REFLECTION-CONDITIONED TEACHER–STUDENT TEST-TIME REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Test-time reinforcement learning aims to adapt large language models on unlabeled queries using only self-generated reasoning trajectories. This setting is attractive when ground-truth labels, executable verifiers, or trusted reward models are unavailable, but it also makes the learning signal fragile. Existing self-consistency-based objectives often use the same query-only rollouts both to define pseudo-rewards and to receive updates, which can turn unreliable agreement into overconfident supervision. Moreover, standard policy-gradient objectives apply a trajectory-level advantage to all tokens, despite the fact that only a subset of tokens is likely to determine the final answer. We propose **RCTS**, a reflection-conditioned teacher–student framework for label-free test-time reasoning adaptation. RCTS uses a single policy snapshot in two contextual roles: query-only student rollouts reveal the model's initial behavior, while reflection-conditioned teacher rollouts provide a structured reference. From the teacher rollouts, RCTS constructs a teacher-weighted answer distribution that assigns soft consistency scores to student trajectories. It further introduces Cross-Group Token Divergence to select student tokens that most differ from the reflection-conditioned teacher distribution. The final update applies mask-selective GRPO only to selected query-only student tokens, regularized by a KL anchor to the base policy. Experiments across mathematical reasoning and knowledge-intensive QA show that RCTS consistently improves the frozen models and is competitive with representative label-free test-time adaptation baselines.

1 INTRODUCTION

Reinforcement learning with verifiable rewards has become an effective paradigm for improving the reasoning ability of large language models Guo et al. (2025); Shao et al. (2024); Yu et al. (2025). However, many deployment-time reasoning problems do not come with ground-truth labels, executable checkers, or trusted reward models Yuan et al. (2023); Oh et al. (2025); Hu et al. (2025). In these settings, the model is usually used as a fixed inference engine, even though its own sampled reasoning attempts may contain useful information for adapting to the current query.

Test-time reinforcement learning (TTRL) provides a natural way to exploit such information. Given an unlabeled query, the model samples multiple trajectories, constructs pseudo-rewards from the sampled answers, and updates itself during inference Zuo et al. (2025); Liu et al. (2025); Du et al. (2026); Wei et al. (2025). This label-free setting is attractive because it does not require additional annotation or task-specific verifiers. Yet it also makes the supervision signal inherently fragile Khattar et al. (2026). A common strategy is to use self-consistency: trajectories whose answers match the dominant answer cluster receive higher rewards Wu et al. (2025); Chen et al. (2024); Majumdar et al. (2026). However, this signal is self-referential. The same query-only rollouts that require correction are also used to define the pseudo-label. As a result, weak majorities and strong consensus may be treated similarly, and a large but unreliable answer cluster can induce an overconfident update. Without a verifier, the objective is not to prove which answer is correct, but to avoid turning unqualified self-agreement into an equally strong learning signal.

A second challenge is token-level credit assignment. Standard policy-gradient objectives assign the same trajectory-level advantage to all tokens in a reasoning trace, which is often too coarse for long-form reasoning. Many tokens are stylistic, connective, or nearly determined by the prefix, while only a subset reflects meaningful reasoning choices. Uniform updates can therefore amplify boilerplate

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and dilute the signal on reasoning-relevant regions. Recent studies suggest that a small subset of high-entropy tokens acts as *forking points* that steer reasoning toward different paths, motivating entropy-based updates at locally uncertain positions Wang et al. (2025); Yang et al. (2026b); Wu et al. (2025). While useful for identifying *where* reasoning branches, local uncertainty does not necessarily reveal *what* problem-specific content deserves credit. High-entropy positions may reflect generic hesitation, verification, or transitions, whereas task-relevant deviations may occur in lower-entropy variables, relations, and intermediate operations. A desirable label-free TTRL method should therefore determine both which sampled answers deserve support and which problem-grounded tokens should receive the update.

These challenges suggest that effective label-free test-time adaptation requires self-supervision that is internal to the model but less self-referential than rewarding agreement among the same query-only rollouts. Our key observation is that behavior under reflection provides such a reference. When prompted to critique and reconsider its sampled attempts, the same model may alter not only its final answer, but also its use of problem-specific concepts, relations, and intermediate operations. Comparing query-only and reflection-conditioned behavior can therefore reveal cross-context deviations not captured by query-only self-consistency or local token uncertainty alone. This reflection-conditioned behavior is not an external teacher or verifier, but serves as a structured reference for deciding how much to trust an answer and where to apply the update.

