Rediscovering Entropy Regularization: Adaptive Coefficient Unlocks Its Potential for LLM Reinforcement Learning

Anonymous ACL submission

Abstract

Reasoning ability has become a defining capability of Large Language Models (LLMs), with Reinforcement Learning with Verifiable Rewards (RLVR) emerging as a key paradigm to enhance it. However, RLVR training often suffers from policy entropy collapse, where the policy becomes overly deterministic, hindering exploration and limiting reasoning performance. While entropy regularization is a common remedy, its effectiveness is highly sensitive to the fixed coefficient, making it unstable across tasks and models. In this work, we revisit entropy regularization in RLVR and argue that its potential has been largely underestimated. Our analysis shows that (i) tasks of varying difficulty demand distinct exploration intensities, and (ii) balanced exploration may require the policy entropy to be maintained within a moderate range below its initial level. Therefore, we propose Adaptive Entropy Regularization (AER) — a framework that dynamically balances exploration and exploitation via three components: difficulty-aware coefficient allocation, initial-anchored target entropy, and dynamic global coefficient adjustment. Experiments on multiple mathematical reasoning benchmarks show that AER consistently outperforms baselines, improving both reasoning accuracy and exploration capability. Codes are available at https://anonymous.4open.science/r/AER_.

1 Introduction

Reasoning ability has become a crucial capability for Large Language Models (LLMs) to solve complex tasks in mathematics and coding. Reinforcement Learning with Verifiable Rewards (RLVR) has recently emerged as an effective paradigm to enhance this capability, driving advances in state-of-the-art models such as OpenAI-o1 and DeepSeek-R1 (Jaech et al., 2024; Guo et al., 2025). However, recent studies observe that *policy entropy collapse* may pose a significant bottleneck in RLVR training (Cui et al., 2025b; He et al., 2025; Cheng et al., 2025; Dai et al., 2025), closely tied to the long-standing exploration–exploitation dilemma (Sutton et al., 1998). Specifically, the model's policy often converges prematurely to a narrow set of exploitative reasoning trajectories, thereby suppressing exploration of the broader solution space (Chen et al., 2025b). This premature convergence typically manifests as a rapid decline in policy entropy during the early stages of training (Yu et al., 2025), trapping the policy in local optima and leading to performance plateaus that constrain the model's overall reasoning potential (Cui et al., 2025b).

A conventional approach in reinforcement learning to alleviate policy entropy collapse is to introduce an entropy regularization term, which explicitly penalizes overly deterministic policies and encourages exploration (Schulman et al., 2017). Despite its simplicity and conceptual appeal, this technique is often omitted in recent RLVR pipelines for LLMs (Yu et al., 2025; Hu et al., 2025b; Liu et al., 2025b; Cui et al., 2025a), as its effectiveness is highly sensitive to the choice of the entropy coefficient. Small coefficients cannot prevent entropy collapse, whereas excessively large coefficients may induce entropy explosion (Cui et al., 2025b; Jiang et al., 2025b). Moreover, a slight change in the base model or dataset may flip the effect of a tuned coefficient from beneficial to harmful (He et al., 2025). Intuitively, the balance between exploration (high entropy) and exploitation (low entropy) should be dynamic throughout training. Fixed coefficients struggle to deal with this evolving trade-off (He et al., 2025; Cui et al., 2025b). This naturally raises the question:

Can we adaptively adjust the coefficient for entropy regularization during RLVR training?

In this work, we revisit entropy regularization in the context of RLVR for LLMs and argue that

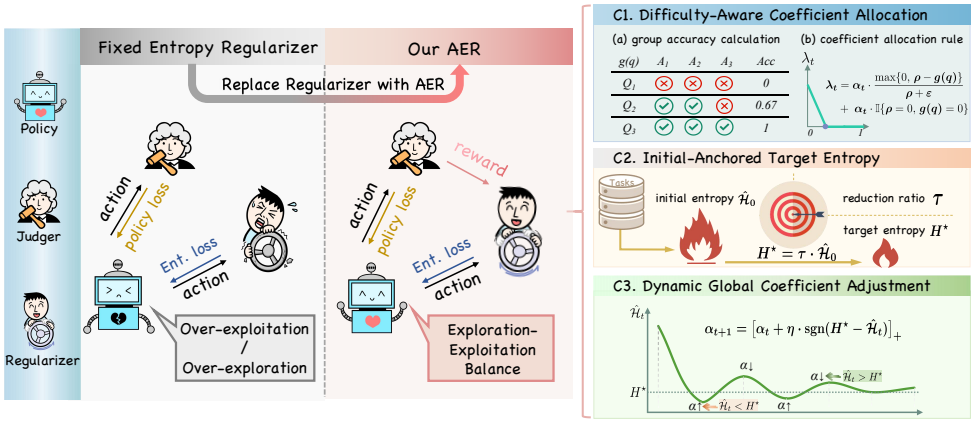


Figure 1: An overview of the AER framework.

its potential has been largely underestimated due to the limitations of fixed-coefficient designs. Motivated by this concern, we conduct preliminary experiments and analysis in Section 3 and have two observations: (*i*) tasks of different difficulty levels require distinct exploration intensities, suggesting the need for a difficulty-aware mechanism that enables sample-level control of entropy regularization; and (*ii*) effective exploration during training requires maintaining the policy entropy at a specific target value below its initial entropy,

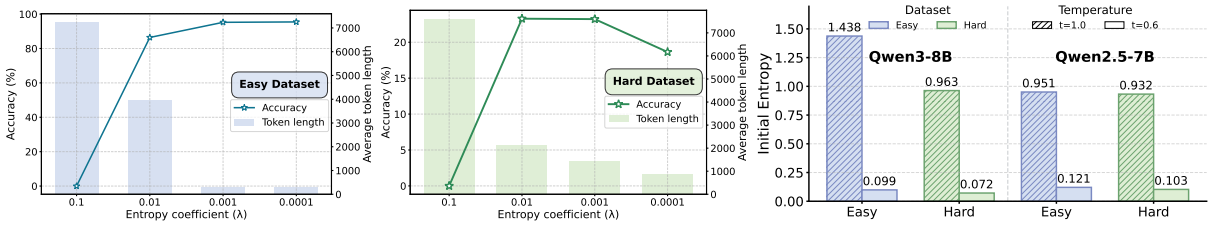
Therefore, we propose *Adaptive Entropy Regularization (AER)* as shown in Figure 1, which dynamically balances exploration and exploitation through adaptive coefficients, including three components: (*i*) *Difficulty-Aware Coefficient Allocation* estimates task difficulty relative to the current policy and assigns sample-level entropy coefficients to achieve fine-grained entropy regularization; (*ii*) *Initial-Anchored Target Entropy* adaptively determines the target entropy value based on each run's initial entropy, maintaining consistent relative exploration budget among different settings; and (*iii*) *Dynamic Global Coefficient Adjustment* adaptively adjusts a global scaling factor for entropy coefficients according to the current policy entropy to ensure that the policy entropy is maintained near the target entropy during training. Together, these components form an adaptive controller that maintains policy entropy within a reasonable range, stabilizing training while retaining balanced exploration. We conducted empirical evaluations on various complex mathematical reasoning benchmarks, and AER showed consistent improvements in reasoning performance and diversity.

