Detecting and Mitigating the Correct-Answer Extinction Window in Test-Time Reinforcement Learning with Majority Voting

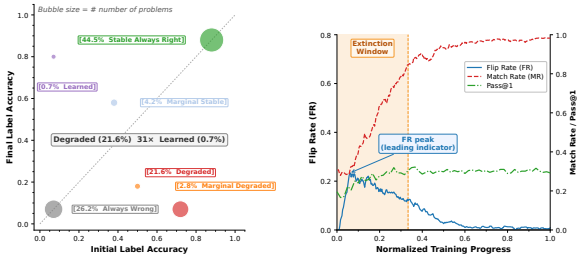
Anonymous ACL submission

Abstract

Test-time reinforcement learning (TTRL) reports substantial accuracy gains on mathematical reasoning benchmarks using majority vote as a pseudo-label signal. We argue these gains are systematically misinterpreted: most reflect sharpening of already-solvable problems rather than genuine learning, while problems corrupted from correct to incorrect outnumber truly learned ones, and this damage is irreversible once majority vote locks onto a wrong answer. Per-problem tracking reveals that correct-answer signals in low-ability problems are briefly active before being permanently suppressed, a phenomenon we term the *Correct-Answer Extinction Window*, with Flip Rate (FR) as its leading indicator. We thus propose TTRL-Guard, a lightweight framework with three mechanisms targeting the extinction window: Flip-Rate-Aware Reward Scaling (FRS) down-weights at-risk updates as FR declines, Minority-Preserving Sampling (MPS) retains gradient signal from minority correct answers, and Risk-Conditioned Sparse Updatings (RCSU) suspends updates on polarized problems. Experiments across three models and four benchmarks show that TTRL-Guard achieves the best average pass@1 on Qwen2.5-7B-Instruct and Qwen3-4B, improves relatively over TTRL by +54% on AIME 2025.

1 Introduction

Reinforcement learning with verifiable rewards (RLVR) has powered recent breakthroughs in LLM reasoning (Guo et al., 2025; Comanici et al., 2025; Yang et al., 2025). However, its reliance on ground-truth labels poses a fundamental scalability ceiling as tasks grow in complexity. This has spurred growing interest in unsupervised alternatives, most notably Test-time reinforcement learning (TTRL) (Zuo et al., 2025), which uses majority voting (MV) over sampled rollouts as pseudo-labels to enable self-improvement without human annotation. Alongside training-free distribution sharpening methods (Ji et al., 2026), unsupervised RL (Zhao et al., 2025; Simonds and Yoshiyama, 2025) is widely celebrated for pushing capability boundaries. Yet a critical question remains unanswered: *Are these models truly learning to reason, or merely learning to agree with themselves?*



(a) Label Migration (b) Training Dynamics

Figure 1: Label Dynamics During Test-Time Reinforcement Learning. (a) Label Migration Patterns Across Training. (b) Training Dynamics: Flip Rate as a Leading Indicator.

To answer this, we conduct a per-problem trajectory analysis across three models (Llama-3.2-3B-Instruct(Grattafiori et al., 2024), Qwen2.5-7B-Instruct (Qwen et al., 2024), Qwen3-4B (Yang et al., 2025)) and two reasoning benchmarks (AIME 2025 (Li et al., 2024), MATH-500 (Hendrycks et al., 2021)), comparing each problem's Initial Label Accuracy (ILA) and Final Label Accuracy (FLA) to classify problems into six categories (Details in Appendix A). As shown in Figure 1a, 44.5% of problems are already solvable before training (Stable Always Right); TTRL merely sharpens their pass@k toward 1.0, consistent with recent theoretical analyses (He et al., 2026). Genuine zero-to-one capability acquisition accounts for a negligible 0.7% (Learned). Meanwhile, 21.6% of initially solvable problems suffer accuracy drops during training (Degraded), sug-

gesting that optimization pressure on hard problems actively erodes performance on easier ones. The headline accuracy gains are thus largely an illusion of learning.

More alarmingly, this illusion masks a hidden cost. We term this *Asymmetric Degradation*: problems corrupted from correct to incorrect outnumber genuinely learned ones by 31× (21.6% vs. 0.7%), yet degradation stays invisible in aggregate validation curves because it is dwarfed by the sharpening of already-solved problems. To understand this mechanism, we trace pseudo-label dynamics via two signals: the Flip Rate (FR), the fraction of problems whose majority-vote answer changes between consecutive steps, and the Match Rate (MR), the fraction of problems where the majority-vote answer matches the ground-truth label. As shown in Figure 1b for Llama-3.2-3B-Instruct on AMC (Li et al., 2024), FR rises sharply in early training while correct answers still win the majority vote on over 31% of problems. As false consensus solidifies, FR collapses and MR locks in the wrong answer, causing pass@1 on the test set to drop irreversibly. We formalize this brief window as the *Correct-Answer Extinction Window*: once FR decays past a critical threshold, the pseudo-label cannot be recovered.

Recent work addresses majority-vote (MV) failures through external tool verification (Liao et al., 2026a), step-wise confidence weighting (Wang et al., 2025), difficulty-aware curricula (Moradi and Mudur, 2026; Yang et al., 2026), and generator-verifier co-evolution (Pan et al., 2026). Despite their diversity, these methods share a common limitation: they treat MV failure as a static property of the input and apply fixed interventions. To intercept this failure without external supervision, we propose **TTRL-Guard**, which monitors the FR as an online, label-free uncertainty signal and intervenes within the Extinction Window via three synergistic mechanisms: Flip-Rate-Aware Reward Scaling (FRS) down-weights updates on unstable pseudo-labels, Minority-Preserving Sampling (MPS) prolongs the survival of correct-minority signals, and Risk-Conditioned Sparse Updating (RCSU) halts updates once false consensus is formed. Experiments across three models and four benchmarks show that TTRL-Guard achieves the best average pass@1 on both Qwen2.5-7B-Instruct and Qwen3-4B, improves relatively over TTRL by +54% on AIME 2025, and cuts the degraded-problem fraction from 60.2% to 28.0% on Llama-3.2-3B-Instruct.

Collectively, these findings reframe TTRL not as a general capability booster but as a sharpening mechanism bounded by the initial label accuracy distribution. Macro-level gains mask a fragile interior: pre-solvable problems benefit while the rest silently degrade, and the window for intervention is narrow. We find this window is most actionable in the moderate capability-difficulty regime (30-70% initial pass@1); in extreme scenarios with near-zero or near-perfect initial accuracy, the extinction window barely opens. TTRL-Guard demonstrates that this window, once identified, can be acted upon without any external supervision, pointing toward a broader principle: reliable unsupervised RL requires not just better pseudo-labels, but dynamic awareness of when those labels can no longer be trusted.

2 Dissecting TTRL Dynamics

We analyze TTRL training on three models (Llama-3.2-3B-Instruct, Qwen2.5-7B-Instruct, Qwen3-4B) and four datasets (AIME 2024, AIME 2025, AMC, MATH-500). Per-batch metrics include Match Rate (MR), the fraction of sampled responses matching the majority-vote winner; Flip Rate (FR), the fraction of problems whose majority-vote answer changes between consecutive steps; and Label Accuracy (LA), the per-problem fraction of responses matching the ground truth. Pass@1 is evaluated on the full test set periodically and serves as the primary aggregate metric.

2.1 Problem Outcome Distribution

We classify each problem into six outcomes: Stable Always Right (AR), Learned, Degraded, Marginal Stable, Marginal Degraded, and Always Wrong (Details in Appendix A). As shown in Figure 2, Degraded problems far outnumber Learned problems by the end of training across most model-benchmark combinations. On MATH-500, Llama-3.2-3B-Instruct reaches 53.8% Degraded versus 0.2% Learned; Qwen2.5-7B-Instruct 13.8% versus 0.6%; Qwen3-4B 0.8% versus 0.4%.

