Understanding and Mitigating Spurious Signal Amplification in Test-Time Reinforcement Learning for Math Reasoning

Yongcan Yu[1,2]*, Lingxiao He[1,3], Jian Liang[1,2]†, Kuangpu Guo[1,4], Meng Wang[3], Qianlong Xie[3], Xingxing Wang[3], Ran He[1,2]

[1]NLPR & MAIS, Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]Meituan [4]University of Science and Technology of China
{yuyongcan0223, liangjian92}@gmail.com

Abstract

Test-time reinforcement learning (TTRL) always adapts models at inference time via pseudo-labeling, leaving it vulnerable to spurious optimization signals from label noise. Through an empirical study, we observe that responses with medium consistency form an ambiguity region and constitute the primary source of reward noise. Crucially, we find that such spurious signals can be even amplified through group-relative advantage estimation. Motivated by these findings, we propose a unified framework, Debiased and Denoised test-time Reinforcement Learning (DDRL), to mitigate spurious signals. Concretely, DDRL first applies a frequency-based sampling strategy to exclude ambiguous samples while maintaining a balanced set of positive and negative examples. It then adopts a debiased advantage estimation with fixed advantages, removing the bias introduced by group-relative policy optimization. Finally, DDRL incorporates a consensus-based off-policy refinement stage, which leverages the rejection-sampled dataset to enable efficient and stable model updates. Experiments on three large language models across multiple mathematical reasoning benchmarks demonstrate that DDRL consistently outperforms existing TTRL baselines. The code is available at https://github.com/yuyongcan/DDRL.

1 Introduction

Reinforcement learning with verifiable rewards (RLVR) has recently emerged as an effective paradigm for improving large language models (LLMs) on structured challenging reasoning tasks, including mathematics and code generation (Lambert et al., 2024; Yue et al., 2025; Jaech et al., 2024; Guo et al., 2025; Yu et al., 2025a; Chen et al., 2025; Yu et al., 2025b). By relying on explicit supervision or rule-based verification, RLVR enables stable optimization and strong task-specific performance. However, its applicability is fundamentally constrained by the availability of reliable labels or verifiable reward functions, which limits its applicability when reliable labels or verifiers are unavailable, especially under distribution shift.

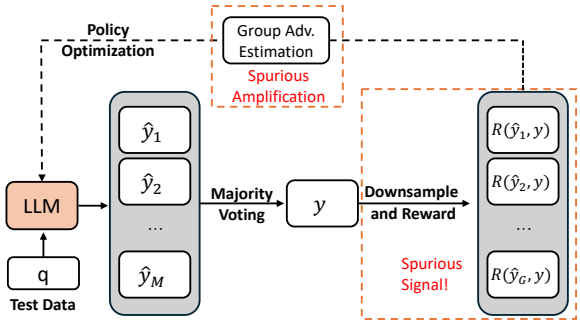


Figure 1: Overview of test-time reinforcement learning (TTRL). The spurious signal is generated during the reward stage with noisy pseudo-labels and amplified in the subsequent group-relative advantage estimation.

To address distribution shifts during inference, Test-Time Reinforcement Learning (TTRL) (Zuo et al., 2025) has emerged as a promising solution. TTRL integrates test-time scaling (TTS) (Muennighoff et al., 2025; Zhang et al., 2025b) with test-time training (TTT) (Sun et al., 2020), enabling parameter updates via unsupervised reinforcement learning (RL). As illustrated in Figure 1, TTRL generates multiple responses for each test query, derives a pseudo-label through majority voting, and optimizes the model using GRPO (Shao et al., 2024) based on these pseudo-labels. Despite its conceptual appeal, TTRL operates in a fundamentally unsupervised regime, where reward signals are derived entirely from the model's own outputs. This design renders TTRL highly vulnerable to spurious reward signals: incorrect responses may inadvertently receive positive rewards, while correct answers might be penalized, thereby distorting learning dynamics.

In this work, we systematically analyze the ori-

*Work done during an internship at Meituan
†Corresponding author

Findings of the Association for Computational Linguistics: ACL 2026, pages 37424–37436

gins of spurious signals and how they propagate through the optimization process. Empirically, we observe a strong correlation between answer frequency and reliability: high-frequency answers are predominantly correct, low-frequency answers are mostly incorrect, while answers with medium sampling frequency tend to be ambiguous and unreliable. These medium-frequency responses constitute a major source of spurious reward signals. Theoretically, we further demonstrate that GRPO's advantage estimation introduces a systematic bias in this unsupervised setting. Specifically, its normalization mechanism disproportionately amplifies spurious rewards in low-consensus regimes.

Motivated by these insights, we propose **D**ebiased and **D**enoised test-time **R**einforcement **L**earning (DDRL), a framework designed to mitigate spurious signals. First, we introduce a balanced confidence-aware sampling strategy that selects rollout samples based on their reliability while maintaining a balanced set of positive and negative examples. Second, we replace group-relative advantage estimation with a bias-corrected scheme that assigns fixed, label-dependent advantages to eliminate the amplification effect. Finally, we incorporate a consensus-based off-policy refinement stage after the RL phase, where a rejection sampling (Zhang et al., 2023) dataset is constructed to enable efficient and stable post-RL optimization.

Demonstrated by experiments on multiple mathematical benchmarks and several LLMs, DDRL achieves significant relative improvements over TTRL, with gains of 15.3% on Qwen2.5-MATH-1.5B and 12.7% on LLaMA-3.1-8B-Instruct. Our contributions are summarized as follows:

- We reveal that medium-frequency samples are particularly prone to inducing noisy rewards. Furthermore, we find that group-relative advantage normalization used in TTRL will amplify these spurious signals subsequently.
- Based on the above findings, we propose DDRL, a framework that consists of a balanced confidence-aware sampling, debiased advantage estimation, and a lightweight consensus-based off-policy refinement to mitigate the spurious signals.
- Extensive experiments on several LLMs and several challenging mathematical benchmarks demonstrate the effectiveness of DDRL in mitigating spurious signals.

2 Understanding Spurious Signals

2.1 Test-Time Reinforcement Learning

Test-time reinforcement learning (TTRL) (Zuo et al., 2025) adapts a pre-trained language model π_θ at inference time via RL, specifically avoiding reliance on ground-truth supervision. Instead, TTRL constructs training signals intrinsically using test-time scaling.