Our contributions are summarized as follows:

- We conduct preliminary analysis to show that exploration should adapt to task difficulty and that balanced exploration may require maintaining the policy entropy within a moderate range below its initial level, motivating adaptive, difficulty-aware entropy regularization.
- We introduce the *Adaptive Entropy Regularization (AER)*, a framework that can dynamically and adaptively adjust the coefficients of entropy regularization to better balance exploration and exploitation throughout training.
- Extensive experiments on various mathematical reasoning benchmarks demonstrate that AER consistently outperforms advanced baselines in both reasoning performance (*pass@1*) and exploration capability (*pass@k*), validating the potential of adaptive entropy regularization in RLVR training.

2 Related Work

Reinforcement Learning for LLMs. Reinforcement learning is an important paradigm for training LLMs (Ouyang et al., 2022; Team et al., 2023; Lee et al., 2024). Recently, reinforcement learning with verifiable rewards (RLVR) has shown remarkable success for advancing the reasoning capabilities of LLMs, demonstrating significant performance in complex tasks such as mathematics and coding (Guo et al., 2025; Jaech et al., 2024; Yang et al., 2025; He et al., 2025; Team et al., 2025; Liu et al., 2025b; Zeng et al., 2025). In addition, a series of actor-only methods further reduce the resource burden and complexity of RLVR (Shao et al., 2024; Li et al., 2024; Hu et al., 2025a; Yu et al., 2025; Zheng et al., 2025a). However, RLVR still suffers from exploration-exploitation dilemma, which manifests as a rapid decrease in policy entropy, thus limiting



(a) The impact of different coefficients on the easy dataset.

(b) The impact of different coefficients on the hard dataset.

(c) Initial entropy under various settings.

Figure 2: **Preliminary experimental results.** (a-b) we show the effect of different entropy coefficients on test accuracy and average token length on easy and difficult datasets, respectively. (c) we demonstrate that different base models, datasets, and sampling temperatures will significantly affect the value of the initial entropy.

the performance of LLMs (Cui et al., 2025b; Cheng et al., 2025; He et al., 2025).

Exploration in Reinforcement Learning. Exploration is a central challenge in reinforcement learning, typically approached through theoretical analysis (Cai et al., 2020; Ishfaq et al., 2021), curiosity-driven signals (Pathak et al., 2017; Burda et al.; Raileanu and Rocktäschel; Henaff et al., 2022), and entropy maximization (Ziebart et al., 2008; Toussaint, 2009). In the context of LLMs, several studies have employed entropy as a performance indicator (Cui et al., 2025b) or as a heuristic for advantage shaping, enhancing the rollout phase, or loss masking (Wang et al., 2025; Cheng et al., 2025; Zheng et al., 2025b; Li et al., 2025d). Entropy regularization or KL penalty helps control policy distributions (He et al., 2025; Liu et al., 2025a), while complementary techniques such as loss reweighting (Wang et al., 2025; Cui et al., 2025b) and clip-higher (Yu et al., 2025) further mitigate entropy collapse. Additional strategies for promoting exploration include adjusting sampling hyperparameters (Chen et al., 2025a), performing self-reflection (Jiang et al., 2025a), leveraging external verification (Zha et al., 2025), emphasizing high-entropy tokens via critical-token training (Wang et al., 2025; Li et al., 2025c; Jiang et al., 2025b), and designing custom intrinsic signals (Li et al., 2025a; Dai et al., 2025; Gao et al., 2025; Song et al., 2025). However, the necessity of entropy regularization in RLVR remains debated, with some studies questioning its impact on exploration effectiveness (Ouyang et al., 2022; Shao et al., 2024; Hu et al., 2025b; Cui et al., 2025b).

3 Preliminary Analysis

Although adding explicit entropy regularization (e.g., an entropy loss term in the objective) is a straightforward, plug-and-play remedy to prevent the policy from becoming overly deterministic and thus mitigate entropy collapse, most recent works of RLVR for LLMs do not include this technique (Hu et al., 2025b; Liu et al., 2025b; Cui et al., 2025a; Yu et al., 2025). Yet the entropy regularization is highly sensitive to coefficients in practice, small coefficients cannot effectively prevent entropy collapse, while large coefficients will lead to entropy explosion, causing training instability or performance degradation as well (Cui et al., 2025b). Moreover, slight changes in the experimental setup can cause the carefully selected coefficients to have the opposite effect (He et al., 2025).

Intuitively, the degree of exploration should correlate with task difficulty: excessive exploration on easy tasks may introduce unnecessary randomness and hinder convergence, whereas difficult tasks often require stronger exploration to escape local optima and discover effective reasoning trajectories (Li et al., 2025a,b). To examine this intuition, we trained Qwen3-4B-Base with GRPO on mathematical datasets of different difficulty levels[1], varying the strength of entropy regularization through different coefficients. As shown in Figure 2a and Figure 2b, increasing the entropy coefficient improves test accuracy on the harder dataset, accompanied by a longer average token length—indicating that moderate promotion of exploration benefits challenging reasoning tasks. In contrast, on the easier dataset, stronger entropy regularization leads to a decline in accuracy, where excessive exploration with longer responses may prevent convergence toward concise and correct reasoning trajectories. These results demonstrate that the optimal level of exploration varies with task

[1]The easy task uses GSM8K, while the hard task consists of a mixture of AIME and AMC datasets.

difficulty, *highlighting the necessity of a difficulty-aware mechanism for entropy regularization*. In addition, when the coefficient is set to an overly large value (i.e., 0.1), both datasets showed a sudden drop in accuracy accompanied by a sharp increase in the average token length, indicating that *excessive exploration may trigger entropy explosion, leading to instability in the training process*.

A recent study (Cui et al., 2025b) establishes an empirical relation between policy entropy $\mathcal{H}$ and downstream performance $\mathcal{R}$, expressed as $\mathcal{R} = -a \exp(\mathcal{H}) + b$, where a and b are fitting coefficients. This indicates an inherent trade-off: policy performance is "purchased" at the cost of entropy. Furthermore, policy entropy decreases monotonically without any entropy intervention (Cui et al., 2025b), which means that effective exploration may require maintaining policy entropy at a "sweet spot" below its initial level to avoid entropy collapse and explosion. He et al. (2025) have similar empirical observations that they monitor policy entropy during training and preventing it from falling below a prespecified target entropy value (e.g., 0.2) from the initial entropy, thereby improving performance. However, as shown in Figure 2c, the initial entropy may vary greatly due to the differences in the base model, training data, and sampling temperature, leading to inconsistent "exploration budget". *This inspires us to design a mechanism that adaptively determines the target entropy value based on the level of initial entropy*.

4 Methodology

Building on these insights, we propose the **Adaptive Entropy Regularization (AER)** framework for RLVR. AER estimates task difficulty with respect to the current policy and adaptively adjusts the entropy coefficient at the sample level. It further sets the target entropy as a fraction of the initial policy entropy and dynamically adjusts a global scaling factor for coefficients to prevent the policy entropy from falling below this target, maintaining effective exploration throughout training.

4.1 Notations and Preliminaries

Formulation. We consider reinforcement learning with verifiable rewards (RLVR) for training LLMs. Given a question-answer pair (q, a) from the dataset $\mathcal{D}$, the policy model $\pi_\theta(\cdot \mid q)$ with parameters θ generates a response $o = (o_1, \ldots, o_T)$. A rule-based reward function $r(q, o) \in \{0, 1\}$ provides binary correctness judgment.