This asymmetry is capacity-dependent: Qwen3-4B is largely immune because 74.4% of its problems are already Stable AR at initialization, so MV reinforces correct answers. Critically, this damage is invisible in Pass@1. Llama-3.2-3B-Instruct's Pass@1 appears stable or even improves slightly, even as 54% of its MATH-500 problems suffer irreversible accuracy drops, demonstrating that ag-

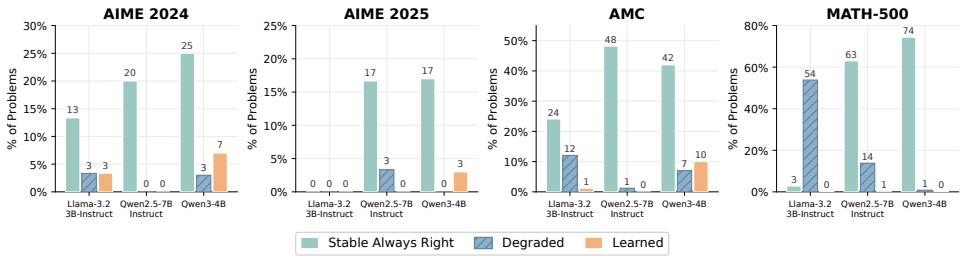


Figure 2: Problem-fate breakdown (Stable AR, Degraded, and Learned) for three models across four benchmarks after TTRL training.

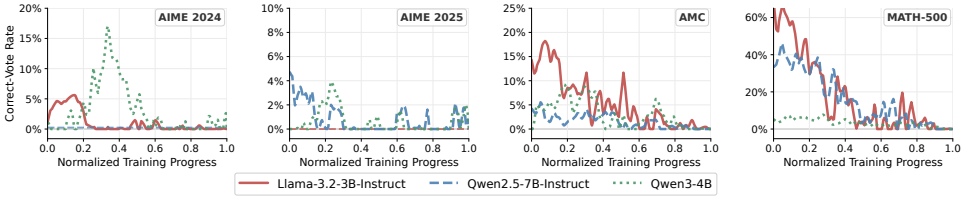


Figure 3: Correct-vote rate over normalized training progress, computed on the Degraded and Always-Wrong problem subsets.

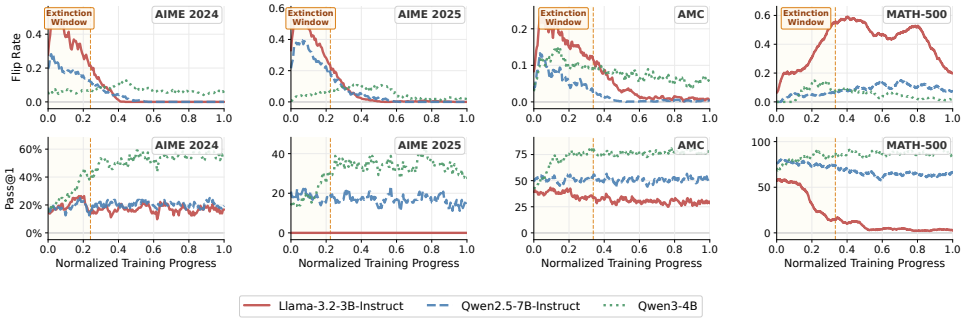


Figure 4: Flip Rate and Pass@1 over normalized training progress for each model across four benchmarks.

gregate accuracy curves mask individual problem degradation.

2.2 The Correct-Answer Extinction Window

For ultimately Degraded or Always Wrong problems, Figure 3 plots the correct-vote rate, the fraction of steps where the correct answer wins majority voting, over normalized training progress. For Llama-3.2-3B-Instruct on MATH-500, this rate starts near 58% but collapses to near zero by the final third of training. Once MV consolidates on the wrong answer, the correct gradient signal is permanently lost. The collapse arrives faster on harder datasets and is less pronounced in stronger models, confirming that the phenomenon is concentrated in weaker models on harder problems.

This irreversibility is a critical observation: we find that problems entering the Degraded category at any point rarely recover to Stable AR status in subsequent training, indicating that once false consensus forms, escape becomes increasingly unlikely under TTRL. This defines the *Correct-Answer Extinction Window*: the brief early-training interval during which correct answers remain viable as minority signals before being permanently suppressed.

2.3 Flip Rate as a Leading Indicator

FR is computed entirely from model samples, providing a label-free early-warning signal for the extinction window. As shown in Figure 4, FR peaks when MV is still unstable, then collapses as consensus forms, before Pass@1 visibly drops. On MATH-500, where MV consensus forms more gradually, FR rises throughout the early phase before subsiding. Critically, during the high-FR phase, Pass@1 curves across models remain bundled; divergence appears only after FR has collapsed. This temporal ordering suggests that FR dynamics track the underlying label stability.

On MATH-500, the extinction window spans the first third of training; on AIME 2024/2025,

it closes earlier due to higher problem difficulty. These observations motivate TTRL-Guard: the FR signal provides a real-time, label-free proxy for when the extinction window is open, enabling targeted intervention before correct-minority signals are permanently suppressed.

3 Method

The analysis in Section 2 identifies the core failure mode of TTRL: during the early extinction window, MV systematically suppresses correct-minority signals, and once the wrong answer's match rate exceeds $1/2$, the pseudo-label becomes locked in an irreversible state. TTRL provides no mechanism to detect or counteract this process. We propose TTRL-Guard, a lightweight framework that intervenes precisely during this critical window by monitoring per-problem label dynamics in real time. As shown in Figure 5, TTRL-Guard acts only on at-risk problems before signals are permanently lost, leaving beneficial sharpening on mastered problems undisturbed and maintaining low computational overhead.

3.1 Preliminary: TTRL with Majority-Voting

TTRL trains π_θ on a fixed unlabeled set $\mathcal{Q} = \{q_i\}_{i=1}^N$. At each step t, the model draws K responses $\{a_i^{(k)}\}_{k=1}^K$ per problem and derives a pseudo-label via majority voting:

$$\hat{y}_i^t = \arg\max_a \sum_{k=1}^{K} \mathbf{1}[a_i^{(k)} = a]. \tag{1}$$

This pseudo-label serves as the reward target, $r(a, \hat{y}_i^t) = \mathbf{1}[a = \hat{y}_i^t]$, and the policy is optimized with GRPO (Shao et al., 2024) to maximize $\mathbb{E}_{a\sim\pi_\theta}[r(a, \hat{y}_i^t)]$. As shown in Section 2, this reward is reliable only while the correct answer holds a majority, a condition that collapses during the extinction window. TTRL-Guard augments this loop with state monitoring per-problem and three targeted interventions.

3.2 Per-Problem State Variables

Building on Eq. (1), two additional step-level statistics are maintained:

$$\mathrm{MR}_i^t = \frac{\max_a \sum_k \mathbf{1}[a_i^{(k)} = a]}{K}, \tag{2}$$

$$\mathrm{FR}_i^t = \frac{1}{W} \sum_{s=t-W+1}^{t} \mathbf{1}[\hat{y}_i^s \neq \hat{y}_i^{s-1}], \tag{3}$$

where MR_i^t (match rate) measures how dominant the leading answer is, and FR_i^t (flip rate) is the fraction of the past W steps in which the pseudo-label changed, taking values in $[0, 1]$.