Given an unlabeled test query q, the model samples N candidate responses $\{y_1, y_2, \ldots, y_N\}$ by repeated sampling $y_i \sim \pi_\theta(\cdot \mid q)$. A consensus pseudo-label y^* is then derived via majority voting over the sampled responses. Based on this pseudo-label, TTRL defines a binary reward function

$$r(y, y^*) = \mathbb{I}(y = y^*), \tag{1}$$

and seeks to maximize the expected reward under the current policy:

$$\max_\theta \ \mathbb{E}_{y \sim \pi_\theta(\cdot|q)} \big[r(y, y^*) \big]. \tag{2}$$

Notably, both the reward signal and the optimization target are induced entirely from model-generated outputs, without external verification.

To optimize π_θ using the sampled rollouts, TTRL adopts GRPO (Shao et al., 2024) as the underlying reinforcement learning algorithm. GRPO computes advantages by normalizing rewards within each sampled group:

$$A_i = \frac{r_i - \mathrm{mean}(r)}{\mathrm{std}(r)}. \tag{3}$$

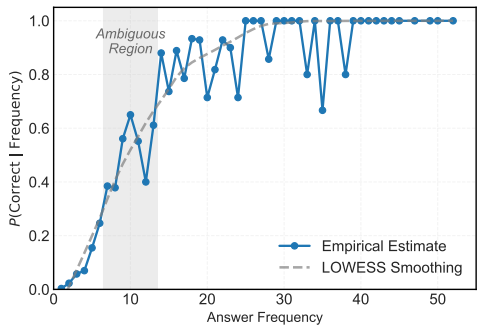
The policy is then updated by maximizing the clipped surrogate objective:

$$\max_\theta \ \mathbb{E}_i \Big[\min \big(f_i(\theta) A_i, \ \mathrm{clip}(f_i(\theta), 1-\epsilon, 1+\epsilon) A_i \big) \Big] \tag{4}$$

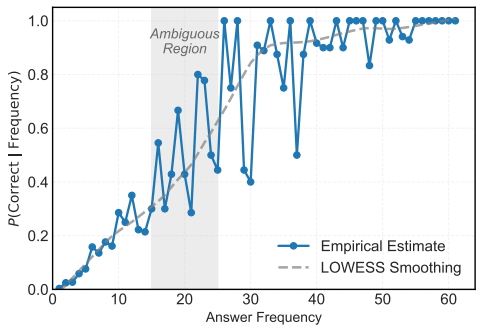
where $f_i(\theta) = \frac{\pi_\theta(y_i|q)}{\pi_{\theta_{\text{old}}}(y_i|q)}$ and ϵ denotes the clipping threshold.

2.2 Origin of Spurious Reward Signals through Answer Frequency

Majority voting provides a convenient mechanism for constructing pseudo-labels at test time, but it does not guarantee alignment with ground-truth labels. As a result, spurious reward signals are often unavoidable when all sampled rollouts are treated equally during optimization. To better understand



(a) Correctness vs. frequency on MATH-500 (Qwen2.5-Math-1.5B).



(b) Correctness vs. frequency on MATH-500 (Qwen2.5-3B).

Figure 2: Answer frequency as an imperfect proxy for reliability. We analyze the relationship between answer sampling frequency and correctness by sampling each prompt 64 times and grouping answers by frequency. High-frequency answers are generally correct, while low-frequency answers are mostly incorrect. In contrast, answers with medium sampling frequency exhibit high variance in correctness (shaded region), forming an ambiguous regime where it is hard to determine the answer correctness. These ambiguous samples constitute a major source of spurious reward signals in TTRL.

when pseudo-labels are reliable, we analyze the relationship between answer correctness and sampling frequency.

As shown in Figure 2, the correctness of a sampled response is strongly correlated with how frequently it appears among repeated generations. High-frequency responses are predominantly correct, while low-frequency responses are mostly incorrect. In contrast, responses with medium sampling frequency exhibit substantial uncertainty: their correctness probability varies sharply and is highly unstable. We provide a Bayesian explanation for this phenomenon in the Appendix C.

These medium-frequency samples constitute a dominant source of spurious reward signals, as they are neither consistently correct nor consistently incorrect. Therefore, ideally, we should minimize the use of these samples to reduce spurious signals. However, standard TTRL treats all sampled rollouts the same in optimization, allowing ambiguous medium-frequency samples to exert a disproportionate influence on the learning signal.

2.3 Amplification of Spurious Signals by Group-Relative Advantage Estimation

GRPO (Shao et al., 2024) has proven highly effective in supervised RL, particularly in settings where reliable labels ensure that positive rewards correspond to correct behaviors. However, its inductive bias implicitly assumes that rare positive samples are informative and trustworthy. In unsupervised RL, this assumption no longer holds.

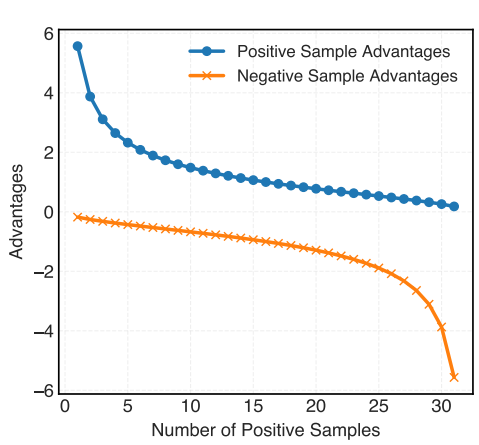
As illustrated in Figure 3, GRPO constructs advantages by normalizing rewards within each group of sampled rollouts. When the number of positive samples is small, this normalization leads to large advantage magnitudes for positive samples. In supervised scenarios, such behavior is desirable: ground-truth supervision guarantees that rare positive samples are correct, allowing the optimizer to emphasize informative signals.

In the context of TTRL, however, a small number of positive samples does not indicate scarcity of valuable supervision. Instead, since rewards are derived from majority-voted pseudo-labels, few positive samples directly reflect low consensus among model-generated responses. Low consensus, in turn, implies high uncertainty about the correctness of the pseudo-label itself. As a result, GRPO assigns large advantages precisely to those samples whose labels are the least reliable.

