *arXiv preprint arXiv:2408.15240*.

Ziqi Zhao, Zhaochun Ren, Jiahong Zou, Liu Yang, Zhiwei Xu, Xuri Ge, Zhumin Chen, Xinyu Ma, Daiting Shi, Shuaiqiang Wang, Dawei Yin, and Xin Xin. 2026. Reinforced efficient reasoning via semantically diverse exploration. *Preprint*, arXiv:2601.05053.

Yixiao Zhou, Yang Li, Dongzhou Cheng, Hehe Fan, and Yu Cheng. 2026. Look inward to explore outward: Learning temperature policy from llm internal states via hierarchical rl. *arXiv preprint arXiv:2602.13035*.

## Appendix

**A Implementation Detials** **13**
A.1 Baselines . . . . . 13
A.2 RL Training Configuration . . . . . 13

**B Further Analysis** **13**
B.1 Exploratory Reasoning Dynamics . . . . . 13
B.2 Cross-domain Generalization . . . . . 13
B.3 Additional Pass@k Results . . . . . 14
B.4 Variance Across Random Seeds . . . . . 14
B.5 Case Study . . . . . 15

**C Theoretical Explanation** **15**
C.1 Problem Formulation: Entropy Collapse . . . . . 15
C.2 Stage 1: Risk-Sensitive Advantage Modulation . . . . . 15
C.3 Stage 2: Adaptive Gradient Stabilization . . . . . 16
C.4 Synthesized Interpretation . . . . . 17

**D Prompt** **17**

**E Dataset License** **17**

**F LLMs Usage Statement** **17**

## A Implementation Detials

### A.1 Baselines

- **Simple-RL-Zoo (Zeng et al., 2025)**: Based on Qwen2.5-Math-7B, trained on the math-level3-5 dataset using the standard GRPO algorithm with rule-based rewards.

- **PRIME-Zero (Cui et al., 2025a)**: An online PRM update approach that leverages implicit process rewards from rollouts and outcome labels without requiring explicit annotations.

- **OpenReasonerZero (Hu et al., 2025)**: A zero-RL baseline on Qwen2.5-7B employing the standard PPO algorithm for policy optimization.

- **Oat-Zero (Liu et al., 2025b)**: Built on Qwen2.5-Math-7B, trained with rule-based rewards using a modified Dr.GRPO algorithm that removes variance terms and applies token-level normalization in the policy loss.

- **GRPO with Entropy Adv. (Cheng et al., 2025)**: Extends RLVR training by incorporating a clipped, gradient-detached entropy term into the advantage function to encourage exploration.

- **KTAE (Sun et al., 2025b)**: A token-level advantage estimation method trained with DAPO, quantifying key-token contributions via statistical association tests and combining them with rollout-level advantages.

These baselines cover applications of fundamental RL algorithms, process-reward-based methods, and algorithms improved for specific tasks like mathematical reasoning, aiming to evaluate the effectiveness and novelty of our method from multiple perspectives.

### A.2 RL Training Configuration

We adopt the VERL framework (Sheng et al., 2024) and train our model using the optimization objective defined in Eq. 11. For both GRPO and DAPO, we use the hyperparameters in Table 4, without using entropy or KL losses. All experiments are conducted on 2 compute nodes, each equipped with 8*80GB GPUs.

| Hyperparameter | Value |
|---|---|
| Optimizer | AdamW |
| Actor learning rate | $1e^{-6}$ |
| Max prompt length | 1024 |
| Max response length | 3072 |
| Training batch size | 512 |
| Samples per prompt | 16 |
| Mini-batch size | 32 |
| Rollout temperature | 1.0 |
| Clip range $\epsilon_{\text{low}}$, $\epsilon_{\text{high}}$ | 0.2, 0.28 |
| UCAS hyperparameter $\alpha$, $\beta$ | 0.25, 0.01 |

Table 4: RL Hyperparameters

## B Further Analysis

### B.1 Exploratory Reasoning Dynamics

To further analyze the effect of UCAS training, we compute the response-level confidence scores of model outputs according to Eq. 6, measured before and after training on Qwen2.5-Math-1.5B across MATH and Olympiad datasets. We focus on the MATH and Olympiad datasets because they contain more samples and a larger number of responses whose correctness changes after training, which makes them well suited for detailed analysis. For comparability, the confidence values are normalized by subtracting the mean and dividing by the standard deviation.