Two cumulative state variables are additionally maintained without any ground-truth labels:

$$\mathrm{HadComp}_i^t = \bigvee_{s\leq t} \mathbf{1}[\mathrm{FR}_i^s > \tau_{\mathrm{FR}}], \tag{4}$$

$$\bar{\mathrm{MR}}_i^t = \frac{1}{\min(t, W)} \sum_{s=t-W+1}^{t} \mathrm{MR}_i^s, \tag{5}$$

where τ_{FR} is a flip-rate threshold and W is a history window. $\mathrm{HadComp}_i^t$ is set permanently once the windowed flip rate exceeds τ_{FR}, distinguishing problems that have ever experienced answer competition from those that have always been dominated by a single answer. $\bar{\mathrm{MR}}_i^t$ is the sliding-mean match rate over the past W steps. As shown in Figure 5 (top), declining FR foreshadows MV lock-in, and $\mathrm{HadComp}$ separates benign consolidation (a Stable AR problem settling on the correct answer) from dangerous consolidation (a wrong answer taking over). These two variables are the sole observables driving all three components.

3.3 Flip-Rate-Aware Reward Scaling

As shown in Figure 5 (top), pseudo-label reliability varies dramatically across the extinction window: during high-FR phases, the correct answer appears in 41.1% of samples, but drops to 3.3% once FR collapses (Llama-3.2-3B-Instruct on MATH-500, Section 2.2). Uniform reward weights amplify wrong-answer gradients at these low-confidence moments. Flip-Rate-Aware Reward Scaling (FRS) down-weights rewards continuously with FR, creating adaptive credibility weights. Importantly, FRS scales only the reward weight, not the pseudo-label, preserving diverse response sampling.

The adjusted reward is

$$\tilde{r}(a,\, \hat{y}_i^t) = r(a,\, \hat{y}_i^t) \cdot w_i^t, \tag{6}$$

where $r(a, \hat{y}_i^t) = \mathbf{1}[a = \hat{y}_i^t]$ is the original binary reward and the weight decomposes as $w_i^t = \alpha_i^t \cdot \gamma_i^t \cdot \delta_i^t$, with

$$\alpha_i^t = 1 - \lambda_1 \cdot \mathrm{FR}_i^t, \tag{7}$$

$$\gamma_i^t = 1 - \lambda_2 \cdot \mathbf{1}[C_1], \tag{8}$$

$$\delta_i^t = 1 - \tfrac{\lambda_2}{2} \cdot \mathbf{1}[C_2]. \tag{9}$$

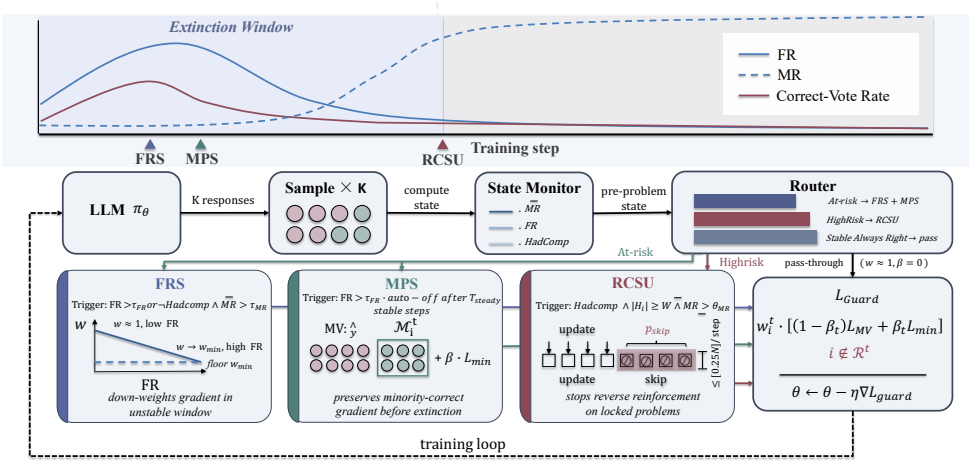


Figure 5: Overview of TTRL-Guard. The top panel illustrates the *Extinction Window*, where the FR peaks early and decays while the correct-vote rate collapses irreversibly. The middle panel depicts the core pipeline for real-time problem monitoring and routing. The bottom panel details the three intervention pathways: at-risk problems enter FRS and MPS concurrently, high-risk cases are routed to RCSU, and stable problems bypass all modules unaltered.

The two trigger conditions are defined as

$$C_1 : \ \mathrm{MR}_i^t > \tau_{\mathrm{MR}} \ \wedge \ \mathrm{FR}_i^t > \tau_{\mathrm{FR}}, \tag{10}$$

$$C_2 : \ \neg\mathrm{HadComp}_i \ \wedge \ |\mathcal{H}_i| \geq W \ \wedge \ \bar{\mathrm{MR}}_i^t > \tau_{\mathrm{MR}}. \tag{11}$$

C_1 and C_2 are mutually exclusive by construction: C_1 requires active $\mathrm{HadComp}_i^t$, while C_2 requires it has never triggered, so at most one of γ_i^t and δ_i^t deviates from 1 per step. α globally reduces reward magnitude during the unstable phase. γ targets the contradictory state of high confidence with high flip rate (C_1), the signature of wrong-answer suppression (Section 2.3). δ lightly penalizes never-challenged problems (C_2), likely stuck in wrong attractors; the trigger count is capped at W to prevent gradient starvation. The weight is clipped at $w_i^t \geq w_{\min}$ to preserve non-zero gradients.

3.4 Minority-Preserving Sampling

Section 2.2 shows that the correct answer survives as a competitive minority early in training before being systematically extinguished. Standard MV treats all non-majority responses as noise and discards their gradient signal entirely. This represents a loss of learnable information during the critical early window when the correct answer is still recoverable. Minority-Preserving Sampling (MPS) preserves this signal by assigning small positive rewards to minority answers that appear with sufficient frequency, effectively retaining them as weak learning targets. Unlike (Pan et al., 2026) and T^{3}RL (Liao et al., 2026b), which require external verification or self-confidence scores, MPS operates entirely label-free, relying only on the relative frequency of responses in the sample.

MPS activates when $\mathrm{FR}_i^t > \tau_{\mathrm{FR}}$ and mixes the standard MV objective with a minority-preservation term. The minority candidate set is

$$\mathcal{M}_i^t = \{a : \mathrm{vote}(a) \geq \lfloor K/4 \rfloor, \ a \neq \hat{y}_i^t\}, \tag{12}$$

where $\mathrm{vote}(a)$ counts responses whose extracted answer equals a. Answers in $\mathcal{M}_i^t$ receive a small positive reward $r_{\min} = \varepsilon \cdot \mathbf{1}[a \in \mathcal{M}_i^t]$, where $\varepsilon \ll 1$ is a minority reward coefficient. Here $\mathcal{L}_{\mathrm{MV}}(q_i)$ denotes the standard GRPO loss under the majority-vote reward $r(a, \hat{y}_i^t)$ (Eq. (1)), and $\mathcal{L}_{\min}(q_i)$ is the minority-preservation loss computed with reward $r_{\min}$ on answers in $\mathcal{M}_i^t$ and zero otherwise. The training objective becomes

$$\mathcal{L}_i^t = (1 - \beta_t)\, \mathcal{L}_{\mathrm{MV}}(q_i) + \beta_t\, \mathcal{L}_{\min}(q_i), \tag{13}$$

where $\beta_t \in [0, \beta_{\max}]$ scales linearly with FR_i^t. Once FR_i^t drops below τ_{FR}, $\beta_t \to 0$ and Eq. (13) recovers TTRL automatically. If FR_i^t stays below τ_{FR} for T_{steady} consecutive steps, minority protection deactivates completely, since a stably dominant answer makes further diversity pressure counterproductive.