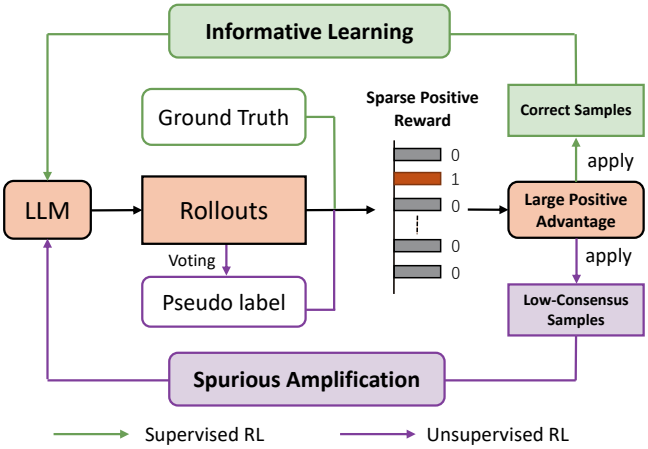
Adv. Estimation	AIME2024	MATH
GRPO	15.8	73.0
GRPO (w/o norm.)	20.6	75.0

Table 1: A preliminary comparison between group-relative and fixed advantage estimation in TTRL using Qwen2.5-Math-1.5B. GRPO (w/o norm.) removes the advantage normalization in GRPO.

This mismatch between GRPO's inductive bias and the semantics of pseudo-labels causes low-consensus and potentially incorrect samples to exert a disproportionate influence on policy updates. Consequently, spurious reward signals introduced during pseudo-label construction are further ampli-



(a) Advantage vs. number of positive samples in GRPO under 32 rollouts. When positive samples are scarce, normalization over group statistics yields large advantage magnitudes.



(b) Conceptual comparison between supervised and unsupervised settings. In supervised RL, sparse positive rewards correspond to rare but reliable signals. In unsupervised TTRL, sparse positive samples indicate low consensus and unreliable pseudo-labels; assigning large advantages to ambiguous samples amplifies spurious signals.

Figure 3: Behavior of group-relative advantage estimation under limited positive samples.

fied during advantage estimation, leading to unstable and misleading optimization dynamics.

To isolate the effect of advantage normalization, we conduct a preliminary comparison between GRPO and a simple fixed advantage scheme. As shown in Table 1, replacing group-relative advantages with constant, label-dependent values already yields consistent improvements across benchmarks. This observation suggests that the instability arises primarily from inappropriate advantage scaling.

3 Mitigating Spurious Signals

Based on the analysis in Section 2, we propose Debiased and Denoised Test-Time Reinforcement Learning (DDRL) to mitigate the spurious signals.

3.1 Balanced Confidence-Aware Sampling

To denoise the potential spurious signal brought by inappropriate reward, we propose *balanced confidence-aware sampling*, which selectively filters noisy samples while preserving informative supervision. Our design is guided by two principles: (i) confidence-aware selection based on sampling frequency, and (ii) balanced exposure to positive and negative samples.

Formally, let $c(y_i)$ denote the occurrence count of response y_i among N sampled rollouts. We use $c(y_i)$ as a confidence indicator, motivated by the empirical observation in Section 2.2 that the conditional correctness probability $P(\text{correct} \mid c)$ is high at extreme values of c and highly unstable in the medium-frequency region. Rather than optimizing over all rollouts, we select a fixed number K of samples for each prompt.

We first determine the number of positive samples as:

$$K^+ = \min\left(c(y^*),\ \left\lfloor \frac{K}{2} \right\rfloor\right), \tag{5}$$

where y^* is the pseudo-label and $\lfloor \cdot \rfloor$ denotes the floor function. This formulation ensures that (1) we do not select more positives than are available, and (2) positive samples never dominate the batch (capped at 50%), thereby enforcing a balanced label distribution. The number of negative samples is then given by:

$$K^- = K - K^+. \tag{6}$$

Based on these quotas, the sampling procedure is defined as follows. **Positive selection**: we select the top-K^+ samples corresponding to the pseudo-label y^*. **Negative selection**: we select the K^- samples with the *lowest* occurrence counts. We treat low-frequency samples as negatives to *mitigate false negatives*. In complex reasoning tasks, high-frequency alternatives may correspond to valid or even correct reasoning paths and should not be penalized. In contrast, rare outliers are statistically more likely to be incorrect hallucinations, making them safe negatives. By discarding medium-frequency responses, we remove the high-variance ambiguous region from optimization.

3.2 Debiased Advantage Estimation

Motivated by the analysis in Section 2.3, to mitigate the spurious amplification introduced by GRPO advantage estimation, we replace group-relative advantage estimation with a bias-corrected, fixed advantage assignment for rollout y_i:

$$A_i = \mathbb{I}(y = y^*) - \mathbb{I}(y \neq y^*), \tag{7}$$

where positive samples receive a constant advantage of $+1$ and negative samples receive -1, independent of the number of positive samples in the group. By decoupling advantage magnitude from group statistics, this formulation eliminates the amplification effect induced by normalization, resulting in more stable and reliable optimization under unsupervised pseudo-labels.

3.3 Consensus-Based Off-Policy Refinement

Although the preceding components reduce spurious signals during on-policy RL, policy updates can still be noisy due to stochastic optimization. However, the consensus here is typically much stronger than in the initial phase. We therefore introduce a lightweight off-policy refinement stage to consolidate the improvements.

Let $\pi_{\theta_{\mathrm{RL}}}$ denote the policy model after the RL phase. For each test query $q \in \mathcal{Q}$, we sample M responses as follows:

$$\{y_1, \ldots, y_M\}, \quad y_j \sim \pi_{\theta_{\mathrm{RL}}}(\cdot \mid q), \tag{8}$$

with $M = 128$ in our experiments. A consensus pseudo-label $y^*(q)$ is obtained via majority voting, and rejection sampling is applied to retain only responses that agree with the consensus:

$$\mathcal{A}(q) = \{(q, y_j) \mid y_j = y^*(q)\}. \tag{9}$$

We then aggregate the accepted samples and yield the off-policy dataset:

$$\mathcal{D}_{\mathrm{op}} = \bigcup_{q \in \mathcal{Q}} \mathcal{A}(q). \tag{10}$$

Starting from θ_{RL}, we perform supervised fine-tuning by maximizing

$$\mathbb{E}_{(q,y)\sim\mathcal{D}_{\mathrm{op}}}[\log \pi_\theta(y \mid q)]. \tag{11}$$

This stage distills high-consensus behaviors of the RL-adapted policy into the model, providing a stable and efficient refinement of test-time updates.

