View-R1: Asymmetric Policy Optimization for Difficulty-Aware Multimodal Reinforcement Learning

Minjie Hong[1,*], **Zirun Guo**[1,*], **Jiabao Zhang**[2,*],
Zehan Wang[1], **Ziang Zhang**[1], **Tao Jin**[1], **Zhou Zhao**[1,†]
[1]Zhejiang University, [2]University of Science and Technology of China
Correspondence: zhaozhou@zju.edu.cn

Abstract

Multimodal Large Language Models (MLLMs) are powerful at integrating diverse data but often struggle with complex reasoning. Reinforcement learning (RL) can enhance reasoning, yet it may cause performance degradation on general tasks and overthinking in MLLMs. We propose Asymmetric Policy Optimization (APO), which separates responses into positive and negative groups. For positive samples, Difficulty-Adaptive Divergence Shaping (DADS) dynamically adjusts the KL weight to stabilize training and preserve knowledge. For negative samples, Suboptimal Trajectory Complexity Regularization (STCR) penalizes overly long responses to reduce overthinking. Applied to Qwen2.5-VL, our model View-R1 achieves a 10.55% improvement in reasoning and outperforms larger models (7–11B) while preserving performance on general tasks and even yielding slight gains. These results highlight the effectiveness and broad applicability of our DADS and STCR techniques for advancing complex multimodal reasoning in MLLMs. Our code is available at https://github.com/Collab-Gen/View-R1.

1 Introduction

Multimodal Large Language Models (MLLMs) (Bai et al., 2025; Chen et al., 2025; Zhu et al., 2025; Peng et al., 2026) have demonstrated strong capabilities across language, vision, and audio modalities, yet they still face challenges in complex reasoning tasks requiring fine-grained multimodal understanding (Guo et al., 2025d; Wang et al., 2026). Recent work (Guo et al., 2025a; Jaech et al., 2024) highlights Reinforcement Learning (RL) as a promising way to enhance reasoning in Large Language Models (LLMs), overcoming the limitations of traditional supervised fine-tuning such as Chain-of-Thought (CoT) dataset construction (Muennighoff et al., 2025). For example, DeepSeek-R1 shows that Reinforcement Learning with Verifiable Rewards (RLVR) can significantly boost mathematical and programming reasoning abilities. Such improvements are also relevant to emerging reasoning-intensive applications (Liu et al., 2026; Zhu et al., 2026b,a; Hong et al., 2025; Guo et al., 2025b).

In the multimodal domain, recent studies (Peng et al., 2025; Huang et al., 2025; Meng et al., 2025) have demonstrated the potential of RL for improving reasoning capabilities. However, multimodal reasoning presents unique challenges, particularly in visual reasoning, where the diversity of problems, uneven difficulty distribution, and the inherent randomness of multimodal inputs make naive RL approaches prone to mode collapse or the generation of verbose yet incorrect chains of thought. Moreover, overthinking and redundant reasoning are more easily triggered in multimodal settings, especially in complex visual tasks, leading to incorrect deductions and degraded overall performance.

Based on the observations above, we propose an **Asymmetric Policy Optimization (APO)** framework that incorporates **Difficulty-Adaptive Divergence Shaping (DADS)** and **Suboptimal Trajectory Complexity Regularization (STCR)** for targeted optimization of positive and negative samples, respectively, thereby enhancing the reasoning capabilities of MLLMs. DADS adaptively adjusts the KL weight of correct samples according to data characteristics, improving training efficiency and preserving the models original knowledge. In parallel, STCR penalizes incorrect answers with excessively long responses, encouraging the model to produce cleaner and more concise reasoning chains.

Using Qwen2.5-VL (Bai et al., 2025) as the

[*]Equal contribution.
[†]Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 30791–30803

base model, we develop our reasoning model **View-R1-3B/7B**, which achieves an average 10.55% improvement on reasoning benchmarks and surpasses several larger open- and closed-source MLLMs. Moreover, unlike other reasoning models that experience performance drops on general multimodal tasks, View-R1-3B/7B maintains consistent gains, demonstrating stronger generalization ability.

Our contributions are summarized as follows:

- We provide a systematic analysis of the KL penalty and overthinking in RL for MLLMs, elucidating their impact on performance and stability.
- We propose an Asymmetric Policy Optimization (APO) framework comprising Difficulty-Adaptive Divergence Shaping and Suboptimal Trajectory Complexity Regularization, which enhance training efficiency, sample utilization, and the clarity of reasoning chains.
- View-R1-3B/7B outperforms a series of larger MLLMs on both reasoning and general benchmarks, indicating the effectiveness and generalization ability of our approach.

2 Related Work

Reinforcement Learning in LLMs. Applying RL to Large Language Models (LLMs) has shifted from environmental mastery to aligning outputs with complex, sometimes ambiguous human preferences via RLHF (Ouyang et al., 2022), which fine-tunes pre-trained models against human judgments rather than external rewards. Proximal Policy Optimization (PPO) (Schulman et al., 2017) remains the standard actor-critic method in RL fine-tuning. Direct Preference Optimization (DPO) (Rafailov et al., 2023) removes the separate reward model by directly optimizing the policy from a preferencepolicy relationship. Group Relative Policy Optimization (GRPO) (Shao et al., 2024) extends PPO by dropping the value head and evaluating groups of outputs, suiting multi-step reasoning and comparative rewards. Group Sequence Policy Optimization (GSPO) (Zheng et al., 2025) defines the importance ratio at the sequence level with sequence-level clipping, outperforming GRPO in performance and efficiency and improving MoE stability. Dr.GRPO (Liu et al., 2025) corrects GRPOs train-longer bias by removing length and variance normalization, improving token efficiency without harming reasoning quality.

Reinforcement Learning in MLLMs. Extending reinforcement learning paradigms from LLMs to MLLMs has emerged as a prominent research direction. Observe-R1 (Guo et al., 2025c) enhances MLLM learning through a difficulty-graded dataset, improved visual observation via multimodal constraints, and a reward system for concise, accurate responses with dynamic weighting. LMM-R1 (Peng et al., 2025) enhances the model's general task performance by conducting two-stage RL training on mixed datasets of pure text and image-text. MM-Eureka (Meng et al., 2025) first trains general reasoning, then introduces KL divergence with domain data to improve performance in specific domains. VL-Rethinker-7B (Wang et al., 2025) enhances GRPO-based reinforcement learning with selective replay and forced rethinking to improve accuracy on multimodal math and science tasks.