3.5 Risk-Conditioned Sparse Updating

Once a wrong pseudo-label locks in, the extinction window closes and FR collapses. Continuing standard updates only accumulates reverse reinforcement. Risk-Conditioned Sparse Updating (RCSU)

skips gradient updates on high-risk problems, halting harmful gradients while maintaining state monitoring. A problem is flagged as high-risk when all three conditions hold: (i) $\mathrm{HadComp}_i^t$ (problem experienced answer competition), (ii) $|\mathcal{H}_i| \geq W$ (sufficient history accumulated), and (iii) $\bar{\mathrm{MR}}_i^t > \theta_{\mathrm{MR}}$ (high confidence in current answer). These identify problems once contested but now re-locked. Criterion (ii) uses history length rather than consecutive counts, since FR fluctuates on hard datasets causing resets before accumulating. Problems are skipped with probability p_{skip} while maintaining state monitoring; stochastic skipping preserves recovery possibility if pseudo-labels flip. At most $\lfloor 0.25N \rfloor$ problems are skipped per step to maintain effective batch size.

3.6 The TTRL-Guard Training Loop

Figure 5 (middle and bottom) illustrates how the three components integrate into a single training loop. At each step, the model generates K responses per problem and the State Monitor updates $\hat{y}_i^t$, MR_i^t, FR_i^t, $\mathrm{HadComp}_i^t$, and $\bar{\mathrm{MR}}_i^t$ for every problem. The Router then assigns each problem to one of three paths: at-risk problems ($\mathrm{FR}_i^t > \tau_{\mathrm{FR}}$) enter FRS and MPS jointly; high-risk problems ($\mathrm{HighRisk}_i^t = 1$) are handled by RCSU; stable AR problems bypass the intervention components with $w_i^t \approx 1$ and $\beta_t \approx 0$, contributing directly to $\mathcal{L}_{\mathrm{Guard}}$ unchanged.

Formally, let $\mathcal{R}^t = \{i : \mathrm{HighRisk}_i^t = 1\}$ denote the high-risk set at step t. The combined objective is

$$\mathcal{L}_{\mathrm{Guard}} = \sum_{i \notin \mathcal{R}^t} w_i^t \cdot \mathcal{L}_i^t, \tag{14}$$

where w_i^t is the FRS weight from Eq. (6), $\mathcal{L}_i^t$ is the per-problem objective from Eq. (13), and problems in $\mathcal{R}^t$ are excluded with probability p_{skip}.

4 Experiments

4.1 Experimental Setup

Models and Datasets. To evaluate the generalization ability of our method across diverse model architectures, we conduct experiments on three models spanning a wide range of parameter scales: Llama-3.2-3B-Instruct (Grattafiori et al., 2024), Qwen2.5-7B-Instruct (Qwen et al., 2024), and Qwen3-4B (Yang et al., 2025). For training and evaluation, we employ four representative mathematical reasoning benchmarks: AIME 2024 (Li et al., 2024), AIME 2025 (Li et al., 2024), AMC (Li et al., 2024), and MATH-500 (Hendrycks et al., 2021).

Method	AIME 2024	AIME 2025	AMC	MATH-500	Average
Llama-3.2-3B-Instruct					
Base Model	4.7	0.0	20.6	43.9	17.3
TTRL	13.3	0.0	31.3	52.1	24.1
CoVerRL	16.9	1.8	34.8	57.0	27.6
SCOPE	17.1	0.8	**38.7**	**61.7**	**29.6**
TTRL (*w/* FRS)	17.1	**3.3**	34.0	54.0	27.1
TTRL (*w/* MPS)	**20.3**	0.8	33.3	55.8	27.6
TTRL (*w/* RCSU)	16.7	0.9	34.8	54.9	26.8
TTRL-Guard	19.5	**3.3**	36.8	57.3	29.2
Qwen2.5-7B-Instruct					
Base Model	18.1	6.4	43.6	77.2	36.3
TTRL	20.0	15.6	52.4	81.0	42.3
CoVerRL	21.5	16.7	**55.0**	80.0	43.3
SCOPE	21.1	14.6	51.1	81.0	42.0
TTRL (*w/* FRS)	20.2	23.3	52.6	82.1	44.6
TTRL (*w/* MPS)	20.5	20.0	53.5	**82.2**	44.1
TTRL (*w/* RCSU)	21.3	20.1	51.2	81.6	43.6
TTRL-Guard	**21.9**	**24.1**	53.8	82.0	**45.5**
Qwen3-4B					
Base Model	3.3	1.8	19.3	52.9	19.3
TTRL	36.7	33.3	71.4	89.1	57.6
CoVerRL	**40.5**	31.8	69.5	90.3	58.0
SCOPE	38.8	35.6	73.0	90.4	59.5
TTRL (*w/* FRS)	35.5	36.1	72.2	90.6	58.6
TTRL (*w/* MPS)	38.0	35.6	**73.6**	90.7	59.5
TTRL (*w/* RCSU)	36.4	35.9	73.2	90.3	59.0
TTRL-Guard	37.8	**36.2**	**73.6**	**91.0**	**59.7**

Table 1: Performance comparison of test-time adaptation methods (pass@1, %). Bold and underlined values indicate the best and second-best results, respectively.

Baselines and Experimental Details. We compare TTRL (Zuo et al., 2025), our proposed TTRL-Guard, and two contemporaneous methods, SCOPE (Wang et al., 2025) and CoVerRL (Pan et al., 2026), which address majority-vote failures through complementary mechanisms. Following TTRL, all runs sample 64 responses per problem for majority voting and downsample 32 for policy updates. Detailed hyperparameter configurations are provided in Appendix B.

Metrics. We report pass@1 calculated over 4 responses per question as the primary accuracy metric (Chen et al., 2021). We also report the Learned/Degraded ratio (L/D), defined as the fraction of problems whose per-problem label accuracy improves during training divided by the fraction that declines, as a diagnostic metric capturing the efficiency of knowledge acquisition relative to knowledge corruption that aggregate accuracy curves conceal.

4.2 Main Results

Table 1 reports pass@1 for all methods across three models and four benchmarks. TTRL-Guard achieves the highest average on Qwen2.5-7B-Instruct (45.5%) and Qwen3-4B (59.7%), and

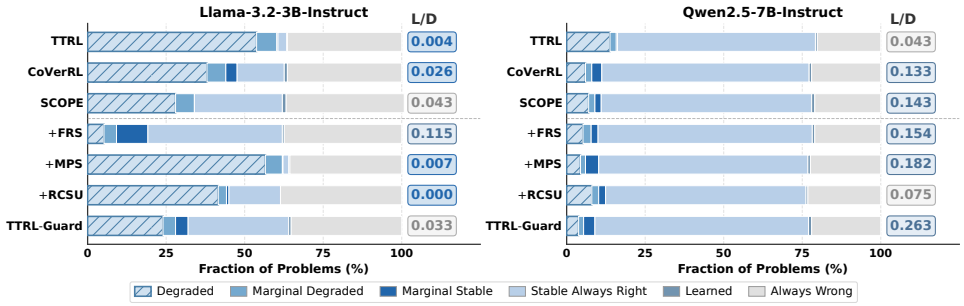


Figure 6: Per-problem distribution at training end on MATH-500. Right-margin L/D values (Learned / Degraded) measure knowledge acquisition efficiency.

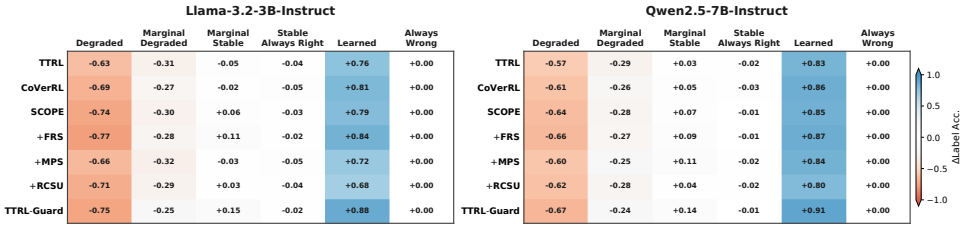


Figure 7: Mean ΔLA per problem category and method on MATH-500 (red = degradation, blue = improvement).

is competitive on Llama-3.2-3B-Instruct, trailing SCOPE by only 0.4% on average without relying on any external confidence estimator.

