Adapt to Thrive! Adaptive Power-Mean Policy Optimization for Improved LLM Reasoning

Anonymous ACL submission

Abstract

Reinforcement Learning with Verifiable Rewards (RLVR) is an essential paradigm that enhances the reasoning capabilities of Large Language Models (LLMs). However, existing methods typically rely on static policy optimization schemes that misalign with the model's evolving reasoning capabilities. To address this issue, we propose **Adaptive Power-Mean Policy Optimization (APMPO)**, which comprises two main innovations: Power-Mean Policy Optimization (PMPO) and Feedback-Adaptive Clipping (FAC). Specifically, PMPO introduces a generalized power-mean objective. This enables the model to adaptively transition from the signal-amplifying behavior of the arithmetic mean to the consistency-enforcing behavior of the geometric mean. FAC adaptively adjusts clipping bounds based on real-time reward statistics to overcome the limitations of static mechanisms. Capitalizing on these innovations, APMPO improves learning dynamics and reasoning performance. Extensive experiments on nine datasets across three reasoning tasks showcase the superiority of APMPO over state-of-the-art RLVR-based baselines. For instance, APMPO boosts the average Pass@1 score on mathematical reasoning benchmarks by 3.0 points compared to GRPO when using Qwen2.5-3B-Instruct.

1 Introduction

> *The measure of intelligence is the ability to change.*
>
> *- Albert Einstein*

Enhancing the reasoning capabilities of Large Language Models (LLMs) (Hurst et al., 2024; Yang et al., 2024a) has become a central research focus. As a promising approach, reinforcement learning (RL) empowers LLMs to surpass simple pattern matching by refining their decision-making strategies based on task-specific feedback (Guo et al., 2025). Within this field, Reinforcement Learning with Verifiable Rewards (RLVR) (Wu et al., 2025; Wen et al., 2025) is recognized for effectively enhancing complex reasoning. Leveraging outcome-based rewards, RLVR has driven notable progress in multiple domains such as mathematics (Chen et al., 2025), coding (Ye et al., 2025), and multimodal reasoning (Wang et al., 2025).

A cornerstone algorithm in RLVR is Group Relative Policy Optimization (GRPO) (Shao et al., 2024), which has gained widespread attention. By eliminating the need for a value model, GRPO reduces memory consumption and streamlines the training process (Ramesh et al., 2024). These benefits have led to several GRPO variants, each designed to address specific limitations in policy optimization (Yu et al., 2025; Zhao et al., 2025). Despite these efforts, existing methods face inherent limitations that impede their wider adoption:

Limitation 1: Static objective functions misalign with the model's evolving reasoning performance. Current RLVR-based methods typically rely on fixed objective functions throughout training, ignoring the need to adapt sensitivity to reward signals at different learning phases. For instance, GRPO (Shao et al., 2024) and DAPO (Yu et al., 2025) use the arithmetic mean, which is highly sensitive to high-reward outliers. While this sensitivity can amplify rare high-reward signals, it often drives the policy to overfit specific solutions and induces entropy collapse. In contrast, GMPO (Zhao et al., 2025) adopts the geometric mean to aggregate rewards from multiple reasoning paths. Since a single low reward can sharply reduce the geometric mean, GMPO tends to discard promising but unstable reasoning paths. This makes GMPO less effective during early training when correct reasoning paths are scarce. Consequently, these methods cannot adjust their sensitivity to match evolving learning dynamics. These observations highlight the need for an *adaptive objective function* that balances the

amplification of high-value signals with sustained policy optimization for improved reasoning.

Limitation 2: Static policy optimization constraints overlook variations in reward signal stability. Standard algorithms such as GRPO employ clipping mechanisms to stabilize training. They typically enforce fixed constraint thresholds irrespective of the statistical stability of reward signals. This static design is suboptimal, since the stability of reward signals varies across training batches. When rewards within a batch are statistically stable, they provide a clearer direction for policy improvement. This allows for more aggressive updates within a wider trust region. Conversely, batches with highly fluctuating rewards reflect ambiguity in the policy's decisions and require tighter trust regions to mitigate unstable policy updates. Ignoring the stability of reward signals can lead to misguided policy updates, ultimately degrading model performance (Huang et al., 2025; Yu et al., 2025; Yoon et al., 2025). This motivates the design of an *adaptive clipping mechanism* that adjusts clipping bounds based on the real-time statistical stability of reward signals.

In light of the above limitations, this work investigates the following key research question:

How to design an adaptive algorithm that aligns learning objectives and policy optimization constraints with the model's learning process?

To answer this question, we propose **Adaptive Power-Mean Policy Optimization (APMPO)**, a novel RLVR-based algorithm designed to strengthen the reasoning capabilities of LLMs. Specifically, APMPO integrates two innovations: (1) **Power-Mean Policy Optimization (PMPO)**: To address *Limitation 1*, PMPO introduces a power-mean formulation that adaptively modulates the objective behavior between the arithmetic and geometric mean objectives. By adaptively balancing these two extremes, PMPO can mitigate training instability and promote the discovery of high-value signals. (2) **Feedback-Adaptive Clipping (FAC)**: To address *Limitation 2*, FAC introduces a feedback-driven mechanism that modulates clipping bounds based on statistical stability of real-time reward signals. In this design, reward signals serve as feedback proxies for evaluating policy reliability. Consistent reward feedback requires larger policy updates, whereas unstable reward feedback demands stricter policy constraints. Collectively, these innovations enable APMPO to achieve adaptive policy optimization. Extensive experiments on nine benchmarks spanning three reasoning tasks demonstrate the superiority of APMPO over state-of-the-art RLVR-based baselines.

Before delving into the details, the main contributions of this work are summarized as follows:

(1) We identify and formalize two critical limitations in current RLVR-based methods, particularly the static nature of their objective functions and clipping mechanisms. These limitations hinder the adaptive tuning of policy optimization strategies in response to changes in model capability.

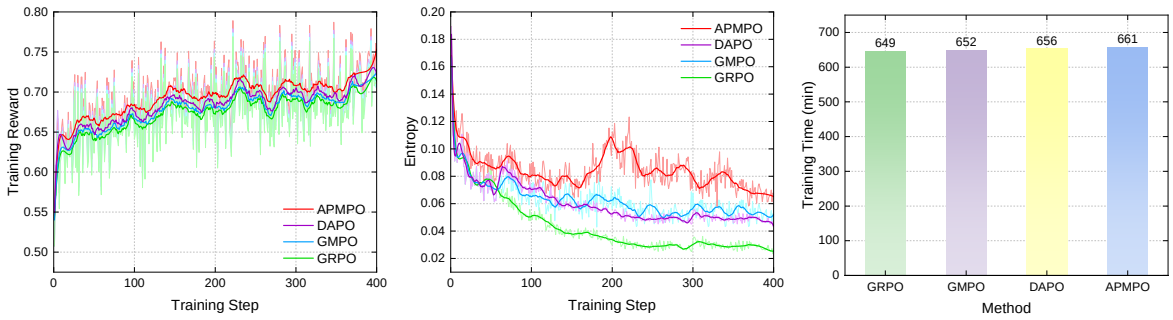
(2) We introduce APMPO, a novel RLVR-based algorithm that integrates a power mean-based objective function with a feedback-adaptive clipping mechanism. This design enables continuous adjustment of policy optimization strategies based on real-time training dynamics.

(3) Extensive experimental results on nine datasets across three tasks substantiate the superiority of APMPO in enhancing the reasoning accuracy of LLMs.

2 Related Work

GRPO (Shao et al., 2024) has become prominent in RLVR. Its core innovation lies in directly estimating advantages from reward scores across multiple sampled solutions, thereby eliminating the need for an explicit value model. To promote stable training, GRPO constrained policy updates via a clipping function controlled by a fixed hyperparameter. This approach has inspired several variants aimed at enhancing the reasoning performance of LLMs (Yu et al., 2025; Liu et al., 2025; Chu et al., 2025; Zhao et al., 2025). For instance, DAPO (Yu et al., 2025) introduced dynamic sampling to explicitly filter out zero-advantage samples, thereby improving gradient quality at the cost of reduced data efficiency. GMPO (Zhao et al., 2025) tackled training instability by optimizing the geometric mean of rewards instead of the arithmetic mean, making it inherently less sensitive to reward outliers. Additional details on these algorithms are provided in Appendix A.

Despite these advances, existing approaches remain constrained by static design choices. Specifically, objective functions and trust-region bounds remain static across all training batches. This static design limits their adaptability to varying reward distributions and training dynamics. As a result, the ability to adaptively adjust policy optimization behavior in response to real-time learning signals is critical for sustained performance improvement.



(a) Training dynamics of reward curves using different RL-based methods. (b) Training dynamics of entropy curves using different RL-based methods. (c) Wall-clock training time using different RL-based methods.

Figure 1: Illustrations of training dynamics in terms of training rewards, policy entropy, and training time using the MATH training dataset.

In light of this, we propose APMPO to promote superior policy optimization.

3 Preliminary Analysis

The motivation for APMPO arises from a preliminary analysis of the training dynamics in GRPO and GMPO, centering on training reward and policy entropy.

3.1 Motivation for PMPO: The Dilemma of Static Objective Functions

Figures 1(a) and (b) reveal the inherent limitations of static objective functions. Specifically, the arithmetic-mean objective in GRPO exhibits *excessive sensitivity to high-reward outliers*. While this sensitivity facilitates early identification of promising signals, it often drives the policy toward early convergence on suboptimal strategies. This is evidenced by a sharp entropy collapse (green curve).

In contrast, GMPO applies a geometric-mean objective to enforce consistency. This approach alleviates entropy collapse and path-specific overfitting, leading to superior policy updates than GRPO (blue curve). Nonetheless, this strict consensus requirement reduces sensitivity. Mathematically, the geometric mean functions as a global filter that weakens the influence of distinct high-value signals. As a result, GMPO risks impeding the acquisition of correct reasoning in early training phases.

These findings emphasize the need for an adaptive objective function that *balances signal amplification with distributional consistency*. This insight directly motivates PMPO, which adaptively interpolates between these distinct learning patterns. A detailed analysis of GRPO and GMPO is presented in Appendix C.

3.2 Motivation for Adaptive Clipping: The Need for Adaptive Trust Regions

Current RL-based methods employ fixed clipping hyperparameters to impose static constraints on policy updates. However, this design is suboptimal given the continual changes in reward distributions. Notably, the raw reward signals in Figure 1(a) (thin lines) fluctuate significantly, reflecting disparate levels in statistical stability across batches. Current methods inherently assume stationary reward statistics, an assumption that rarely holds in practice. Consequently, static clipping bounds fail to adapt to the reward distributional shifts. They can be overly restrictive for batches exhibiting high reward stability and excessively permissive for those with unstable reward patterns. Therefore, the policy struggles to converge during stable phases and remains exposed to detrimental updates during unstable phases. These observations motivate the design of FAC, which adaptively modulates clipping bounds based on the real-time statistical properties of the reward signal.

4 Methodology

As depicted in Figure 2, APMPO is proposed to enhance reasoning capabilities of LLMs. It consists of Power-Mean Policy Optimization (PMPO) and Feedback-Adaptive Clipping (FAC). In the following, some notations and preliminaries are briefly introduced before each of its key innovations is elaborated in detail.

4.1 Notations and Preliminaries

Group Sampling. For each input query q, a group of G responses is sampled from the old policy $\pi_{\theta_{\text{old}}}$:

$$\{o_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(\cdot|q) \tag{1}$$

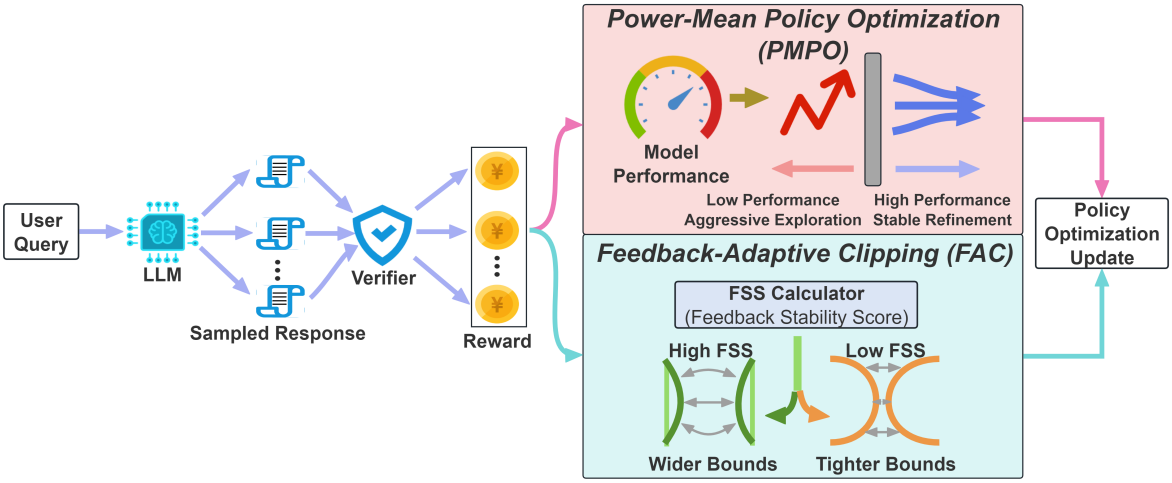


Figure 2: Illustration of APMPO, which consists of Power-Mean Policy Optimization (PMPO) and Feedback-Adaptive Clipping (FAC).

where each o_i comprises a sequence of tokens $\{o_{i,1}, o_{i,2}, \dots\}$, and a scalar reward R_i is assigned to each completed response.

Group-Normalized Advantage. The advantage $\hat{A}_i$ is computed for each sample given as:

$$\hat{A}_i = \frac{R_i - \mu_R}{\sigma_R + \delta} \tag{2}$$

where μ_R and σ_R denote the mean and standard deviation of rewards within the group.

Importance Sampling Ratio. For each token $o_{i,t}$, the importance sampling ratio $r_{i,t}(\theta)$ measures the change in action probability between the current policy π_θ and the old policy $\pi_{\theta_{\text{old}}}$:

$$r_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q, o_{i,<t})} \tag{3}$$

4.2 Power-Mean Policy Optimization (PMPO)

Existing RL-based methods typically employ static objective functions that exhibit contrasting limitations. Concretely, the arithmetic-mean objective in GRPO induces overly aggressive policy updates by prioritizing outliers, thereby causing mode collapse. Conversely, conservative formulations such as GMPO overly penalize reward variance and hinder the amplification of high-reward outliers.

To resolve this dilemma, we introduce **Power-Mean Policy Optimization (PMPO)** that adaptively interpolates between these two learning patterns. The key insight is that the degree of policy-update aggressiveness can be explicitly controlled by the choice of the mean operator. This design builds on the theoretical insight that both the arithmetic and geometric means are special cases of the generalized power mean (see Appendix D). Accordingly, PMPO employs a generalized power mean where the exponent p is adaptively determined by the model's real-time performance. Specifically, we use the average reward of each batch $\mu_R \in [0, 1]$, which reflects the model's learning progress. To facilitate a smooth transition from signal discovery to stable refinement, we adopt an exponential decay function for the exponent p:

$$p = \exp(-\gamma \cdot \mu_R) \tag{4}$$

where γ regulates the sensitivity of p to performance variations. This formulation naturally implements a performance-driven scheduling strategy:

(1) **Exploration Phase (Low μ_R):** When performance is low, p approaches 1. This encourages signal amplification by prioritizing high-reward outliers, thereby facilitating the rapid discovery of rare correct solutions.

(2) **Consolidation Phase (High μ_R):** As performance improves, p decays toward 0. This transitions the policy into a consistency-enforcing phase, prioritizing reward stabilization to regularize the policy against overfitting.