3 Methodology

3.1 Preliminaries

Reinforcement Learning with Verifiable Reward. Group Relative Policy Optimization (GRPO) (Shao et al., 2024) is a variant of Proximal Policy Optimization (PPO) (Schulman et al., 2017), which eliminates the value function and estimates the advantage in a group manner. Specifically, given a question-answer pair (q, a), GRPO samples a group G of responses $\{o_i\}_{i=1}^{G}$. Following DeepSeek-R1 (Guo et al., 2025a), we use the verifiable reward, which is divided into the accuracy reward r_i^{accuracy} and the format reward r_i^{format}. The accuracy reward is the final accuracy of a verifiable task and the format reward is granted when the model follows the format constraint such as `<think></think><answer></answer>`. Therefore, the total reward r_i of the response o_i can be computed as:

$$r_i = r_i^{\text{accuracy}} + \lambda r_i^{\text{format}} \tag{1}$$

where λ is the coefficient to adjust the weight. In GRPO, the reward r_i is normalized in a group manner as follows:

$$\hat{A}_{i,t} = \frac{r_i - \text{mean}(\{r_i\}_{i=1}^{G})}{\text{std}(\{r_i\}_{i=1}^{G})} \tag{2}$$

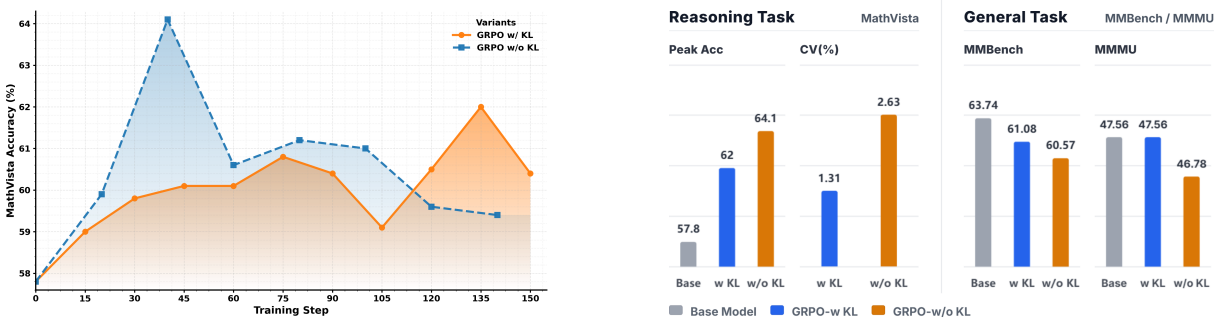

Figure 1: (a) Comparison of the performance on MathVista during the training process. (b) Comparison of the performance and stability on the reasoning and general benchmarks. CV denotes the coefficient of variation of the performance during the training process.

Then, GRPO adopts a clipped objective function with the KL penalty term similar to PPO:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\,\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\Big(\min\big(r_{i,t}(\theta)\hat{A}_{i,t}, \text{clip}\big(r_{i,t}(\theta),\,1-\varepsilon,1+\varepsilon\big)\hat{A}_{i,t}\big) - \beta\,\mathbb{D}_{\text{KL}}(\pi_\theta\,\|\,\pi_{\text{ref}})\Big)\Bigg] \tag{3}$$

where $r_{i,t}(\theta)$ is the importance sampling ratio (Schulman et al., 2017). In this work, we use GRPO as our base approach for RL training. Furthermore, our improvements focus on $\hat{A}_{i,t}$ and β.

3.2 Difficulty-Adaptive Divergence Shaping

KL or Not? The KL penalty in Equation 3 is added for the stable parameter update, which can prevent the model from collapsing. However, several recent studies (Meng et al., 2025; Yu et al., 2025) propose to remove the KL penalty in the original GRPO for better performance. We pose the question: *Does removing the KL divergence lead to unintended consequences?* To answer this question and understand the KL penalty in the original GRPO, we conduct several experiments.

Empirically, we compare models trained with and without the KL penalty on the MathVista (Lu et al., 2024) dataset, as shown in Fig. 1(a). The results show that the model without the KL penalty performs better in most training stages, mainly because it can explore reasoning strategies more freely with fewer constraints, achieving higher performance and faster convergence in the early phase. However, as training progresses, the performance of the model without the KL penalty gradually declines, while the model with the KL penalty continues to improve steadily. This is because removing the KL penalty increases the divergence from the reference model, leading to forgetting effects and a degradation in reasoning ability. Fig. 1(b) further confirms this: although the model with the KL penalty performs slightly worse on reasoning tasks, it consistently outperforms the other on general benchmarks. Moreover, stability analysis based on the coefficient of variation (standard deviation divided by mean) shows that the model without the KL penalty exhibits greater fluctuations during training(Fig. 1(b)), whereas the model with the penalty remains more stable.

From the above analysis, we have several key findings. Firstly, removing the KL penalty helps the model perform better and learn faster during the RL training on the reasoning tasks. However, it fails to retain some prior knowledge which leads to performance degradation on the general benchmarks. Secondly, the KL penalty makes the training more stable and helps to improve the reasoning abilities while also retaining the generalization capabilities of the MLLM, but the reasoning ability does not improve as much as that without the KL penalty.