Group Relative Policy Optimization (GRPO). GRPO (Shao et al., 2024) extends proximal policy optimization (Schulman et al., 2017) to the group sampling setting and eliminates the need for a value network. For each question q, a group of G candidate responses $\{o_i\}_{i=1}^G$ is sampled. Then, the advantage of the i-th response is calculated by normalizing the group-level rewards $\{R_i\}_{i=1}^G := \{r(q, o_1), \ldots, r(q, o_G)\}$:

$$\hat{A}_{i,t} = \frac{r(q, o_i) - \text{mean}(\{R_i\}_{i=1}^G)}{\text{std}(\{R_i\}_{i=1}^G)}. \tag{1}$$

GRPO adopts a clipped objective and combines a KL penalty term:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^G\sim\pi_{\theta\text{ old}}(\cdot|q)} \left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\Big(\min\Big(w_{i,t}(\theta)\hat{A}_{i,t}, \quad \text{clip} \left(w_{i,t}(\theta), 1-\varepsilon, 1+\varepsilon\right)\hat{A}_{i,t}\Big) - \beta D_{\text{KL}}\left(\pi_\theta \| \pi_{\text{ref}}\right)\Big)\right] \tag{2}$$

where $w_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})}$, π_{ref} denotes the reference policy and β controls the strength of the KL penalty.

Entropy Regularization. To alleviate the policy entropy collapse, a common approach is to explicitly incorporate entropy regularization as an auxiliary objective. Formally, for an autoregressive policy π_θ, the token-level entropy at step t is defined as

$$\mathcal{H}_t = -\sum_{a\in\mathcal{V}} \pi_\theta(a \mid x, y_{<t}) \log \pi_\theta(a \mid x, y_{<t}), \tag{3}$$

where $\mathcal{V}$ denotes the vocabulary. The sequence-level entropy is the average over response positions:

$$\mathcal{H}(\pi_\theta(\cdot \mid x)) = \frac{1}{T}\sum_{t=1}^{T}\mathcal{H}_t, \tag{4}$$

with T being the response length. The entropy regularization objective is given by

$$\mathcal{J}_{\text{ent}}(\theta) = \gamma\, \mathbb{E}_{(x,y)\sim\mathcal{D}}[\mathcal{H}(\pi_\theta(\cdot \mid x))], \tag{5}$$

where γ controls the regularization strength.

Finally, the overall training objective combines this entropy term with the policy optimization objective (e.g., GRPO):

$$\mathcal{J}_{\text{total}}(\theta) = \mathcal{J}_{\text{GRPO}}(\theta) + \mathcal{J}_{\text{ent}}(\theta). \tag{6}$$

4.2 Adaptive Entropy Regularization (AER)

AER dynamically balances exploration and exploitation in RLVR training by incorporating three complementary mechanisms: (i) **Difficulty-Aware Entropy Allocation**, which adjusts sample-level entropy coefficients according to task difficulty; (ii) **Initial-Anchored Target Entropy**, which adaptively defines the target exploration level based on the model's initial entropy; and (iii) **Dynamic Global Coefficient Adjustment**, which continuously tunes the overall regularization strength to stabilize global policy entropy. Together, these components form a self-regulating system that allocates exploration where needed, adapts to different setups, and maintains stable entropy dynamics throughout training. Formally, the training objective at iteration t is:

$$\begin{aligned}\mathcal{J}_{\text{AER}}(\theta) = \; & \mathcal{J}_{\text{GRPO}}(\theta) \\ & + \; \mathbb{E}_{(x,y)\sim\mathcal{D}}\big[\lambda_t(x,y)\,\mathcal{H}(\pi_\theta(\cdot \mid x))\big],\end{aligned} \tag{7}$$

where $\mathcal{H}(\pi_\theta(\cdot \mid q))$ denotes the sequence-level entropy and $\lambda_t(q,o)$ represents the adaptive coefficient for each sample.

C1. Difficulty-Aware Coefficient Allocation. As discussed in Section 3, harder questions require stronger exploration to uncover potential reasoning trajectories, whereas excessive exploration on easy questions can hinder convergence. To achieve this balance, we introduce a mechanism to estimate each question's difficulty relative to the current policy during training. GRPO naturally supports this estimation: for each question q, the policy generates a group of m candidate responses $\{o_j\}_{j=1}^m$, from which we compute the *group accuracy* as

$$g(q) = \frac{1}{m}\sum_{j=1}^{m} r(q,o_j), \quad r(q,o_j) \in \{0,1\}. \tag{8}$$

Allocation rule. Given a pivot accuracy $\rho \in [0,1]$ and a small constant $\varepsilon > 0$ (typically 10^{-8}), we define a *difficulty-aware entropy coefficient* as

$$\begin{aligned}\lambda_t(q,o_j) = \alpha_t \cdot & \frac{\max\{0,\; \rho - g(q)\}}{\rho + \varepsilon} \\ & + \; \alpha_t \cdot \mathbb{I}\{\rho = 0,\; g(q) = 0\},\end{aligned} \tag{9}$$

where α_t is a *global entropy scaling factor* controlling the overall strength of entropy regularization, which is adaptively updated through the dynamic adjustment mechanism introduced in C3. The coefficient $\lambda_t(q,o_j)$ is positive only for hard questions ($g(q) \le \rho$) and zero otherwise. Additionally, it allocates larger entropy coefficients to more difficult questions based on the difference between the current $g(q)$ and ρ to encourage more exploration.

C2. Initial-Anchored Target Entropy. As analyzed in Section 3, maintaining the policy entropy within a certain "sweet spot" below its initial value can appropriately promote exploration and stabilize training. However, the initial policy entropy $\widehat{\mathcal{H}}_0$ varies substantially across base models, datasets, and sampling temperature, making it difficult to directly specify a specific value as the target entropy. Therefore, we seek an adaptive mechanism that can determine the value of the target entropy based on the initial entropy at each run.

Definition. Let $\widehat{\mathcal{H}}_0$ denote the policy entropy at initialization (obtained in the first training step). We define the *initial-anchored target entropy* as

$$H^\star = \tau \cdot \widehat{\mathcal{H}}_0, \qquad \tau \in (0,1), \tag{10}$$

where τ is a predefined reduction ratio representing the desired relative decrease from the initial entropy. By anchoring the target entropy to the initial entropy, it adaptively calibrates the effective "exploration budget" across different setups, thereby eliminating the need for repeated hyperparameter tuning and improving cross-run stability.

C3. Dynamic Global Coefficient Adjustment. While C1 determines *where* to allocate exploration and C2 specifies *how much* entropy should be retained, the global policy entropy can still drift from the target $H^\star$ as training proceeds. Recall that α_t in Equation 9 serves as a global scaling factor that controls the overall strength of entropy regularization. Instead of keeping α_t fixed as a hyperparameter, AER updates it dynamically according to the observed policy entropy (He et al., 2025). This mechanism follows the classical principle of *closed-loop control* (Åström and Murray, 2021), in which a system continuously measures its output, compares it to a target, and adjusts its input to reduce the deviation.

Definition. Let $\widehat{\mathcal{H}}_t$ denote the batch-level policy entropy at iteration t. With step size $\eta > 0$, the global entropy scaling factor α_t in Equation 9 is updated as

$$\alpha_{t+1} = \big[\alpha_t + \eta \cdot \operatorname{sgn}(H^\star - \widehat{\mathcal{H}}_t)\big]_+, \tag{11}$$

where $[z]_+ = \max(z,0)$. If $\widehat{\mathcal{H}}_t < H^\star$, α_t increases to encourage exploration; if $\widehat{\mathcal{H}}_t > H^\star$, α_t decreases to suppress excessive exploration.