This adaptive mechanism empowers PMPO to continuously adjust its learning dynamics, seamlessly interpolating between local sensitivity and global stability within a unified framework. The complete objective function formulation is detailed in Section 4.4 (See Appendix E for more analysis).

4.3 Feedback-Adaptive Clipping (FAC)

Current RL-based approaches typically utilize a fixed clipping ratio ϵ to constrain policy updates,

thereby enforcing a uniform trust region across all batches. Nonetheless, this static design cannot adapt to the fluctuating statistical stability of training batches. Consequently, a fixed ϵ is overly restrictive for statistically stable batches while remaining overly permissive for noisy batches.

To address this limitation, we propose **Feedback-Adaptive Clipping (FAC)**, which adaptively adjusts clipping bounds based on the real-time reward distributions. FAC assesses the reward feedback to estimate its suitability for guiding policy updates. The key insight is that a high-stability feedback batch exhibits a strong average reward signal and high internal consistency. Accordingly, we formalize this notion by using the *Feedback Stability Score (FSS)* as follows:

$$\text{FSS} = \frac{\mu_R}{\sigma_R + \delta} \tag{5}$$

where μ_R and σ_R are the mean and standard deviation of the rewards within a batch, respectively. δ is a small constant for numerical stability. Notably, FSS serves as a critical non-linear scaling factor that modulates the effective signal magnitude. Intuitively, a higher FSS signifies a high-confidence batch with stable feedback, thereby justifying an expanded trust region to accelerate learning. Conversely, a low FSS implies that the signal is dominated by high uncertainty, necessitating a conservative clipping bound to alleviate potential policy degradation. Additional analysis on FSS is shown in Appendix G.

Based on the computed FSS, the clipping bound is adaptively updated. Specifically, FSS is mapped to an adaptive upper clipping bound ϵ_{ada} given as:

$$\epsilon_{\text{ada}} = \epsilon_{\text{min}} + (\epsilon_{\text{max}} - \epsilon_{\text{min}}) \cdot \tanh(\text{FSS}) \tag{6}$$

where ϵ_{min} and ϵ_{max} are the predefined minimum and maximum clipping bounds, respectively. The $\tanh(\cdot)$ function smoothly maps unbounded FSS values to the range $(0, 1)$, enabling the clipping bound to adaptively widen in stable phases and narrow in unstable phases. A detailed analysis of FAC is provided in Appendix F.

4.4 RL Training

PMPO and FAC are jointly integrated to construct the final objective of APMPO. The training process proceeds in three logically connected steps:

Step 1: An adaptive clipping function $\rho_{i,t}(\theta)$ is defined to establish an asymmetric trust region. While the lower bound is fixed to ensure baseline stability, the upper bound is adaptively modulated by ϵ_{ada} from Eq. (6) to adapt to signal quality:

$$\rho_{i,t}(\theta) = \max\left[1 - \epsilon_{\text{low}}, \min(r_{i,t}(\theta), 1 + \epsilon_{\text{ada}})\right] \tag{7}$$

where ϵ_{low} is a fixed lower bound to prevent excessive policy changes. This mechanism allows the policy to aggressively exploit high-quality signals while remaining conservative under noisy feedback. Further analysis on this asymmetric design is shown in Appendix K.4.

Step 2: To satisfy the non-negative constraint of the power-mean operator, the computation is decoupled into a non-negative magnitude term and a directional term. The token-level magnitude $\phi_{i,t}(\theta)$ is defined as:

$$\phi_{i,t}(\theta) = \left|\min\left(r_{i,t}(\theta)\hat{A}_i, \rho_{i,t}(\theta)\hat{A}_i\right)\right| \tag{8}$$

Step 3: Finally, APMPO aggregates the token-level magnitudes via the power-mean operator and introduces a directional control term. The per-sequence objective is defined as:

$$\mathcal{J}_i(\theta) = \underbrace{\left[\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}(\phi_{i,t}(\theta))^p\right]^{1/p}}_{\text{PMPO}} \cdot \underbrace{\text{sgn}(\hat{A}_i)}_{\text{Directional Control}} \tag{9}$$

where $\text{sgn}(\hat{A}_i) \in \{-1, 1\}$ ensures that positive advantages drive policy maximization while negative advantages incur penalties. The complete objective function $\mathcal{J}(\theta)$ is then given as:

$$\mathcal{J}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\mathcal{J}_i(\theta) - \beta D_{\text{KL}}(\pi_\theta || \pi_{\theta_{\text{ref}}}) \tag{10}$$

where $\pi_{\theta_{\text{ref}}}$ is the reference policy. A detailed gradient derivation, convergence analysis, and pseudocode are provided in Appendices H, I, and J, respectively.

5 Experiments

In this study, we conducted a systematic evaluation to address the following research questions:

RQ1: How does APMPO compare to state-of-the-art RLVR-based baselines? **RQ2:** Does APMPO achieve superior training performance than previous methods? **RQ3:** What are the contributions of PMPO and FAC to overall performance? **RQ4:** How sensitive is APMPO to different values of γ, ϵ_{min}, and ϵ_{max}?

Method	Math500	AIME24	AIME25	AMC23	Minerva	Olympiad	Avg.
			Qwen2.5-Math-1.5B-Instruct				
Base	74.2	10.0 / 16.7	3.3 / 10.0	47.5 / 70.0	28.7	35.2	33.2 / 32.2
GRPO	$75.2_{\pm 0.4}$	$13.3_{\pm 0.0}$ / $16.7_{\pm 0.0}$	$13.3_{\pm 0.0}$ / $16.7_{\pm 0.0}$	$52.5_{\pm 1.4}$ / $75.0_{\pm 0.0}$	$29.4_{\pm 0.3}$	$39.0_{\pm 0.6}$	$37.1_{\pm 0.5}$ / $36.1_{\pm 0.0}$
DAPO	$77.2_{\pm 0.5}$	$16.7_{\pm 1.9}$ / $23.3_{\pm 0.0}$	$16.7_{\pm 1.9}$ / $23.3_{\pm 0.0}$	$57.5_{\pm 0.0}$ / $80.0_{\pm 1.4}$	$29.0_{\pm 0.4}$	$40.4_{\pm 0.6}$	$39.6_{\pm 0.9}$ / $42.2_{\pm 0.5}$
GMPO	$76.6_{\pm 0.4}$	$13.3_{\pm 0.0}$ / $20.0_{\pm 1.9}$	**20.0**$_{\pm 0.0}$ / **26.7**$_{\pm 0.0}$	$55.0_{\pm 1.4}$ / $82.5_{\pm 0.0}$	$30.1_{\pm 0.3}$	$38.7_{\pm 0.5}$	$39.0_{\pm 0.4}$ / $43.1_{\pm 0.6}$
APMPO	**78.0**$_{\pm 0.3}$	**20.0**$_{\pm 0.0}$ / **30.0**$_{\pm 0.0}$	$16.7_{\pm 0.0}$ / **26.7**$_{\pm 0.0}$	**62.5**$_{\pm 0.0}$ / **85.0**$_{\pm 0.0}$	**30.5**$_{\pm 0.2}$	**42.4**$_{\pm 0.3}$	**41.7**$_{\pm 0.1}$ / **47.2**$_{\pm 0.0}$
			Qwen2.5-3B-Instruct				
Base	62.0	0.0 / 3.3	0.0 / 6.7	35.0 / 55.0	24.3	29.1	25.1 / 21.7
GRPO	$66.0_{\pm 0.4}$	$6.7_{\pm 0.0}$ / $13.3_{\pm 1.9}$	$6.7_{\pm 0.0}$ / $13.3_{\pm 0.0}$	$40.0_{\pm 0.0}$ / $60.0_{\pm 1.4}$	$25.4_{\pm 0.3}$	$31.5_{\pm 0.4}$	$29.4_{\pm 0.2}$ / $28.9_{\pm 1.1}$
DAPO	$67.6_{\pm 0.4}$	$6.7_{\pm 1.9}$ / $16.7_{\pm 0.0}$	**10.0**$_{\pm 0.0}$ / **20.0**$_{\pm 0.0}$	$45.0_{\pm 1.4}$ / $65.0_{\pm 0.0}$	$26.8_{\pm 0.4}$	$32.6_{\pm 0.5}$	$31.5_{\pm 0.8}$ / $33.9_{\pm 0.0}$
GMPO	$66.8_{\pm 0.5}$	$6.7_{\pm 0.0}$ / $16.7_{\pm 0.0}$	**10.0**$_{\pm 0.0}$ / $16.7_{\pm 1.9}$	$42.5_{\pm 0.0}$ / $60.0_{\pm 1.4}$	$26.1_{\pm 0.3}$	$32.2_{\pm 0.4}$	$30.7_{\pm 0.2}$ / $31.1_{\pm 1.1}$
APMPO	**68.4**$_{\pm 0.2}$	**10.0**$_{\pm 0.0}$ / **20.0**$_{\pm 0.0}$	**10.0**$_{\pm 0.0}$ / **20.0**$_{\pm 0.0}$	**45.0**$_{\pm 0.0}$ / **70.0**$_{\pm 0.0}$	**27.9**$_{\pm 0.2}$	**33.2**$_{\pm 0.3}$	**32.4**$_{\pm 0.1}$ / **36.7**$_{\pm 0.0}$
			DeepSeek-R1-Distill-Qwen-1.5B				
Base	64.6	6.7 / 26.7	13.3 / 40.0	47.5 / 70.0	24.6	30.9	31.3 / 45.6
GRPO	$75.4_{\pm 0.5}$	$13.3_{\pm 0.0}$ / $33.3_{\pm 1.9}$	$20.0_{\pm 1.9}$ / $43.3_{\pm 0.0}$	$57.5_{\pm 1.4}$ / $82.5_{\pm 0.0}$	$29.8_{\pm 0.3}$	$43.2_{\pm 0.5}$	$39.9_{\pm 0.8}$ / $53.0_{\pm 0.6}$
DAPO	$79.8_{\pm 0.4}$	$20.0_{\pm 1.9}$ / $46.7_{\pm 0.0}$	$23.3_{\pm 0.0}$ / **50.0**$_{\pm 0.0}$	$60.0_{\pm 0.0}$ / $90.0_{\pm 1.4}$	$30.1_{\pm 0.4}$	$43.8_{\pm 0.5}$	$42.8_{\pm 0.5}$ / $62.2_{\pm 0.5}$
GMPO	$76.6_{\pm 0.6}$	$16.7_{\pm 0.0}$ / $43.3_{\pm 1.9}$	$23.3_{\pm 1.9}$ / $46.7_{\pm 0.0}$	$62.5_{\pm 1.4}$ / $87.5_{\pm 0.0}$	$30.9_{\pm 0.5}$	$44.8_{\pm 0.6}$	$42.5_{\pm 0.8}$ / $59.2_{\pm 0.6}$
APMPO	**81.6**$_{\pm 0.3}$	**23.3**$_{\pm 0.0}$ / **50.0**$_{\pm 0.0}$	**26.7**$_{\pm 0.0}$ / **50.0**$_{\pm 0.0}$	**65.0**$_{\pm 0.0}$ / **92.5**$_{\pm 0.0}$	**32.7**$_{\pm 0.2}$	**46.6**$_{\pm 0.4}$	**46.0**$_{\pm 0.2}$ / **64.2**$_{\pm 0.0}$

Table 1: Experimental results on multiple mathematical reasoning benchmarks. The results are reported as mean and standard deviation across 3 random seeds (format: $\text{Mean}_{\pm \text{Std}}$). The best results are highlighted in bold. Note that "$\cdot/\cdot$" indicates "Pass@1/Pass@16", and the last column indicates "Average Pass@1 / Average Pass@16".

5.1 Settings

Models. In this work, Qwen2.5-Math-1.5B-Instruct (Yang et al., 2024b), Qwen2.5-3B-Instruct (Yang et al., 2024a), and DeepSeek-R1-Distill-Qwen-1.5B (Guo et al., 2025) were employed for mathematical reasoning. For SQL generation and multi-modal reasoning, we utilized Qwen2.5-Coder-3B-Instruct (Hui et al., 2024) and Qwen2.5-VL-3B-Instruct (Bai et al., 2025), respectively.

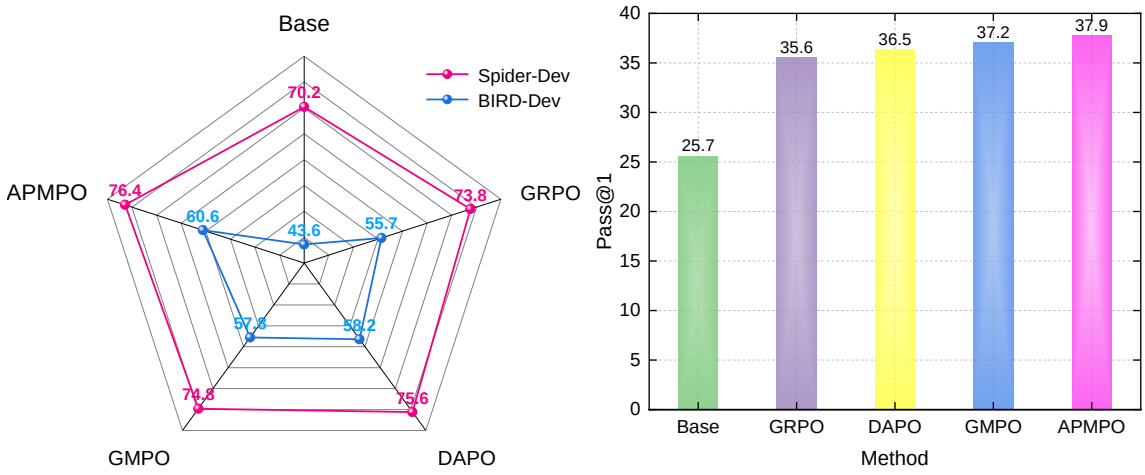
Datasets. For mathematical reasoning, training was performed on the MATH dataset (Hendrycks et al., 2021). Moreover, MATH500 (Hendrycks et al., 2021), AIME24 (Li et al., 2024a), AIME25 (Codeforces), AMC23 (Ouyang et al., 2022), Minerva (Lewkowycz et al., 2022), and OlympiadBench (Huang et al., 2024) were used for evaluation. For SQL generation, BIRD-Train (Li et al., 2024b) served as the training set, and we evaluated on Spider-Dev (Yu et al., 2018) and BIRD-Dev (Li et al., 2024b). For multi-modal reasoning, we used Geometry3K (Lu et al., 2021), which included dedicated training and test subsets. Further details of models and datasets are given in Appendix B.

Implementation Details. For all experiments, we employed a binary reward function where each response received a reward of 1 if it was correct and 0 otherwise. During training, the coefficient of KL loss term was $\beta = 0.001$. The batch size and the number of rollouts were 512 and 8, and we used a sampling temperature of 1.0. We used the AdamW optimizer (Zhou et al., 2024) with a learning rate of 1×10^{-6} and trained for 400 steps. For APMPO, γ, ϵ_{min}, ϵ_{max}, ϵ_{low}, and δ were fixed at 0.8, 0.2, 0.4, 0.2, and 1×10^{-6}, respectively. During evaluation, the sampling temperature was set to 0.6, and Pass@1 was used as the primary metric. Pass@16 was also reported for small-sized datasets (*i.e.*, AIME24, AIME25, and AMC23). For SQL generation, the reward was based on execution accuracy. All experiments were conducted on four NVIDIA GeForce A100 40GB GPUs.