Redesign of the KL penalty. Based on the empirical findings, we need to figure out where to explore and how much to explore, as well as where to retain and how much to retain, in order to achieve the best performance. To this end, we propose the **Difficulty-Adaptive Divergence Shaping** (DADS) strategy. From Equation 3, we can observe that for every response, the weighting term β has the same value, which means that it imposes equal constraints on all samples. Such a strategy is clearly inappropriate, as different sam-

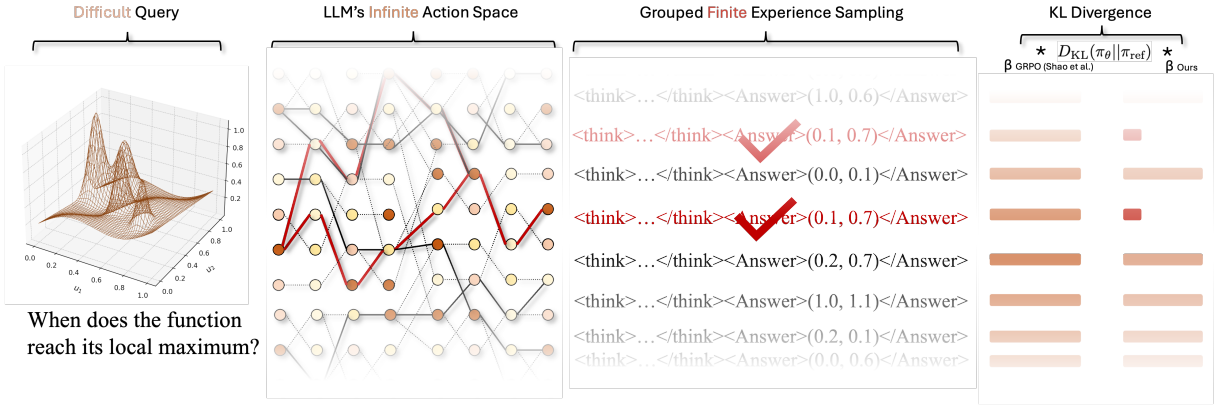

Figure 2: Illustration of GRPO and our method dealing with difficult query conditions. DADS adaptively adjusts the KL divergence penalty based on the test-time difficulty of a sample, calculated from a group of sampled responses.

ples and responses have varying training value for improving the reasoning abilities and retaining the prior knowledge. Therefore, DADS is proposed to adaptively shape the KL divergence based on the test-time difficulty of the data sample. Fig. 2 presents the outline of DADS. Specifically, given a question-answer pair (q, a), the current policy samples a group G of responses $\{o_i\}_{i=1}^{G}$. Then, we define the test-time difficulty as:

$$d = \frac{\sum_{i=1}^{G} \mathbb{1}(r_i^{\text{accuracy}} = 0)}{G} \tag{4}$$

where $\mathbb{1}$ is an indicator function that equals 1 when $r_i^{\text{accuracy}} = 0$, and 0 otherwise. We define the difficulty as the failure rate estimated from G on-policy rollouts, with further justification provided in Appendix 5. In GRPO, for high-difficulty positive samples, even when they provide advantages in subsequent computations, the KL divergence penalty still constrains the probabilistic deviation between the actor and reference models, reducing the training efficiency of correct samples and making it difficult to learn problems that the reference model struggles with. To address this, we aim to improve the training efficiency of positive samples and impose a weaker KL constraint on those with higher difficulty, thereby overcoming the limitations of the reference model. Accordingly, we adaptively map the difficulty d to the KL weight term:

$$\beta_i^{dads} = \begin{cases} 1 - \mathrm{e}^{\mathrm{e}(d-1)}\beta & \text{if } r_i^{\text{accuracy}} = 1 \text{ and } d \neq 0 \\ \beta & \text{otherwise} \end{cases} \tag{5}$$

To reduce notation, we henceforth use the shorthand $f(d) = 1 - e^{e(d-1)}$ to denote this mapping.

We want $f(d)$ to have several necessary properties: (1) On the interval $[0, 1]$, $f(d)$ is a positive-valued function that strictly monotonically decreases from 1 approaching 0. (2) Non-negativity serves as a constraint that helps prevent the actor model from hacking the loss in undesired ways, such as by uniformly decreasing token probabilities. (3) Low initial slopes and terminal fast decays unbind the model under extremely difficult queries. It's worth noting that groups with all-correct samples are filtered out, as decaying the KL divergence of experiences without advantage introduces greater training instability. We select $f(d) = 1 - \mathrm{e}^{\mathrm{e}(d-1)}$ as the primary function. In addition we explore several variants in Fig. 6 and analyze them in Sec. 4.4.

3.3 Suboptimal Trajectory Complexity Regularization

Wrong answers often involve overthinking. In various GRPO-based studies (Shao et al., 2024; Huang et al., 2025; Meng et al., 2025; Zhou et al., 2025), a significant increase in response length during training is commonly observed and often interpreted as evidence of enhanced reasoning capability. However, Dr. GRPO (Liu et al., 2025) and DAPO (Yu et al., 2025) suggest that this phenomenon may stem from the inherent bias of the GRPO objective function. As shown in Fig. 3, our experiments on shuffled and difficulty-graded datasets reveal that incorrect responses are consistently longer than correct ones, with the length gap continuously widening as training progresses; meanwhile, on the shuffled dataset, the length of correct responses remains nearly unchanged

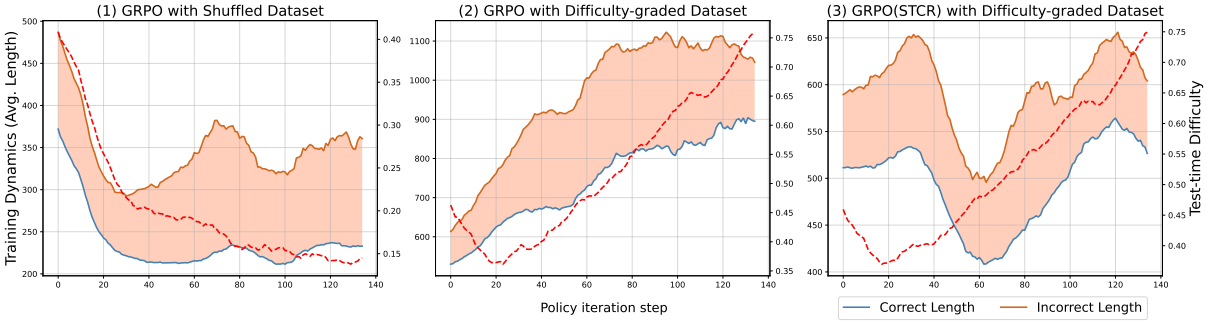

Figure 3: Comparison of response length dynamics during training: Original GRPO on difficulty-graded datasets and shuffled datasets, versus our STCR method applied to difficulty-graded data.

throughout training.

As shown in Fig. 4, the original GRPO algorithm assigns equal weight to all samples in the final loss computation. Consequently, tokens in longer responses contribute less to the overall loss, leading to a divergence in output length over timeincorrect responses tend to become longer, while correct ones grow shorter. This length bias is particularly problematic for MLLMs that integrate multimodal information, as it can exacerbate issues such as verbosity and repetition (Yu et al., 2025).