Based on the correctness of the responses before and after training, the samples are categorized into three groups: (i) consistently correct (1→1), (ii) correct before but incorrect after (1→0), (iii) incorrect before but correct after (0→1), and (iiii) incorrect before and incorrect after (0→0). Figure 4 illustrates the distribution of these categories, where each point represents model's response to a given problem.

From Figure 4, we observe that for many problems correctly solved only after UCAS training (0→1), the model's confidence notably increases. In contrast, for problems that remain unsolved before and after training (0→0), the model tends to reduce its confidence, suggesting a more calibrated estimation of its own uncertainty.

### B.2 Cross-domain Generalization

To assess the generality of UCAS beyond mathematical reasoning, we conduct additional experiments on three diverse benchmarks: **Leet-Code** (Guo et al., 2024) (code generation), **Live-**

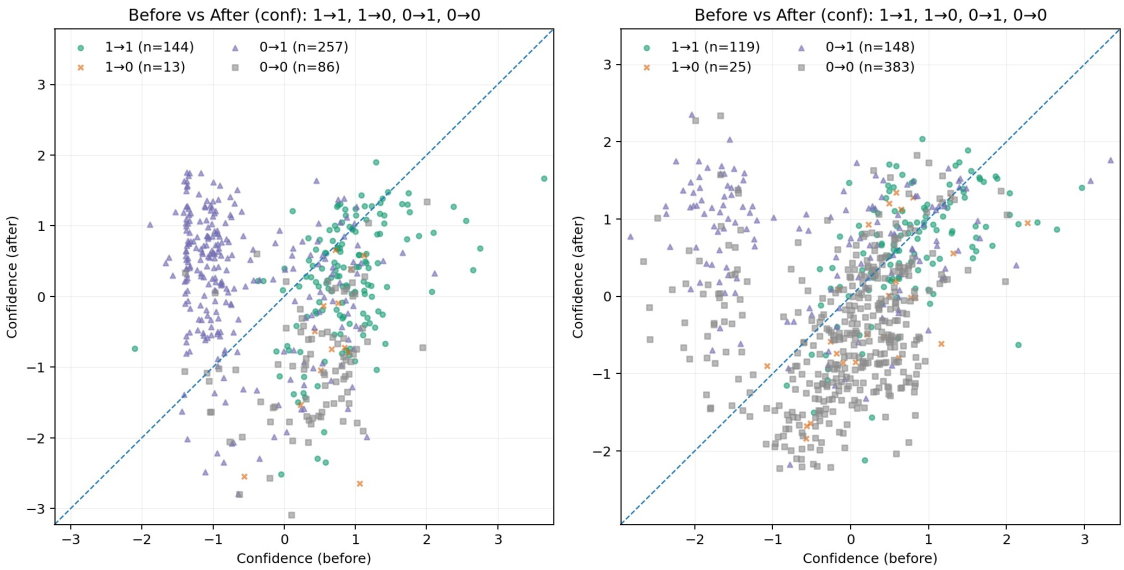


Figure 4: Confidence dynamics before and after UCAS training on the MATH and Olympiad datasets.

| Method | LeetCode (Pass@1) | LiveCode (Pass@1) | MMLU (Acc) | Avg |
|---|---|---|---|---|
| Base Model | 11.7 | 5.7 | 65.7 | 27.7 |
| DAPO | 18.3 | 9.2 | 67.3 | 31.6 |
| UCAS (Ours) | **23.6** (+5.3) | **14.8** (+5.6) | **70.8** (+3.5) | **36.4** (+4.8) |

Table 5: Generalizing UCAS from math-only training to evaluations on non-math tasks.