5.2 Experimental Results

5.2.1 Main Results (RQ1)

As presented in Table 1, APMPO consistently outperforms all baselines across mathematical reasoning benchmarks. For instance, on Qwen2.5-Math-1.5B-Instruct, APMPO achieves an average score of 41.7, surpassing the strongest baseline (DAPO) by 2.1 points. This consistent superiority across diverse model sizes and architectures confirms the superiority of APMPO. Additionally, we evaluate APMPO on SQL generation and multi-modal reasoning to showcase its broad applicability. The results in Figure 3 show consistent improvements in SQL generation and multi-modal reasoning tasks. For instance, APMPO achieves the highest Pass@1 scores of 75.4 and 57.6 on Spider-Dev and BIRD-Dev, respectively. Further results are shown in



(a) Experimental results on Spider-Dev and BIRD-Dev by using different RL-based methods.

(b) Experimental results on Geometry3K by using different RL-based methods.

Figure 3: Experimental results on (a) SQL generation, and (b) multi-modal reasoning tasks.

Appendix L.1.

5.2.2 Analysis of Training Dynamics (RQ2)

Training Rewards and Entropy. Figure 1(a) indicates that APMPO (red curve) achieves higher training rewards than all baselines, which correlates with the policy entropy trends in Figure 1(b). While other methods exhibit a notable entropy collapse, APMPO maintains higher entropy throughout the majority of training. This sustained entropy signifies better regularization, allowing the model to explore diverse reasoning paths before converging. In essence, APMPO alleviates the early collapse of GRPO and and mitigates GMPO's overly conservative exploration, leading to stronger final performance. Further analysis on the output diversity of APMPO is shown in Appendix K.6.

Training Efficiency. Figure 1(c) presents the computational efficiency of all compared methods on Qwen2.5-Math-1.5B-Instruct. The adaptive mechanisms in APMPO are computationally lightweight, where computing p and ϵ_{ada} requires only batch-level statistics (*i.e.*, mean and standard deviation). As a result, APMPO incurs negligible computational overhead while delivering substantial performance gains. A further analysis on training efficiency is provided in Appendix K.1.

5.2.3 Ablation Study (RQ3)

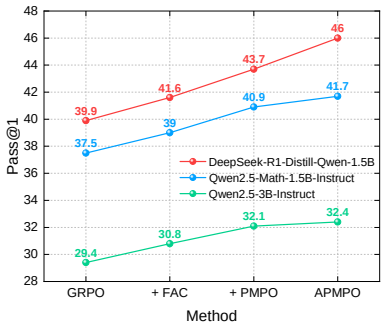
Efficacy of PMPO and FAC. As shown in Figure 4(a), an ablation study is performed on mathematical reasoning to quantify the contribution of each component. Starting from GRPO, incorporating FAC alone yields consistent performance gains, indicating that adaptively adjusting clipping bounds benefited training. In contrast, applying PMPO alone leads to larger improvements, suggesting that handling the exploration-stability trade-off via an adaptive objective is the primary source of these gains. Combining PMPO and FAC achieves the highest scores across all benchmarks, thereby showcasing a clear synergistic effect. A more in-depth analysis and additional results are shown in Appendices K.2 and L.2, respectively.

Ablation of FSS Components. The full FSS is compared with variants using only the numerator (μ_R) and the denominator ($1/\sigma_R$). Results in Figure 4(b) reveal the superiority of the complete FSS. Specifically, relying solely on μ_R offers limited sensitivity to variations in reward stability. Conversely, dependence only on $1/\sigma_R$ favors batches where reward signals show minimal variation, potentially promoting consistent failure. These findings indicate that the efficacy of FSS stems from its specific formulation, which amplifies high-confidence successful signals while suppressing unreliable or consistently incorrect outcomes. Further analyses of FSS and additional results are shown in Appendices K.3 and L.3, respectively.

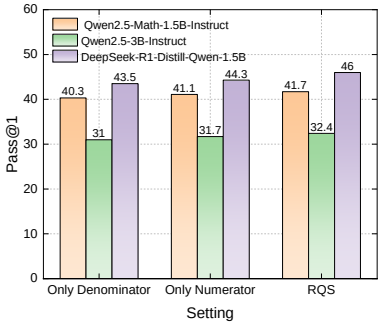
Ablation of Adaptive Exponent Formulation. The exponential-decay formulation for the adaptive exponent p (Eq. (4)) is compared against a simplified linear-decay variant defined as:

$$p_{\text{linear}} = 1 - \gamma \cdot \mu_R \tag{11}$$

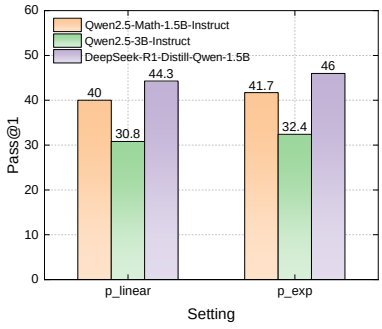
Figure 4(c) shows that the exponential-decay formulation outperforms the linear-decay variant.



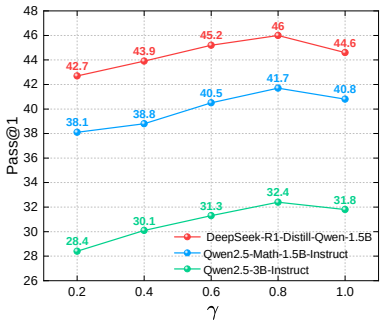
(a) Experimental results using different components of APMPO.



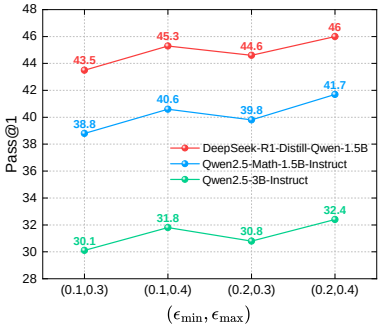
(b) Experimental results using different variants of FSS.



(c) Experimental results using different variants of adaptive exponent function.



(d) Experimental results using different values of γ.



(e) Experimental results using different values of $(\epsilon_{\text{min}}, \epsilon_{\text{max}})$.

Figure 4: Ablation studies and sensitivity analysis. Results are reported as average Pass@1 scores on mathematical reasoning benchmarks. (a) Contribution of each component (PMPO and FAC). (b-c) Effectiveness of different design choices for FSS and the adaptive exponent function. (d-e) Sensitivity analysis for γ and $(\epsilon_{\text{min}}, \epsilon_{\text{max}})$.

This result suggests that the efficacy of PMPO stems from its smooth and asymptotic transition, which avoids abrupt shifts in the learning objective and supports stable policy updates. Further analysis and additional results are provided in Appendices K.5 and L.4, respectively.

5.2.4 Sensitivity Analysis (RQ4)

γ in PMPO. A sensitivity analysis is performed to assess the impact of γ in PMPO. We vary γ and report average Pass@1 scores on mathematical reasoning datasets. As shown in Figure 4(d), a small γ induces a slow transition, where the objective remains GRPO-like for an extended period. Under this setting, heightened sensitivity to reward outliers leads to constrained exploration. Conversely, a large γ induces a fast shift to a GMPO-like objective, thereby excessively suppressing high-reward outliers and risking early convergence to suboptimal solutions. Based on the results, $\gamma = 0.8$ is adopted in this study. More experimental results are provided in Appendix L.5.

$(\epsilon_{\text{min}}, \epsilon_{\text{max}})$ in FAC. We further assess the reasoning performance of APMPO under variations in the clipping bounds $(\epsilon_{\text{min}}, \epsilon_{\text{max}})$. Figure 4(e) illustrates Pass@1 scores on mathematical reasoning datasets. While APMPO reaches its peak performance using the clipping bound $(0.2, 0.4)$, other tested alternatives produces comparable results, including the wider range of $(0.1, 0.4)$ and the narrower range of $(0.2, 0.3)$. These findings demonstrate that APMPO remains robust to the choice of clipping bounds. In this work, $(0.2, 0.4)$ is selected, and more results are provided in Appendix L.6.

6 Conclusion

This work introduces APMPO, which is designed to enhance the reasoning capabilities of LLMs from an adaptive perspective. To overcome the limitations of static objective functions and rigid clipping strategies in previous methods, APMPO incorporates Power-Mean Policy Optimization (PMPO) and Feedback-Adaptive Clipping (FAC). Extensive experiments on several reasoning benchmarks validate the efficacy of APMPO, showcasing a superior learning process compared to state-of-the-art RLVR-based baselines.

Limitations

Despite the promising results achieved by APMPO, there are several limitations that warrant further attention. (1) Due to limited computational resources, the experiments were constrained to models with 1.5B and 3B parameters. However, the observed results consistently demonstrate the effectiveness of APMPO across these scales. Meanwhile, it is expected that the proposed adaptive mechanisms would yield comparable or even greater benefits when applied to larger models exhibiting more complex emergent behaviors. (2) While our experiments span multiple domains, the core of the method depends on the availability of accurate and verifiable outcome-based rewards. The applicability of APMPO to domains where such reward signals are difficult to define remains a promising direction for future work.

Ethical Considerations

We provide the following ethical statements: (1) The efficacy of APMPO was evaluated using open-source LLMs. These models require substantial computational resources, which may contribute to increased carbon dioxide emissions and high energy consumption. (2) All fine-tuned models in this work are derived from publicly released, open-source architectures. No proprietary or confidential models were used. (3) The datasets used for training and evaluation are publicly available, ensuring that no data privacy concerns arise.

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Huayu Chen, Kaiwen Zheng, Qinsheng Zhang, Ganqu Cui, Yin Cui, Haotian Ye, Tsung-Yi Lin, Ming-Yu Liu, Jun Zhu, and Haoxiang Wang. 2025. Bridging supervised learning and reinforcement learning in math reasoning. *arXiv preprint arXiv:2505.18116*.

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

MAA Codeforces. American invitational mathematics examination-aime 2024, 2024.

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*.

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*.

Guanhua Huang, Tingqiang Xu, Mingze Wang, Qi Yi, Xue Gong, Siheng Li, Ruibin Xiong, Kejiao Li, Yuhao Jiang, and Bo Zhou. 2025. Low-probability tokens sustain exploration in reinforcement learning with verifiable reward. *arXiv preprint arXiv:2510.03222*.

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

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. 2024. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

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

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

Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Ruiying Geng, Nan Huo, et al. 2024b. Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls. *Advances in Neural Information Processing Systems*, 36.

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

Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-chun Zhu. 2021. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning. In

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 6774–6786.

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.

Shyam Sundhar Ramesh, Yifan Hu, Iason Chaimalas, Viraj Mehta, Pier Giuseppe Sessa, Haitham Bou Ammar, and Ilija Bogunovic. 2024. Group robust preference optimization in reward-free rlhf. *Advances in Neural Information Processing Systems*, 37:37100–37137.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. 2015. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR.

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

Yikun Wang, Yibin Wang, Dianyi Wang, Zimian Peng, Qipeng Guo, Dacheng Tao, and Jiaqi Wang. 2025. Geometryzero: Improving geometry solving for llm with group contrastive policy optimization. *arXiv preprint arXiv:2506.07160*.

Xumeng Wen, Zihan Liu, Shun Zheng, Zhijian Xu, Shengyu Ye, Zhirong Wu, Xiao Liang, Yang Wang, Junjie Li, Ziming Miao, et al. 2025. Reinforcement learning with verifiable rewards implicitly incentivizes correct reasoning in base llms. *arXiv preprint arXiv:2506.14245*.

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*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024a. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

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

Yufan Ye, Ting Zhang, Wenbin Jiang, and Hua Huang. 2025. Process-supervised reinforcement learning for code generation. *arXiv preprint arXiv:2502.01715*.

Hee Suk Yoon, Eunseop Yoon, Mark A Hasegawa-Johnson, Sungwoong Kim, and Chang D Yoo. 2025. Confpo: Exploiting policy model confidence for critical token selection in preference optimization. In *Forty-second International Conference on Machine Learning*.

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*.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, et al. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3911–3921.

Yuzhong Zhao, Yue Liu, Junpeng Liu, Jingye Chen, Xun Wu, Yaru Hao, Tengchao Lv, Shaohan Huang, Lei Cui, Qixiang Ye, et al. 2025. Geometric-mean policy optimization. *arXiv preprint arXiv:2507.20673*.

Pan Zhou, Xingyu Xie, Zhouchen Lin, and Shuicheng Yan. 2024. Towards understanding convergence and generalization of adamw. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

A Appendix A: Description of the Compared Methods

This section provides a formal description of the policy optimization methods compared in our work. All methods estimate advantage based on a group of sampled rollouts, thereby eliminating the need for a separate value model.

A.1 GRPO

GRPO is the foundational method, which first samples G responses for each query, assigns rewards R via a rule-based reward function, and estimates the corresponding advantages as in Eq. (2). Finally, the model parameters are updated as follows:

$$\mathcal{J}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\min\Big(r_{i,t}\hat{A}_i, \text{clip}(r_{i,t}, 1-\epsilon, 1+\epsilon)\hat{A}_i\Big) \tag{12}$$

where ϵ was fixed at 0.2 in our implementation.

A.2 DAPO

The primary innovations in DAPO lie in advanced clipping mechanism and the Token Level Mean loss (TLM). Specifically, DAPO decouples the clipping range into asymmetric bounds, which focuses on enhancing exploration and mitigating entropy collapse. Moreover, since longer sequences exert a disproportionate influence on gradient update in GRPO, DAPO standardizes token counts across the entire batch. Collectively, the objective function of DAPO is formally given as:

$$\mathcal{J}(\theta) = \frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\min\Big(r_{i,t}\hat{A}_i, \text{clip}(r_{i,t}, 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}})\hat{A}_i\Big) \tag{13}$$

where the upper clipping bound ϵ_{high} is set higher than the lower bound ϵ_{low}. Following the original work (Yu et al., 2025), ϵ_{low} and ϵ_{high} were set to 0.2 and 0.28, respectively.

A.3 GMPO

GMPO addresses the training instability caused by outlier rewards by changing the optimization objective. Instead of maximizing the arithmetic mean of token-level importance-weighted rewards, GMPO maximizes their geometric mean. While the advantage calculation remains the same as in Eq. (2), the objective function for a single response o_i is conceptually structured to optimize for the geometric mean of importance sampling ratios:

$$\mathcal{J}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\left\{\prod_{t=1}^{|o_i|}\min\Big(r_{i,t}^{\text{sgn}(\hat{A}_i)}, \text{clip}(r_{i,t}^{\text{sgn}(\hat{A}_i)}, \epsilon_1, \epsilon_2)\Big)\right\}^{\frac{1}{|o_i|}}\hat{A}_i \tag{14}$$

where ϵ_1 and ϵ_2 were set to $e^{-0.4}$ and $e^{0.4}$, respectively. This structural change provides more stable policy updates in the presence of reward outliers.

B Appendix B: Details of Models and Benchmarks

In the following, the links regarding the models and benchmarks used in this work are provided. Notably, all datasets adopted in this study are publicly available under the CC BY-SA 4.0 license.