4 Experiments

4.1 Setup

Benchmarks and Models. We evaluate DDRL on three widely used mathematical reasoning benchmarks: AIME 2024 (Li et al., 2024a), AMC (Li et al., 2024a), and MATH-500 (Hendrycks et al., 2021). To assess robustness across model architectures and scales, we conduct experiments on a diverse set of models, including a base LLM (Qwen2.5-3B (Qwen et al., 2025)), an instruction-tuned LLM (Llama-3.1-8B-Instruct (Grattafiori et al., 2024)), and a math-specialized model (Qwen2.5-Math-1.5B (Yang et al., 2024)). **Baselines.** We compare DDRL against representative test-time training methods. Specifically, we include **TTRL** (Zuo et al., 2025), which performs unsupervised reinforcement learning using majority-vote pseudo-labels, and **ETMR** (Liu et al., 2025), which improves rollout diversity by selectively forking trajectories at high-entropy tokens. We also report results of the original pretrained models without any test-time adaptation, referred to as **No Adaptation**.

Implementation Details and Evaluation Protocol. All experiments are implemented using the `verl` framework (Sheng et al., 2024). For pseudo-label estimation, we sample $N = 64$ rollouts per prompt with temperature 1.0 for math-specialized models and 0.6 for other models, following prior work (Zuo et al., 2025). These rollouts are then downsampled to 32 trajectories for training. We optimize models using AdamW with a cosine learning rate schedule, where the peak learning rate is set to 5×10^{-7} for Qwen models and 2×10^{-7} for the Llama model to account for scale differences. The maximum generation length is fixed to 3072 tokens. The number of training episodes is set to 10, 30, and 80 for MATH-500, AMC, and AIME 2024, respectively, proportional to dataset size. In the off-policy refinement stage, we apply rejection sampling independently for each prompt and retain a fixed number of high-confidence samples (4 for MATH-500 and 16 for AMC and AIME 2024). We then perform supervised fine-tuning for 5 epochs with a learning rate of 1×10^{-5}. For evaluation, we report pass@1 following DeepSeek-R1, where 16 responses are generated per prompt with temperature 0.6 and top-p $= 0.95$, and accuracy is averaged across prompts.

Model	Name	AIME 2024	AMC	MATH-500	Avg
Qwen2.5-Math-1.5B	No Adaptation	7.7	28.6	32.7	23.0
	TTRL (Zuo et al., 2025)	15.8	48.9	73.0	45.9
	ETMR (Liu et al., 2025)	21.0	50.8	76.9	49.6
	DDRL	**25.0**	**52.9**	**80.7**	**52.9**
	Δ	↑19.0%	↑4.1%	↑4.9%	↑6.7%
Qwen2.5-Base-3B	No Adaptation	4.4	24.5	53.2	27.4
	TTRL (Zuo et al., 2025)	7.9	40.7	72.2	40.3
	ETMR (Liu et al., 2025)	9.2	41.7	71.7	40.9
	DDRL	**10.0**	**42.2**	**72.7**	**41.6**
	Δ	↑8.2%	↑1.2%	↑1.4%	↑1.7%
Llama-3.1-8B-Instruct	No Adaptation	4.6	23.3	48.6	25.5
	TTRL (Zuo et al., 2025)	10.0	32.3	63.7	35.3
	ETMR (Liu et al., 2025)	**16.9**	35.4	59.5	37.3
	DDRL	13.3	**38.6**	**67.6**	**39.8**
	Δ	↓21.3%	↑9.0%	↑13.6%	↑6.7%

Table 2: Performance (pass@1) comparison among DDRL and baseline methods. Δ represents the percentage performance gain achieved by our method compared to ETMR. The best results are in bold.

Model	BCS	DAE	COR	AIME 2024	AMC	MATH-500	Avg
Qwen2.5-Math-1.5B	-	-	-	15.8	48.9	73.0	45.9
	✓	-	-	17.3	46.9	74.1	46.1
	✓	✓	-	20.2	46.3	74.6	47.0
	✓	✓	✓	25.0	52.9	80.7	52.9
Qwen2.5-Base-3B	-	-	-	7.9	40.7	72.2	40.3
	✓	-	-	6.7	41.3	72.0	40.0
	✓	✓	-	10.0	42.2	72.3	41.5
	✓	✓	✓	10.0	42.2	72.7	41.6
Llama-3.1-8B-Instruct	-	-	-	10.0	32.3	63.7	35.3
	✓	-	-	13.3	38.4	64.4	38.7
	✓	✓	-	13.3	38.5	67.4	39.7
	✓	✓	✓	13.3	38.6	67.6	39.8

Table 3: Ablation study of DDRL. BCS, DAE, and COR denote balanced confidence-aware sampling, debiased advantage estimation, and consensus-based off-policy refinement, respectively. The results demonstrate the complementary effects of reducing spurious signals, correcting advantage bias, and consolidating learned behaviors.

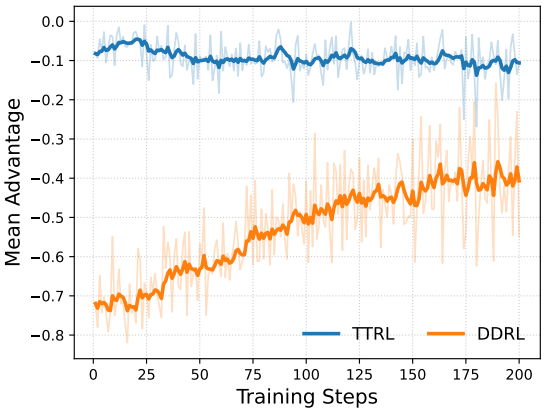
4.2 Main Results

Table 2 compares DDRL with baseline methods using pass@1 across three models and multiple mathematical reasoning benchmarks.

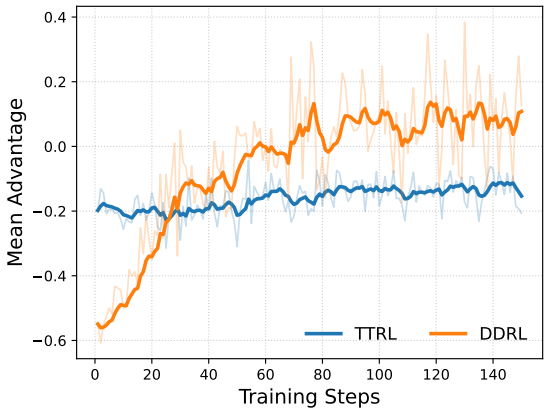
Across all settings, DDRL consistently improves over TTRL and, in most cases, over the stronger ETMR baseline. On Qwen2.5-Math-1.5B, DDRL achieves substantial gains over ETMR, improving performance by $+19.0\%$ on AIME 2024, $+4.1\%$ on AMC, and $+4.9\%$ on MATH-500, which corresponds to a $+6.7\%$ average improvement. These results indicate that mitigating spurious reward signals is particularly effective when the base model already exhibits strong reasoning ability.