Method	AIME24		AIME25		AMC23		MATH500		Avg.	
	pass@1	*pass@32*	*pass@1*	*pass@32*	*pass@1*	*pass@32*	*pass@1*	*pass@32*	*pass@1*	*pass@32*
Qwen3-4B-Base										
Base	9.0	40.0	3.2	27.0	34.6	86.7	52.2	92.0	24.8	61.5
GRPO	19.7	42.0	15.4	32.3	59.9	87.8	80.5	93.7	43.9	64.0
w/ Clip-Higher	20.4	32.0	20.0	40.3	59.6	89.7	81.1	93.5	45.3	63.9
w/ Ent-Adv	21.0	40.9	13.3	31.6	49.2	86.7	79.2	91.3	40.7	62.6
w/ KL-Cov	24.5	**54.6**	20.4	39.2	57.2	84.9	80.3	95.5	45.6	68.6
w/ Clip-Cov	24.0	52.5	**22.5**	41.4	65.9	92.4	**87.8**	95.9	50.1	70.6
w/ AER (Ours)	**25.2**	49.6	22.1	**48.9**	**70.2**	**94.8**	86.7	**96.6**	**51.1**	**72.5**
Qwen3-8B-Base										
Base	11.5	48.0	8.8	35.5	45.0	88.7	67.2	94.2	33.1	66.6
GRPO	20.5	47.0	18.5	34.3	62.8	88.4	82.0	94.1	46.0	66.0
w/ Clip-Higher	27.1	61.5	21.1	46.9	68.5	92.7	88.3	97.0	51.3	74.5
w/ Ent-Adv	23.2	45.0	15.8	35.2	62.5	80.6	83.5	93.4	46.3	63.6
w/ KL-Cov	26.9	58.4	20.8	42.7	65.3	92.1	80.6	96.7	48.4	72.5
w/ Clip-Cov	30.8	62.1	24.4	46.9	74.5	94.1	**90.4**	**97.8**	55.0	75.2
w/ AER (Ours)	**31.4**	**63.2**	**25.1**	**48.9**	**75.6**	**94.8**	89.4	97.0	**55.4**	**76.0**

Table 1: **Overall performance comparison.** We compared the *pass@1* and *pass@32* performance between different methods using the Qwen3-4B-Base and Qwen3-8B-Base models on four mathematical reasoning benchmarks.

This dynamic global coefficient adjustment prevents both *entropy collapse* and *entropy explosion*. By continuously monitoring and correcting deviations, AER maintains policy entropy near $H^\star$, forming a dynamic self-regulating mechanism that sustains balanced exploration without much manual tuning of hyperparameters.

Workflow of AER. Each training iteration of AER proceeds as follows: (i) estimate the group accuracy $g(q)$ for each question via Equation 8; (ii) compute difficulty-aware coefficients $\lambda_t(q, o)$ via Equation 9; (iii) optimize $\mathcal{J}_{\text{AER}}(\theta)$ in Equation 7; and (iv) update α_t according to Equation 11. This closed-loop process adaptively allocates and regulates exploration, maintaining stable entropy dynamics throughout RLVR training.

5 Experiments

5.1 Experiment Setup

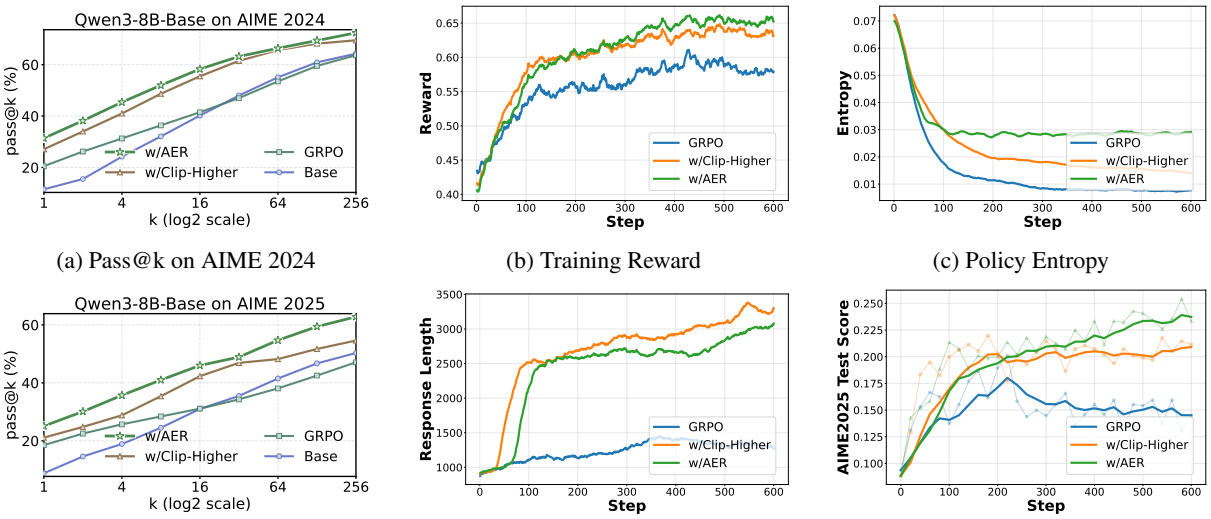
Configurations. We adopt the widely used Qwen series open-source models (Yang et al., 2025) for validating the efficacy of the proposed method, including Qwen3-4B-Base and Qwen3-8B-Base. All models are trained on the open-source DeepscaleR dataset (Luo et al., 2025). We use the Hugging Face verification tool `math_verify` (Kydlíček and Face, 2025) to automatically check the correctness of model answers. Detailed hyperparameter configuration can be found in Appendix A.

Evaluations. We follow standard protocols to evaluate mathematical reasoning and select widely adopted benchmarks: AIME24 (MAA, 2024), AIME25 (MAA, 2025), AMC23 (AI-MO, 2024) and MATH500 (Hendrycks et al., 2021). We report pass@1 to evaluate performance and pass@k to evaluate exploration capability (Yue et al., 2025). Greater exploration improves the likelihood of finding a correct reasoning path within k attempts.

Baseline methods. We compare AER against several baselines, including the standard GRPO (Shao et al., 2024), GRPO with the Clip-Higher (Yu et al., 2025) and several advanced exploration-oriented methods: Ent-Adv (Cheng et al., 2025), which introduces an entropy-based advantage term to encourage longer reasoning trajectories, and Clip-Cov and KL-Cov (Cui et al., 2025b), which regularize policy updates through token–entropy covariance to stabilize training.

5.2 Main Results

Overall Performance. Table 1 presents the main results on four mathematical reasoning benchmarks. Across both model scales, **AER** generally achieves higher performance (*pass@1*) and stronger exploration capability (*pass@32*) than the baselines. For the Qwen3-4B-Base model, AER yields an average **+7.2%** improvement in *pass@1* over vanilla GRPO and **+1.0%** over the best baseline (Clip-Cov). In terms of exploration, AER improves the average *pass@32* by **+8.5%** compared



(a) Pass@k on AIME 2024 (b) Training Reward (c) Policy Entropy

(d) Pass@k on AIME 2025 (e) Avg. Response Length (f) Test Score on AIME25

Figure 3: **Pass@k and training dynamics (2×3 grid).** Left column: pass@k on AIME24 (a) and AIME25 (d) as k scales. Right two columns: training dynamics—reward (b), policy entropy (c), response length (e)—and test score on AIME25 (f) over steps.