(1) **Models:**

- Qwen2.5-Math-1.5B-Instruct: https://huggingface.co/Qwen/Qwen2.5-Math-1.5B-Instruct
- Qwen2.5-3B-Instruct: https://huggingface.co/Qwen/Qwen2.5-3B-Instruct
- DeepSeek-R1-Distill-Qwen-1.5B: https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-1.5B
- Qwen2.5-Coder-3B-Instruct: https://huggingface.co/Qwen/Qwen2.5-Coder-3B-Instruct
- Qwen2.5-VL-3B-Instruct: https://huggingface.co/Qwen/Qwen2.5-VL-3B-Instruct

(2) **Benchmarks:**

- MATH: (12000 training samples) https://huggingface.co/datasets/HuggingFaceH4/MATH
- MATH500: (500 evaluation samples) https://huggingface.co/datasets/HuggingFaceH4/MATH-500
- AIME24: (30 evaluation samples) https://huggingface.co/datasets/HuggingFaceH4/aime_2024

- AIME25: (30 evaluation samples) https://huggingface.co/datasets/HuggingFaceH4/aime_2025
- AMC23: (40 evaluation samples) https://huggingface.co/datasets/math-ai/amc23
- Minerva: (272 evaluation samples) https://huggingface.co/datasets/svc-huggingface/minerva-math
- OlympiadBench: (674 evaluation samples) https://huggingface.co/datasets/knoveleng/OlympiadBench
- Spider: (1034 evaluation samples) https://yale-lily.github.io/spider
- BIRD: (9428 and 1534 samples for training and evaluation https://bird-bench.github.io/
- Geometry3K: (2100 and 601 samples for training and evaluation) https://huggingface.co/datasets/hiyouga/geometry3k

C Appendix C: Gradient Analysis of GRPO and GMPO

In this section, we provide a gradient derivation for GRPO and GMPO. We omit clipping mechanisms and KL penalties to isolate the impact of each objective function on the learning signal, specifically regarding sensitivity to high-reward outliers.

C.1 Preliminaries

Let G denote the group size. For the i-th sample o_i sampled from the group ($i \in \{1, \ldots, G\}$), let $|o_i|$ be its token length. We define $r_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t}|o_{i,<t})}{\pi_{\text{old}}(o_{i,t}|o_{i,<t})}$ and $\psi_{i,t}(\theta) = \nabla_\theta \log \pi_\theta(o_{i,t}|o_{i,<t})$.

C.2 Gradient Derivation of GRPO (Arithmetic Mean)

Definition 1 (GRPO Objective). *The objective function of GRPO based on arithmetic mean is defined as:*

$$\mathcal{J}_{GRPO}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\left(\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} r_{i,t}(\theta)\right)\cdot \hat{A}_i \quad (15)$$

Proposition 1 (Linear Scaling of GRPO). *The gradient of $\mathcal{J}_{GRPO}$ scales linearly with the advantage value $\hat{A}_i$.*

Proof. Differentiating $\mathcal{J}_{\text{GRPO}}$ with respect to θ:

$$\begin{aligned}\nabla_\theta \mathcal{J}_{\text{GRPO}} &= \frac{1}{G}\sum_{i=1}^{G}\frac{\hat{A}_i}{|o_i|}\sum_{t=1}^{|o_i|}\nabla_\theta r_{i,t}(\theta) \\ &= \frac{1}{G}\sum_{i=1}^{G}\frac{\hat{A}_i}{|o_i|}\sum_{t=1}^{|o_i|}\frac{\nabla_\theta \pi_\theta(\cdot)}{\pi_{\text{old}}(\cdot)}\end{aligned} \quad (16)$$

Using the identity $\frac{\nabla \pi}{\pi_{old}} = \frac{\pi}{\pi_{old}}\nabla \log \pi = r_{i,t}(\theta)\psi_{i,t}(\theta)$ yields:

$$\nabla_\theta \mathcal{J}_{\text{GRPO}} \approx \frac{\hat{A}_i}{|o_i|}\sum_{t=1}^{|o_i|}\underbrace{r_{i,t}(\theta)}_{\text{Local Weight}}\cdot\psi_{i,t}(\theta) \quad (17)$$

Therefore, the gradient magnitude is directly proportional to $\hat{A}_i$, and outliers with large $\hat{A}_i$ disproportionately dominate the gradient update. □

C.3 Gradient Derivation of GMPO (Geometric Mean)

Definition 2 (GMPO Objective). *GMPO calculates the geometric mean of the importance sampling ratios over the sequence length, scaled by the advantage. This formulation fundamentally alters the weighting mechanism of individual token gradients compared to GRPO.*

$$\mathcal{J}_{GMPO}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\underbrace{\left\{\prod_{t=1}^{|o_i|} r_{i,t}(\theta)\right\}^{\frac{1}{|o_i|}}}_{U_i(\theta)}\cdot\hat{A}_i \quad (18)$$

Proposition 2 (Global vs. Local Gradient Weighting). *The gradient derivations reveal a structural divergence in weighting mechanisms. While GRPO assigns a local weight (i.e., $r_{i,t}$) to each gradient step, GMPO assigns a global weight (i.e., U_i) to every token in the sequence.*

Proof. We analyze the gradient contribution of a single sample i. Applying the log-derivative trick $\nabla f = f\nabla \log f$:

$$\begin{aligned}\nabla_\theta \mathcal{J}_i(\theta) &= \hat{A}_i \cdot \nabla_\theta U_i(\theta) \\ &= \hat{A}_i \cdot U_i(\theta)\cdot\nabla_\theta\left(\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\log r_{i,t}(\theta)\right) \\ &= \frac{\hat{A}_i}{|o_i|}\sum_{t=1}^{|o_i|}\underbrace{U_i(\theta)}_{\text{Global Weight}}\cdot\psi_{i,t}(\theta)\end{aligned} \quad (19)$$

□

Corollary 1 (**Holistic Stability vs. Local Aggressiveness**). *The disparity in coefficient terms between GRPO and GMPO aligns with the empirical analysis in Figures 1(a) and (b):*

- ***GRPO (Local Sensitivity):*** *The weight for token t corresponds directly to its individual ratio $r_{i,t}(\theta)$, implying that an extreme ratio at a single token induces a disproportionately large update for that specific token. Consequently, GRPO becomes highly sensitive to local fluctuations and prone to high variance.*
- ***GMPO (Holistic View):*** *The weight for token t is $U_i(\theta)$, representing the geometric mean of all ratios in the sequence. This mechanism imposes a holistic perspective where the update strength is determined by the consistency of the entire generation path. Even if a single token exhibits an extreme ratio, the geometric mean dampens its influence to promote stability. While GMPO effectively suppresses outliers, it inevitably restricts the model's capacity to reinforce pivotal token-level breakthroughs.*

This analytical contrast motivates a dynamic exponent p in PMPO to switch between local sensitivity and holistic stability depending on training phase.

D Appendix D: Theoretical Analysis on PMPO

In this section, we provide a theoretical analysis of the PMPO objective. Remarkably, PMPO establishes a unified framework connecting the arithmetic mean-based objective (*e.g.*, GRPO) and the geometric mean-based objective (*e.g.*, GMPO). This allows for dynamic adjustment of policy optimization strategy by varying the exponent p.

D.1 Preliminaries

We begin by formally establishing the necessary mathematical definitions.

Definition 3 (Power Mean). *For a set of non-negative real numbers $X = \{x_1, x_2, \ldots, x_n\}$, the generalized power mean with exponent $p \in \mathbb{R}\backslash\{0\}$ is defined as:*

$$M_p(X) = \left(\frac{1}{n}\sum_{t=1}^{n} x_t^p\right)^{\frac{1}{p}} \tag{20}$$

The cases for $p \to 1$ and $p \to 0$ are of special interest:

- *The Arithmetic Mean is defined as:*

$$M_1(X) = \frac{1}{n}\sum_{t=1}^{n} x_t \tag{21}$$

- *The Geometric Mean is defined as:*

$$M_0(X) = \lim_{p\to 0} M_p(X) = \left(\prod_{t=1}^{n} x_t\right)^{\frac{1}{n}} \tag{22}$$

Definition 4 (Per-Sequence PMPO Objective). *The full per-sequence PMPO objective for a given response o_i is defined by:*

$$\mathcal{J}_i(\theta) = sgn(\hat{A}_i) \cdot M_p(\Phi_i(\theta)) \tag{23}$$

where $\Phi_i(\theta) = \{\phi_{i,1}(\theta), ..., \phi_{i,|o_i|}(\theta)\}, \phi_{i,t}(\theta) \geq 0$ is the set of token-level magnitudes. Each $\phi_{i,t}(\theta)$ is defined in Eq. (8).

D.2 Asymptotic Behavior of the PMPO Objective

We now formally demonstrate that the PMPO objective encompasses arithmetic and geometric mean-based counterparts as limiting cases, establishing its role as a unifying theoretical framework.

Proposition 3 (Convergence to the GRPO Objective). *As the exponent $p \to 1$, the per-sequence PMPO objective $\mathcal{J}_i(\theta)$ asymptotically converges to the standard GRPO objective.*

Proof. By direct substitution of $p = 1$ and $n = |o_i|$ into the power mean definition (*i.e.*, Eq. (20)), we get the arithmetic mean:

$$\lim_{p\to 1} M_p(\Phi_i(\theta)) = M_1(\Phi_i(\theta)) = \frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \phi_{i,t}(\theta) \tag{24}$$

Substituting this into the full PMPO objective yields:

$$\lim_{p\to 1} \mathcal{J}_i(\theta) = \text{sgn}(\hat{A}_i) \cdot \left(\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \phi_{i,t}(\theta)\right)$$

$$= \frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \text{sgn}(\hat{A}_i) \cdot |\min(r_{i,t}(\theta)\hat{A}_i, \rho_{i,t}(\theta)\hat{A}_i)| \tag{25}$$

In particular, $\text{sgn}(X) \cdot |X| = X$. Since the term $\min(r_{i,t}(\theta)\hat{A}_i, \rho_{i,t}(\theta)\hat{A}_i)$ inherently preserves the direction of $\hat{A}_i$, it follows that:

$$\begin{aligned} &\text{sgn}(\hat{A}_i) \cdot |\min(r_{i,t}(\theta)\hat{A}_i, \rho_{i,t}(\theta)\hat{A}_i)| \\ &= \min(r_{i,t}(\theta)\hat{A}_i, \rho_{i,t}(\theta)\hat{A}_i) \end{aligned} \tag{26}$$

Therefore, the objective simplifies to:

$$\lim_{p\to 1} \mathcal{J}_i(\theta) = \frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \min(r_{i,t}(\theta)\hat{A}_i, \rho_{i,t}(\theta)\hat{A}_i) \tag{27}$$

This is precisely the GRPO objective. As a result, PMPO exactly recovers GRPO as a special case. □

Proposition 4 (Convergence to the GMPO-Style Objective)**.** *In the theoretical limit as $p \to 0^+$, the per-sequence PMPO objective $\mathcal{J}_i(\theta)$ converges to a geometric mean-based formulation, recovering the strict consistency constraints characteristic of GMPO.*

Proof 1. *First, we establish the limit of the power mean term. Let $J = \lim_{p\to 0^+} M_p(\phi_{i,t}(\theta))$. This limit presents an indeterminate form of type 1^∞. To resolve this, we analyze the limit of $\ln(J)$. For notational simplicity, let $n = |o_i|$ and $x_t = \phi_{i,t}(\theta)$:*

$$\begin{aligned} \ln(J) &= \lim_{p\to 0^+} \ln\left[\left(\frac{1}{n}\sum_{t=1}^{n} x_t^p\right)^{\frac{1}{p}}\right] \\ &= \lim_{p\to 0^+} \frac{\ln\left(\frac{1}{n}\sum_{t=1}^{n} x_t^p\right)}{p} \\ &\quad \textit{(Indeterminate form } \tfrac{0}{0}\textit{, applying L'Hôpital's Rule)} \\ &= \lim_{p\to 0^+} \frac{\frac{d}{dp}\left(\ln\left(\frac{1}{n}\sum_{t=1}^{n} x_t^p\right)\right)}{\frac{d}{dp}(p)} \\ &= \lim_{p\to 0^+} \frac{\frac{1}{\frac{1}{n}\sum_{t=1}^{n} x_t^p} \cdot \left(\frac{1}{n}\sum_{t=1}^{n} x_t^p \ln(x_t)\right)}{1} \\ &= \frac{1}{\frac{1}{n}\sum_{t=1}^{n} x_t^0} \cdot \left(\frac{1}{n}\sum_{t=1}^{n} x_t^0 \ln(x_t)\right) \\ &= \frac{1}{1} \cdot \left(\frac{1}{n}\sum_{t=1}^{n} \ln(x_t)\right) \\ &= \frac{1}{n}\sum_{t=1}^{n} \ln(x_t) \\ &= \ln\left(\left(\prod_{t=1}^{n} x_t\right)^{\frac{1}{n}}\right) \end{aligned} \tag{28}$$

By exponentiating both sides, we obtain:

$$J = \left(\prod_{t=1}^{n} x_t\right)^{\frac{1}{n}} = \left(\prod_{t=1}^{|o_i|} \phi_{i,t}(\theta)\right)^{\frac{1}{|o_i|}} \tag{29}$$

Substituting this back into the full PMPO objective:

$$\lim_{p\to 0^+} \mathcal{J}_i(\theta) = sgn(\hat{A}_i) \cdot \left(\prod_{t=1}^{|o_i|} \phi_{i,t}(\theta)\right)^{\frac{1}{|o_i|}} \tag{30}$$

Therefore, the PMPO objective converges to a geometric aggregation of token-level magnitudes.

D.3 Conclusion

By tuning p based on model's performance, PMPO fluidly interpolates between a high-sensitivity pattern and a high-stability pattern. This allows it to effectively balance the pursuit of high-reward outliers with stable policy refinement, thereby providing a more adaptive learning mechanism.

E Appendix E: Theoretical Justification of Gradient Dynamics in PMPO

In this section, we provide a unified theoretical framework to analyze the superiority of PMPO. We employ *Gradient Sensitivity Analysis* to showcase how PMPO adaptively balances signal amplification and noise suppression.

E.1 Gradient Sensitivity and the Crossover Point

We analyze the policy update dynamics by examining the *effective gradient contribution* of the advantage signal A, which is modeled as a transformation function $\Psi(A)$.

(1) For GRPO (arithmetic mean), the objective is linear in A (*i.e.*, $\mathcal{J} \sim \Sigma A_i$). Therefore, the gradient contribution is directly proportional to the magnitude of the advantage itself, which is given as $\Psi_{GRPO}(A) = A$.

(2) For PMPO (power mean), the objective involves the p-th power of advantages (*i.e.*, $\mathcal{J} \sim (\Sigma A_i^p)^{1/p}$). Crucially, when computing the gradient with respect to the policy parameters, the chain rule introduces a factor proportional to A^{p-1} scaling the original gradient direction. Therefore, the effective gradient scaling factor behaves as $\Psi_{PMPO}(A) \propto A^p$ where $p \in (0, 1]$.

Definition 5 (Gradient Sensitivity)**.** *The sensitivity of the policy update with respect to the advantage signal magnitude A is defined as $S(A) = \frac{\partial \Psi(A)}{\partial A}$.*

For GRPO, $S_{GRPO}(A) = 1$ (constant sensitivity). For APMPO, $S_{PMPO}(A) = pA^{p-1}$.

Proposition 5 (Dual-Phase Sensitivity and Unique Crossover)**.** *For any fixed adaptive parameter $p \in (0, 1)$, there exists a unique Crossover Point A^* such that PMPO exhibits two distinct behaviors relative to GRPO:*

$$A^* = p^{\frac{1}{1-p}} \tag{31}$$

The sensitivity ratio $\rho(A) = \frac{S_{PMPO}(A)}{S_{GRPO}(A)}$ satisfies:

1. ***Signal Boosting Phase** ($A < A^*$)**:** In this phase, $\rho(A) > 1$. Gradients for low-advantage samples are amplified relative to the linear baseline, preventing gradient vanishing in rare but correct reasoning paths that may not yield maximum rewards.*

2. ***Outlier Damping Phase** ($A > A^*$)**:** In this phase, $\rho(A) < 1$. Gradients for high-advantage outliers are attenuated. This functions as a gradient-clipping mechanism, preventing any single sample from destabilizing the policy update.*

Proof. We analyze the ratio $\rho(A) = pA^{p-1}$. To find the crossover point where sensitivities are equal, we set $\rho(A) = 1$:

$$\begin{aligned} pA^{p-1} = 1 \implies A^{p-1} &= \frac{1}{p} \\ \implies A &= \left(\frac{1}{p}\right)^{\frac{1}{p-1}} = p^{\frac{1}{1-p}} \end{aligned} \tag{32}$$

Let $f(A) = pA^{p-1}$. Since $p \in (0, 1)$, the exponent $p - 1 < 0$. As a result, $f(A)$ is strictly monotonically decreasing on $(0, \infty)$.