Model-capability-dependent improvements. On stronger models (Qwen3-4B and Qwen2.5-7B-Instruct), TTRL-Guard achieves consistent gains: 59.7% average on Qwen3-4B (leading on three of four benchmarks) and +54% relative improvement on AIME 2025 for Qwen2.5-7B-Instruct, outperforming single-component baselines by 0.9% on average. On weaker models (Llama-3.2-3B-Instruct), TTRL-Guard produces gains even in extreme regimes: reaching 3.3% on AIME 2025 (where TTRL yields 0%), and remaining competitive on MATH-500 and AMC (57.3% and 36.8%) compared to external-signal methods (61.7% and 38.7%), all without relying on external confidence estimators.

Ablation study. Each component contributes in a model-dependent manner. On Llama-3.2-3B-Instruct, FRS is the dominant single component, raising the average to 27.1% with the only consistent gains on AIME 2025, a result not matched by MPS or RCSU in isolation. MPS provides the largest lift on AIME 2024, while RCSU yields modest but broad improvements across benchmarks. On Qwen2.5-7B-Instruct, all three components are effective: FRS leads on AIME 2025 and ties for best on MATH-500; MPS achieves the top MATH-500 result; RCSU performs best on AIME 2024. On Qwen3-4B, where TTRL already performs strongly, each single component improves over TTRL, and the full TTRL-Guard combination matches or exceeds all three ablations, confirming that FRS, MPS, and RCSU are complementary rather than redundant.

4.3 Per-Problem Analysis

Pass@1 masks a harm-to-benefit asymmetry that only TTRL-Guard resolves. Figure 6 reveals what aggregate curves conceal. On Llama-3.2-3B-Instruct, TTRL leaves over 60% of problems Degraded while genuinely learning fewer than 1%, yielding L/D $= 0.004$, a severe asymmetry invisible in validation pass@1. Adding FRS alone cuts the degraded fraction sharply and raises L/D to 0.12, confirming that label protection is the dominant lever, yet pass@1 rises only modestly: shielding without a fresh learning signal cannot unlock further gains. TTRL-Guard combines FRS with MPS and RCSU to supply that signal, achieving the best pass@1 among all TTRL variants with L/D $= 0.03$ and the lowest degraded fraction. SCOPE's higher accuracy traces entirely to sharpening already-solvable problems, not to protecting degraded ones; it achieves a higher score by widening the harm gap rather than closing it. On Qwen2.5-7B-Instruct, TTRL-Guard achieves the lowest degraded fraction across all methods, beat-

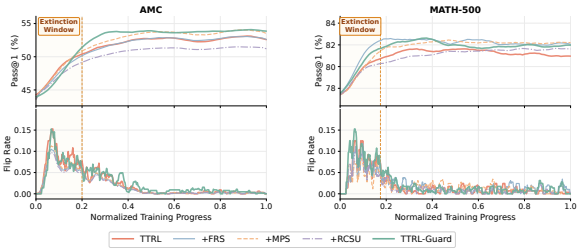


Figure 8: Training dynamics of TTRL-Guard variants on AMC and MATH-500 datasets for Qwen2.5-7B-Instruct.

ing both SCOPE and CoVerRL, confirming that all three mechanisms act synergistically once the base model has sufficient initial capacity.

Accuracy degradation is concentrated and severe within the Degraded group. Figure 7 measures the depth of accuracy shift within each problem category. On Llama-3.2-3B-Instruct under standard TTRL, problems in the Degraded group experience a mean ΔLA of -0.63: originally solvable problems driven close to zero by false-consensus reinforcement, representing irreversible capability regression. Marginal Stable problems show near-zero shifts, indicating training fails to consolidate them. TTRL-Guard is the only configuration yielding positive shifts on both models, suggesting MPS extends the correct-minority survival window long enough for gradient updates to take effect. On Qwen2.5-7B-Instruct, the Degraded group is smaller with accordingly attenuated ΔLA values, consistent with fewer high-risk problems at initialization.

Training dynamics reveal the Extinction Window. Figure 8 exposes a striking dissociation: all five methods converge to near-identical final Pass@1 values, yet their Flip Rate trajectories diverge sharply during the Extinction Window. Guard-equipped methods suppress the FR peak earlier and more aggressively, and their accuracy advantage materialises precisely after this window closes (e.g., TTRL-Guard reaches 53.8% vs. TTRL's 52.4% on AMC). This temporal ordering provides causal evidence that controlling answer-flip dynamics during the Extinction Window is the operative mechanism behind TTRL-Guard's improvements.

5 Related Work

Unsupervised RLVR. Supervised RLVR has proven effective for reasoning alignment with exact feedback (Guo et al., 2025), but its reliance on ground-truth labels limits broader applicability. Unsupervised RLVR addresses this with self-generated proxy rewards, either from single-model signals such as self-certainty and entropy (Zhao et al., 2025; Agarwal et al., 2026), or from ensemble agreement via majority voting and semantic consistency (Zuo et al., 2025; Zhang et al., 2026; Zhou et al., 2025). A complementary direction grounds rewards externally through unlabeled corpora or generation-verification asymmetries (Dong et al., 2025; Simonds and Yoshiyama, 2025). A fundamental limitation of intrinsic approaches is that majority-vote signals can actively reinforce incorrect behaviors when consensus is corrupted (Zhang et al., 2025b), a failure our work identifies as the Correct-Answer Extinction Window.

Test-Time Training. Test-time training (Zhang et al., 2025a) adapts model parameters at inference time to handle distribution shifts (Sun et al., 2024), recently extended to LLMs (Hardt and Sun, 2024). Test-Time Reinforcement Learning (Zuo et al., 2025) advances this by using majority-vote pseudo-labels to enable self-improvement without human annotation. Subsequent work (Tan et al., 2026) addresses majority-vote failures through external tool verification (Liao et al., 2026a), confidence weighting (Wang et al., 2025), difficulty-aware curricula (Moradi and Mudur, 2026; Yang et al., 2026), and generator-verifier co-evolution (Pan et al., 2026). Despite their diversity, these methods treat pseudo-label failure as a static property, leaving the dynamic degradation process during training unexamined.

6 Conclusion

TTRL's accuracy gains are systematically misinterpreted: most reflect sharpening of already-solvable problems, while corrupted problems far outnumber genuinely learned ones, a disparity invisible in aggregate curves. Per-problem trajectory analysis identifies the *Correct-Answer Extinction Window* as the underlying mechanism, with FR as a reliable leading indicator. TTRL-Guard intervenes within this window via three complementary mechanisms, substantially reducing degradation and achieving state-of-the-art performance among label-free TTRL variants, motivating future research toward training strategies robust to untrustworthy pseudo-labels.

Limitations

TTRL-Guard's effectiveness depends critically on the presence of correct-minority signals during the early training phase, which constrains its applicability to three regimes:

Low initial capability. When base model accuracy is near zero, the majority vote provides insufficient signal for minority-correct answers to exist. The FR signal becomes unreliable due to low answer diversity, limiting TTRL-Guard's ability to identify the extinction window.

High initial capability. When most problems are already solvable at initialization, the extinction window barely manifests, and problems rarely experience the competition phase. FRS and MPS have minimal effect, with improvements coming primarily from RCSU's selective skipping rather than extinction-window protection.