**CodeBench** (Jain et al., 2024) (competitive programming), and **MMLU** (Hendrycks et al., 2020) (general task reasoning). As shown in Table 5, despite being trained solely on mathematical reasoning data, UCAS consistently outperforms the strong DAPO baseline across all non-math tasks.

This strong transferability suggests that our uncertainty-aware exploration mechanism is a broadly applicable principle. By unlocking exploration for high-uncertainty paths, UCAS improves performance not just in calculation but also in the multi-step logical planning required for programming and general reasoning, demonstrating gains beyond the mathematical domain.

## B.3 Additional Pass@k Results

To further substantiate the exploration and diversity claims beyond AIME24, we evaluate pass@k on AMC and MATH-500 using Qwen2.5-Math-7B. As shown in Tables 6 and 7, UCAS exhibits consistently stronger scaling than the DAPO baseline as the sampling budget increases. The improvement is particularly pronounced on AMC, where UCAS maintains a clear advantage across all $k$ values from 4 to 64. On MATH-500, UCAS also yields higher pass@k than DAPO, indicating that the broader search behavior induced by uncertainty-aware shaping transfers beyond a single benchmark.

| Method | Pass@4 | Pass@8 | Pass@16 | Pass@32 | Pass@64 |
|---|---|---|---|---|---|
| Base Model | 65.06 | 74.70 | 80.72 | 84.34 | 87.95 |
| DAPO | 72.56 | 82.78 | 86.64 | 88.10 | 89.50 |
| UCAS | **77.80** | **87.24** | **90.12** | **92.16** | **93.58** |

Table 6: Additional pass@k results on AMC with Qwen2.5-Math-7B. UCAS consistently outperforms the DAPO baseline as the generation budget increases.

| Method | Pass@4 | Pass@8 | Pass@16 |
|---|---|---|---|
| Base Model | 74.40 | 82.60 | 88.80 |
| DAPO | 89.20 | 91.20 | 92.80 |
| UCAS | **91.40** | **92.10** | **93.40** |

Table 7: Additional pass@k results on MATH-500 with Qwen2.5-Math-7B. UCAS preserves an advantage over DAPO at each sampling budget.

## B.4 Variance Across Random Seeds

To assess the statistical reliability of the gains, we conduct four independent training and evaluation runs for both DAPO and UCAS using different random seeds on Qwen2.5-Math-7B. Table 8 reports the mean and standard deviation across runs. UCAS achieves higher mean performance than DAPO on every benchmark while maintaining comparable, and in several cases smaller, standard deviations. These results indicate that the improvements reported in the main paper are not driven by a single favorable run and that the method remains stable across random initializations.

### B.5 Case Study

As shown in Figure 6, the baseline model exhibits a critical failure in logical modeling. While it correctly identifies the symmetry of the hyperbola and the basic distance formula, it treats the vertices $B$ and $D$ as independent points on the hyperbola symmetric about the origin. Crucially, it fails to incorporate the *rhombus constraint*, which necessitates that the diagonals $AC$ and $BD$ must be perpendicular. This omission reduces the problem to a trivial minimization of the $x$-coordinate ($x^2 = 20$), resulting in an incorrect lower bound of 80.

In contrast, the model trained with UCAS (Response Parts 1 and 2) demonstrates a significantly higher degree of structural awareness. Rather than performing a simple substitution, the model engages in self-reflection to identify the implicit geometric constraints of the problem. It explicitly parameterizes the diagonals as perpendicular lines and performs a domain verification to ensure they actually intersect the hyperbola. This rigorous reasoning process allows the model to discover the hidden domain of the slope parameter and use monotonicity analysis to reach the correct ground-truth answer.

The baseline's failure highlights the tendency of standard algorithms to converge on superficial "safe" trajectories that ignore implicit constraints. UCAS, by rewarding exploration of high-uncertainty paths that maintain semantic consistency, enables the model to derive complex geometric dependencies and reach the correct ground-truth value of 480 via rigorous logical deduction and code verification.