- For $A < A^*$, $f(A) > f(A^*) = 1$, implying signal boosting.
- For $A > A^*$, $f(A) < f(A^*) = 1$, implying outlier suppression.

This establishes PMPO as a *non-linear filter* that enhances weak signals while suppressing noise. □

E.2 Adaptive Stability Analysis

A critical feature of PMPO is the adaptive nature of p, *i.e.*, $p = \exp(-\gamma \mu_R)$. We demonstrate that this mechanism automatically transitions the learning strategy from exploration to stabilization.

Proposition 6 (Asymptotic Stability)**.** *As the policy performance improves ($\mu_R \uparrow$), the adaptive parameter $p \to 0$. Consequently, the crossover point A^* approaches zero:*

$$\lim_{p \to 0^+} A^* = 0 \tag{33}$$

Proof. Let $L = \lim_{p\to 0^+} \ln(A^*) = \lim_{p\to 0^+} \frac{\ln p}{1-p}$. As $p \to 0^+$, $\ln p \to -\infty$ and $1 - p \to 1$. As a result, $L \to -\infty$. Since $\ln(A^*) \to -\infty$, it follows that $A^* \to 0$. □

Corollary 2 (Phased Learning Dynamics)**.** *This proposition implies a theoretical guarantee for the learning schedule:*

- ***Early Training (Low** $\mu_R \implies p \approx 1$**):** This indicates minimal non-linear distortion. In this regime, PMPO operates almost linearly (similar to GRPO), allowing high-reward outliers to exert full influence without suppression. This is crucial for signal discovery, enabling the model to rapidly capture the earliest correct reasoning paths.*

- ***Late Training (High** $\mu_R \implies p \approx 0$**):** The crossover point $A^* \to 0$. The Outlier Damping Phase then dominates the entire signal space ($A > A^*$ for almost all A). In this regime, PMPO applies strong attenuation to high-advantage signals, effectively driving the policy to optimize for consistency rather than pursuing isolated high rewards.*

F Appendix F: Theoretical Analysis on FAC

This section provides a theoretical analysis demonstrating the superiority of FAC over static clipping mechanisms. The core argument posits that by adaptively adjusting the clipping bound based on fluctuated reward distributions, FAC circumvents the inherent trade-offs of fixed constraints.

F.1 Preliminaries

Definition 6 (Static and Adaptive Clipping)**.** *A static clipping mechanism constrains the importance sampling ratio $r_{i,t}(\theta)$ to a fixed interval*

$[1 - \epsilon_{static1}, 1 + \epsilon_{static2}]$, *imposing a uniform constraint across all updates. In contrast, FAC defines an adaptive interval* $[1 - \epsilon_{low}, 1 + \epsilon_{ada}]$, *where the upper bound* ϵ_{ada} *is adaptively adjusted.*

Definition 7 (Feedback Stability Score)**.** *We utilize the FSS from Eq. (5) as a quantitative proxy for signal stability within a batch.*

We make the following assumption:

Assumption 1 (FSS as a Quality Proxy)**.** *FSS is positively correlated with the stability of the positive learning signal. High FSS implies high-fidelity reinforcement signals, whereas low FSS indicates unstable signals necessitating caution.*

F.2 Design Rationale for Asymmetric Clipping

A pivotal design feature in FAC is its asymmetric clipping mechanism, characterized by a *fixed lower bound* $1-\epsilon_{\text{low}}$ and an *adaptive upper bound* $1+\epsilon_{\text{ada}}$. This architecture stems from the distinct roles of positive and negative feedback in reasoning tasks.

Adaptive Upper Bound for Conservative Reinforcement. The upper bound regulates positive reinforcement updates (*i.e.*, $\hat{A} > 0$). The primary risk in this context involves overfitting to spurious correlations common in reward-sparse reasoning tasks. The stability of positive signals is contingent upon batch statistics. Concretely, a high FSS indicates high-fidelity feedback, justifying a larger update to accelerate convergence. Conversely, a low FSS implies that positive signals may represent outliers within a noisy distribution. In such cases, tightening the upper bound mitigates the risk of the model confidently committing to spurious success. Consequently, modulating ϵ_{ada} via FSS enforces *risk aversion* under low-quality signals.

Fixed Lower Bound for Decisive Pruning. In contrast, the lower bound governs negative penalty updates (*i.e.*, $\hat{A} < 0$). A fixed ϵ_{low} is employed based on the premise that *negative signals are inherently more stable than positive ones*. Even in low FSS batches, reasoning paths yielding incorrect answers constitute definitive signals warranting penalization. Adaptively tightening the lower bound alongside the upper bound during low-FSS phases would effectively freeze the policy, thereby hindering error correction. By maintaining a fixed lower bound, a consistent mechanism for policy adjustment is preserved, enabling the decisive pruning of incorrect reasoning paths even when confidence for positive reinforcement is insufficient. This asymmetry establishes a *"Conservative Reinforcement, Decisive Pruning"* dynamic essential for improved reasoning.

F.3 Analysis of the Adaptive Mechanism

The efficacy of FAC stems from its ability to map signal quality to an optimal trust region.

Lemma 1 (Properties of the FAC Mapping Function)**.** *The mapping from FSS to the adaptive clipping bound* ϵ_{ada} *is bounded, monotonic, and smooth.*

Proof. We establish these properties for the mapping function:

(1) **Boundedness**: The $\tanh(\cdot)$ function maps FSS $\in (0, \infty)$ to $(0, 1)$. This constrains ϵ_{ada} to the strictly positive interval $(\epsilon_{\text{min}}, \epsilon_{\text{max}})$, preventing vanishing or exploding trust regions.

(2) **Monotonicity**: Since $\tanh(\cdot)$ is strictly increasing, ϵ_{ada} scales positively with FSS. This aligns with our design rationale that higher FSS warrants a larger trust region.

(3) **Smoothness**: The differentiability of $\tanh(\cdot)$ ensures smooth transitions in ϵ_{ada}, thereby preventing abrupt shifts in learning dynamics. □

Proposition 7 (FAC Overcomes the Static Clipping Dilemma)**.** *Static clipping strategies face an intrinsic dilemma, where a small* ϵ_{static} *ensures stability but retards learning on good batches, while a large* ϵ_{static} *accelerates learning but risks instability on noisy batches. FAC resolves this by adaptively tuning* ϵ_{ada} *based on FSS.*

Proof. We analyze the behavior of FAC in distinct scenarios:

(1) **Case 1: Stable Reward Signal (FSS $\gg 0$)**: The batch exhibits high-reward trajectories. As FSS $\to \infty$, $\tanh(\text{FSS}) \to 1$, and $\epsilon_{\text{ada}} \to \epsilon_{\text{max}}$. FAC automatically expands the trust region, enabling aggressive exploitation of high-quality signals. A static method would unnecessarily restrict this valid update and slow down convergence.

(2) **Case 2: Unstable Reward Signal (FSS ≈ 0)**: The batch exhibits pronounced reward fluctuations, suggesting that positive advantage signals are likely unreliable. As FSS $\to 0$, $\tanh(\text{FSS}) \to 0$, and $\epsilon_{\text{ada}} \to (\epsilon_{\text{min}} + \epsilon_{\text{max}})/2$. FAC tightens the trust region to limit updates driven by unreliable samples. This mitigates overfitting to unstable learning signals, whereas a fixed static bound would risk policy degradation. □

Remark 1 (Transition in Intermediate FSS Regime)**.** *Beyond the above two extremes, the behavior of FAC in the intermediate regime (i.e., $0 < FSS < 1$) offers a critical advantage. Recall that $\tanh(x) \approx x$ for small x, the adaptive threshold ϵ_{ada} scales approximately linearly with FSS given as:*

$$\epsilon_{ada} \approx \epsilon_{min} + (\epsilon_{max} - \epsilon_{min}) \cdot FSS \tag{34}$$

As FSS improves, FAC linearly relaxes the trust region. This allows for fine-grained modulation of the update step size, effectively stabilizing training during the critical phase where the model transitions from exploration to consistent reasoning.

G Appendix G: Adaptive Trust Region and Monotonic Improvement

This section provides the theoretical underpinning for FAC. We demonstrate that maximizing the lower bound of monotonic policy improvement necessitates adaptively scaling the trust region size inversely with reward volatility.

G.1 Summary of Notations

For clarity, Table 2 summarizes the key mathematical notations utilized in this analysis.

G.2 Theoretical Analysis

Lemma 2 (Error Bound under Reward Uncertainty)**.** *Let $J(\pi)$ denote the true expected return and $\hat{L}(\pi)$ be the empirical surrogate objective constructed from sampled rewards. Assuming the reward estimation noise is bounded by its standard deviation σ_R, the approximation error is bounded by:*

$$\left|J(\pi) - \hat{L}(\pi)\right| \leq \beta \cdot D_{KL}(\pi \parallel \pi_{old}) + \alpha \cdot \sigma_R \tag{35}$$

where $\beta > 0$ is the Lipschitz constant related to the maximum advantage, and $\alpha > 0$ scales with the importance sampling weights.

Proof. We decompose the total error into the *policy interpolation error* (due to distribution shift) and the *estimation error* (due to reward noise).

$$\begin{aligned} |J(\pi) - \hat{L}(\pi)| \leq & \underbrace{|J(\pi) - L_{true}(\pi)|}_{\text{(I) Interpolation Error}} \\ & + \underbrace{|L_{true}(\pi) - \hat{L}(\pi)|}_{\text{(II) Estimation Error}} \end{aligned} \tag{36}$$

Part I: Bounding the Policy Interpolation Error. The difference between the true objective and the theoretical surrogate is bounded by the quadratic variation of the policy (Schulman et al., 2015). Utilizing Pinsker's inequality to relate Total Variation divergence to KL divergence, we have:

$$|J(\pi) - L_{true}(\pi)| \leq C \cdot \max_s D_{KL}(\pi_{old}(\cdot|s) \parallel \pi(\cdot|s)) \tag{37}$$

Setting $\beta = C$, this yields the first term $\beta \cdot D_{KL}(\pi \parallel \pi_{old})$. (A detailed derivation of this bound is provided in the subsequent subsection).

Part II: Bounding the Estimation Error. The empirical surrogate $\hat{L}$ deviates from L_{true} due to advantage estimation noise. Let $\hat{A}(s,a) = A_{\pi_{old}}(s,a) + \xi(s,a)$, where ξ is noise with variance σ_R^2.

$$\begin{aligned} |L_{true}(\pi) - \hat{L}(\pi)| &= \left| \mathbb{E}_\tau \left[\frac{\pi(a|s)}{\pi_{old}(a|s)} (A_{\pi_{old}} - \hat{A}) \right] \right| \\ &= |\mathbb{E}_\tau \left[r_t(\theta) \cdot \xi_t \right]| \end{aligned} \tag{38}$$

Applying the Cauchy-Schwarz inequality $|\mathbb{E}[XY]| \leq \sqrt{\mathbb{E}[X^2]\mathbb{E}[Y^2]}$ and assuming $r_t(\theta)$ are well-behaved within the trust region, the error is dominated by the noise magnitude:

$$|L_{true}(\pi) - \hat{L}(\pi)| \leq \underbrace{\sqrt{\mathbb{E}[r_t^2]}}_{\alpha} \cdot \underbrace{\sqrt{\mathbb{E}[\xi_t^2]}}_{\sigma_R} = \alpha \cdot \sigma_R \tag{39}$$

Combining Parts I and II completes the proof. □

G.3 Detailed Derivation of the Interpolation Bound

(Note: This subsection expands on Part I of Lemma 2.)

Proof. The derivation unfolds in three steps:

Step 1: Performance Difference Lemma. The performance difference between two policies is given by:

$$J(\pi) - J(\pi_{old}) = \mathbb{E}_{\tau \sim \pi} \left[\sum_{t=0}^{\infty} \gamma^t A_{\pi_{old}}(s_t, a_t) \right] \tag{40}$$

where γ is the discount factor. The surrogate $L_{true}(\pi)$ approximates this by sampling trajectories from π_{old}:

$$L_{true}(\pi) = J(\pi_{old}) + \mathbb{E}_{\tau \sim \pi_{old}} \left[\sum_{t=0}^{\infty} \gamma^t \rho_t A_{\pi_{old}}(s_t, a_t) \right] \tag{41}$$

Notation	Description
π_{old}, π	The policy before and after the update step.
$J(\pi)$	The **true expected return** of policy π in the environment.
$L_{true}(\pi)$	The **theoretical surrogate objective** using exact advantage values.
$\hat{L}(\pi)$	The **empirical surrogate objective** using sampled advantages.
$\mathcal{J}(\theta)$	Shorthand for the objective function parameterized by θ.
σ_R	The standard deviation of the reward estimation noise.
$D_{KL}(\cdot \parallel \cdot)$	KL divergence measuring the distance between policies.
ϵ	The clipping bound hyperparameter used in algorithms such as GRPO.
β	Coefficient reflecting the value function curvature (*i.e.*, policy interpolation cost).
α	Scaling factor for the estimation noise.
γ	Discount factor in RL.

Table 2: Summary of notations used in the theoretical derivation.

The discrepancy $|J(\pi) - L_{true}(\pi)|$ arises solely from the distributional shift in different distributions.

Step 2: Bounding via Total Variation. Let $\epsilon_{tv} = \max_s D_{TV}(\pi \parallel \pi_{old})$. The L_1 distance between state distributions is bounded by $2(1 - (1 - \epsilon_{tv})^t)$. Given a maximum advantage ϵ_{adv}, the accumulated error is:

$$|J(\pi) - L_{true}(\pi)| \leq \frac{2\gamma\epsilon_{adv}}{(1-\gamma)^2} \cdot \epsilon_{tv} \tag{42}$$

Step 3: Relating to KL Divergence. Using the Pinsker Inequality $D_{TV} \leq \sqrt{D_{KL}/2}$ and standard quadratic approximations, we define the penalty coefficient $\beta = \frac{2\gamma\epsilon_{adv}}{(1-\gamma)^2}$. This yields:

$$|J(\pi) - L_{true}(\pi)| \leq \beta \cdot D_{KL}(\pi \parallel \pi_{old}) \tag{43}$$

□

Theorem 1 (Optimality of Adaptive Clipping)**.** *To maximize the lower bound of monotonic policy improvement (i.e., $J(\pi_{new}) \geq J(\pi_{old})$), the clipping bound ϵ must be adaptively scaled inversely with reward variance:*

$$\epsilon \propto \frac{1}{f(\sigma_R)} \approx \mathit{FSS} \tag{44}$$

Proof. Combining Lemma 2 with the policy improvement identity, the lower bound of the true improvement is:

$$\Delta J(\pi) \geq \underbrace{\Delta L(\pi)}_{\text{Surrogate Gain}} - \underbrace{(\beta \cdot D_{KL}(\pi \parallel \pi_{old}) + \alpha \cdot \sigma_R)}_{\text{Total Penalty}} \tag{45}$$

In GRPO, the clipping parameter ϵ serves as a hard constraint on policy divergence, specifically $D_{KL} \approx \mathcal{O}(\epsilon^2)$. To guarantee monotonic improvement ($\Delta J(\pi) > 0$), the Surrogate Gain must outweigh the Total Penalty.