On Qwen2.5-Base-3B, DDRL yields consistent but more moderate improvements across all benchmarks, resulting in a $+1.7\%$ average gain over ETMR. This suggests that while DDRL is broadly effective, the magnitude of improvement depends on the capacity of the underlying model.

On LLaMA-3.1-8B-Instruct, DDRL substantially improves performance on AMC and MATH-500, with gains of $+9.0\%$ and $+13.6\%$, respectively. Although DDRL underperforms ETMR on AIME 2024, it remains consistently stronger than TTRL and achieves the best average performance among all methods.



(a) AIME 2024 with Qwen2.5-Math-1.5B.



(b) MATH-500 with Qwen2.5-Math-1.5B.

Figure 4: Comparison of training dynamics between TTRL and DDRL via mean advantage.

4.3 Ablation Study

Table 3 reports an ablation study analyzing the individual and combined effects of the proposed components in DDRL.

Applying balanced confidence-aware sampling (BCS) alone improves performance on Qwen2.5-Math-1.5B and LLaMA-3.1-8B-Instruct, indicating that filtering medium-frequency, ambiguous samples effectively suppresses spurious reward signals. However, on Qwen2.5-Base-3B, BCS alone leads to a slight degradation, suggesting that reducing noisy supervision without correcting the optimization dynamics can limit learning for some models.

Incorporating debiased advantage estimation (DAE) consistently improves performance across all models. Compared to BCS-only training, DAE stabilizes optimization by removing frequency-dependent advantage amplification, recovering the performance drop on Qwen2.5-Base-3B and yielding clear gains on the other models. This demonstrates that debiasing advantage estimation is essential for robust learning under unsupervised reinforcement learning with pseudo-labels.

Adding consensus-based off-policy refinement (COR) further improves results in all settings. The largest gains are observed on Qwen2.5-Math-1.5B, where high-consensus behaviors learned during reinforcement learning can be effectively consolidated. On Qwen2.5-Base-3B and LLaMA-3.1-8B-Instruct, COR provides smaller but consistent improvements, indicating that this stage primarily serves as a stabilization and refinement step rather than a primary driver of performance.

4.4 Additional Analysis

Advantage Dynamics. Figure 4 compares the training dynamics of TTRL and DDRL by tracking the mean advantage during optimization on AIME 2024 and MATH-500. Since the advantage reflects the relative contribution of positive and negative samples, its magnitude and sign provide insight into how learning signals evolve over training.

On AIME 2024, the mean advantage under TTRL remains close to zero throughout training, indicating limited differentiation between positive and negative samples' contributions. In contrast, DDRL maintains a consistently negative mean advantage, suggesting that optimization is dominated by negative samples. Given the high difficulty of AIME problems and the low reliability of early pseudo-labels, this behavior suppresses premature reinforcement of incorrect positive signals.

On MATH-500, DDRL exhibits a distinct transition in advantage dynamics. Early in training, the mean advantage is strongly negative, reflecting conservative learning when pseudo-label confidence is low. As training progresses, the mean advantage gradually increases and becomes positive, indicating a shift toward learning from positive samples as pseudo-label quality improves. This adaptive transition is not observed in TTRL.

Analysis of the Off-Policy Stage. To assess the efficiency and effectiveness of the consensus-based off-policy refinement stage, we compare it with extending the on-policy RL phase under comparable computational budgets. Experiments are conducted on Qwen2.5-MATH-1.5B by allocating additional RL training time corresponding to approximately one-tenth of the original RL budget

Dataset	Setting	Epoch	Time (↓)	Pass@1 (↑)
AIME	BCS + DAE	–	–	20.2
	+ Additional RL	8	15 min	19.2
	+ COR (Ours)	5	**3 min**	**25.0**
AMC	BCS + DAE	–	–	46.3
	+ Additional RL	3	42 min	46.9
	+ COR (Ours)	5	**3 min**	**52.9**
MATH	BCS + DAE	–	–	74.6
	+ Additional RL	1	72 min	74.4
	+ COR (Ours)	5	**5 min**	**80.7**

Table 4: Comparison between additional on-policy RL training and the consensus-based off-policy refinement stage on Qwen2.5-MATH-1.5B. "BCS + DAE" denotes the model performance after the RL stage without any additional training.

for each dataset.

As reported in Table 4, additional on-policy RL yields limited or inconsistent improvements and can even degrade performance suggested on AIME 2024. In contrast, the off-policy refinement stage consistently achieves larger and more stable gains with substantially lower training cost. Training for only five SFT epochs (3-5 minutes) improves performance across all datasets, outperforming extended RL runs that require up to 72 minutes. These results indicate that consolidating high-consensus behaviors via off-policy supervision is a more efficient and reliable alternative to prolonging on-policy reinforcement learning.

5 Related Work

5.1 Test-Time Training

Test-time training (Sun et al., 2020; Liang et al., 2025; Yu et al., 2025c) refers to a paradigm where a model is adapted during inference by updating its parameters using self-supervised or pseudo-supervised signals available at test time, without access to ground-truth labels. In early test-time training (TTT) paradigms, pseudo-labeling (Liang et al., 2020) and entropy minimization (Wang et al., 2021; Yu et al., 2024, 2023) are the primary techniques used to enhance a model's generalization ability under distribution shift during inference-time.

In the context of LLMs, LMSI (Huang et al., 2023) uses the high-confidence chain-of-thought (CoT) trajectories with majority answers to SFT the language model for improving both in-domain and out-of-domain performance. Similarly, SEALONG (Li et al., 2024b) scores the sampled trajectories with Minimum Bayes Risks and then applies SFT and preference optimization. TLM (Hu et al., 2025) constructs a test-time learning paradigm for LLMs that adapts to domain shifts by minimizing the input perplexity of unlabeled test data, with sample-efficient selection and LoRA-based updates to mitigate instability. TTRL (Zuo et al., 2025) combines the TTT and TTS, using the majority voting answer to conduct RL on unlabeled data.