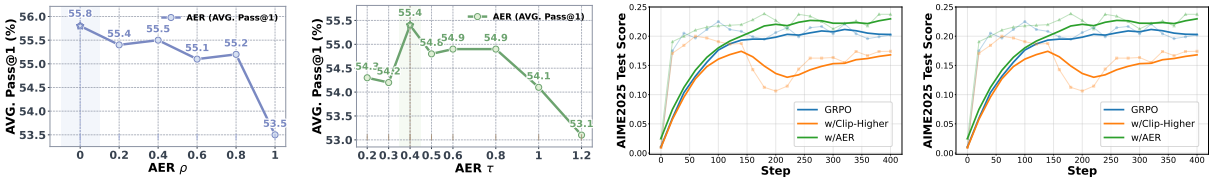
Qwen3-8B-Base	AIME24	AIME25	AMC23	MATH500
GRPO	20.5	18.5	62.8	82.0
w/ Fixed Coeff.	23.4	21.9	63.9	86.1
w/ Adaptive Coeff. (C2&C3)	29.4	21.4	73.8	89.4
w/ AER (C1&C2&C3)	**31.4**	**25.1**	**75.6**	**89.4**

Table 2: Ablation study of different components.

to vanilla GRPO and **+1.9%** compared to Clip-Cov. A similar trend is observed for the Qwen3-8B-Base model, where AER reaches the highest average scores—**55.4%** on *pass@1* and **76.0%** on *pass@32*. The gains are more noticeable on challenging benchmarks such as AIME24 and AIME25, suggesting that difficulty-aware entropy allocation may better promote exploration on harder reasoning tasks. Overall, the results indicate that AER can enhance reasoning performance and exploration diversity, highlighting its potential to further leverage entropy regularization in RLVR training for LLMs.

Pass@k Analysis. Following prior work (Yue et al., 2025; Cheng et al., 2025; Jiang et al., 2025b), we further examine the exploration boundary of different methods through pass@k performance, which reflects the model's upper bound of reasoning capability. Specifically, we evaluate the Qwen3-8B-Base model on two challenging benchmarks—AIME2024 and AIME2025—extending k up to 256 to assess how performance scales with increased rollout attempts. As shown in Figure 3a, AER consistently achieves higher pass@k performance across all k, indicating stronger exploration and reasoning diversity. The advantage becomes more pronounced as k increases on the AIME25 dataset, where AER maintains a clear margin over baselines throughout the curve. This trend suggests that AER more effectively broadens the search space of reasoning trajectories, enabling the policy to discover a larger proportion of correct solutions under extended sampling. Detailed Quantitative results can be found in Appendix C.1.

Analysis of Training Dynamics. In Figure 3b, the training reward of AER reaches a higher level than the baselines, suggesting that our method facilitates more effective policy improvement during training. As shown in Figure 3c, the policy entropy of naive GRPO drops sharply within the first 100 steps, indicating early entropy collapse. Although the baseline slightly mitigates this effect, its entropy still gradually declines over steps. AER maintains the policy entropy at a moderate target level throughout training, which help sustain consistent and balanced exploration. Despite maintaining higher entropy, AER produces shorter average response lengths than the baseline as shown in Figure 3e, implying that our method guides exploration more effectively without unnecessary verbosity. Finally, Figure 3f shows that AER dynamically maintains a stable policy entropy throughout training and allocates more exploration to difficult samples, thereby facilitating the policy to escape local optima and continuously improve performance. Overall, these results suggest that AER promotes



(a) AER ρ ablation (b) AER τ ablation (c) Qwen3-8B Test Score (d) Qwen2.5-7B Test Score

Figure 4: **AER ablations and generalization.** (a) Pivot accuracy ρ ablation (C1, τ=0.4). (b) Reduction ratio τ ablation (C2, ρ=0.2). (c) Generalization on DAPO_MATH_17K: Qwen3-8B-Base test score. (d) Generalization on DAPO_MATH_17K: Qwen2.5-7B-Base test score.

stable entropy dynamics, balanced exploration, and sustained performance gains during RLVR training.

5.3 Ablation Study

Effectiveness of different components. As shown in Table 2, building upon GRPO, introducing entropy regularization with a tuned fixed coefficient improves performance, suggesting that moderate exploration is beneficial. Then applying C2 & C3 brings further gains, particularly on challenging benchmarks such as AIME24, highlighting the importance of maintaining an appropriate level of policy entropy throughout training. When the C1 is further incorporated, the full AER framework achieves the best performance, e.g., improving AIME25 performance from 21.4 to 25.1.

Ablation of the pivot accuracy (ρ) in C1. Fixing τ=0.4, we sweep ρ to examine how it allocates exploration (Figure 4a). The *Avg. pass@1* curve increases as ρ decreases from 1.0 to 0.0, peaking at **55.8** when ρ=0, and degrading once $\rho \geq 0.6$ (down to **53.5** at ρ=1.0). The monotonic $1 \rightarrow 0$ rise confirms our *difficulty-aware* design: lowering ρ gates positives to truly hard cases, focusing entropy where exploration helps while sparing easy ones. Although C3 stabilizes the global entropy around the target, an overly large ρ makes many questions "hard," diluting per-sample selectivity and weakening the signal-to-noise ratio of entropy injection. Accordingly, we recommend a **small–moderate** pivot and adopt ρ<=0.2 by default.

Ablation of the reduction ratio (τ) in C2. Fixing ρ=0.2, we sweep τ to study how anchoring the target entropy to the initial entropy affects performance (Figure 4b). The *Avg. pass@1* curve exhibits a clear sweet spot at τ=0.4 (**55.4**), with a mild plateau for $\tau \in [0.5, 0.6]$ and degradation in both directions: *under*-reduction at $\tau \leq 0.3$ and *over*-preservation at high-entropy settings $\tau \geq 0.8$. These results substantiate the empirical premise that *moderate* entropy reduction accompanies accuracy gains: compared with naïve GRPO, AER stabilizes the global entropy *near* the prescribed band, retaining sufficient diversity while avoiding premature collapse. Pushing τ too high inflates entropy beyond the useful range, diluting the signal for exploitation; pushing it too low curtails beneficial exploration. In practice, we thus adopt $\tau \in [0.4, 0.6]$ as default.

Generalization under different experimental settings. To test whether AER's closed-loop entropy control transfers across data and architectures without retuning, we replace DEEPSCALER with DAPO_MATH_17K and train **Qwen3-8B-Base** and **Qwen2.5-7B-Base** using the same default hyperparameters (ρ=0.2, τ=0.4). The test-score trajectories in Figure 4c and Figure 4d show that AER achieves faster early gains and maintains a clear, persistent margin over *Clip-Higher* throughout training, while converging to a higher (or comparable) asymptote than *GRPO*. This behavior indicates that the initial-anchored target entropy (C2) together with dynamic global scaling (C3) adapts the effective exploration level to the new data distribution and model backbone.

6 Conclusion

This paper revisits entropy regularization in reinforcement learning with verifiable rewards and identifies fixed-coefficient schemes—often prone to entropy collapse or explosion—as a key limitation to reasoning performance. We introduce *Adaptive Entropy Regularization (AER)*, which adaptively balances exploration and exploitation throughout training. On multiple mathematical reasoning benchmarks, AER consistently improves both performance and exploration capability, revealing that adaptive coefficient control is crucial to rediscovering the potential of entropy regularization for LLM reinforcement learning.