We analyze the impact of increasing σ_R:

- As σ_R increases, the penalty $\alpha \cdot \sigma_R$ grows, consuming a larger portion of the potential gain.
- To maintain $\Delta J > 0$, the remaining penalty term $\beta \cdot D_{KL}$ must be minimized.
- Since $D_{KL} \propto \epsilon^2$, this necessitates reducing ϵ.

Mathematically, the optimal trust region ϵ^* that balancing these terms satisfies $\epsilon^* \propto 1/\sigma_R$. FAC operationalizes this via $\epsilon_{ada} \propto \text{FSS} \approx \frac{\mu}{\sigma_R}$. This mechanism tightens the clipping bound when σ_R is high and relaxes it when σ_R is low. □

H Appendix H: Gradient Derivation of the APMPO Objective

In this section, we provide a detailed derivation for the gradient of the APMPO objective function $\mathcal{J}(\theta)$ with respect to the model parameters θ. The overall objective is the average of per-sample objectives:

$$\mathcal{J}(\theta) = \frac{1}{G}\sum_{i=1}^{G} \mathcal{J}_i(\theta) \tag{46}$$

The gradient is computed as:

$$\nabla_\theta \mathcal{J}(\theta) = \frac{1}{G}\sum_{i=1}^{G} \nabla_\theta \mathcal{J}_i(\theta) \tag{47}$$

H.1 Per-Sample Objective Function

The per-sample objective $L_i(\theta)$ is defined as:

$$\mathcal{J}_i(\theta) = \left[\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}(\phi_{i,t}(\theta))^p\right]^{\frac{1}{p}} \cdot \text{sgn}(\hat{A}_i) \quad (48)$$

where $\phi_{i,t}(\theta)$ is the token-level magnitude:

$$\phi_{i,t}(\theta) = \left|\min\left(r_{i,t}(\theta)\hat{A}_i, \rho_{i,t}(\theta)\hat{A}_i\right)\right| \quad (49)$$

H.2 Applying the Chain Rule

We apply the chain rule to compute $\nabla_\theta \mathcal{J}_i(\theta)$. For simplicity, let $M_p(\theta) = \left[\frac{1}{|o_i|}\sum_t(\phi_{i,t}(\theta))^p\right]^{\frac{1}{p}}$. The objective is $\mathcal{J}_i(\theta) = M_p(\theta) \cdot \text{sgn}(\hat{A}_i)$. Since $\text{sgn}(\hat{A}_i)$ is a constant scalar, the gradient is:

$$\nabla_\theta \mathcal{J}_i(\theta) = \text{sgn}(\hat{A}_i) \cdot \nabla_\theta M_p(\theta) \quad (50)$$

Now we compute the gradient of the power-mean term $M_p(\theta)$. Let $S(\theta) = \frac{1}{|o_i|}\sum_t(\phi_{i,t}(\theta))^p$, so $M_p(\theta) = (S(\theta))^{\frac{1}{p}}$. We can derive that:

$$\begin{aligned}
\nabla_\theta M_p(\theta) &= \frac{1}{p}(S(\theta))^{\frac{1}{p}-1} \cdot \nabla_\theta S(\theta) \\
&= \frac{1}{p}(S(\theta))^{\frac{1-p}{p}} \cdot \frac{1}{|o_i|}\sum_t\left[p(\phi_{i,t}(\theta))^{p-1} \cdot \nabla_\theta \phi_{i,t}(\theta)\right] \\
&= (M_p(\theta))^{1-p} \cdot \frac{1}{|o_i|}\sum_t\left[(\phi_{i,t}(\theta))^{p-1} \cdot \nabla_\theta \phi_{i,t}(\theta)\right]
\end{aligned} \quad (51)$$

H.3 Gradient of $\phi_{i,t}(\theta)$

Let $U_{i,t}(\theta) = \min(r_{i,t}(\theta)\hat{A}_i, \rho_{i,t}(\theta)\hat{A}_i)$, we can know that $\phi_{i,t}(\theta) = |U_{i,t}(\theta)|$. Using the chain rule and the subgradient of the absolute value function ($\frac{d|x|}{dx} = \text{sgn}(x)$), we get:

$$\nabla_\theta \phi_{i,t}(\theta) = \text{sgn}(U_{i,t}(\theta)) \cdot \nabla_\theta U_{i,t}(\theta) \quad (52)$$

The term $U_{i,t}(\theta)$ is a minimum of two terms. Both $r_{i,t}(\theta)$ and $\rho_{i,t}(\theta)$ are positive ratios close to 1. Therefore, the sign of $U_{i,t}(\theta)$ is determined solely by the sign of $\hat{A}_i$. Therefore, we can know that $\text{sgn}(U_{i,t}(\theta)) = \text{sgn}(\hat{A}_i)$.

Next, the subgradient of the $\min$ function is given by:

$$\nabla_\theta U_{i,t}(\theta) = \begin{cases} \hat{A}_i \cdot \nabla_\theta r_{i,t}(\theta) & \text{if } r_{i,t}(\theta) < \rho_{i,t}(\theta) \\ \hat{A}_i \cdot \nabla_\theta \rho_{i,t}(\theta) & \text{if } \rho_{i,t}(\theta) < r_{i,t}(\theta) \end{cases} \quad (53)$$

Substituting back into Eq. (52):

$$\nabla_\theta \phi_{i,t}(\theta) = \text{sgn}(\hat{A}_i) \cdot \nabla_\theta U_{i,t}(\theta) \quad (54)$$

$$= \begin{cases} \text{sgn}(\hat{A}_i) \cdot \hat{A}_i \cdot \nabla_\theta r_{i,t}(\theta) & \text{if } r_{i,t}(\theta) < \rho_{i,t}(\theta) \\ \text{sgn}(\hat{A}_i) \cdot \hat{A}_i \cdot \nabla_\theta \rho_{i,t}(\theta) & \text{if } \rho_{i,t}(\theta) < r_{i,t}(\theta) \end{cases} \quad (55)$$

H.4 Assembling the Final Gradient

Now we substitute the gradient of M_p (Eq. (51)) back into the main gradient expression (Eq. (50)).

$$\nabla_\theta \mathcal{J}_i(\theta) = \text{sgn}(\hat{A}_i) \cdot (M_p(\theta))^{1-p} \cdot \frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left[(\phi_{i,t}(\theta))^{p-1} \cdot \nabla_\theta \phi_{i,t}(\theta)\right] \quad (56)$$

Next, we substitute the subgradient of $\phi_{i,t}(\theta)$ from Eq. (55) into the expression above. This yields:

$$\nabla_\theta \mathcal{J}_i(\theta) = \text{sgn}(\hat{A}_i) \cdot (M_p(\theta))^{1-p} \cdot \frac{1}{|o_i|} \cdot \sum_{t=1}^{|o_i|}\left[(\phi_{i,t}(\theta))^{p-1} \cdot \underbrace{\text{sgn}(\hat{A}_i) \cdot \nabla_\theta U_{i,t}(\theta)}_{\nabla_\theta \phi_{i,t}(\theta)}\right] \quad (57)$$

Since $\text{sgn}(\hat{A}_i) \cdot \text{sgn}(\hat{A}_i) = (\text{sgn}(\hat{A}_i))^2 = 1$, this simplifies the expression to:

$$\nabla_\theta \mathcal{J}_i(\theta) = (M_p(\theta))^{1-p} \cdot \frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left[(\phi_{i,t}(\theta))^{p-1} \cdot \nabla_\theta U_{i,t}(\theta)\right] \quad (58)$$

where $\nabla_\theta U_{i,t}(\theta)$ is given in Eq. (53).

We define the token-level weight $w_{i,t}(\theta)$ as:

$$w_{i,t}(\theta) = (M_p(\theta))^{1-p} \cdot (\phi_{i,t}(\theta))^{p-1} \quad (59)$$

We can derive that:

$$\nabla_\theta \mathcal{J}_i(\theta) = \frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left[w_{i,t} \cdot \nabla_\theta U_{i,t}(\theta)\right] \quad (60)$$

Assuming for simplicity that $r_{i,t} < \rho_{i,t}$, The final gradient expression becomes:

$$\begin{aligned}
\nabla_\theta L_i(\theta) &\approx \frac{1}{|o_i|}\sum_t\left[w_{i,t}(\theta) \cdot \hat{A}_i \cdot \nabla_\theta r_{i,t}(\theta)\right] \\
&\approx \frac{1}{|o_i|}\sum_t\left[w_{i,t}(\theta) \cdot \hat{A}_i \cdot r_{i,t}(\theta) \cdot \nabla_\theta \log \pi_\theta(o_{i,t}|o_{i,<t})\right]
\end{aligned} \quad (61)$$

where the last step uses the log-derivative technique $\nabla_\theta r_{i,t}(\theta) = r_{i,t}(\theta) \cdot \nabla_\theta \log \pi_\theta(o_{i,t}|o_{i,<t})$.

H.5 Analysis

The final gradient form in Eq. (61) can be interpreted as follows.

Direction. The gradient direction is determined by $\hat{A}_i$ and $\nabla_\theta \log \pi_\theta$.

Magnitude Modulation. The core innovation of APMPO lies in the adaptive weight $w_{i,t}(\theta)$, particularly its token-level component $(\phi_{i,t}(\theta))^{p-1}$.

- **When $p \to 1$ (GRPO-like):** We can know that $(\phi_{i,t}(\theta))^{p-1} \to 1$. The weights $w_{i,t}(\theta)$ become approximately uniform across tokens, and the gradient resembles the standard GRPO gradient.
- **When $p \to 0$ (GMPO-like):** We can know that $(\phi_{i,t}(\theta))^{p-1} = 1/\phi_{i,t}(\theta)$. This means that tokens with a larger update magnitude ϕ receive smaller weight in the gradient summation. This leads to a smoother gradient update.

This derivation shows that the APMPO objective results in a well-behaved gradient. Its core mechanism is to adaptively re-weight the contribution of each token to the total gradient, thereby smoothly interpolating between an aggressive and a conservative update strategy based on the exponent p.

I Appendix I: Convergence Analysis of APMPO

In this section, we provide a theoretical analysis of the convergence properties of APMPO. To address the challenge that the power mean p and clipping threshold ϵ_{ada}) evolve adaptively with the policy, we treat the adaptive hyperparameters as functions of the policy parameters θ and analyze the convergence of the implicit composite objective.

I.1 Problem Setup

Let η represent the vector of adaptive hyperparameters $\eta = [p, \epsilon_{ada}]^\top$. In APMPO, these parameters are determined by the reward statistics of the current policy π_θ. We explicitly denote this dependency as $\eta(\theta)$.

APMPO optimizes a parameterized objective $f(\theta; \eta)$. However, since η is updated continuously, the *true* implicit objective we aim to maximize is:

$$\mathcal{F}(\theta) \coloneqq f(\theta; \eta(\theta)) \tag{62}$$

The parameter update follows stochastic gradient descent (SGD) using a stochastic estimator g_k. Crucially, standard RL compute the gradient of f assuming η is fixed, ignoring the path dependence of η on θ. The update rule is:

$$\theta_{k+1} = \theta_k - \alpha_k g_k \tag{63}$$

where g_k estimates the partial gradient $\nabla_\theta f(\theta; \eta)|_{\eta=\eta(\theta_k)}$.

I.2 Assumptions and Plausibility Analysis

Assumption 2 (Smoothness of the Parametric Objective)**.** *For any fixed η, the function $f(\cdot; \eta)$ is L_f-smooth. Furthermore, the gradient is bounded by the clipping mechanism:*

$$\|\nabla_\theta f(\theta; \eta)\| \le M_f \tag{64}$$

Remark 2 (Plausibility). *This is structurally enforced by APMPO's design. FAC constrains the sampling ratio $r_{i,t}(\theta)$ and limits the gradient magnitude via ϵ-clipping. This explicit constraint ensures the objective behaves as a locally Lipschitz function.*

Assumption 3 (Stochastic Gradient Oracle)**.** *The stochastic gradient g_k is an unbiased estimator of the partial gradient with bounded variance:*

1. $\mathbb{E}[g_k|\theta_k] = \nabla_\theta f(\theta_k; \eta(\theta_k))$.
2. $\mathbb{E}[\|g_k - \nabla_\theta f(\theta_k; \eta(\theta_k))\|^2] \le \sigma^2$.

Assumption 4 (Lipschitz Continuity of Adaptive Parameters)**.** *The mapping from policy parameters to adaptive hyperparameters, $\theta \mapsto \eta(\theta)$, is L_η-Lipschitz continuous. Additionally, the objective $f(\theta; \eta)$ is smooth with respect to η, with bounded gradient $\|\nabla_\eta f\| \le M_\eta$.*

Remark 3 (Plausibility: Smoothness of Expectation). *The parameters p and ϵ_{ada} are derived from reward statistics $\mu_R(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}[R(\tau)]$ and $\sigma_R(\theta)$. Since the policy π_θ is a smooth function of θ and rewards are bounded, the expectation $\mu_R(\theta)$ is Lipschitz continuous w.r.t. θ. Since $p(\mu_R)$ (exponential) and $\epsilon_{ada}(\sigma_R)$ (tanh) are smooth bounded functions, the composite mapping $\eta(\theta)$ preserves Lipschitz continuity.*

I.3 Convergence Proof

We analyze the convergence of the true implicit objective $\mathcal{F}(\theta)$. The key challenge is that our update direction g_k approximates the *partial* gradient $\nabla_\theta f$, whereas the true gradient of $\mathcal{F}(\theta)$ includes a total derivative term.

By the Chain Rule, the true gradient is:

$$\nabla\mathcal{F}(\theta) = \nabla_\theta f(\theta;\eta) + \nabla_\eta f(\theta;\eta)^\top \nabla_\theta \eta(\theta) \quad (65)$$

Let $B(\theta) \coloneqq \nabla_\eta f(\theta;\eta)^\top \nabla_\theta \eta(\theta)$ denote the approximation bias. Under Assumption 4, this bias is bounded:

$$\|B(\theta)\| \le \|\nabla_\eta f\| \, \|\nabla_\theta \eta\| \le M_\eta L_\eta \coloneqq C_{bias} \quad (66)$$

Therefore, our algorithm performs SGD with a *biased* gradient estimator.

Theorem 2 (Convergence with Bounded Bias)**.** *Under Assumptions 2–4, with a learning rate schedule satisfying $\sum \alpha_k = \infty$ and $\sum \alpha_k^2 < \infty$, the algorithm converges to a neighborhood of a stationary point of $\mathcal{F}(\theta)$. Specifically:*

$$\liminf_{k\to\infty} \mathbb{E}\left[\|\nabla\mathcal{F}(\theta_k)\|^2\right] \le K \cdot C_{bias}^2 \quad (67)$$

for some constant K, implying convergence up to the limit of the adaptive drift.