## C Theoretical Explanation

In this section, we provide a theoretical analysis of the Uncertainty-aware Advantage Shaping (UCAS) method. We demonstrate that its heuristic components—exponential advantage modulation and min-max logit penalties—arise naturally from principles of risk-sensitive importance weighting and adaptive gradient regularization. This analysis establishes UCAS not as an ad-hoc collection of tricks, but as a coherent algorithmic framework for countering entropy collapse in sparse-reward RLVR.

### C.1 Problem Formulation: Entropy Collapse

In RLVR, reward signals are typically sparse and binary. A known pathology in this setting is *Entropy Collapse* (Cui et al., 2025b), where the policy $\pi_\theta$ prematurely converges to a deterministic distribution, severely limiting exploration. Mathematically, as the policy becomes deterministic, its entropy $H(\pi_\theta(\cdot|s)) \to 0$. Consequently, the Kullback-Leibler divergence from the policy to the uniform distribution $U(\mathcal{V})$ approaches its maximum:

$$D_{\mathrm{KL}}(\pi_\theta \| U) = \log |\mathcal{V}| - H(\pi_\theta) \to \log |\mathcal{V}|. \quad (12)$$

This divergence quantifies the policy's deviation from a maximally exploratory prior. Standard RLVR algorithms lack an explicit mechanism to penalize this deviation, often converging to local optima that exploit a narrow set of high-confidence but potentially suboptimal reasoning paths.

### C.2 Stage 1: Risk-Sensitive Advantage Modulation

The first stage applies a confidence-dependent modulation $W(\hat{\mathcal{C}}_i)$ to the advantage estimates. We first clarify that the response-level confidence used in UCAS is implemented entirely in logit space, and then interpret the resulting modulation as a form of risk-sensitive importance weighting.

**Logit-Space Confidence and Its KL Interpretation.** For token $t$ in trajectory $o_i$, let $p_{i,t}(v) = \mathrm{softmax}(\ell_{i,t})_v$ and $U(v) = 1/|\mathcal{V}|$. The token-level confidence used by UCAS is

$$C_{i,t} = \mathrm{LSE}(\ell_{i,t}) - \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \ell_{i,t,v}. \quad (13)$$

Using $\log p_{i,t}(v) = \ell_{i,t,v} - \mathrm{LSE}(\ell_{i,t})$, we obtain

$$\begin{aligned} \mathrm{KL}(U \| p_{i,t}) &= \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \left( \log \frac{1}{|\mathcal{V}|} - \log p_{i,t}(v) \right) \\ &= \mathrm{LSE}(\ell_{i,t}) - \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \ell_{i,t,v} \\ &\quad - \log |\mathcal{V}|. \end{aligned} \quad (14)$$

Therefore,

$$C_{i,t} = \mathrm{KL}(U \| p_{i,t}) + \log |\mathcal{V}|. \quad (15)$$

This equivalence shows that UCAS preserves the intended uncertainty signal while avoiding direct probability-space evaluation near zero probabilities. Compared with entropy-based quantities driven by

| Method | AIME24 | MATH-500 | AMC | Minerva | Olympiad | Avg |
|---|---|---|---|---|---|---|
| Qwen2.5-Math-7B (Base) | 11.0 | 69.0 | 45.8 | 21.3 | 28.4 | 35.1 |
| DAPO | $28.1_{\pm 3.9}$ | $79.4_{\pm 2.2}$ | $57.6_{\pm 3.8}$ | $32.4_{\pm 2.6}$ | $43.4_{\pm 2.2}$ | 48.18 |
| UCAS | $\mathbf{40.6}_{\pm 3.5}$ | $\mathbf{84.6}_{\pm 1.6}$ | $\mathbf{65.3}_{\pm 3.3}$ | $\mathbf{36.7}_{\pm 1.9}$ | $\mathbf{46.5}_{\pm 2.3}$ | **54.74** |

Table 8: Performance variance across four independent trials on Qwen2.5-Math-7B. UCAS consistently improves the mean score over DAPO with comparable variance.

expectation under $p_{i,t}$, $\mathrm{KL}(U \| p_{i,t})$ is more sensitive to support contraction and therefore better aligned with our goal of detecting premature distributional collapse.