Based on this observation, we argue that the length-biased loss in GRPO encourages excessively long generations, wasting inference resources and weakening the effective learning signal. To address this issue and promote more concise and accurate responses, we propose a novel regularization method: Suboptimal Trajectory Complexity Regularization (STCR).

We do not completely remove the length bias, as the tendency of correct answers to become concise is desirable for reducing inference cost and improving stability; instead, we retain this effect while preventing incorrect responses from growing unboundedly long.

The core idea of STCR is to introduce a penalty term that scales the advantage estimates $\hat{A}_{i,t}$ for suboptimal responses based on their length relative to the average length of correct ones. Specifically, we penalize incorrect responses whose lengths significantly exceed the mean length of correct responses, as they primarily contribute to length bias and gradient dilution. Let L_i denote the length of the i-th trajectory o_i ($L_i = |o_i|$), and $L^{\text{acc}}_{\text{mean}}$ the average length of correct responses in the current batch or training window. For incorrect trajectories with $L_i > L^{\text{acc}}_{\text{mean}}$, a length-dependent scaling coefficient α_i is applied to $\hat{A}_{i,t}$, computed as:

$$\alpha_i = 2 - \mu^{L^{\text{acc}}_{\text{mean}} - L_i} \tag{6}$$

where μ is a hyperparameter slightly greater than 1. When $L_i > L^{\text{acc}}_{\text{mean}}$, the exponent $L^{\text{acc}}_{\text{mean}} - L_i$ becomes negative, causing $\mu^{L^{\text{acc}}_{\text{mean}} - L_i}$ to decrease toward 0 as L_i grows. Consequently, α_i increases toward 2, thereby intensifying the penalty on overly long incorrect sequences. For trajectories where $L_i \leq L^{\text{acc}}_{\text{mean}}$ or for any correct trajectory, we set $\alpha_i = 1$, applying no scaling.

The modified advantage term, $\hat{A}^{stcr}_{i,t}$, used in the objective function is:

$$\hat{A}^{stcr}_{i,t} = \begin{cases} \hat{A}_{i,t} \cdot \alpha_i & \text{if } r_i^{accuracy} = 0 \text{ and } L_i > L^{\text{acc}}_{\text{mean}} \\ \hat{A}_{i,t} & \text{otherwise} \end{cases} \tag{7}$$

By multiplying $\hat{A}_{i,t}$ by $\alpha_i > 1$ for long incorrect sequences, we intensify their negative advantages. This effectively increases the penalty for generating tokens within these overly long, incorrect responses. This targeted scaling amplifies the learning signal, strongly discouraging the model from producing verbose, incorrect outputs.

4 Experiments

4.1 Experimental Setup

In our experiments, we use Qwen2.5-VL-3B/7B-Instruct (Bai et al., 2025) as our base model. We use multiple datasets to train our model, including MathVision (Wang et al., 2024), We-Math (Qiao et al., 2024), SceMQA (Liang et al., 2024), PolyMath (Gupta et al., 2024), GeoQA+ (Cao and Xiao, 2022), FigureQA (Kahou et al., 2018), UniGeo (Chen et al., 2022), TabMWP (Lu et al., 2023), ScienceQA (Lu et al., 2022) and CLEVR-Math (Lindström and Abraham, 2022). These

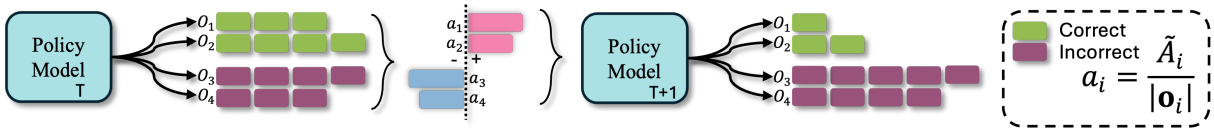

Figure 4: Illustration of the length bias in GRPO. When calculating the effective advantage, the division of $\hat{A}_i$ by the response length $|o_i|$ introduces a length bias to the token-level advantage. This bias preferentially rewards shorter sequences for positive samples and penalizes shorter sequences for negative samples. Over successive iterations, this mechanism can induce a notable divergence in the model's output length, ultimately leading to a phenomenon where incorrect answers progressively lengthen while correct answers become shorter.

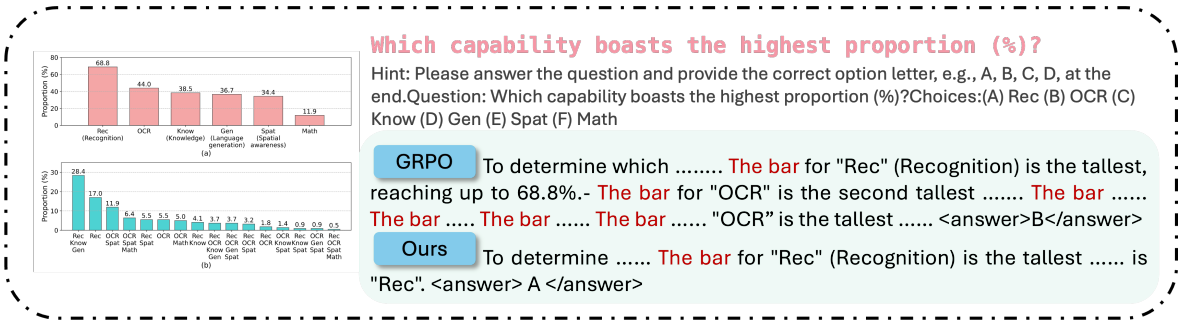

Figure 5: An example demonstrating how our method, by incorporating DADS and STCR, avoids the overthinking seen in the original GRPO, leading to a more concise and correct reasoning process.

datasets cover a wide range of domains from reasoning tasks to multimodal general tasks. We construct these datasets with incrementally increasing difficulty according to Observe-R1 (Guo et al., 2025c).

In the training stage, the train batch size is set to 64 and the rollout batch size is set to 128. We use a learning rate of 1e-6 and sample 8 responses for each question. The reward trade-off λ is set to 0.5 following (Meng et al., 2025). Besides, we use 1.0001 as the hyperparameter value for μ.