Proof. Since $\mathcal{F}$ is a composition of smooth functions, we assume it is L-smooth. From the Descent Lemma:

$$\mathcal{F}(\theta_{k+1}) \le \mathcal{F}(\theta_k) + \langle \nabla\mathcal{F}(\theta_k), \theta_{k+1} - \theta_k \rangle + \frac{L}{2}\|\theta_{k+1} - \theta_k\|^2 \quad (68)$$

Substituting the update $\theta_{k+1} - \theta_k = -\alpha_k g_k$:

$$\mathcal{F}(\theta_{k+1}) \le \mathcal{F}(\theta_k) - \alpha_k \langle \nabla\mathcal{F}(\theta_k), g_k \rangle + \frac{L\alpha_k^2}{2}\|g_k\|^2 \quad (69)$$

Taking expectations conditioned on θ_k. Note that $\mathbb{E}[g_k] = \nabla_\theta f = \nabla\mathcal{F}(\theta_k) - B(\theta_k)$ (from Eq. (65)).

$$\begin{aligned}\mathbb{E}[\mathcal{F}(\theta_{k+1})] &\le \mathcal{F}(\theta_k) - \alpha_k \langle \nabla\mathcal{F}(\theta_k), \nabla\mathcal{F}(\theta_k) - B(\theta_k) \rangle \\ &\quad + \frac{L\alpha_k^2}{2}\mathbb{E}[\|g_k\|^2] \\ &= \mathcal{F}(\theta_k) - \alpha_k \|\nabla\mathcal{F}(\theta_k)\|^2 \\ &\quad + \alpha_k \langle \nabla\mathcal{F}(\theta_k), B(\theta_k) \rangle + \frac{L\alpha_k^2}{2} M^2 \end{aligned} \quad (70)$$

Using Young's Inequality $\langle a, b \rangle \le \frac{1}{2}\|a\|^2 + \frac{1}{2}\|b\|^2$:

$$\begin{aligned}\mathbb{E}[\mathcal{F}(\theta_{k+1})] &\le \mathcal{F}(\theta_k) - \alpha_k \|\nabla\mathcal{F}(\theta_k)\|^2 \\ &\quad + \frac{\alpha_k}{2}\|\nabla\mathcal{F}(\theta_k)\|^2 + \frac{\alpha_k}{2}\|B(\theta_k)\|^2 \\ &\quad + \mathcal{O}(\alpha_k^2) \\ &= \mathcal{F}(\theta_k) - \frac{\alpha_k}{2}\|\nabla\mathcal{F}(\theta_k)\|^2 + \frac{\alpha_k}{2} C_{bias}^2 \\ &\quad + \mathcal{O}(\alpha_k^2) \end{aligned} \quad (71)$$

Rearranging and summing over $k = 0$ to N:

$$\sum_{k=0}^{N} \frac{\alpha_k}{2}\mathbb{E}[\|\nabla\mathcal{F}(\theta_k)\|^2] \le \mathcal{F}(\theta_0) - \mathbb{E}[\mathcal{F}(\theta_{N+1})] + \sum_{k=0}^{N} \frac{\alpha_k}{2} C_{bias}^2 + \sum_{k=0}^{N} \mathcal{O}(\alpha_k^2) \quad (72)$$

s As $N \to \infty$, for the LHS to remain consistent with the RHS (bounded objective), the gradient norm cannot remain arbitrarily large. The algorithm drives the gradient norm down until it is dominated by the bias term C_{bias}.

Practically, as the policy converges, the reward distribution statistics stabilize, meaning $\nabla_\theta \eta(\theta) \to 0$. Consequently, the bias $C_{bias} \to 0$, recovering standard convergence to a stationary point. □

Remark 4 (Quantitative Bound on Gradient Bias)**.** *While the qualitative boundedness of the bias term $B(\theta)$ suffices for the convergence proof, we can explicitly quantify this bound to analyze the convergence rate. Based on Assumption 4, we have $\|\nabla_\theta \eta\| \le L_\eta$. Furthermore, assuming the objective function's sensitivity to η is bounded such that $\|\nabla_\eta J(\theta, \eta)\| \le M_\eta$, applying the Cauchy-Schwarz inequality yields:*

$$\|B(\theta)\| = \|\nabla_\theta \eta \cdot \nabla_\eta J\| \le \|\nabla_\theta \eta\| \cdot \|\nabla_\eta J\| \le L_\eta \cdot M_\eta \quad (73)$$

This quantitative bound explicitly connects the approximation error to the stability of the adaptive parameter. It implies that as the policy stabilizes (i.e., $\nabla_\theta \eta \to 0$ and $L_\eta \to 0$ locally), the bias term vanishes asymptotically, ensuring that APMPO recovers the exact policy gradient properties in the limit.

J Appendix J: Pseudocode for APMPO

Based on the description in Section 4, the pseudocode for APMPO is presented in Algorithm 1, which outlines its key steps and facilitates the reproducibility of our method.

K Appendix K: Further Analysis

K.1 Analysis of Training Efficiency

The empirical results in Figure 1(c) demonstrate that APMPO achieves significant performance gains with negligible additional wall-clock time compared to GRPO. This efficiency is justified by a computational complexity analysis of a single training step.

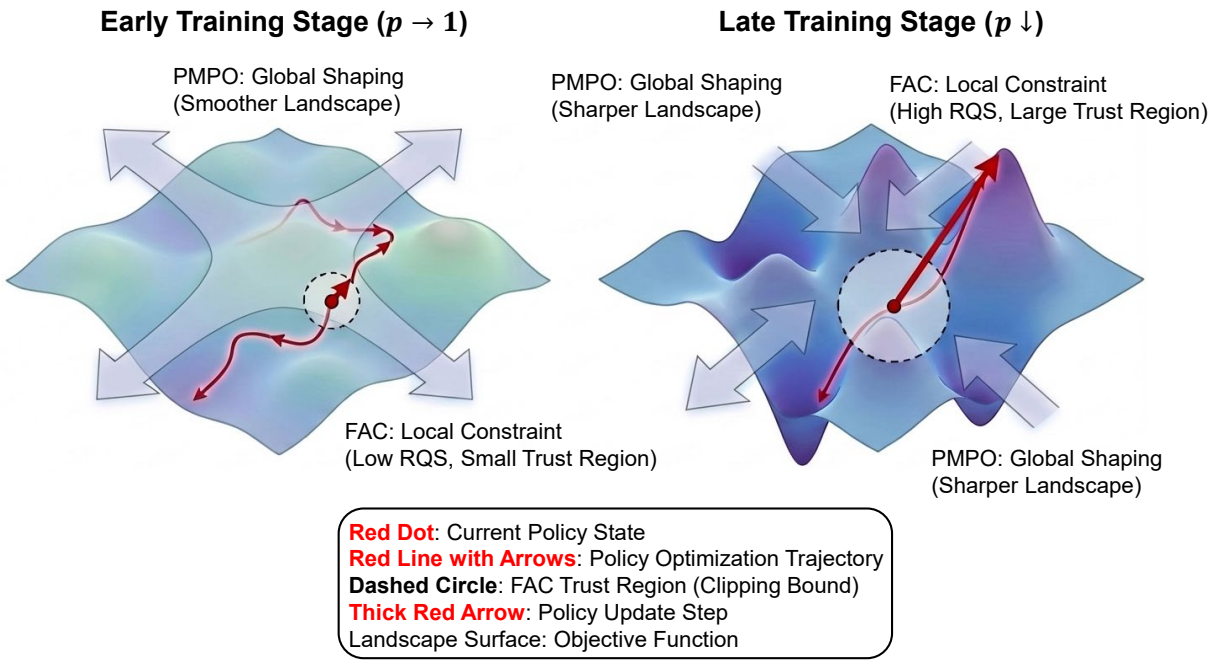


Figure 5: Illustration of the synergy of PMPO and FAC.

Let B be the batch size and G be the number of rollouts for each query. The total number of sampled responses per batch is $N = B \times G$. Let L denote the average sequence length. The computational cost of one training step for GRPO can be broken down as follows:

- **Rollout:** This involves N forward passes to generate responses. The cost is given by:

$$\mathcal{T}_{\text{rollout}} = N \cdot \mathcal{T}_{\text{forward}} \approx O(B \cdot G \cdot L) \quad (74)$$

  where $\mathcal{T}_{\text{forward}}$ is the cost of one forward pass. This phase constitutes the most computationally expensive component of the process.

- **Advantage Calculation:** For GRPO, this requires computing the mean μ_R and standard deviation σ_R of rewards across N samples, followed by normalizing rewards for all N samples. The complexity is given by:

$$\mathcal{T}_{\text{adv\_grpo}} = O(B \cdot G) = O(N) \quad (75)$$

- **Policy Update:** This involves a single forward and backward pass using the N sampled responses to estimate the policy gradient. The cost is given by:

$$\mathcal{T}_{\text{update}} = \mathcal{T}_{\text{forward}} + \mathcal{T}_{\text{backward}} \approx O(B \cdot G \cdot L) \quad (76)$$

The total complexity for GRPO is therefore given as:

$$\mathcal{T}_{\text{GRPO}} \approx \mathcal{T}_{\text{rollout}} + \mathcal{T}_{\text{adv\_grpo}} + \mathcal{T}_{\text{update}} \approx O(B \cdot G \cdot L) \quad (77)$$

Next, the additional computations introduced by APMPO are analyzed:

- **PMPO Overhead:** To compute the adaptive exponent p, the computation requires the batch-level average reward μ_R. This statistic is already obtained in GRPO during advantage normalization. The subsequent $\exp(\cdot)$ operation is a single scalar computation, with a complexity of $O(1)$.

$$\mathcal{T}_{\text{overhead\_pmpo}} = O(1) \quad (78)$$

- **FAC Overhead:** To compute FSS, the calculation requires the mean μ_R and standard deviation σ_R within the batch. This involves iterating through N advantage values. The complexity is:

$$\mathcal{T}_{\text{overhead\_fac}} = O(N) \quad (79)$$

The total additional complexity introduced by APMPO is $\mathcal{T}_{\text{overhead\_apmpo}} = \mathcal{T}_{\text{overhead\_pmpo}} + \mathcal{T}_{\text{overhead\_fac}} \approx O(N)$.

Finally, the total complexity of APMPO is:

$$\mathcal{T}_{\text{APMPO}} = \mathcal{T}_{\text{GRPO}} + O(N) \approx O(B \cdot G \cdot L) + O(B \cdot G) \quad (80)$$

Algorithm 1 Adaptive Power-Mean Policy Optimization (APMPO)

Input: Policy model π_θ, old policy model $\pi_{\theta_{\text{old}}}$, reference model $\pi_{\theta_{\text{ref}}}$, training data $\mathcal{D}$ comprising queries q;
Required: Learning rate η, KL divergence regularizer β, group size per query G, total training steps T, PMPO sensitivity parameter γ, FAC clipping bounds $(\epsilon_{\text{min}}, \epsilon_{\text{max}})$, fixed lower bound ϵ_{low}, numerical stability constant δ;
Output: Optimized policy model π_θ^*;

for $t = 1$ to T **do**
  $\pi_{\theta_{\text{old}}} \leftarrow \pi_\theta$; ▷ Update old policy
  Sample a query $q \sim \mathcal{D}$;
  Initialize lists for responses $\mathcal{O} \leftarrow []$, rewards $\mathcal{R} \leftarrow []$, advantages $\mathcal{A} \leftarrow []$;
  Sample G responses $\{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot|q)$;
  Obtain verifiable rewards $\{R_i\}_{i=1}^G$ for each response;
  $\mu_R \leftarrow \frac{1}{G}\sum_{i=1}^G R_i$; ▷ Group average reward for PMPO
  $\sigma_R \leftarrow \sqrt{\frac{1}{G}\sum_{i=1}^G (R_i - \mu_R)^2}$; ▷ Group reward standard deviation for PMPO
  for $i = 1$ to G **do**
    $\hat{A}_i \leftarrow (R_i - \mu_R)/\sigma_R$; ▷ Compute group-normalized advantage
    Append $\hat{A}_i$ to $\mathcal{A}$;
  end for
  /* — ***Core of APMPO: Adaptive Mechanisms*** — */
  // PMPO: Determine adaptive exponent p
  $p \leftarrow \exp(-\gamma \cdot \mu_R)$; ▷ Eq. (4)
  // FAC: Determine adaptive clipping bound ϵ_{ada}
  $\text{FSS} \leftarrow \mu_R/(\sigma_R + \delta)$; ▷ Eq. (5)
  $\epsilon_{\text{ada}} \leftarrow \epsilon_{\text{min}} + (\epsilon_{\text{max}} - \epsilon_{\text{min}}) \cdot \text{FSS}$; ▷ Eq. (6)
  /* — ***Objective Function Construction*** — */
  Initialize total loss $\mathcal{J}_{\text{APMPO}}(\theta) \leftarrow 0$;
  for $i = 1$ to G **do** ▷ Iterate over all samples in the group
    Let o_i be the i-th response with length $|o_i|$ and advantage $\hat{A}_i$;
    Initialize token-level magnitude sum $S_i \leftarrow 0$;
    for $t = 1$ to $|o_i|$ **do**
      $r_{i,t}(\theta) \leftarrow \frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})}$; ▷ Importance sampling ratio
      $\rho_{i,t}(\theta) \leftarrow \max(1 - \epsilon_{\text{low}}, \min(r_{i,t}(\theta), 1 + \epsilon_{\text{ada}}))$; ▷ Eq. (7)
      $\phi_{i,t}(\theta) \leftarrow |\min(r_{i,t}(\theta) \cdot \hat{A}_i, \rho_{i,t}(\theta) \cdot \hat{A}_i)|$; ▷ Eq. (8)
      $S_i \leftarrow S_i + (\phi_{i,t}(\theta))^p$;
    end for
    $\mathcal{J}_i(\theta) \leftarrow \left(\frac{1}{|o_i|} S_i\right)^{1/p} \cdot \text{sgn}(\hat{A}_i)$; ▷ Eq. (9)
    $\mathcal{J}_{\text{APMPO}}(\theta) \leftarrow \mathcal{J}_{\text{APMPO}}(\theta) + \mathcal{J}_i(\theta)$;
  end for
  $\mathcal{J}_{\text{total}}(\theta) \leftarrow \frac{1}{G}\mathcal{J}_{\text{APMPO}}(\theta) - \beta D_{\text{KL}}(\pi_\theta||\pi_{\theta_{\text{ref}}})$; ▷ Final objective to maximize
  $\theta \leftarrow \theta + \eta \nabla_\theta \mathcal{J}_{\text{total}}(\theta)$; ▷ Update policy parameters via gradient ascent
end for

In the context of LLMs, where the sequence length L is typically large, the additional $O(N)$ overhead remains negligible. This confirms the empirical finding that the adaptive mechanisms in APMPO are computationally lightweight.

K.2 Analyzing the Synergy of PMPO and FAC

The superior performance of APMPO stems from the hierarchical synergy between PMPO and FAC in regulating optimization dynamics.

As illustrated in Figure 5, PMPO functions at the strategic level by shaping the global optimization landscape. By conditioning the power parameter p on the reward mean μ_R, PMPO determines the learning regime, enabling smooth transitions between aggressive signal exploration and conservative policy consolidation.

Conversely, FAC operates at the tactical level, controlling the trust-region constraints. Guided by FSS, FAC modulates the magnitude of policy updates generated by PMPO, ensuring that step sizes are adaptively calibrated according to the stability of the reward signal.

This two-tiered control mechanism underpins APMPO's advantage. While PMPO alone can adapt the update direction, it remains sensitive to random fluctuations in the reward signal, which may cause instability. FAC mitigates this issue by imposing tighter constraints when reward signals are unreliable. Conversely, FAC alone promotes stability but lacks the capacity to reshape the gradient landscape. Their integration allows APMPO to achieve learning that is strategically adaptive.

K.3 Further Analysis of FSS Components

The empirical superiority of the composite FSS formulation over its isolated components (*i.e.*, the numerator μ_R or the denominator $1/\sigma_R$) underscores the necessity of a holistic metric for feedback quality.

Specifically, relying solely on μ_R provides a linear learning signal that biases the optimization toward batches with high mean reward, irrespective of their stability. This approach risks overfitting to batches containing high-reward outliers, leading to potentially erroneous policy updates. Conversely, an objective emphasizing stability (*i.e.*, high $1/\sigma_R$) tends to be overly conservative, rewarding consistently poor behavior. Over-penalizing instability can hinder the emergence of complex reasoning patterns that typically arise during early exploratory stages.