Let $y_i = \mathrm{sign}(\hat{A}_i) \in \{+1, -1\}$ indicate the favorability of a trajectory. To encourage robust exploration, we seek a weighting distribution $w$ over trajectories within a mini-batch that emphasizes informative cases—specifically, those where the outcome $y_i$ and the confidence $\hat{\mathcal{C}}_i$ are *anti-correlated* (e.g., correct but uncertain).

We derive this by solving for the maximum entropy distribution $w$ that shifts the expected signed confidence to a target value $\mu^*$, lower than its empirical mean under a uniform prior:

$$\max_{w \in \Delta^{G-1}} \quad H(w) = -\sum_{i=1}^{G} w_i \log w_i \tag{16}$$

$$\text{s.t.} \quad \sum_{i=1}^{G} w_i = 1, \quad \sum_{i=1}^{G} w_i (y_i \hat{\mathcal{C}}_i) = \mu^*. \tag{17}$$

Solving this via Lagrange multipliers yields the Boltzmann distribution:

$$w_i^* \propto \exp\big(-\alpha \cdot y_i \cdot \hat{\mathcal{C}}_i\big), \tag{18}$$

where $\alpha > 0$ controls the intensity of the shift.

**Connection to UCAS.** The modulated advantage in UCAS, $\hat{A}_i^{\mathrm{mod}} = \hat{A}_i \cdot \exp(-\alpha y_i \hat{\mathcal{C}}_i)$, is directly proportional to this optimal importance weight $w_i^*$. Thus, Stage 1 amplifies updates from trajectories valuable for exploration (correct/uncertain) and strongly penalizes those indicative of overfitting (incorrect/confident).

### C.3 Stage 2: Adaptive Gradient Stabilization

Stage 2 addresses entropy collapse at the token level via an asymmetric penalty term. We interpret this as a mechanism for adaptive gradient stabilization that suppresses confident errors while safeguarding positive trajectories.

**Rationale for Min-Max Normalization.** The penalty uses the Min-Max normalized logit $\hat{\ell}_{i,t} \in [0, 1]$. To ensure numerical stability and fit within column constraints, we denote the sequence-level extrema as $\ell_i^{\min} = \min_k \ell_{i,k}$ and $\ell_i^{\max} = \max_k \ell_{i,k}$. The normalized logit is defined as:

$$\hat{\ell}_{i,t} = \frac{\ell_{i,t} - \ell_i^{\min}}{\ell_i^{\max} - \ell_i^{\min} + \epsilon}. \tag{19}$$

This design serves two critical functions: 1. **Scale Invariance:** It removes dependence on the absolute logit scale, which varies across layers and training stages. 2. **Relative Peak Detection:** It measures how peaked the distribution is for the chosen token *relative to other steps in the same sequence*. A value of $\hat{\ell}_{i,t} \approx 1$ indicates the model is maximally confident at step $t$ compared to its temporal neighbors, signaling a local collapse point.

**Effect on Optimization Dynamics.** The composite advantage is

$$\hat{A}_{i,t}^{\mathrm{UCAS}} = \begin{cases} \max(0, \hat{A}_i^{\mathrm{mod}} - \beta \hat{\ell}_{i,t}), & \hat{A}_i^{\mathrm{mod}} \geq 0, \\ \hat{A}_i^{\mathrm{mod}} - \beta \hat{\ell}_{i,t}, & \hat{A}_i^{\mathrm{mod}} < 0. \end{cases} \tag{20}$$

For unfavorable trajectories with $\hat{A}_i^{\mathrm{mod}} < 0$, the gradient update becomes

$$\nabla \mathcal{J} \propto \mathbb{E}\left[\big(\hat{A}_i^{\mathrm{mod}} - \beta \hat{\ell}_{i,t}\big) \nabla \log \pi(o_{i,t}|s)\right]. \tag{21}$$

This acts as a soft, adaptive gradient damper. When $\hat{\ell}_{i,t}$ is large, the effective driving signal becomes more negative, applying stronger suppression to confidently wrong transitions and preventing aggressive probability concentration on error-prone tokens. For favorable trajectories with $\hat{A}_i^{\mathrm{mod}} \geq 0$, the clamp ensures that the token-level penalty can attenuate positive updates but cannot reverse them. This context-aware regularization applies strong pressure only where needed while preserving non-negative learning signals for globally beneficial reasoning paths.