5.2 Unsupervised Reinforcement Learning

TTRL (Zuo et al., 2025) proposes to perform reinforcement learning using unlabeled data. Recently, a growing body of work has extended this paradigm (Prabhudesai et al., 2025; Zhao et al., 2025; Wu et al., 2025; Zhang et al., 2025a; Liu et al., 2025; Yu et al., 2025d; Wei et al., 2025; Yan et al., 2026). MM-UPT (Wei et al., 2025) extends the TTRL to a multi-modal setting. RENT (Prabhudesai et al., 2025) and INTUITOR (Zhao et al., 2025) both use the model's own confidence as the reward signal for RL. ETTRL (Liu et al., 2025) proposes entropy-fork tree majority rollout to fork rollout only at the token with high entropy to form a diverse set and reshape the advantage according to the entropy. Similarly, SPINE (Wu et al., 2025) forks the rollout only at high entropy tokens and only applies loss on the fork token. RESTRAIN (Yu et al., 2025d) applies a negative rollout penalization when the model's prediction is in a low consensus and weights each prompt with an offline estimated confidence. In contrast to these approaches, which primarily focus on designing alternative rewards or rollout strategies, our work analyzes how spurious signals arise and are amplified during optimization in unsupervised settings. We show that advantage estimation itself can introduce systematic bias when pseudo-label reliability is low, and propose a unified framework that jointly addresses the problem.

6 Conclusion

In this work, we investigate the prevailing spurious signals in TTRL. We first empirically demonstrate that ambiguous medium-frequency samples are the major source of spurious signals and group-relative advantage estimation amplifies these signals, leading to a misguided optimization. Based on these insights, we propose debiased and denoised test-time reinforcement learning, which denoises the spurious signals through a confidence-aware sampling and applies a debiased advantage estimation. Moreover, a lightweight consensus-based off-policy re-

finement stage is introduced to further enhance the consensus. Extensive experiments on multiple language models and mathematical reasoning benchmarks demonstrate that DDRL consistently outperforms existing TTRL-based methods under comparable computational budgets. These results highlight the importance of controlling uncertainty and bias in test-time optimization and point to more robust directions for unsupervised reinforcement learning at inference time.

Limitations

Our experiments focus on mathematical reasoning benchmarks with relatively well-defined correctness criteria. It remains an open question whether the proposed framework generalizes to more open-ended generation tasks, such as dialogue or creative writing, where pseudo-label ambiguity may manifest differently.

To eliminate spurious signal amplification introduced by group-relative normalization, DDRL adopts a simple fixed, label-dependent advantage assignment. While this design is effective and stabilizes optimization in unsupervised settings, it represents a deliberately conservative choice. More expressive advantage formulations (e.g., confidence-adaptive scaling or uncertainty-aware advantage shaping) may further improve learning efficiency and robustness. We leave the exploration of more sophisticated advantage designs to future work.

Acknowledgements

This research is supported by the National Natural Science Foundation of China under Grants-(62276256, U2441251). We thank Dong Yan at NLPR for early discussions and feedback on this project. Besides, we also extend our sincere thanks to the anonymous reviewers for their constructive suggestions. We thank Meituan for providing academic exchange and hardware support in this work.

References

Yang Chen, Zhuolin Yang, Zihan Liu, Chankyu Lee, Peng Xu, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025. Acereason-nemotron: Advancing math and code reasoning through reinforcement learning. *arXiv preprint arXiv:2505.16400*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. In *Proc. NeurIPS*.

Jinwu Hu, Zitian Zhang, Guohao Chen, Xutao Wen, Chao Shuai, Wei Luo, Bin Xiao, Yuanqing Li, and Mingkui Tan. 2025. Test-time learning for large language models. In *Proc. ICML*.

Jiaxin Huang, Shixiang Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2023. Large language models can self-improve. In *Proc. EMNLP*.

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*.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and 1 others. 2024. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, and 1 others. 2024a. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*.

Siheng Li, Cheng Yang, Zesen Cheng, Lemao Liu, Mo Yu, Yujiu Yang, and Wai Lam. 2024b. Large language models can self-improve in long-context reasoning. *arXiv preprint arXiv:2411.08147*.

Jian Liang, Ran He, and Tieniu Tan. 2025. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, 133(1):31–64.

Jian Liang, Dapeng Hu, and Jiashi Feng. 2020. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *Proc. ICML*.

Jia Liu, ChangYi He, YingQiao Lin, MingMin Yang, FeiYang Shen, and ShaoGuo Liu. 2025. Ettrl: Balancing exploration and exploitation in llm test-time reinforcement learning via entropy mechanism. *arXiv preprint arXiv:2508.11356*.

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

Mihir Prabhudesai, Lili Chen, Alex Ippoliti, Katerina Fragkiadaki, Hao Liu, and Deepak Pathak. 2025. Maximizing confidence alone improves reasoning. *arXiv preprint arXiv:2505.22660*.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

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

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

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. 2020. Test-time training with self-supervision for generalization under distribution shifts. In *Proc. ICML*.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2021. Tent: Fully test-time adaptation by entropy minimization. In *Proc. ICLR*.

Lai Wei, Yuting Li, Chen Wang, Yue Wang, Linghe Kong, Weiran Huang, and Lichao Sun. 2025. Unsupervised post-training for multi-modal llm reasoning via grpo. *arXiv preprint arXiv:2505.22453*.

Jianghao Wu, Yasmeen George, Jin Ye, Yicheng Wu, Daniel F Schmidt, and Jianfei Cai. 2025. Spine: Token-selective test-time reinforcement learning with entropy-band regularization. *arXiv preprint arXiv:2511.17938*.

Dong Yan, Jian Liang, Yanbo Wang, Shuo Lu, Ran He, and Tieniu Tan. 2026. What if consensus lies? selective-complementary reinforcement learning at test time. In *Proc. ACL*.

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

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

Yongcan Yu, Lingxiao He, Shuo Lu, Lijun Sheng, Yinuo Xu, Yanbo Wang, Kuangpu Guo, Jianjie Cheng, Meng Wang, Qianlong Xie, and 1 others. 2025b. Reassessing the role of supervised fine-tuning: An empirical study in vlm reasoning. *arXiv preprint arXiv:2512.12690*.

Yongcan Yu, Lijun Sheng, Ran He, and Jian Liang. 2023. Benchmarking test-time adaptation against distribution shifts in image classification. *arXiv preprint arXiv:2307.03133*.

Yongcan Yu, Lijun Sheng, Ran He, and Jian Liang. 2024. Stamp: Outlier-aware test-time adaptation with stable memory replay. In *Proc. ECCV*.