By integrating both components via the specific formulation, FSS allows FAC to distinguish between reliable high-quality signals and unreliable fluctuations. This enables the expansion of clipping bounds only when the learning signal is genuinely trustworthy, thereby ensuring a principled balance between learning efficacy and policy stability.

K.4 Analysis on Clipping Asymmetry

To justify the design choice of FAC, we conducted an ablation study comparing our asymmetric clipping strategy (*i.e.*, Fixed Lower Bound) against a fully symmetric variant, where both the upper and lower bounds are adaptively adjusted by FSS. The expression of the variant is given as:

$$\rho_{i,t}(\theta) = \max\left[1 - \epsilon_{\text{ada}}, \min(r_{i,t}(\theta), 1 + \epsilon_{\text{ada}})\right] \tag{81}$$

As shown in Table 3, our default FAC consistently outperformed the symmetric adaptive variant. This performance gap stemmed from the distinct roles of the clipping bounds. While an adaptive upper bound allowed the model to aggressively capitalize on high-quality consensus (*i.e.*, high FSS), an adaptive lower bound creates instability. Relaxing the lower bound allowed the policy to reduce the probability of certain tokens, potentially leading to excessive policy shifts. By keeping the lower bound fixed, FAC acted as a safety anchor, enabling aggressive positive reinforcement while preventing destabilizing negative updates. This confirmed that an asymmetric trust region was more suitable for reasoning tasks.

K.5 Details of Adaptive Exponent Formulation

The superior performance of the exponential-decay formulation for the adaptive exponent p underscores the importance of achieving a smooth and asymptotic transition within the learning objective. Specifically, the primary limitation of the linear variant lies in its constant rate of change. The exponent p responds to fluctuations in μ_R with a linear sensitivity determined by γ. This approach treats all changes in reward equally, failing to distinguish between minor reward fluctuations and significant performance shifts that warrant different response magnitudes.

In contrast, the exponential-decay formulation provides two key advantages. First, it ensures a smooth and non-linear transition, where the rate of change gradually decreases as performance improves. Second, p asymptotically approaches but never reaches zero. This property guarantees persistent exploratory pressure, preventing the policy from collapsing into overly deterministic behavior. Owing to its inherent smoothness and non-saturating behavior, the exponential-decay formulation represents a more stable mechanism for adaptive policy optimization.

K.6 Empirical Analysis of Output Diversity

To substantiate the claim that APMPO mitigates the entropy collapse observed in GRPO and circumvents the conservatism of GMPO, the output diversity of models on the MATH500 dataset was analyzed. Notably, 8 solutions were generated for each prompt using three complementary metrics:

- **Self-BLEU:** The BLEU-4 score of each generated solution was calculated against others in the same sampled group. A lower Self-BLEU signifies reduced lexical overlap, suggesting that the model avoids simply repeating memorized templates.
- **Average Cosine Similarity (ACS):** Pairwise ACS was computed between the sentence embeddings[1] of all generated solutions per prompt. Lower ACS values indicate greater semantic diversity in the reasoning process.
- **GPT-based Diversity Score:** GPT-5 was employed to evaluate the distinctness of valid

[1] https://huggingface.co/Qwen/Qwen3-Embedding-4B

Method	Math500	AIME24	AIME25	AMC23	Minerva	Olympiad	Avg.
Qwen2.5-Math-1.5B-Instruct							
FSS (Adaptive Lower)	77.5	**20.0**	16.7	60.0	30.1	42.0	41.1
FSS (Fixed Lower)	**78.0**	**20.0**	**16.7**	**62.5**	**30.5**	**42.4**	**41.7**
Qwen2.5-3B-Instruct							
FSS (Adaptive Lower)	68.2	**10.0**	**10.0**	42.5	27.6	32.9	31.8
FSS (Fixed Lower)	**68.4**	**10.0**	**10.0**	**45.0**	**27.9**	**33.2**	**32.4**
DeepSeek-R1-Distill-Qwen-1.5B							
FSS (Adaptive Lower)	81.4	20.0	23.3	62.5	32.4	46.3	44.4
FSS (Fixed Lower)	**81.6**	**23.3**	**26.7**	**65.0**	**32.7**	**46.6**	**46.0**

Table 3: Ablation study on the asymmetry of FAC. We compare our default asymmetric design (Fixed Lower Bound) against a symmetric variant where both bounds are adaptive. The best results are highlighted in bold.

reasoning paths. For correct solutions, a score from 0 to 3 was assigned based on mathematical distinctness (detailed in the *GPT-based prompt*). A higher score implies successful exploration of fundamentally different solving strategies.

Method	ACS (↓)	Self-BLEU (↓)	GPT-Diversity (↑)
GRPO	0.84	0.76	1.22
DAPO	0.79	0.71	1.35
GMPO	0.75	0.65	1.41
APMPO	**0.66**	**0.54**	**1.62**

Table 4: Diversity metrics on the MATH-500 dataset using Qwen2.5-Math-1.5B-Instruct. Comparisons are made against state-of-the-art RLVR-based baselines. ↓ indicates lower is better, and ↑ indicates higher is better.

Analysis. As presented in Table 4, GRPO exhibited the highest ACS and Self-BLEU, confirming the "mode collapse" phenomenon indicated by the entropy curves in Figure 1(b). The arithmetic mean objective in GRPO disproportionately reinforced the first successful path, thereby inducing early policy convergence to a single solution pattern. While DAPO and GMPO improved diversity, it remained more conservative than APMPO. Crucially, APMPO achieved superior performance across all diversity metrics, further validating the efficacy of PMPO and FAC.

GPT-based Prompt. To ensure rigorous evaluation, the following prompt was employed for the GPT-based Diversity Score:

"Given the following set of correct solutions to a math problem, evaluate the diversity of the reasoning methods used. Assign a single integer score from 0 to 3 based on the following criteria:

- ***0:** Solutions use identical logic and phrasing (High Repetition).*
- ***1:** Solutions use the same underlying logic but different phrasing.*
- ***2:** Solutions use slightly different logical steps or intermediate derivations.*
- ***3:** Solutions employ fundamentally different mathematical approaches (e.g., Coordinate Geometry vs. Synthetic Geometry, or Induction vs. Direct Proof).*

Output only the single numerical score."

L Appendix J: More Experimental Results

L.1 Results on SQL Generation and Multi-modal Reasoning

The experimental results on SQL generation (Spider and BIRD) and multi-modal reasoning (Geometry3K) are shown in Table 5.

L.2 Effectiveness of PMPO and FAC

The experimental results regarding the efficacy of different components in APMPO are presented in Table 6.

L.3 Ablation of FSS Components

The experimental results regarding different FSS components are shown in Table 7.

L.4 Ablation of Adaptive Exponent Formulation

The experimental results regarding the adaptive exponent formulation in PMPO are presented in Table 8.

Method	Geometry3K	Spider	BIRD
Base	25.7	70.2	43.6
GRPO	$35.6_{\pm 0.8}$	$73.8_{\pm 0.5}$	$55.7_{\pm 0.9}$
DAPO	$36.5_{\pm 0.7}$	$75.6_{\pm 0.6}$	$58.2_{\pm 0.8}$
GMPO	$37.2_{\pm 0.6}$	$74.8_{\pm 0.5}$	$57.8_{\pm 0.7}$
APMPO	$\mathbf{37.9}_{\pm 0.5}$	$\mathbf{76.4}_{\pm 0.4}$	$\mathbf{60.6}_{\pm 0.6}$

Table 5: Generalization performance on multi-modal reasoning (Geometry3K) and SQL generation (Spider, BIRD). The results are given as mean and standard deviation across 3 random seeds.

Method	Math500	AIME24	AIME25	AMC23	Minerva	Olympiad	Avg.
Qwen2.5-Math-1.5B-Instruct							
GRPO	75.2	13.3	13.3	52.5	29.4	39.0	37.5
+ FAC	76.4	16.7	13.3	57.5	29.8	40.1	39.0
+ PMPO	77.2	**20.0**	**16.7**	60.0	30.1	41.4	40.9
APMPO	**78.0**	**20.0**	**16.7**	**62.5**	**30.5**	**42.4**	**41.7**
Qwen2.5-3B-Instruct							
GRPO	66.0	6.7	6.7	40.0	25.4	31.5	29.4
+ FAC	67.2	6.7	**10.0**	42.5	26.5	32.0	30.8
+ PMPO	67.8	**10.0**	**10.0**	**45.0**	27.2	32.6	32.1
APMPO	**68.4**	**10.0**	**10.0**	**45.0**	**27.9**	**33.2**	**32.4**
DeepSeek-R1-Distill-Qwen-1.5B							
GRPO	75.4	13.3	20.0	57.5	29.8	43.2	39.9
+ FAC	77.6	16.7	20.0	60.0	30.9	44.1	41.6
+ PMPO	79.4	20.0	23.3	62.5	31.6	45.1	43.7
APMPO	**81.6**	**23.3**	**26.7**	**65.0**	**32.7**	**46.6**	**46.0**

Table 6: Ablation study on multiple mathematical reasoning benchmarks using different modules. The best results are highlighted in bold, and we reported Pass@1 score.

L.5 Ablation of γ in PMPO

The experimental results regarding the parameter γ in PMPO are shown in Table 9. Note that $(\epsilon_{\text{min}}, \epsilon_{\text{max}}) = (0.2, 0.4)$ was kept when comparing different values of γ.

L.6 Ablation of $(\epsilon_{\text{min}}, \epsilon_{\text{max}})$ in FAC

The experimental results regarding the parameters $(\epsilon_{\text{min}}, \epsilon_{\text{max}})$ in FAC are shown in Table 10. Note that $\gamma = 0.8$ was kept when comparing different values of $(\epsilon_{\text{min}}, \epsilon_{\text{max}})$.

Method	Math500	AIME24	AIME25	AMC23	Minerva	Olympiad	Avg.
Qwen2.5-Math-1.5B-Instruct							
Only σ_R	77.2	16.7	16.7	60.0	29.8	41.4	40.3
Only μ_R	77.6	**20.0**	**16.7**	60.0	30.1	42.0	41.1
FSS	**78.0**	**20.0**	**16.7**	**62.5**	**30.5**	**42.4**	**41.7**
Qwen2.5-3B-Instruct							
Only σ_R	67.6	6.7	**10.0**	42.5	26.8	32.6	31.0
Only μ_R	67.8	**10.0**	**10.0**	42.5	27.2	32.8	31.7
FSS	**68.4**	**10.0**	**10.0**	**45.0**	**27.9**	**33.2**	**32.4**
DeepSeek-R1-Distill-Qwen-1.5B							
Only σ_R	80.8	20.0	20.0	62.5	32.0	45.8	43.5
Only μ_R	81.2	20.0	23.3	62.5	32.4	46.1	44.3
FSS	**81.6**	**23.3**	**26.7**	**65.0**	**32.7**	**46.6**	**46.0**

Table 7: Ablation study on multiple mathematical reasoning benchmarks using different variants of FSS. The best results are highlighted in bold, and we reported Pass@1 score.

Method	Math500	AIME24	AIME25	AMC23	Minerva	Olympiad	Avg.
Qwen2.5-Math-1.5B-Instruct							
Linear p	77.2	16.7	**16.7**	57.5	29.8	41.8	40.0
Ours	**78.0**	**20.0**	**16.7**	**62.5**	**30.5**	**42.4**	**41.7**
Qwen2.5-3B-Instruct							
Linear p	68.0	6.7	**10.0**	40.0	27.2	32.8	30.8
Ours	**68.4**	**10.0**	**10.0**	**45.0**	**27.9**	**33.2**	**32.4**
DeepSeek-R1-Distill-Qwen-1.5B							
Linear p	81.2	**23.3**	23.3	60.0	32.0	46.1	44.3
Ours	**81.6**	**23.3**	**26.7**	**65.0**	**32.7**	**46.6**	**46.0**

Table 8: Ablation study on multiple mathematical reasoning benchmarks using different p formulations. The best results are highlighted in bold, and we reported Pass@1 score.

γ	Math500	AIME24	AIME25	AMC23	Minerva	Olympiad	Avg.
Qwen2.5-Math-1.5B-Instruct							
0.2	76.8	13.3	13.3	55.0	28.6	41.4	38.1
0.4	77.2	13.3	13.3	57.5	29.4	41.8	38.8
0.6	77.6	16.7	**16.7**	60.0	30.1	42.1	40.5
0.8	**78.0**	**20.0**	**16.7**	**62.5**	**30.5**	**42.4**	**41.7**
1.0	77.8	**20.0**	13.3	**62.5**	29.8	41.5	40.8
Qwen2.5-3B-Instruct							
0.2	67.3	3.3	6.7	35.0	25.7	32.5	28.4
0.4	67.6	6.7	6.7	40.0	26.5	32.8	30.1
0.6	68.2	6.7	**10.0**	42.5	27.2	32.9	31.3
0.8	**68.4**	**10.0**	**10.0**	**45.0**	**27.9**	**33.2**	**32.4**
1.0	68.0	**10.0**	6.7	45.0	27.6	**33.2**	31.8
DeepSeek-R1-Distill-Qwen-1.5B							
0.2	79.6	20.0	20.0	60.0	31.3	45.1	42.7
0.4	80.3	20.0	23.3	62.5	31.6	45.6	43.9
0.6	81.1	**23.3**	23.3	**65.0**	32.4	46.1	45.2
0.8	**81.6**	**23.3**	**26.7**	**65.0**	**32.7**	**46.6**	**46.0**
1.0	80.8	20.0	26.7	62.5	32.0	45.8	44.6

Table 9: Ablation study on multiple mathematical reasoning benchmarks using different values of γ. The best results are highlighted in bold, and we reported Pass@1 score.

$(\epsilon_{\text{min}}, \epsilon_{\text{max}})$	**Math500**	**AIME24**	**AIME25**	**AMC23**	**Minerva**	**Olympiad**	**Avg.**
			Qwen2.5-Math-1.5B-Instruct				
$(0.1, 0.3)$	77.2	16.7	13.3	55.0	29.4	41.2	38.8
$(0.1, 0.4)$	77.8	**20.0**	13.3	60.0	30.1	42.1	40.6
$(0.2, 0.3)$	77.6	16.7	13.3	60.0	29.8	41.4	39.8
$(0.2, 0.4)$	**78.0**	**20.0**	**16.7**	**62.5**	**30.5**	**42.4**	**41.7**
			Qwen2.5-3B-Instruct				
$(0.1, 0.3)$	67.6	6.7	6.7	40.0	26.5	32.8	30.1
$(0.1, 0.4)$	68.0	**10.0**	**10.0**	42.5	27.2	32.9	31.8
$(0.2, 0.3)$	67.8	6.7	**10.0**	40.0	27.2	32.9	30.8
$(0.2, 0.4)$	**68.4**	**10.0**	**10.0**	**45.0**	**27.9**	**33.2**	**32.4**
			DeepSeek-R1-Distill-Qwen-1.5B				
$(0.1, 0.3)$	80.7	20.0	23.3	60.0	31.6	45.8	43.5
$(0.1, 0.4)$	81.2	20.0	**26.7**	**65.0**	32.4	46.3	45.3
$(0.2, 0.3)$	80.4	**23.3**	23.3	62.5	32.0	46.1	44.6
$(0.2, 0.4)$	**81.6**	**23.3**	**26.7**	**65.0**	**32.7**	**46.6**	**46.0**

Table 10: Ablation study on multiple mathematical reasoning benchmarks using different values of $(\epsilon_{\text{min}}, \epsilon_{\text{max}})$. The best results are highlighted in bold, and we reported Pass@1 score.