### C.4 Synthesized Interpretation

Combining both stages, UCAS optimizes a policy subject to two complementary, uncertainty-aware regularizers:

$$\pi^* \approx \arg\max_{\pi} \Big( \mathbb{E}_{\tau}[R(\tau)] + \lambda_1 \mathcal{R}_{\text{macro}}(\pi) + \lambda_2 \mathcal{R}_{\text{micro}}(\pi) \Big). \tag{22}$$

**Macro Regularizer $\mathcal{R}_{\text{macro}}$ (Stage 1)** re-weights trajectory gradients to favor correctness with lower confidence. **Micro Regularizer $\mathcal{R}_{\text{micro}}$ (Stage 2)** acts as an asymmetric constraint on token-level logits, suppressing confident errors while preserving non-negative updates on favorable trajectories. This dual-level framework aligns the global objective (robust correctness) with stable local optimization dynamics.

## D Prompt

As shown in Figure 5, we use the same prompt template (Qwen-Math template) for both RL training and evaluation.

## E Dataset License

Our training data is based on the MATH dataset (Hendrycks et al., 2021), and we evaluate our model on AIME24 (LI et al., 2024), MATH-500 (Hendrycks et al., 2021), AMC (LI et al., 2024), Minerva (Lewkowycz et al., 2022), and Olympiad-Bench (Huang et al., 2024). We strictly adhere to the licenses associated with each dataset.

## F LLMs Usage Statement

We employed a Large Language Model (LLM) to assist exclusively in the editorial stage of manuscript preparation. Its role was limited to refining phrasing, correcting grammar, and enhancing clarity and readability across different sections. The LLM had no involvement in formulating research ideas, designing experiments, or conducting analyses. All scientific contributions and findings are entirely the work of the authors. The authors have ensured that the use of the LLM complies with ethical standards, avoiding plagiarism and scientific misconduct.

## Prompt templates of RL training and Evaluation

```
<|im_start|>system\nPlease reason step by step, and put your final answer
within \\boxed{}.<|im_end|>\n<|im_start|>user\n{question}<|im_end|>\n<|im
start|>assistant\n
```

Figure 5: Prompt template in RL training and evaluation.

## Question and Ground Truth

**Question:** Let $A$, $B$, $C$, and $D$ be points on the hyperbola $\frac{x^2}{20} - \frac{y^2}{24} = 1$ such that $ABCD$ is a rhombus whose diagonals intersect at the origin. Find the greatest real number that is less than $BD^2$ for all such rhombi.

**Answer:** 480

## Response from RL Baseline

To solve this problem, we need to understand the properties of the hyperbola and the rhombus. The hyperbola is given by the equation $\frac{x^2}{20} - \frac{y^2}{24} = 1$. The diagonals of the rhombus intersect at the origin, which means the rhombus is symmetric with respect to the origin. This implies that if $(x_1, y_1)$ is a point on the hyperbola, then $(-x_1, -y_1)$, $(x_1, -y_1)$, and $(-x_1, y_1)$ are also points on the hyperbola. The diagonals of the rhombus are the line segments connecting these points.