Yongcan Yu, Yanbo Wang, Ran He, and Jian Liang. 2025c. Test-time immunization: A universal defense framework against jailbreaks for (multimodal) large language models. *arXiv preprint arXiv:2505.22271*.

Zhaoning Yu, Will Su, Leitian Tao, Haozhu Wang, Aashu Singh, Hanchao Yu, Jianyu Wang, Hongyang Gao, Weizhe Yuan, Jason Weston, and 1 others. 2025d. Restrain: From spurious votes to signals–self-driven rl with self-penalization. *arXiv preprint arXiv:2510.02172*.

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

Kongcheng Zhang, Qi Yao, Shunyu Liu, Yingjie Wang, Baisheng Lai, Jieping Ye, Mingli Song, and Dacheng Tao. 2025a. Consistent paths lead to truth: Self-rewarding reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.08745*.

Qiyuan Zhang, Fuyuan Lyu, Zexu Sun, Lei Wang, Weixu Zhang, Wenyue Hua, Haolun Wu, Zhihan Guo, Yufei Wang, Niklas Muennighoff, and 1 others. 2025b. A survey on test-time scaling in large language models: What, how, where, and how well? *arXiv preprint arXiv:2503.24235*.

Yifan Zhang, Jingqin Yang, Yang Yuan, and Andrew Chi-Chih Yao. 2023. Cumulative reasoning with large language models. *arXiv preprint arXiv:2308.04371*.

Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. 2025. Learning to reason without external rewards. *arXiv preprint arXiv:2505.19590*.

Yuxin Zuo, Kaiyan Zhang, Li Sheng, Shang Qu, Ganqu Cui, Xuekai Zhu, Haozhan Li, Yuchen Zhang, Xinwei Long, Ermo Hua, and 1 others. 2025. Ttrl: Test-time reinforcement learning. *arXiv preprint arXiv:2504.16084*.

A Implementation Detail on Input Prompt Format

Following TTRL (Zuo et al., 2025), we add the reasoning prompt to the model:

Reasoning Cue

```
Please reason step by step, and put your
final answer within \boxed{}.
```

For Math-specialized models, we use it as a system prompt; for other models, we append the cue to the question.

B Why is DDRL Effective in Mitigating Spurious Signals?

It is widely recognized that spurious learning signals in TTRL primarily originate from erroneous pseudo-labels. Since DDRL also relies on pseudo-labels for optimization, a natural question arises: how does DDRL mitigate spurious signals without eliminating pseudo-labeling altogether?

We argue that spurious signals introduced during the reward construction stage stem from two distinct sources: *false negatives* and *false positives*. DDRL is designed to explicitly suppress the impact of both.

False Negatives. False negatives correspond to correct answers that are mistakenly assigned negative rewards. DDRL mitigates this issue through balanced confidence-aware sampling, which prioritizes samples with extreme answer frequencies. As shown in Figure 2, low-frequency answers are overwhelmingly incorrect, while correct answers are more likely to appear at higher frequencies. By preferentially selecting low-frequency samples as negative examples, DDRL substantially reduces the probability of assigning negative rewards to potentially correct answers, thereby suppressing false negatives.

False Positives. False positives arise when incorrect answers are incorrectly assigned positive rewards due to unreliable pseudo-labels. Although such samples cannot be entirely avoided, DDRL limits their influence in two ways. First, low-consensus samples—where false positives are more likely—have a lower probability of being selected during training. Second, DDRL assigns fixed, label-dependent advantages rather than amplifying advantages through group-relative normalization. As a result, even when false positives are sampled, their contribution to policy updates is bounded and progresses more slowly. This effect is reflected in the stabilized mean advantage dynamics shown in Figure 4, demonstrating that DDRL effectively suppresses the impact of false positive signals during optimization.

C The Probability Analysis of the Answer Frequencies and Correctness

Probabilistic Interpretation. Let $x \in [0, 1]$ denote the sampling frequency (self-consistency) of an answer under repeated sampling, and let C and $\bar{C}$ denote the events that the answer is correct or incorrect, respectively. The solid curves in Figure 5 correspond to the estimated conditional densities

$$p_c(x) \triangleq p(x \mid C), \qquad p_i(x) \triangleq p(x \mid \bar{C}), \quad (12)$$

while the dashed curve represents the posterior probability $p(C \mid x)$.

By Bayes' rule, the posterior probability of an answer being correct given its sampling frequency is given by

$$\begin{aligned} p(C \mid x) &= \frac{p(x \mid C)\, p(C)}{p(x \mid C)\, p(C) + p(x \mid \bar{C})\, p(\bar{C})} \\ &= \frac{\pi_c\, p_c(x)}{\pi_c\, p_c(x) + \pi_i\, p_i(x)}, \end{aligned} \quad (13)$$

where $\pi_c = p(C)$ and $\pi_i = p(\bar{C}) = 1 - \pi_c$ denote the prior probabilities.

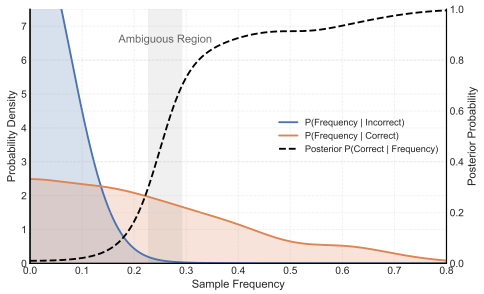
Rewriting the posterior in log-odds form yields

$$\log \frac{p(C \mid x)}{1 - p(C \mid x)} = \log \frac{\pi_c}{\pi_i} + \log \frac{p_c(x)}{p_i(x)}, \quad (14)$$

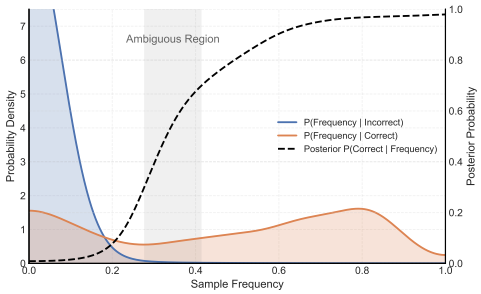
which shows that the shape of the dashed curve is entirely determined by the likelihood ratio $p_c(x)/p_i(x)$.

As shown in Figure 5, incorrect answers dominate the low-frequency regime where $p_i(x) \gg p_c(x)$, resulting in $p(C \mid x) \approx 0$. Conversely, correct answers increasingly dominate at high frequencies where $p_c(x) \gg p_i(x)$, and the posterior saturates near one. Crucially, in the intermediate frequency interval—highlighted as the *ambiguous region*—the two conditional densities substantially overlap, yielding a likelihood ratio close to one. In this regime, the posterior probability lies near the decision boundary and becomes highly sensitive to small variations in sampling frequency.