For evaluation, we use three reasoning benchmarks: MathVista (Lu et al., 2024), MathVerse (Zhang et al., 2024a), and MMK12 (Meng et al., 2025). For general multimodal evaluation, we use MMStar (Chen et al., 2024) and MMMU (Yue et al., 2024). We report MMMU results on the validation set. All reported results are based on a single training run due to computational constraints.

4.2 Main Results

To comprehensively validate the effectiveness of our method, we compare View-R1-3B/7B with various MLLMs, including closed-source, open-source general, and reasoning-oriented models. As shown in Table 1, despite having only 7B parameters, View-R1-7B achieves the best performance on MathVista, MathVerse, and MMStar, and ranks second on MMK12, surpassing several closed-source and reasoning models with 7–11B parameters. Moreover, View-R1-3B attains the best results among all models with 3B parameters or fewer across all five benchmarks, demonstrating the superior efficiency and effectiveness of our approach in enhancing the reasoning capabilities of MLLMs.

At the same time, we observe that most reasoning models tend to perform worse after reinforcement learning (RL) training compared to their original base models. In contrast, View-R1-3B/7B continues to exhibit consistent improvements on MMMU, validating the effectiveness of our method in preserving the original knowledge of MLLMs while enhancing reasoning ability.

We also compare APO with concurrent group policy optimization baselines, including GRPO, DrGRPO, and GSPO, under the same backbone and evaluation protocol (Table 4). While these methods improve multimodal reasoning to varying degrees, they often show weaker general-task retention. By comparison, View-R1-3B achieves the strongest overall performance, improving both reasoning (MathVista/MathVerse) and general capability (MMStar/MMMU).

In addition, we provide an illustrative example in Fig. 5. As shown, the base model lacks a complete reasoning process and sufficient detail, result-

Model	Year	Params	Reasoning Benchmark			General Benchmark	
			MathVista	MathVerse	MMK12	MMStar	MMMU
Closed-Source MLLMs							
GPT-4o (OpenAI, 2024)	2024	-	63.8	37.6	49.9	63.9	**62.8**
GPT-4V (OpenAI, 2023)	2024	-	49.4	39.4	47.2	49.7	53.8
GPT-4o-mini (OpenAI, 2024)	2024	-	52.5	37.5	39.7	54.8	60.0
Open-Source MLLMs							
InternVL3-2B (Zhu et al., 2025)	2025	2B	57.0	25.3	23.4	60.7	48.6
InternVL2.5-4B (Chen et al., 2025)	2024	4B	64.1	27.7	35.6	58.7	51.8
InternVL2.5-8B (Chen et al., 2025)	2024	8B	64.4	39.5	45.6	62.8	56.2
Qwen2.5-VL-3B (Bai et al., 2025)	2025	3B	57.8	34.7	43.5	54.1	47.6
Qwen2.5-VL-7B (Bai et al., 2025)	2025	7B	64.8	46.7	45.3	64.1	48.4
Open-Source Reasoning MLLMs							
Mulberry-7B (Yao et al., 2024)	2025	7B	63.1	45.1	34.2	61.3	51.5
InternVL2.5-4B-MPO (Chen et al., 2025)	2024	4B	65.3	35.8	10.9	58.7	51.8
LMM-R1 (Peng et al., 2025)	2025	3B	63.2	41.5	49.9	58.0	49.9*
R1-VL-2B (Zhang et al., 2025)	2025	2B	52.1	26.2	22.4	49.8	36.7
R1-VL-7B (Zhang et al., 2025)	2025	7B	63.5	40.0	32.7	60.0	43.3
R1-Onevision-7B (Yang et al., 2025)	2025	7B	64.1	46.4	39.8	61.9	44.4
VL-Rethinker-7B (Wang et al., 2025)	2025	7B	70.2	50.0	**62.7**	63.6	40.1
Curr-ReFT-3B (Deng et al., 2025)	2025	3B	60.4	31.4	42.4	53.6	48.8
Curr-ReFT-7B (Deng et al., 2025)	2025	7B	69.2	41.6	50.8	63.5	54.2
DeepEyes-7B (Zhang et al., 2024b)	2025	7B	70.8	47.7	54.0	63.7	53.6
View-R1-3B	2025	**3B**	66.2*	41.6*	47.0*	61.3*	48.9
View-R1-7B	2025	**7B**	**71.2**	**51.7**	55.7	**65.4**	49.1

Table 1: Comparison with recent MLLMs on reasoning and general benchmarks. **Bold** = best; underline = second best. number* = best-performing model with $\leq$ 3B parameters.

Model	MathVista					MathVerse
	TQA	AR	SR	VQA	ALL	
Base Model	60.1	51.0	73.8	47.5	57.8	35.0
+ GRPO-w-KL	58.2	61.7	69.4	55.3	60.5	37.5
+ GRPO-w/o-KL	58.9	61.8	78.1	54.7	64.1	38.9
+ GRPO-w-STCR	58.2	62.9	78.4	53.1	64.5	40.0
+ GRPO-w-DADS	**65.2**	60.9	**80.1**	54.2	66.1	40.1
View-R1-3B	64.8	**63.1**	78.6	**54.9**	**66.2**	**41.6**

Table 2: Ablation study of View-R1-3B on reasoning benchmarks. MathVista sub-scores: Textbook Question Answering (TQA), Arithmetic Reasoning (AR), Statistical Reasoning (SR), Visual Question Answering (VQA). **Bold**: best result.

ing in an incorrect final answer. The GRPO model generates a detailed reasoning path but includes excessive redundant information and outputs, leading to an overthinking issue that ultimately produces an incorrect answer. In contrast, APO generates an accurate and concise step-by-step reasoning process, effectively avoiding the overthinking problem.

4.3 Ablation Study

To investigate the contribution of each proposed component to the overall performance of our model, we conduct comprehensive ablation experiments. Starting with the base model, we progressively introduce GRPO, the standard KL penalty, STCR, and DADS. The results are summarized in Table 2.

Incorporating STCR into the GRPO framework yields notable performance improvements. Specifically, GRPO enhanced with STCR achieves scores of 64.5 on MathVista and 40.0 on MathVerse, surpassing both the basic GRPO and GRPO with the standard KL penalty. This clearly demonstrates STCR's effectiveness in mitigating the length bias inherent in GRPO (Fig. 3(3)), promoting the generation of concise and accurate responses, especially for incorrect attempts, thereby enhancing learning efficiency.