The length of the diagonal $BD$ can be calculated using the distance formula. Since the diagonals intersect at the origin, the length of the diagonal $BD$ is the distance between $(x_1, y_1)$ and $(-x_1, -y_1)$. The distance formula gives us:

$$BD = \sqrt{(x_1 - (-x_1))^2 + (y_1 - (-y_1))^2} = \sqrt{(2x_1)^2 + (2y_1)^2} = 2\sqrt{x_1^2 + y_1^2}.$$

Thus, $BD^2 = 4(x_1^2 + y_1^2)$. Since $(x_1, y_1)$ lies on the hyperbola, we have $\frac{x_1^2}{20} - \frac{y_1^2}{24} = 1$. We can express $y_1^2$ in terms of $x_1^2$:

$$y_1^2 = 24\left(\frac{x_1^2}{20} - 1\right) = \frac{24x_1^2}{20} - 24 = \frac{6x_1^2}{5} - 24.$$

Substituting this into the expression for $BD^2$, we get:

$$BD^2 = 4\left(x_1^2 + \frac{6x_1^2}{5} - 24\right) = 4\left(\frac{5x_1^2 + 6x_1^2}{5} - 24\right) = 4\left(\frac{11x_1^2}{5} - 24\right) = \frac{44x_1^2}{5} - 96.$$

To find the greatest real number that is less than $BD^2$ for all such rhombi, we need to find the minimum value of $BD^2$. The minimum value of $BD^2$ occurs when $x_1^2$ is minimized. Since $x_1^2$ must be greater than or equal to 20 (because $\frac{x_1^2}{20} \geq 1$), the minimum value of $x_1^2$ is 20. Substituting $x_1^2 = 20$ into the expression for $BD^2$, we get:

$$BD^2 = \frac{44 \cdot 20}{5} - 96 = \frac{880}{5} - 96 = 176 - 96 = 80.$$

Therefore, the greatest real number that is less than $BD^2$ for all such rhombi is 80. Let's confirm this with Python code.

```python
# Define the function for BD^2 in terms of x1^2
def bd_squared(x1_squared):
    return (44 * x1_squared) / 5 - 96

# The minimum value of x1^2 is 20
x1_squared_min = 20
bd_squared_min = bd_squared(x1_squared_min)
print(bd_squared_min)
```

```
80.0
```

The greatest real number that is less than $BD^2$ for all such rhombi is $\boxed{80}$.

Figure 6: **Reasoning Case Study.** The baseline model (above) fails to account for the perpendicular constraint of the rhombus diagonals, reducing the problem towards a wrong path.

**Response from Our Method (Part 1)** (Continued from previous page)

To solve this, I need to find the lower bound of the length of diagonal $BD$ of a rhombus inscribed in the hyperbola $\frac{x^2}{20} - \frac{y^2}{24} = 1$.

**Step 1: Analyzing the Geometry**

Let's think about the properties of the rhombus. Since the hyperbola is symmetric about the origin, and the rhombus is inscribed, its center must be the origin $(0, 0)$. Let the vertices be $A, B, C, D$. The diagonals $AC$ and $BD$ must intersect at the origin.

*Let's try* to define the lines containing the diagonals. Crucially, the diagonals of a rhombus are perpendicular. If I let the slope of line $AC$ be $m$, then the line equation is $y = mx$. Consequently, the slope of line $BD$ must be $-1/m$, and its equation is $y = -\frac{1}{m}x$.

**Step 2: Checking Constraints (Wait, do they strictly exist?)**

I need to ensure that these lines actually intersect the hyperbola. Not all lines passing through the origin intersect a hyperbola. Let's substitute $y = kx$ into the hyperbola equation:

$$\frac{x^2}{20} - \frac{k^2x^2}{24} = 1 \implies x^2\left(\frac{1}{20} - \frac{k^2}{24}\right) = 1$$

For $x$ to be a real number, we must have $x^2 > 0$. This implies:

$$\frac{1}{20} - \frac{k^2}{24} > 0 \implies \frac{1}{20} > \frac{k^2}{24} \implies k^2 < \frac{24}{20} = \frac{6}{5}$$

*Wait, let me double-check.* Does this apply to both diagonals? Yes, for the rhombus to exist, all four vertices must be on the hyperbola. So, the condition must hold for both the slope of $AC$ (which is $m$) and the slope of $BD$ (which is $-1/m$).