This behavior can be further characterized by the



(a) Answer distribution on MATH-500 with Qwen2.5-Math-1.5B.



(b) Answer distribution on MATH-500 with Qwen2.5-3B.

Figure 5: Conditional answer distributions reveal ambiguous frequency regions in self-consistency in the Figure 2.

derivative

$$\frac{d}{dx}p(C \mid x) = p(C \mid x)\big(1 - p(C \mid x)\big)\frac{d}{dx}\log\frac{p_c(x)}{p_i(x)}, \tag{15}$$

which explains why the posterior changes most rapidly in the ambiguous region while flattening at both extremes. Overall, this figure demonstrates that sampling frequency provides reliable confidence signals only in low- and high-frequency regimes, whereas medium-frequency answers inherently induce uncertainty and constitute a major source of noisy supervision in test-time reinforcement learning.

C.1 Additional Results

We extended the analysis to a larger AIME collection (AIME 1983–2024, 933 problems) in Figure 6. The same frequency–correctness pattern emerges: answers in the medium-frequency regime exhibit substantially higher uncertainty, while low-(<=5) and high-(>=45) frequency regions remain relatively reliable. This confirms that the key assumption behind BCS is not specific to MATH-500 but generalizes to other datasets.

Method	AIME 2024	AMC	MATH-500	**Avg**
Qwen2.5-Math-7B	12.9	35.6	46.7	31.7
+TTRL	40.2	68.1	83.4	63.9
+DDRL	**40.3**	**69.0**	**86.7**	**65.3**

Table 5: Additional Results on Qwen2.5-Math-7B.

We evaluate a larger variant within the same family, Qwen2.5-Math-7B, to better isolate scaling effects in Table 5. DDRL continues to show consistent improvements over TTRL at this larger scale

Refinement Size (compared to original setting)	AIME 2024	AMC	MATH-500
1x (original setting)	25.0	52.9	80.7
2x	25.2	53.0	80.7
4x	25.3	53.0	80.8

Table 6: Sensitivity analysis of refinement size on mathematical reasoning benchmarks.

To investigate the impact of the refinement size on model performance, we conduct a sensitivity analysis by scaling the original setting (1x) to 2x and 4x Table 6. As shown in the table, our method demonstrates remarkable robustness to this hyperparameter. Scaling up the refinement size yields highly marginal improvements across all benchmarks. Specifically, the accuracy on AIME 2024 only increases slightly from 25.0 to 25.3, while performance on AMC and MATH-500 saturates almost immediately, showing negligible gains (≤ 0.1). These results indicate that the model's capabilities are already fully elicited at the 1x scale. Given the linear increase in computational overhead associated with larger refinement sizes, the extremely low marginal benefit justifies our choice of 1x as the default setting. It achieves an optimal trade-off, maintaining strong reasoning performance while minimizing inference costs.

Algorithm 1 Debiased and Denoised Test-Time Reinforcement Learning (DDRL)

Require: Test queries $\mathcal{Q}$, pretrained policy π_{θ_0}, number of rollouts N, selected samples per query K, RL epochs E_{RL}, off-policy rollout size M, SFT epochs E_{SFT}

Ensure: Adapted policy π_θ

Initialize policy $\pi_\theta \leftarrow \pi_{\theta_0}$

▷ **On-policy Test-Time RL**

for $e = 1$ **to** E_{RL} **do**
  for each query $q \in \mathcal{Q}$ **do**
    Sample N rollouts $\{y_i\}_{i=1}^{N}$, where $y_i \sim \pi_\theta(\cdot \mid q)$
    Obtain pseudo-label y^* via majority voting
    Compute frequency $c(y_i)$ for each rollout

▷ **Balanced Confidence-Aware Sampling**

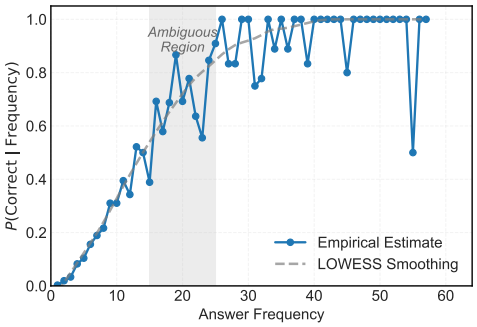
    $K^+ \leftarrow \min(c(y^*), \lfloor K/2 \rfloor)$
    $K^- \leftarrow K - K^+$
    Select K^+ positive samples with label y^*
    Select K^- negative samples with lowest frequencies

▷ **Debiased Advantage Estimation**

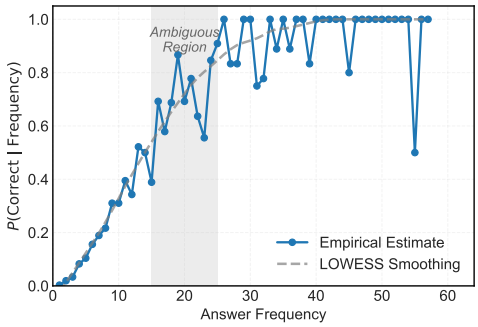
    for each selected rollout y **do**
      $A(y) \leftarrow \mathbb{I}(y = y^*) - \mathbb{I}(y \neq y^*)$
    end for
    Update π_θ using policy gradient with advantages $\{A(y)\}$
  end for
end for

▷ **Consensus-Based Off-Policy Refinement**

Initialize off-policy dataset $\mathcal{D}_{\mathrm{op}} \leftarrow \emptyset$
for each query $q \in \mathcal{Q}$ **do**
  Sample M rollouts $\{y_j\}_{j=1}^{M}$ from π_θ
  Obtain consensus label $y^*(q)$ via majority voting
  Add (q, y_j) to $\mathcal{D}_{\mathrm{op}}$ if $y_j = y^*(q)$
end for
for $e = 1$ **to** E_{SFT} **do**
  Fine-tune π_θ on $\mathcal{D}_{\mathrm{op}}$ with supervised learning
end for
return π_θ



(a) Correctness vs. frequency on AIME (Qwen2.5-Math-1.5B).



(b) Correctness vs. frequency on AIME (Qwen2.5-3B).

Figure 6: Answer frequency as an imperfect proxy for reliability on AIME (1985-2024).