Hyperparameter sensitivity. The extinction window's characteristics (peak FR timing, decay rate) vary significantly with problem difficulty. Hyperparameters optimized on one dataset may degrade performance on others, necessitating dataset-specific tuning.

TTRL-Guard is most effective in the moderate capability-difficulty regime (approximately 30-70% initial pass@1), where the extinction window is both present and sufficiently non-trivial for intervention.

References

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. 2026. The unreasonable effectiveness of entropy minimization in llm reasoning. *Advances in Neural Information Processing Systems*, 38:107150–107180.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Qingxiu Dong, Li Dong, Yao Tang, Tianzhu Ye, Yutao Sun, Zhifang Sui, and Furu Wei. 2025. Reinforcement pre-training. *arXiv preprint arXiv:2506.08007*.

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: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Moritz Hardt and Yu Sun. 2024. Test-time training on nearest neighbors for large language models. In *International Conference on Learning Representations*, volume 2024, pages 54625–54640.

Bingxiang He, Yuxin Zuo, Zeyuan Liu, Shangziqi Zhao, Zixuan Fu, Junlin Yang, Cheng Qian, Kaiyan Zhang, Yuchen Fan, Ganqu Cui, and 1 others. 2026. How far can unsupervised rlvr scale llm training? *arXiv preprint arXiv:2603.08660*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Xiaotong Ji, Rasul Tutunov, Matthieu Zimmer, and Haitham Bou Ammar. 2026. Scalable power sampling: Unlocking efficient, training-free reasoning for llms via distribution sharpening. *arXiv preprint arXiv:2601.21590*.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, and 1 others. 2024. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*, 13(9):9.

Ruotong Liao, Nikolai Röhrich, Xiaohan Wang, Yuhui Zhang, Yasaman Samadzadeh, Volker Tresp, and Serena Yeung-Levy. 2026a. Tool verification for test-time reinforcement learning. *arXiv preprint arXiv:2603.02203*.

Ruotong Liao, Nikolai Röhrich, Xiaohan Wang, Yuhui Zhang, Yasaman Samadzadeh, Volker Tresp, and Serena Yeung-Levy. 2026b. Tool verification for test-time reinforcement learning. *arXiv preprint arXiv:2603.02203*.

Mohammad Mahdi Moradi and Sudhir Mudur. 2026. Disctt: Consensus-guided self-curriculum for efficient test-time adaptation in reasoning. *arXiv preprint arXiv:2603.05357*.

Teng Pan, Yuchen Yan, Zixuan Wang, Ruiqing Zhang, Guiyang Hou, Wenqi Zhang, Weiming Lu, Jun Xiao, and Yongliang Shen. 2026. Coverrl: Breaking the consensus trap in label-free reasoning via generator-verifier co-evolution. *arXiv preprint arXiv:2603.17775*.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, and 1 others. 2024. Qwen2.5 technical report. *arXiv 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*.

Toby Simonds and Akira Yoshiyama. 2025. Ladder: Self-improving llms through recursive problem decomposition. *arXiv preprint arXiv:2503.00735*.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, and 1 others. 2024. Learning to (learn at test time): Rnns with expressive hidden states. *arXiv preprint arXiv:2407.04620*.

Lit Sin Tan, Junzhe Chen, Xiaolong Fu, Lichen Ma, Junshi Huang, Jianzhong Shi, Yan Li, and Lijie Wen. 2026. Meta-ttrl: A metacognitive framework for self-improving test-time reinforcement learning in unified multimodal models. *arXiv preprint arXiv:2603.15724*.

Weiqin Wang, Yile Wang, Kehao Chen, and Hui Huang. 2025. Beyond majority voting: Towards fine-grained and more reliable reward signal for test-time reinforcement learning. *arXiv preprint arXiv:2512.15146*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Chengyi Yang, Zhishang Xiang, Yunbo Tang, Zongpei Teng, Chengsong Huang, Fei Long, Yuhan Liu, and Jinsong Su. 2026. Ttcs: Test-time curriculum synthesis for self-evolving. *arXiv preprint arXiv:2601.22628*.

Qingyang Zhang, Haitao Wu, Changqing Zhang, Peilin Zhao, and Yatao Bian. 2026. Right question is already half the answer: Fully unsupervised llm reasoning incentivization. *Advances in neural information processing systems*, 38:67345–67372.

Tianyuan Zhang, Sai Bi, Yicong Hong, Kai Zhang, Fujun Luan, Songlin Yang, Kalyan Sunkavalli, William T Freeman, and Hao Tan. 2025a. Test-time training done right. *arXiv preprint arXiv:2505.23884*.

Yanzhi Zhang, Zhaoxi Zhang, Haoxiang Guan, Yilin Cheng, Yitong Duan, Chen Wang, Yue Wang, Shuxin Zheng, and Jiyan He. 2025b. No free lunch: Rethinking internal feedback for llm reasoning. *arXiv preprint arXiv:2506.17219*.

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

Yujun Zhou, Zhenwen Liang, Haolin Liu, Wenhao Yu, Kishan Panaganti, Linfeng Song, Dian Yu, Xiangliang Zhang, Haitao Mi, and Dong Yu. 2025. Evolving language models without labels: Majority drives selection, novelty promotes variation. *arXiv preprint arXiv:2509.15194*.

Yuxin Zuo, Kaiyan Zhang, Shang Qu, Li Sheng, Xuekai Zhu, Biqing Huang, and 1 others. 2025. Ttrl: Test-time reinforcement learning. *arXiv preprint arXiv:2504.16084*.

A Problem Category Definitions

To systematically analyze label dynamics during test-time reinforcement learning, we classify each problem according to how its pseudo-label accuracy evolves over training. We define the Initial Label Accuracy (ILA) as the mean label accuracy over the first three checkpoints and the Final Label Accuracy (FLA) as the mean over the last five. Comparing these two quantities partitions all problems into six mutually exclusive categories, summarized in Table 2.

The six categories reflect qualitatively distinct behavioral trajectories. Stable Always Right (SAR) problems are reliably solved from the outset and remain so throughout training. At the opposite extreme, Always Wrong (AW) problems are never mastered. Learned (LN) problems capture genuine knowledge acquisition, where the model initially fails but eventually succeeds. Degraded (DG) problems represent the reverse, where initially correct responses deteriorate as training progresses. The two marginal categories cover problems with intermediate initial accuracy: Marginal Degraded (MDG) exhibits a moderate decline, while Marginal Stable (MS) shows no significant change in either direction.

B Implementation Details

Table 3 summarizes the hyperparameters shared across all models, datasets, and TTRL-Guard variants. Models were trained for 60 epochs on AIME

Category	Condition	Description
Stable Always Right	ILA ≥ 0.7 and FLA ≥ 0.6	Stable convergence with consistently high accuracy
Degraded	ILA ≥ 0.5 and FLA $<$ ILA $- 0.2$	Initially correct but accuracy drops significantly
Learned	ILA < 0.15 and FLA ≥ 0.5	Initially unsolvable but mastered by end of training
Marginal Degraded	$0.15 \leq$ ILA < 0.7 and FLA $<$ ILA $- 0.15$	Mid-range accuracy with notable decline
Marginal Stable	$0.15 \leq$ ILA < 0.7 and FLA $\geq$ ILA $- 0.15$	Mid-range accuracy with no significant change
Always Wrong	ILA < 0.15 and FLA < 0.5	Consistently unsolvable throughout training

Table 2: Problem categorization based on label accuracy dynamics.