Next, we evaluate the effect of DADS by integrating it into the GRPO framework, which yields notable performance improvements66.1 on MathVista and 40.1 on MathVerse. Compared with

Model	MathVista	MathVerse	MMMU
InternVL3-2B	56.7	25.4	47.3
+ APO (ours)	**65.4**	**28.7**	**48.9**

Table 3: APO improves InternVL3-2B under the same training/evaluation protocol.

Method	MathVista	MathVerse	MMStar	MMMU
Qwen2.5-VL-3B-Instruct	57.8	34.7	54.1	47.6
w-GRPO	64.1	38.9	57.9	46.8
w-DrGRPO	62.8	40.4	58.7	45.8
w-GSPO	58.7	39.6	54.2	45.9
View-R1-3B (ours)	**66.2**	**41.6**	**61.3**	**48.9**

Table 4: Comparison with GRPO/Dr.GRPO/GSPO.

GRPO using only the standard KL penalty, this highlights the crucial role of DADS. These results confirm that adaptively adjusting KL divergence based on sample difficulty allows freer exploration of challenging but correct samples while maintaining stability through stronger constraints on simpler or incorrect ones, as shown in Fig. 2. Ultimately, our full model, View-R1-3B, achieves the best overall performance, scoring 66.2 on MathVista and 41.6 on MathVerse, and ranks best or near-best across MathVista sub-benchmarks. The combined use of STCR and DADS effectively overcomes the limitations of standard GRPO, improving both performance and training stability: STCR mitigates length bias, while DADS optimizes the explorationstability balance through adaptive KL shaping.

4.4 Further Analysis

The core of DADS is the projection function $f(d)$, which maps difficulty d to a scaling factor on the KL weight β. To assess its effect, we compare linear and cubic variants with the original KL penalty and with training without a KL term. Fig. 6 shows performance on MathVista under these settings. Training without KL (KL-w/o) yields fast initial gains but later declines, while the original KL (KL-Original) improves steadily but slowly. Adaptive DADS seeks to balance these advantages.

We evaluate peak performance (Peak Overall Accuracy on MathVista) and training stability, measured by the Coefficient of Variation (CV) across training steps. Fig. 6(c) summarizes these results. The DADS variants reveal clear trade-offs: the linear function $f(d) = 1 - d$ improves peak accuracy (63.5%) over KL-Original but shows lower stability (CV = 2.52%). The cubic function

Method	AIME 2024	MATH-500
Base (Qwen2.5-3B-Instruct)	7.03	62.4
+ GRPO	7.31	64.8
+ Dr.GRPO	7.34	65.2
+ APO (ours)	**7.55**	**66.6**

Table 5: Text-only LLM results.

achieves a better balance, attaining higher peak accuracy (64.6%) than both KL-Original and KL-Linear, while also offering improved stability (CV = 2.15%) compared with KL-w/o, KL-Linear, and slightly better than KL-DADS.

Our KL-DADS variant with $f(d) = 1 - \mathrm{e}^{\mathrm{e}(d-1)}$ achieves the highest peak accuracy on MathVista (66.1%). Although its stability (CV = 2.29%) is slightly lower than the original KL and cubic variants, it provides a better overall trade-off, clearly outperforming the no-KL setting and other adaptive strategies. By assigning higher penalties to easier samples and rapidly decaying them for harder ones, it balances exploration and stability and mitigates catastrophic forgetting.

Furthermore, as shown in Fig. 3, we compare the original GRPO with our STCR-enhanced model across different datasets. The original GRPO shows a growing divergence between the lengths of incorrect and correct responses during training, reflecting an overthinking phenomenon. In contrast, STCR effectively mitigates this trend, leading to more concise and stable reasoning.

In contrast, applying STCR on the difficulty-graded dataset keeps the length gap between incorrect and correct responses much more stable, remaining within roughly 100 tokens. This indicates that STCR effectively regularizes suboptimal trajectory length. Moreover, as described in Sec. 3, the hyperparameter μ in the α_i scaling factor controls this length gap, enabling fine-grained adjustment of the regularization strength.

STCR not only helps control response length but also enhances training efficiency and model performance. As shown in the ablation study in Table 2, applying STCR to the base GRPO model increases the MathVista ALL score from 64.1 to 64.5 and the MathVerse score from 38.9 to 40. This result indicates that by penalizing lengthy and incorrect responses, STCR focuses policy updates on more effective reasoning paths, encouraging the model to generate cleaner and more concise reasoning chains, thereby improving learning efficiency and overall performance.

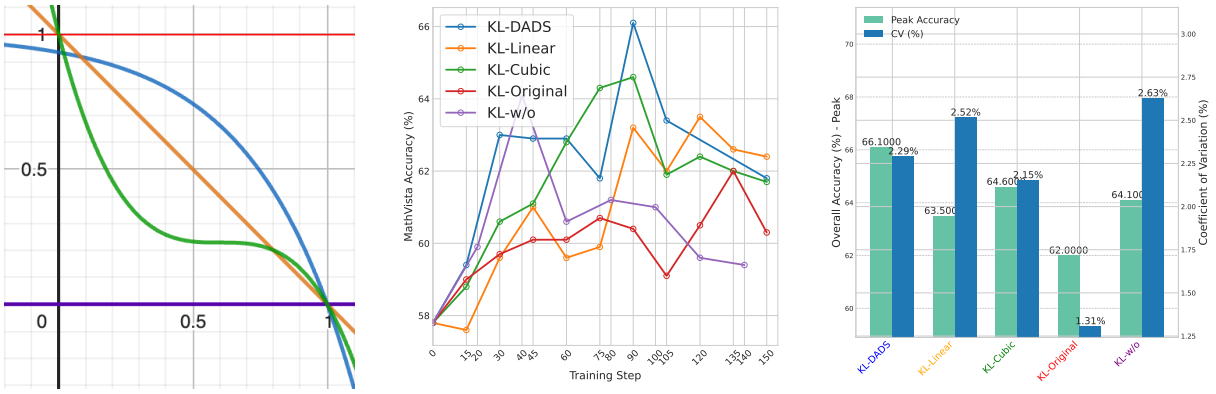

Figure 6: (a) Different curve variants of $f(d)$ in Difficulty-Adaptive Divergence Shaping (DADS). (b) Performance comparison on MathVista during training. (c) Peak performance and training stability (Coefficient of Variation, CV) for different DADS variants. Lower CV indicates higher stability.