Limitations

We discuss two practical limitations of our work. (1) Due to limited computational resources, our experiments are restricted to 4B, 7B, and 8B backbones; nevertheless, the results consistently demonstrate the efficacy of AER across these model sizes. (2) In line with most open-source studies, we train primarily on mathematical datasets because they are easy to obtain and provide accurate, verifiable rewards. This is not an important issue, and we believe they can achieve better results if more training datasets with verifiable rewards for general domains are available.

Ethics Statement

We have carefully considered the ethical implications of our research and provide the following statements:(1) Throughout this study, we have strictly followed established ethical guidelines, ensuring that our findings are reported honestly, transparently, and with full accuracy. (2) No sensitive or confidential information was used at any stage of our research. All data and materials utilized are suitable for public release. (3) The datasets employed in our experiments originate from publicly available and peer-reviewed scientific sources, supporting the transparency and reproducibility of our work. (4) We offer detailed descriptions of the datasets and the hyper-parameter configurations used in our experiments to ensure the reproducibility and clarity of our results. (5) In the interest of openness and to support future research, we have made our code available anonymously on GitHub and will fully open source it following the acceptance of our paper.

References

AI-MO. 2024. Amc 2023. https://huggingface.co/datasets/AI-MO/aimo-validation-amc.

Karl Johan Åström and Richard Murray. 2021. *Feedback systems: an introduction for scientists and engineers*. Princeton university press.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations*.

Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. 2020. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pages 1283–1294. PMLR.

Zhipeng Chen, Yingqian Min, Beichen Zhang, Jie Chen, Jinhao Jiang, Daixuan Cheng, Wayne Xin Zhao, Zheng Liu, Xu Miao, Yang Lu, et al. 2025a. An empirical study on eliciting and improving r1-like reasoning models. *arXiv preprint arXiv:2503.04548*.

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. *arXiv 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. *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, et al. 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, et al. 2025b. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*.

Runpeng Dai, Linfeng Song, Haolin Liu, Zhenwen Liang, Dian Yu, Haitao Mi, Zhaopeng Tu, Rui Liu, Tong Zheng, Hongtu Zhu, et al. 2025. Cde: Curiosity-driven exploration for efficient reinforcement learning in large language models. *arXiv preprint arXiv:2509.09675*.

Jingtong Gao, Ling Pan, Yejing Wang, Rui Zhong, Chi Lu, Qingpeng Cai, Peng Jiang, and Xiangyu Zhao. 2025. Navigate the unknown: Enhancing llm reasoning with intrinsic motivation guided exploration. *arXiv preprint arXiv:2505.17621*.

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

Jujie He, Jiacai Liu, Chris Yuhao Liu, Rui Yan, Chaojie Wang, Peng Cheng, Xiaoyu Zhang, Fuxiang Zhang, Jiacheng Xu, Wei Shen, et al. 2025. Skywork open reasoner 1 technical report. *arXiv preprint arXiv:2505.22312*.

Mikael Henaff, Roberta Raileanu, Minqi Jiang, and Tim Rocktäschel. 2022. Exploration via elliptical episodic bonuses. In *Advances in Neural Information Processing Systems*.

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*.

Jian Hu, Jason Klein Liu, Haotian Xu, and Wei Shen. 2025a. Reinforce++: An efficient rlhf algorithm with robustness to both prompt and reward models. *arXiv preprint arXiv:2501.03262*.

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

Haque Ishfaq, Qiwen Cui, Viet Nguyen, Alex Ayoub, Zhuoran Yang, Zhaoran Wang, Doina Precup, and Lin Yang. 2021. Randomized exploration in reinforcement learning with general value function approximation. In *International Conference on Machine Learning*, pages 4607–4616. PMLR.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Yuhua Jiang, Yuwen Xiong, Yufeng Yuan, Chao Xin, Wenyuan Xu, Yu Yue, Qianchuan Zhao, and Lin Yan. 2025a. Pag: Multi-turn reinforced llm self-correction with policy as generative verifier. *arXiv preprint arXiv:2506.10406*.

Yuxian Jiang, Yafu Li, Guanxu Chen, Dongrui Liu, Yu Cheng, and Jing Shao. 2025b. Rethinking entropy regularization in large reasoning models. *arXiv preprint arXiv:2509.25133*.

Hynek Kydlíček and Hugging Face. 2025. Math-verify. https://github.com/huggingface/Math-Verify.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Ren Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, et al. 2024. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback. In *International Conference on Machine Learning*, pages 26874–26901. PMLR.

Ang Li, Zhihang Yuan, Yang Zhang, Shouda Liu, and Yisen Wang. 2025a. Know when to explore: Difficulty-aware certainty as a guide for llm reinforcement learning. *arXiv preprint arXiv:2509.00125*.

Jiazheng Li, Hong Lu, Kaiyue Wen, Zaiwen Yang, Jiaxuan Gao, Hongzhou Lin, Yi Wu, and Jingzhao Zhang. 2025b. Questa: Expanding reasoning capacity in llms via question augmentation. *arXiv preprint arXiv:2507.13266*.

Qingbin Li, Rongkun Xue, Jie Wang, Ming Zhou, Zhi Li, Xiaofeng Ji, Yongqi Wang, Miao Liu, Zheming Yang, Minghui Qiu, et al. 2025c. Cure: Critical-token-guided re-concatenation for entropy-collapse prevention. *arXiv preprint arXiv:2508.11016*.

Ziheng Li, Zexu Sun, Jinman Zhao, Erxue Min, Yongcheng Zeng, Hui Wu, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Xu Chen, et al. 2025d. Staying in the sweet spot: Responsive reasoning evolution via capability-adaptive hint scaffolding. *arXiv preprint arXiv:2509.06923*.

Ziniu Li, Tian Xu, Yushun Zhang, Zhihang Lin, Yang Yu, Ruoyu Sun, and Zhi-Quan Luo. 2024. Remax: a simple, effective, and efficient reinforcement learning method for aligning large language models. In *Proceedings of the 41st International Conference on Machine Learning*, pages 29128–29163.

Mingjie Liu, Shizhe Diao, Ximing Lu, Jian Hu, Xin Dong, Yejin Choi, Jan Kautz, and Yi Dong. 2025a. Prorl: Prolonged reinforcement learning expands reasoning boundaries in large language models. *arXiv preprint arXiv:2505.24864*.

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*.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, et al. 2025. Deepscaler: Surpassing o1-preview with a 1.5 b model by scaling rl. *Notion Blog*.

MAA. 2024. American invitational mathematics examination - aime.

MAA. 2025. American invitational mathematics examination - aime.

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

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. 2017. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR.

Roberta Raileanu and Tim Rocktäschel. Ride: Rewarding impact-driven exploration for procedurally-generated environments. In *International Conference on Learning Representations*.

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

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

Yuda Song, Julia Kempe, and Remi Munos. 2025. Outcome-based exploration for llm reasoning. *arXiv preprint arXiv:2509.06941*.

Richard S Sutton, Andrew G Barto, et al. 1998. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

Marc Toussaint. 2009. Robot trajectory optimization using approximate inference. In *Proceedings of the 26th annual international conference on machine learning*, pages 1049–1056.

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

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. 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. In *Second Conference on Language Modeling*.

Kaiwen Zha, Zhengqi Gao, Maohao Shen, Zhang-Wei Hong, Duane S Boning, and Dina Katabi. 2025. Rl tango: Reinforcing generator and verifier together for language reasoning. *arXiv preprint arXiv:2505.15034*.