Hyperparameter	Value
Training	
Train batch size	8
PPO mini-batch size	1
PPO micro-batch size	2
learning rate	5×10^{-7}
LR warmup ratio	0.03 (cosine)
KL coefficient	0.00
Rollout	
Samples per prompt K	32
Votes per prompt	64
Max prompt length	1,024 tokens
Max response length	3,072 tokens
Training temperature	0.6
Validation samples N	4
FRS	
λ_1	0.5
λ_2	0.3
τ_{FR}	0.3
τ_{MR}	0.6
Min. weight $w_{\min}$	0.1
MPS	
$\beta_{\max}$	0.3
Minority threshold	$\lfloor K/4 \rfloor = 8$
Steady-exit steps T_{steady}	3
RCSU	
History window W	5
θ_{MR}	0.5
Skip probability p_{skip}	0.7
Max skip per step	25% of N

Table 3: Hyperparameters used in all TTRL-Guard experiments.

2024/2025 and 30 epochs on AMC and MATH-500. All reported metrics represent the average of 5 independent runs.

The *extinction window* is not a hyperparameter but an empirical phenomenon: a brief early-training phase in which the correct answer still holds a competitive minority before majority voting locks irreversibly onto the wrong answer. TTRL-Guard does not require its boundaries to be specified; the flip-rate signal FR_i^t serves as a real-time proxy, activating FRS and MPS while the window is open and handing off to RCSU once it closes.

C Parameter Sensitivity

Figure 9 reports the sensitivity of each component to its primary hyperparameter, evaluated uniformly on Qwen2.5-7B-Instruct / MATH-500. Both Pass@1 and the L/D ratio (Learned% / Degraded%) are shown; the shaded band marks the recommended default.

FRS: flip-rate threshold τ_{FR}. Figure 9a sweeps $\tau_{\text{FR}} \in [0.10, 0.50]$. Both metrics peak at the default τ_{FR}=0.30, with a total swing of 2.5 pp in Pass@1. Small values ($\tau_{\text{FR}} \leq 0.15$) over-trigger FRS, suppressing correct-minority gradients alongside noisy ones and degrading both Pass@1 and L/D. Large values ($\tau_{\text{FR}} \geq 0.40$) prevent FRS from firing on mildly unstable problems, allowing wrong-answer lock-in to proceed unchecked and collapsing L/D sharply. The method is robust for $\tau_{\text{FR}} \in [0.20, 0.40]$.

RCSU: history window W. Figure 9b sweeps $W \in \{2, 3, 5, 8, 12, 20\}$. The default W=5 achieves the best Pass@1 (81.6%) and L/D (0.075). Short windows ($W \leq 3$) produce noisy match-rate estimates, causing HighRisk flags to fire on transient spikes and incorrectly skipping learnable problems, which suppresses Learned% and hurts accuracy. Large windows ($W \geq 12$) detect reward lock-in too slowly; wrong-answer gradients accumulate before skipping activates, driving Degraded% up and Pass@1 down. L/D is more sensitive to W than Pass@1, confirming that RCSU primarily guards training dynamics quality.

MPS: maximum minority weight $\beta_{\max}$. Figure 9c sweeps $\beta_{\max} \in [0.05, 0.50]$. At $\beta_{\max}$=0.05, the minority reward is negligible and results approach the plain TTRL baseline (80.5%); L/D remains near its unguarded level, confirming that minority-correct gradients are not being preserved. Both metrics peak at the default $\beta_{\max}$=0.30. Above 0.30, excessive upweighting amplifies incorrect minority answers (any response with $\geq \lfloor K/4 \rfloor$

votes qualifies for $\mathcal{M}_i^t$, not only correct ones), introducing gradient noise that degrades both Pass@1 and L/D.

D Computational Overhead Analysis

A critical concern when introducing per-problem state tracking is the potential increase in training latency. We report the relative change in wall-clock time per training step compared to TTRL (positive values indicate slowdown; negative values indicate speedup) in Table 4. FRS incurs modest overhead from per-problem state monitoring, and MPS adds a small additional loss branch; both introduce only positive overhead regardless of model size. RCSU, by contrast, skips gradient updates for high-risk problems, reducing the number of tokens processed during the backward pass. Its speedup grows with model size, as the backward pass becomes increasingly dominant for larger models. On Llama-3.2-3B-Instruct, the skip savings are limited, so the TTRL-Guard pipeline incurs a small net overhead of +3.8%. On Qwen2.5-7B-Instruct, RCSU's skipping outweighs the overhead of FRS and MPS, yielding a net speedup of 29.3% for the full pipeline.

Method	Llama-3.2-3B	Qwen2.5-7B
TTRL	—	—
+ FRS	+5.3%	+5.0%
+ MPS	+13.4%	+13.2%
+ RCSU	−6.0%	−34.9%
TTRL-Guard	+3.8%	−29.3%

Table 4: Relative change in wall-clock time per step vs. TTRL on MATH-500. Negative values indicate speedup.

E The Capability-Difficulty Mismatch

To demonstrate the consistency of asymmetric degradation and the effectiveness of TTRL-Guard across varying task difficulties, Table 5 reports the complete per-problem fate distributions for Llama-3.2-3B-Instruct and Qwen2.5-7B-Instruct across AIME 2024, AIME 2025, AMC, and MATH-500.

The results reveal a consistent structural pattern: degradation severity is strongly correlated with the capability-difficulty mismatch between the model and the dataset. When a dataset is entirely beyond a model's capability (e.g., Llama-3.2-3B-Instruct on AIME 2025, where almost all problems fall into Always Wrong), majority voting rarely produces a false consensus, so TotalDeg remains near zero and TTRL-Guard has little room to intervene. At the other extreme, when a strong model faces an easy dataset (e.g., Qwen2.5-7B-Instruct on AMC), most problems are already stably correct, leaving only a small fraction susceptible to degradation. The most severe degradation occurs in the moderate-mismatch regime, exemplified by Llama-3.2-3B-Instruct on MATH-500, where TotalDeg reaches 60.2% under TTRL. In this regime, the model is capable enough to generate plausible but incorrect answers that attract a majority, yet not capable enough to recover once the wrong pseudo-label locks in. TTRL-Guard reduces TotalDeg by 32.2 percentage points in this setting, from 60.2% to 28.0%, while simultaneously increasing Stable AR from 2.8% to 32.0%, confirming that the interventions protect knowledge rather than merely suppressing updates.

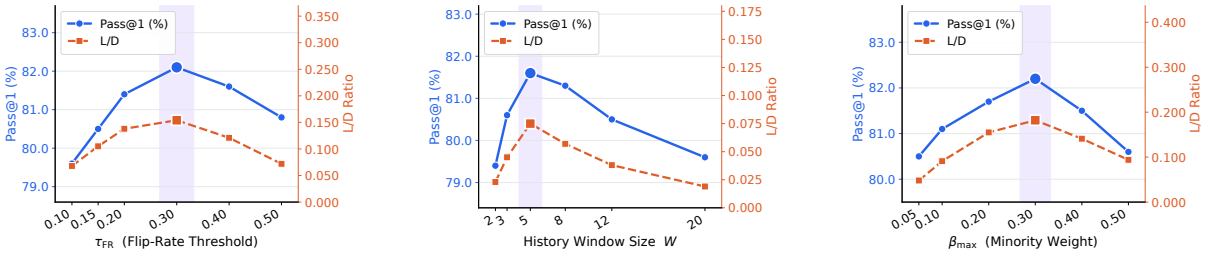
F Microscopic Dynamics of the Consensus Trap

To further illustrate the mechanics of Asymmetric Degradation, we isolate the trajectory of problems that fall into the Degraded category (initially correct, but corrupted by the end of training).