Generalization to Other Backbones. We evaluate APO on additional model backbones under the same training and evaluation protocol. Table 3 reports results on InternVL3-2B, and Table 5 reports results on a text-only LLM.

InternVL3-2B backbone. As shown in Table 3, applying APO to InternVL3-2B yields consistent gains on multimodal reasoning benchmarks (MathVista and MathVerse) while maintaining a slight improvement on the general benchmark MMMU. This suggests that APO generalizes beyond the Qwen2.5-VL family and remains effective on a different MLLM architecture.

Text-only LLM backbone. We further test APO on a text-only LLM using AIME 2024 and MATH-500 (Table 5). APO improves both benchmarks over GRPO and Dr.GRPO, indicating that the proposed training principles extend beyond multimodal settings and can benefit text-only reasoning as well.

5 Conclusion

We propose a novel Asymmetric Policy Optimization (APO) framework composed of Difficulty-Adaptive Divergence Shaping (DADS) and Suboptimal Trajectory Complexity Regularization (STCR) to enhance the reasoning capabilities of MLLMs. DADS dynamically adjusts the KL penalty to improve reasoning performance while maintaining generalization, whereas STCR penalizes overly long incorrect outputs to encourage concise and coherent reasoning. Built upon Qwen2.5-VL, View-R1-3B/7B achieves significant performance gains across multiple reasoning and multimodal benchmarks, demonstrating the effectiveness and generality of APO in enhancing reasoning while preserving generalization.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under Grant No. U24A20326.

Limitations

Due to computational and budget constraints, we were unable to scale our models beyond 7B parameters. Consequently, the performance improvements of the proposed Asymmetric Policy Optimization (APO) method were verified only on mid-sized MLLMs. Scaling the method to larger architectures could alter the optimization dynamics and lead to different trade-offs between reasoning improvement and general-task retention.

Ethical Considerations

Potential Risks. Reinforcement learning may cause overconfidence in reasoning, leading models to persist in errors when visual inputs are noisy. Consequently, deploying this method in high-risk scenarios (e.g., autonomous driving or medical assistance) requires rigorous safety measures. These results are for research purposes only.

Data and Model Compliance. We strictly follow the open-source licenses (e.g., Apache-2.0, CC BY-NC 4.0) of all datasets and models used. This work is conducted for non-commercial research, with all sources properly cited.

Privacy and Harmful-Content Review. We performed preprocessing and review of the data, including the screening and removal of personally identifiable information (PII) and potentially offensive or discriminatory content. The benchmarks used primarily originate from public academic resources, synthetic graphics, or educational questionanswer corpora; we did not identify entries containing recognizable personal sensitive information.

Use of AI Assistance. During manuscript preparation, an AI assistant was used for language polishing and formatting suggestions (e.g., grammatical fluency and reference formatting).

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, and 1 others. 2025. Qwen2.5-vl technical report. *Preprint*, arXiv:2502.13923.

Jie Cao and Jing Xiao. 2022. An augmented benchmark dataset for geometric question answering through dual parallel text encoding. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1511–1520, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Jiaqi Chen, Tong Li, Jinghui Qin, Pan Lu, Liang Lin, Chongyu Chen, and Xiaodan Liang. 2022. Unigeo: Unifying geometry logical reasoning via reformulating mathematical expression. *Preprint*, arXiv:2212.02746.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and 1 others. 2024. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, and 1 others. 2025. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *Preprint*, arXiv:2412.05271.

Huilin Deng, Ding Zou, Rui Ma, Hongchen Luo, Yang Cao, and Yu Kang. 2025. Boosting the generalization and reasoning of vision language models with curriculum reinforcement learning. *Preprint*, arXiv:2503.07065.

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

Hongcheng Guo, Zheyong Xie, Shaosheng Cao, Boyang Wang, Weiting Liu, Zheyu Ye, Zhoujun Li, Zuozhu Liu, and Wei Lu. 2025b. Pet-bench: Benchmarking the abilities of large language models as e-pets in social network services. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pages 6402–6407.

Zirun Guo, Minjie Hong, and Tao Jin. 2025c. Observer-r1: Unlocking reasoning abilities of mllms with dynamic progressive reinforcement learning. *arXiv preprint arXiv:2505.12432*.

Zirun Guo, Minjie Hong, Feng Zhang, Kai Jia, and Tao Jin. 2025d. Thinking with programming vision: Towards a unified view for thinking with images. *Preprint*, arXiv:2512.03746.

Himanshu Gupta, Shreyas Verma, Ujjwala Anantheswaran, Kevin Scaria, Mihir Parmar, Swaroop Mishra, and Chitta Baral. 2024. Polymath: A challenging multi-modal mathematical reasoning benchmark. *Preprint*, arXiv:2410.14702.

Minjie Hong, Zetong Zhou, Zirun Guo, Ziang Zhang, Ruofan Hu, Weinan Gan, Jieming Zhu, and Zhou Zhao. 2025. Generative reasoning recommendation via llms. *Preprint*, arXiv:2510.20815.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Yao Hu, and Shaohui Lin. 2025. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, and 1 others. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Samira Ebrahimi Kahou, Vincent Michalski, Adam Atkinson, Akos Kadar, Adam Trischler, and Yoshua Bengio. 2018. Figureqa: An annotated figure dataset for visual reasoning. *Preprint*, arXiv:1710.07300.

Zhenwen Liang, Kehan Guo, Gang Liu, Taicheng Guo, Yujun Zhou, Tianyu Yang, Jiajun Jiao, Renjie Pi, Jipeng Zhang, and Xiangliang Zhang. 2024. Scemqa: A scientific college entrance level multimodal question answering benchmark. *Preprint*, arXiv:2402.05138.

Adam Dahlgren Lindström and Savitha Sam Abraham. 2022. Clevr-math: A dataset for compositional language, visual and mathematical reasoning. *Preprint*, arXiv:2208.05358.

Weiting Liu, Han Wu, Yufei Kuang, Xiongwei Han, Tao Zhong, Jianfeng Feng, and Wenlian Lu.