Let's check the constraints for both:

1. For $AC$: $m^2 < \frac{6}{5}$
2. For $BD$: $(-\frac{1}{m})^2 < \frac{6}{5} \implies \frac{1}{m^2} < \frac{6}{5} \implies m^2 > \frac{5}{6}$

So, combining these, the range for $m^2$ is $\frac{5}{6} < m^2 < \frac{6}{5}$. This looks correct. If $m^2$ is outside this range, one of the diagonals won't hit the hyperbola.

**Step 3: Formulating the Target Function**

I need to minimize $BD^2$. Let's express the coordinates of point $B$. Point $B$ lies on the line $y = -\frac{1}{m}x$. Let's set $k_{bd} = -1/m$. From the intersection equation earlier:

$$x_B^2 = \frac{1}{\frac{1}{20} - \frac{k_{bd}^2}{24}}$$

The squared length $BD^2$ is $(2 \cdot OB)^2 = 4(x_B^2 + y_B^2) = 4x_B^2(1 + k_{bd}^2)$. Let's try to simplify this expression. Substituting $k_{bd} = -1/m$, note that $k_{bd}^2 = 1/m^2$. Let $u = m^2$. Then the slope squared for BD is $1/u$.

Wait, calculating with $1/u$ might be messy. Let's stick to the slope of $BD$ directly. Let $S$ = slope of $BD$. Then $S^2 = 1/m^2$. From the inequality $\frac{5}{6} < m^2 < \frac{6}{5}$, we have:

$$\frac{5}{6} < \frac{1}{S^2} < \frac{6}{5} \implies \frac{5}{6} < S^2 < \frac{6}{5}$$

Actually, the problem is symmetric. It doesn't matter which diagonal is which. Let's just say the slope squared of the diagonal of interest is $k^2$, where $k^2 \in (5/6, 6/5)$.

Let's re-calculate $BD^2$ with slope $k$:

$$BD^2 = 4x^2(1 + k^2) = \frac{4(1 + k^2)}{\frac{1}{20} - \frac{k^2}{24}} = \frac{4(1 + k^2)}{\frac{6-5k^2}{120}} = \frac{480(1 + k^2)}{6 - 5k^2}$$

**Step 4: Finding the Minimum (Trial and Computation)**

Now I need to minimize $f(t) = \frac{480(1+t)}{6-5t}$ for $t \in (5/6, 6/5)$. Let's analyze the monotonicity. As $t$ increases, the numerator $(1 + t)$ increases. The denominator $(6 - 5t)$ decreases (and stays positive since $t < 1.2$). Since the numerator is growing and the denominator is shrinking, the whole fraction must be strictly increasing.

*Let me verify this quickly.* If the function is increasing, the minimum value should be approached as $t \to 5/6$. Wait, the interval is open $(5/6, 6/5)$. So we strictly can't reach $5/6$. The question asks for the "greatest real number less than $BD^2$". This implies we are looking for the infimum.

So I just need to plug in $t = 5/6$ to find the limit. Let's use Python to verify the calculation and ensure I haven't made a silly arithmetic error.

**Response from Our Method (Part 2)**

```
import numpy as np

def bd_squared(t):
    # derived formula: 480(1+t) / (6-5t)
    return (480 * (1 + t)) / (6 - 5 * t)

# The range for slope squared is (5/6, 6/5)
lower_bound_t = 5/6

# Calculate the limit value
limit_val = bd_squared(lower_bound_t)
print(f"Limit at lower bound: {limit_val}")

# Let's also check a value slightly above 5/6 to confirm it's increasing
test_val = bd_squared(5/6 + 0.001)
print(f"Test value (slightly inside): {test_val}")
```

```
Limit at lower bound: 480.0
Test value (slightly inside): 481.309...
```

**Step 5: Final Conclusion**
The code confirms the limit is 480 and the function is increasing. Since $BD^2$ is strictly increasing on the interval, $BD^2 > 480$ for all valid rhombi. Thus, the greatest real number less than $BD^2$ is 480.

The greatest real number that is less than $BD^2$ for all such rhombi is $\boxed{480}$.