When tracing the step-by-step voting distribution of a typical Degraded problem, we observe a distinct **Scissor Effect** that drives the Extinction Window, as visualized in Figure 10. The dynamics unfold in three phases:

1. **Phase 1 (Competition).** In the early training steps, the correct answer and a highly plausible incorrect answer co-exist, competing closely for the majority vote. As shown in Figure 10a, aggregated over all three models (Llama-3.2-3B-Instruct, Qwen2.5-7B-Instruct, and Qwen3-4B) on AIME 2024, AMC, and MATH-500, roughly 30% of Degraded problems still have a wrong-answer Match Rate below 0.5 in the early phase, meaning the correct answer retains a chance to win the majority vote.

2. **Phase 2 (Suppression).** Due to the winner-takes-all nature of the standard GRPO loss, any step in which the incorrect answer wins the majority vote heavily penalises the correct answer. The incorrect answer systematically suppresses the correct answer's log-probabilities, producing a visible crossover. Figure 10b quantifies this collapse: the correct-answer survival space $1 - \overline{\mathrm{MR}}$, av-



(a) Sensitivity to flip-rate threshold τ_{FR}.

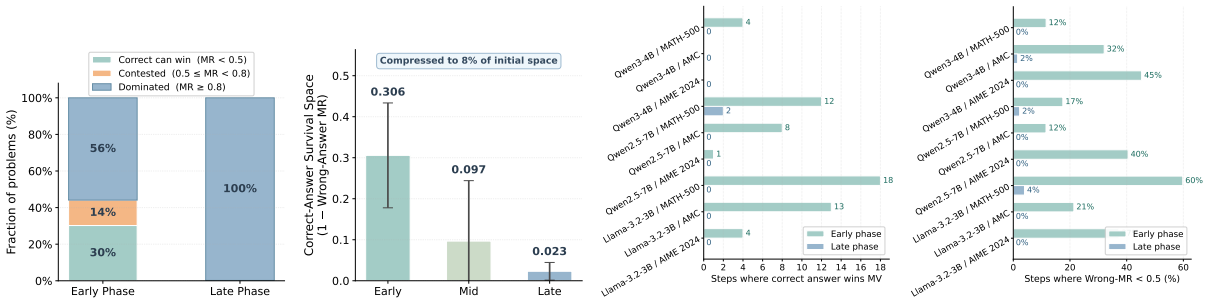
(b) Sensitivity to history window size W.

(c) Sensitivity to minority weight $\beta_{\max}$.

Figure 9: Hyperparameter sensitivity for each TTRL-Guard component (Qwen2.5-7B-Instruct / MATH-500). Shaded band marks the default value and larger marker highlights the optimal point.

Dataset	Method	Degraded	Marg. Deg.	Total Deg.	Learned	Stable AR	Always Wrong
			Llama-3.2-3B-Instruct				
MATH-500	TTRL	53.8%	6.4%	**60.2%**	0.2%	2.8%	36.4%
	TTRL-Guard	24.0%	4.0%	**28.0%**	0.8%	32.0%	35.2%
AMC	TTRL	12.0%	3.6%	**15.6%**	1.2%	24.1%	55.4%
	TTRL-Guard	5.0%	3.0%	**8.0%**	2.4%	28.3%	55.4%
AIME 2024	TTRL	3.3%	3.3%	**6.6%**	3.3%	13.3%	76.7%
	TTRL-Guard	0.0%	3.3%	**3.3%**	6.7%	16.7%	73.3%
			Qwen2.5-7B-Instruct				
MATH-500	TTRL	13.8%	2.0%	**15.8%**	0.6%	63.0%	20.2%
	TTRL-Guard	3.8%	1.6%	**5.4%**	1.0%	68.0%	22.0%
AMC	TTRL	1.2%	1.2%	**2.4%**	0.0%	48.2%	47.0%
	TTRL-Guard	0.0%	1.2%	**1.2%**	1.2%	50.6%	44.6%
AIME 2025	TTRL	3.3%	0.0%	**3.3%**	0.0%	13.3%	76.7%
	TTRL-Guard	0.0%	3.3%	**3.3%**	3.3%	20.0%	73.3%

Table 5: Per-problem fate distributions for Llama-3.2-3B-Instruct and Qwen2.5-7B-Instruct across four datasets. TotalDeg is the sum of Degraded and Marginal Degraded.



(a) Wrong-answer MR distribution, early vs. late training.

(b) Correct-answer survival space across training phases.

(c) Steps where the correct answer wins the majority vote.

(d) Extinction window width per model and dataset.

Figure 10: The Scissor Effect underlying the Correct-Answer Extinction Window. (a) Even in early training, most problems are already dominated by the wrong answer; by late training this fraction approaches 100%. (b) The correct-answer survival space collapses monotonically across phases. (c, d) The window of correct-answer competitiveness is narrow and confined to the early phase, motivating MPS to capture these signals before vanishing.

eraged across all model and dataset combinations, shrinks monotonically from the early to the late phase, compressing to a small fraction of its initial value.

3. **Phase 3 (Lock-in).** Once the Match Rate of the incorrect answer crosses a critical threshold, the flip rate collapses and the correct answer's sampling probability drops near zero, finalising irreversible degradation. Figures 10c and 10d confirm this pattern across all nine model-dataset pairs: the number of steps in which the correct answer wins the majority vote, and the fraction of steps in

which the wrong-answer MR remains below 0.5, both drop sharply from the early phase to the late phase. The extinction window is therefore narrow, transient, and consistent across architectures and difficulty levels.

TTRL provides no mechanism to detect or counteract this crossover. TTRL-Guard addresses this gap through MPS, which maintains a non-zero gradient for the correct answer as long as it retains any minority presence in the rollout batch. This delays the crossover and extends the window in which the model can self-correct before degradation becomes irreversible.

G Artifact Citations and Licensing

We use three language models from Hugging Face: Llama-3.2-3B-Instruct (Grattafiori et al., 2024) (Meta, licensed under Llama 2 Community License[1]), Qwen2.5-7B-Instruct (Qwen et al., 2024) (Alibaba, Apache 2.0 license[2]), and Qwen3-4B (Yang et al., 2025) (Alibaba, Apache 2.0 license[3]). All models are used solely for research purposes in accordance with their respective licenses. The four mathematical reasoning benchmarks (AIME 2024/2025, AMC, MATH-500) are standard academic evaluation sets available in the public domain for research use. We do not modify, redistribute, or deploy these artifacts outside research contexts.

H Artifact Use Consistency

All artifacts are used strictly for their intended research purposes. Llama-3.2-3B-Instruct and Qwen models are intended for research and development by their creators, and our use is limited to academic experimentation on standard mathematical benchmarks. The AIME, AMC, and MATH-500 datasets are designed for academic evaluation and are used accordingly. The methods and findings presented in this paper are intended for research advancement and academic publication, not for production deployment. Any future application of TTRL-Guard to real-world systems would require additional validation, safety testing, and compliance with applicable licensing and regulatory requirements.

[1] https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct/tree/main

[2] https://huggingface.co/Qwen/Qwen2.5-7B-Instruct/blob/main/LICENSE

[3] https://huggingface.co/Qwen/Qwen3-4B/blob/main/LICENSE

I Computational Infrastructure

All experiments were conducted on standard GPU clusters. Models were trained for 60 epochs on AIME 2024/2025 and 30 epochs on AMC and MATH-500. During inference, we sampled 64 responses per problem for majority voting and 32 for policy updates. The total training cost across 3 models and 4 benchmarks was approximately 40 GPU-hours on 8 A100-80G GPUs.

J AI Assistant Disclosure

This paper was developed with limited AI assistance: Claude (Anthropic) for implementation and debugging of TTRL-Guard algorithm, ChatGPT (OpenAI) for English language refinement and manuscript polishing, and Gemini (Google) for background research on related work. All core algorithmic decisions, experimental design, hyperparameter tuning, and result interpretation were conducted independently by the researchers. Full responsibility for accuracy and validity of all claims rests with the authors.