2026. Automated optimization modeling via a localizable error-driven perspective. *arXiv preprint arXiv:2602.11164*.

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, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *Preprint*, arXiv:2310.02255.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Preprint*, arXiv:2209.09513.

Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. 2023. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. *Preprint*, arXiv:2209.14610.

Fanqing Meng, Lingxiao Du, Zongkai Liu, Zhixiang Zhou, Quanfeng Lu, Daocheng Fu, Botian Shi, Wenhai Wang, Junjun He, Kaipeng Zhang, and 1 others. 2025. Mm-eureka: Exploring visual aha moment with rule-based large-scale reinforcement learning. *arXiv preprint arXiv:2503.07365*.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. 2025. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*.

OpenAI. 2023. Gpt-4v(ision) system card.

OpenAI. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

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

Yingzhe Peng, Gongrui Zhang, Miaosen Zhang, Zhiyuan You, Jie Liu, Qipeng Zhu, Kai Yang, Xingzhong Xu, Xin Geng, and Xu Yang. 2025. Lmm-r1: Empowering 3b lmms with strong reasoning abilities through two-stage rule-based rl. *arXiv preprint arXiv:2503.07536*.

Yuxin Peng, Zishuo Wang, Geng Li, Xiangtian Zheng, Sibo Yin, and Hulingxiao He. 2026. A survey on fine-grained multimodal large language models. *Chinese Journal of Electronics*, 35(2):769–801.

Runqi Qiao, Qiuna Tan, Guanting Dong, Minhui Wu, Chong Sun, Xiaoshuai Song, Zhuoma GongQue, Shanglin Lei, Zhe Wei, Miaoxuan Zhang, Runfeng Qiao, Yifan Zhang, Xiao Zong, Yida Xu, Muxi Diao, Zhimin Bao, Chen Li, and Honggang Zhang. 2024. We-math: Does your large multimodal model achieve human-like mathematical reasoning? *Preprint*, arXiv:2407.01284.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

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

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

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. 2025. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *Preprint*, arXiv:2504.08837.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. 2024. Measuring multimodal mathematical reasoning with math-vision dataset. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Zehan Wang, Ziang Zhang, Jiayang Xu, Jialei Wang, Tianyu Pang, Chao Du, HengShuang Zhao, and Zhou Zhao. 2026. Orient anything v2: Unifying orientation and rotation understanding. *Preprint*, arXiv:2601.05573.

Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, and 1 others. 2025. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. *arXiv preprint arXiv:2503.10615*.

Huanjin Yao, Jiaxing Huang, Wenhao Wu, Jingyi Zhang, Yibo Wang, Shunyu Liu, Yingjie Wang, Yuxin Song, Haocheng Feng, Li Shen, and 1 others. 2024. Mulberry: Empowering mllm with o1-like reasoning and reflection via collective monte carlo tree search. *arXiv preprint arXiv:2412.18319*.

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

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, and 1 others. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567.

Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. 2025. R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. *arXiv preprint arXiv:2503.12937*.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Peng Gao, and 1 others. 2024a. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? *arXiv preprint arXiv:2403.14624*.

Ruohong Zhang, Bowen Zhang, Yanghao Li, Haotian Zhang, Zhiqing Sun, Zhe Gan, Yinfei Yang, Ruoming Pang, and Yiming Yang. 2024b. Improve vision language model chain-of-thought reasoning. *Preprint*, arXiv:2410.16198.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. 2025. Group sequence policy optimization. *Preprint*, arXiv:2507.18071.

Hengguang Zhou, Xirui Li, Ruochen Wang, Minhao Cheng, Tianyi Zhou, and Cho-Jui Hsieh. 2025. R1-zero's" aha moment" in visual reasoning on a 2b non-sft model. *arXiv preprint arXiv:2503.05132*.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, and 1 others. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *Preprint*, arXiv:2504.10479.

Wei Zhu, Zhiwen Tang, and Kun Yue. 2026a. Symphony: Synergistic multi-agent planning with heterogeneous language model assembly. *arXiv preprint arXiv:2601.22623*.

Wei Zhu, Lixing Yu, Hao-Ren Yao, Zhiwen Tang, and Kun Yue. 2026b. Task-aware llm council with adaptive decision pathways for decision support. *arXiv preprint arXiv:2601.22662*.

Why Choose Failure Rate as the Difficulty Metric

Our decision to define difficulty as the failure rate estimated from G on-policy rollouts is deliberate and motivated by the requirements of Difficulty-Adaptive Divergence Shaping (DADS).

Real time, model capability dependent difficulty. DADS aims to adaptively allocate training signals according to the models current capability. Consequently, the difficulty metric must reflect the models present performance rather than a static property of the data. The failure rate computed from the models own on-policy rollouts naturally satisfies this requirement: it directly arises from the models current behavior and evolves dynamically as training progresses, enabling stable and responsive adaptive scheduling.

Limitations of dataset-provided difficulty annotations. Although some datasets provide difficulty labels, such annotations are typically dataset-specific, depend on human-defined criteria, and lack a unified scale across tasks or domains. More importantly, these difficulty labels are inherently static, whereas difficulty is always relative to the models current ability. A sample labeled as hard may become trivial after a few training iterations, making static annotations unsuitable for adaptive optimization.

Limitations of teacher-based difficulty estimation. Estimating difficulty using a teacher models confidence introduces substantial computational overhead, as it requires additional forward passes of a large model for each sample. Moreover, the teachers competence profile may not align with that of the student model, leading to domain mismatch. As a result, teacher confidence may fail to accurately reflect the student models actual failure likelihood, potentially producing misleading difficulty signals.

Limitations of entropy or disagreement-based metrics. Policy entropy or model disagreement does not directly correspond to problem difficulty. Entropy is often influenced by task structure rather than inherent difficulty; for example, open-ended generation tasks naturally exhibit higher entropy due to multiple valid outputs, while well-defined mathematical problems may have low entropy despite being difficult. In addition, entropy lacks natural bounds and typically requires maintaining large-scale sampling statistics, which can be unstable during early training stages. These properties make entropy-based metrics unsuitable for DADS.

Overall, the failure rate from on-policy rollouts provides a simple, stable, and model-aligned difficulty signal that directly reflects the models current capability, making it well suited for adaptive KL shaping in DADS.