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

Tianyu Zheng, Tianshun Xing, Qingshui Gu, Taoran Liang, Xingwei Qu, Xin Zhou, Yizhi Li, Zhoufutu Wen, Chenghua Lin, Wenhao Huang, et al. 2025b. First return, entropy-eliciting explore. *arXiv preprint arXiv:2507.07017*.

Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al. 2008. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA.

A Implementation Details

A.1 Training Setup

We conduct all experiments on a single node equipped with **32× NVIDIA H100 (80 GB, SXM)** GPUs interconnected via NVLink. Unless otherwise noted, all methods are trained under the same computational budget (wall-clock hours and effective batch size) and with identical data pre-processing and evaluation protocols to ensure fair comparison.

RL stack and integration. For the reinforcement learning (RL) stage, we adopt VERL, a post-training framework tailored to large language models (LLMs) with verifiable or preference-based feedback. VERL provides modular APIs that integrate seamlessly with mainstream LLM infrastructures (e.g., PyTorch FSDP and Megatron-LM) and inference engines (e.g., vLLM), enabling (i) memory-efficient sharding for optimizer states and activations, (ii) flexible rollout orchestration, and (iii) pluggable algorithm components (e.g., GRPO, DAPO, and our variants). This design reduces engineering overhead and makes experimental factors (algorithmic choices, hyperparameters, and decoding policies) *isolatable and reproducible*.

Distributed training and precision. We use mixed-precision training (bf16 where supported, with automatic casting and safe gradient scaling) and distributed data parallelism via FSDP-style full sharding of parameters, gradients, and optimizer states. Gradient accumulation is employed to match the effective global batch sizes reported in Table 3. Checkpointing is performed at fixed intervals to support fault tolerance and ablations with matched training budgets.

Inference engine for rollouts and evaluation. All online rollouts and offline evaluations are executed with VLLM, an efficient LLM inference engine that supports asynchronous batching and distributed serving with a paged key–value (KV) cache. Using a single, shared engine for both training-time rollouts and test-time evaluation minimizes distribution shift induced by heterogeneous runtimes. Decoding configurations (temperature, nucleus/top-p, maximum length, stop rules) are held *constant across methods* within each experiment; pass@k metrics use a fixed k (default k=32) unless otherwise specified.

Reproducibility. We fix random seeds for data sampling, parameter initialization, and decoder sampling; we further report the exact learning rate, context length, evaluation interval, and method-specific knobs in Table 3. All results are averaged over the same evaluation protocol to control for stochasticity in sampling-based metrics.

A.2 Hyperparameters

Notation. In Table 3, `data_train_batch_size` denotes the *effective* number of sequences per optimizer step after gradient accumulation (i.e., the global batch size across 32 GPUs). `ppo_mini_batch_size` specifies the per-update minibatch for PPO-style objectives. The `kl` column is the coefficient of the KL regularizer (set to 0 for fully KL-free variants). `length` is the maximum sequence length (tokens) and is kept identical during training and evaluation to avoid truncation bias. `eval_step` is the validation interval (in optimizer steps). Method-specific switches are listed under `Others`.

Method-specific parameters:

- **Clip-Higher:** Uses asymmetric clipping ratios with $c_l = 0.2$ (lower bound) and $c_h = 0.28$ (upper bound) to allow more aggressive updates for positive advantages while constraining negative ones.
- **Ent-Adv:** Implements entropy-based advantage estimation with scaling factor $\alpha = 0.4$ and temperature parameter $\kappa = 2.0$ to balance exploration and exploitation.
- **KL-Cov:** Adds KL divergence covariance regularization with KL coefficient $\lambda_{kl} = 1.0$ and covariance ratio $\rho_{kl} = 0.002$ to stabilize training dynamics.
- **Clip-Cov:** Combines clipping with covariance regularization using symmetric clipping ratios $c_l = c_h = 1.0$ and covariance ratio $\rho_{clip} = 0.0002$ for enhanced stability.

Method	data_train_batch_size	ppo_mini_batch_size	kl	length	lr	epoch	eval_step	Others
GRPO	128	32	0.0	8k	1e-6	30	20	–
w/Clip-Higher	128	32	0.0	8k	1e-6	30	20	c_l=0.2, c_h=0.28
w/Ent-Adv	128	32	0.0	8k	1e-6	30	20	α=0.4, κ=2.0
w/KL-Cov	128	32	0.0	8k	1e-6	30	20	λ_{kl}=1.0, ρ_{kl}=0.002
w/Clip-Cov	128	32	0.0	8k	1e-6	30	20	c_l=1.0, c_h=1.0, ρ_{clip}=0.0002
w/AER	128	32	0.0	8k	1e-6	30	20	t=0.6, β=0.2, η=0.005

Table 3: **Training configurations.** We keep the compute budget and decoding policy matched across methods. The KL coefficient controls regularization strength in PPO-style objectives; DAPO removes KL entirely. Clip-Higher uses asymmetric clipping ratios (c_l, c_h). Ent-Adv introduces entropy-based advantage estimation with scaling factor α and temperature κ. KL-Cov and Clip-Cov add covariance regularization with coefficients ρ_{kl} and ρ_{clip} respectively. AER (Adaptive Entropy Regulation) introduces adaptive entropy control with target entropy t, difficulty threshold β, and adaptation rate η.

- **AER:** Implements adaptive entropy regulation with target entropy $t = 0.6$, difficulty threshold $\beta = 0.2$ for distinguishing hard vs. easy samples, and adaptation rate $\eta = 0.005$ for dynamic parameter adjustment. The method automatically maintains the target entropy level through adaptive α parameter updates.

Scheduling and optimization. Unless stated otherwise, we use AdamW with learning rate 1×10^{-6} (see Table 3) and align the number of epochs so that the *total* number of optimizer updates is comparable across baselines. Evaluation is triggered every `eval_step` updates to monitor both pass@ 1 (accuracy) and pass@ k (diversity under multi-sample decoding). For RL methods, rollout budgets (number of samples per prompt) are matched at training and validation time to ensure apples-to-apples comparisons.

Protocol fairness. All ablations modify a *single* factor at a time (e.g., toggling KL, changing t in APARL) while keeping the data curriculum, tokenizer, context length, and decoding policy fixed. This controls confounders and isolates the effect of exploration–exploitation regularization on pass@ 1 and pass@ k.

Checkpoint selection. For each method, we perform validation on the AIME2025 set every 20 optimizer steps (i.e., `eval_step`=20) using the same decoding configuration as at test time. Unless otherwise stated, the checkpoint used for reporting is the one that attains the highest pass@ 1 on AIME2025; all metrics in the main tables are computed from this selected checkpoint.

B Detailed Description of Benchmarks

To fairly evaluate mathematical reasoning ability, we need to use benchmarks that cover different types of problems, various levels of difficulty, and a range of math topics. When choosing datasets, we focus on the following points in Table 4:

Dataset	Core Description	Key Characteristics
AIME '24	High school Olympiad-level assessment from American Invitational Mathematics Examination	• 15 complex competition problems • Algebra/Geometry/Number theory focus • 3-hour time constraint design • Multi-step reasoning verification
AIME '25	High school Olympiad-level assessment from American Invitational Mathematics Examination	• 15 complex competition problems • Algebra/Geometry/Number theory focus • 3-hour time constraint design • Multi-step reasoning verification
GSM8K	Elementary school math word problem benchmark	• 8,500 graded problems • Natural language scenarios • Basic arithmetic operations • Step-by-step solution validation
MATH-500	Advanced mathematics evaluation set by OpenAI	• 500 curated problems • Formal mathematical notation • Non-standard solution analysis • Cross-domain evaluation
AMC 2023	American Mathematics Competitions system	• Tiered assessment structure • Hybrid question types • Curriculum alignment verification • Official difficulty metrics

Table 4: Comparison of Mathematical Competition Datasets

Links:

AIME '24: https://huggingface.co/datasets/HuggingFaceH4/aime_2024;
AIME '25: https://huggingface.co/datasets/HuggingFaceH4/aime_2025;
GSM8K: https://huggingface.co/datasets/openai/gsm8k;
MATH-500: https://huggingface.co/datasets/HuggingFaceH4/MATH-500;
AMC 2023: https://huggingface.co/datasets/AI-MO/aimo-validation-amc

Method	AIME2024: Pass@k (%)								
	k=1	k=2	k=4	k=8	k=16	k=32	k=64	k=128	k=256
Base	11.5	15.5	24.2	32.1	40.2	48.0	55.1	60.9	64.3
GRPO	20.5	26.2	31.3	36.4	41.5	47.0	53.5	59.5	63.7
w/Clip-Higher	27.1	34.0	41.0	48.7	55.5	61.5	65.7	68.2	69.5
w/AER	**31.4**	**38.2**	**45.4**	**52.0**	**58.3**	**63.2**	**66.4**	**69.4**	**72.4**

Table 5: **Qwen3-8B-Base on AIME2024**. Exact Pass@ k values underlying the main-text curve.

Method	AIME2025: Pass@k (%)								
	k=1	k=2	k=4	k=8	k=16	k=32	k=64	k=128	k=256
Base	8.8	14.6	18.9	24.5	31.0	35.5	41.5	46.7	50.3
GRPO	18.5	22.5	25.7	28.4	31.1	34.3	38.1	42.5	47.1
w/Clip-Higher	21.1	24.8	28.8	35.4	42.3	46.9	48.2	51.7	54.6
w/AER	**25.1**	**30.1**	**35.7**	**41.0**	**46.0**	**48.9**	**54.7**	**59.4**	**62.8**

Table 6: **Qwen3-8B-Base on AIME2025**. Exact Pass@ k values underlying the main-text curve.

C Detailed Results

C.1 Pass@k as a Function of k (Qwen3 - 8B - Base)

Table 5–6 report the exact values underlying the Pass@k–vs.–k plots on AIME2024 and AIME2025. We follow the uniform evaluation setup and the checkpoint-selection protocol in the main text (validation every 20 steps on AIME2025 and selecting the checkpoint with the highest Pass@ 1 per method). All numbers are percentages and are monotone non-decreasing in k as expected.

Statistical remarks. (1) **Small-k accuracy.** At k=32, *w/AER* attains 63.2% on AIME2024 and 48.9% on AIME2025, improving over *Base* (48.0%, 35.5%) by +15.2/+13.4 points and over *GRPO* (47.0%, 34.3%) by +16.2/+14.6 points; it also surpasses *w/Clip-Higher* (61.5%, 46.9%) by +1.7/+2.0 points. (2) **Exploration headroom.** The gain from k=32 to k=256 for *w/AER* is +9.2 points on AIME2024 (63.2→72.4) and +13.9 on AIME2025 (48.9→62.8), exceeding *w/Clip-Higher* on both splits (+8.0, +7.7), indicating a larger pool of viable solutions at higher k. (3) **Sample efficiency via k_{50}.** Define $k_{50} = \min\{k : \text{Pass@}k \geq 50\%\}$. On AIME2024, k_{50} is 8 for *w/AER*, 16 for *w/Clip-Higher*, and 64 for both *Base* and *GRPO*. On AIME2025, k_{50} is 64 for *w/AER*, 128 for *w/Clip-Higher*, 256 for *Base*, while *GRPO* does not reach 50% by k=256. These trends are consistent with the intended effect of adaptive entropy regularization.

C.2 Ablation on Reduction Ratio τ (AER, ρ=0.2)

We ablate the target-entropy ratio $\tau \in \{0.2, 0.3, 0.4, 0.5, 0.6, 0.8, 1.0, 1.2\}$ with ρ fixed at 0.2 on Qwen3-8B-Base. Results in Table 7 show that overall accuracy peaks near τ≈0.4 (Avg. = 55.4), and degrades as τ increases into the high-entropy regime (τ≥1.0: 54.1/53.1). On the hard splits (AIME2024/2025), moderate entropy reduction improves accuracy up to τ=0.4 (31.4/25.1), whereas pushing τ higher reduces performance on AIME2025. On AMC23 and MATH500, the trend is milder; AMC23 peaks at τ=0.8 (76.6), but the overall average remains best at τ=0.4. These observations support the existence of a favorable entropy band centered around $\tau \approx 0.4$.

C.3 Ablation on Difficulty Threshold ρ (AER, τ=0.4)

We ablate the difficulty threshold ρ (formerly denoted β; cf. Section §4.2) while fixing the target-entropy ratio at τ=0.4. Recall that samples with group accuracy $g(q) \leq \rho$ receive a positive entropy bonus (hard set). Table 8 shows that **narrower hard sets** (smaller ρ) generally yield higher overall accuracy: the best average is obtained at ρ=0.0 (55.8), and performance degrades when ρ is made overly inclusive (e.g.,

AER (β=0.2)	**AIME2024**	**AIME2025**	**AMC23**	**MATH500**	**Avg.**
τ=0.2	30.8	23.8	73.7	88.9	54.3
τ=0.3	29.8	23.2	75.2	88.8	54.2
τ=0.4	**31.4**	**25.1**	75.6	**89.4**	**55.4**
τ=0.5	30.5	24.0	75.6	89.3	54.8
τ=0.6	**31.4**	23.2	75.9	89.2	54.9
τ=0.8	30.0	24.1	**76.6**	88.9	54.9
τ=1.0	31.1	23.2	72.8	89.2	54.1
τ=1.2	28.7	20.7	74.2	88.8	53.1

Table 7: **AER target-entropy ablation (ρ=0.2).** Exact accuracy (%). **Avg.** is the mean over the four datasets. Best values per column are in bold.

AER (τ=0.4)	**AIME2024**	**AIME2025**	**AMC23**	**MATH500**	**Avg.**
ρ=0.0	**33.1**	24.6	75.6	**89.7**	**55.8**
ρ=0.2	31.4	25.1	75.6	89.4	55.4
ρ=0.4	30.2	**26.1**	**76.4**	89.3	55.5
ρ=0.6	32.1	23.7	75.1	89.5	55.1
ρ=0.8	29.5	**26.1**	75.9	89.4	55.2
ρ=1.0	29.4	21.4	73.8	89.4	53.5

Table 8: **AER ablation on the difficulty threshold ρ (fixed τ=0.4).** Exact accuracy (%). **Avg.** is the mean over the four datasets. Best values per column are in bold.

ρ=1.0, 53.5). This indicates that allocating the entropy budget only to the *genuinely* difficult prompts preserves exploitation on easier ones and avoids diluting exploration across too many samples. At the dataset level, AIME2025 and AMC23 achieve their peaks at moderate ρ (0.4/0.8 and 0.4, respectively), suggesting a mild benefit from targeted—but not indiscriminate—expansion of the hard set.