We propose **RCTS** (**R**eflection-**C**onditioned **T**eacher–**S**tudent reinforcement learning), a self-supervised test-time RL framework without ground-truth labels, external verifiers, auxiliary reward models, or separate teacher networks. RCTS uses a single frozen policy snapshot in two contextual roles. Query-only rollouts act as *student* trajectories, while reflection-conditioned rollouts act as a *teacher* reference that critiques and reconsiders the model's previous attempts. Thus, the teacher–student distinction is contextual rather than architectural. This contextual view enables two complementary forms of pseudo-supervision. At the answer level, reflection-conditioned teacher rollouts provide soft support over candidate answers, so student trajectories are rewarded by teacher support rather than query-only majority voting. At the token level, RCTS contrasts query-only and reflection-conditioned token usage to identify student-specific cross-context deviations rather than relying solely on local uncertainty. It updates only selected student tokens instead of the full reasoning trace. The teacher therefore provides answer-level support and a token-level contrastive reference, while optimization remains a label-free GRPO-style update on query-only student trajectories without directly imitating teacher outputs. Our contributions are summarized as follows:

- We identify two coupled limitations of label-free TTRL: self-referential answer supervision from query-only rollouts and coarse trajectory-level credit assignment over reasoning traces.
- We propose **RCTS**, a reflection-conditioned teacher–student framework in which a single policy snapshot plays two contextual roles, requiring no ground-truth labels, external verifiers, auxiliary reward models, or separate teacher networks at adaptation time.
- We introduce two complementary adaptation signals: a teacher-weighted answer support signal for confidence-aware trajectory supervision, and **Cross-Group Token Divergence** (CGTD), which contrasts query-only and reflection-conditioned token usage to prioritize student-specific deviations beyond local uncertainty-based token selection.

Experiments on mathematical reasoning and knowledge-intensive QA benchmarks show that RCTS improves frozen models and achieves competitive performance with representative label-free test-time adaptation baselines across multiple models and metrics.

2 Related Work

Label-Free Adaptation and Test-Time Reinforcement Learning. Label-free adaptation derives proxy supervision from model predictions, including entropy, perturbation consistency, and sample agreement Hosseini et al. (2025); Agarwal et al. (2025); Zhao et al. (2025); Singh et al. (2025); Wang & Heckel (2026); Yang et al. (2026a). These signals avoid labels and task-specific verifiers but remain vulnerable to uncertainty and miscalibration. For reasoning tasks, TTRL derives pseudo-rewards from sampled trajectories. TTRL uses majority-vote self-consistency Zuo et al. (2025), while subsequent methods incorporate response entropy, distribution-aware rewards, multimodal policy optimization, or embedding-based novelty Liu et al. (2025); Du et al. (2026); Wei et al. (2025); Zhou et al. (2025).

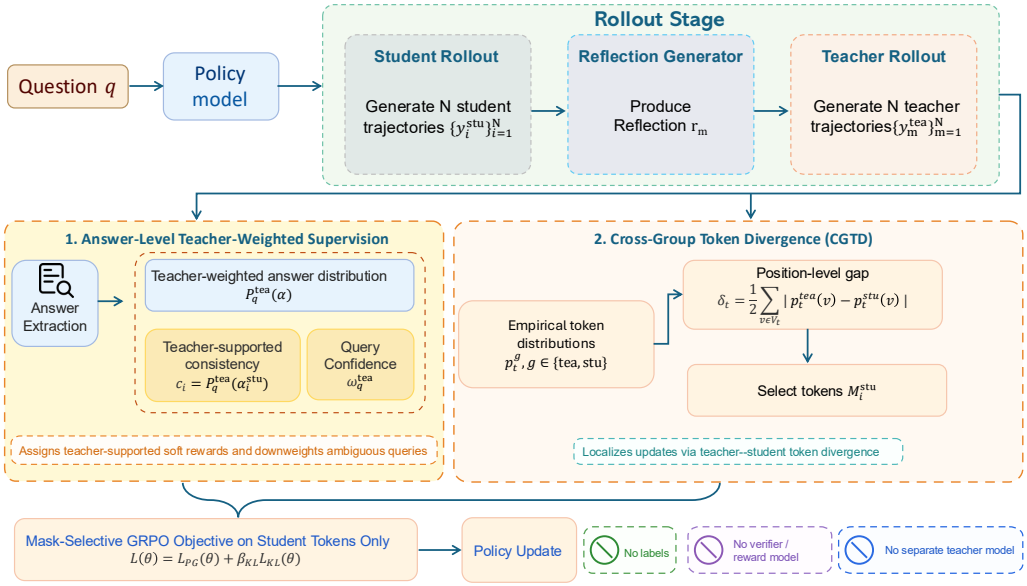

Figure 1: Overview of RCTS. The frozen rollout policy samples query-only student trajectories, generates reflections, and then produces reflection-conditioned teacher trajectories. The teacher group provides teacher-weighted answer support and query-level confidence, while CGTD selects student-specific deviation tokens by contrasting student and teacher token distributions. RCTS then performs mask-selective GRPO only on selected query-only student tokens.

Other approaches introduce external teachers, judges, or synthetic curricula Jayalath et al. (2025); Yang et al. (2026a). SCRL and SPINE further address noisy consensus and coarse credit assignment through selective pseudo-labeling or token-selective optimization Yan et al. (2026); Wu et al. (2025). RCTS differs by using the same policy snapshot in two contextual roles: query-only student rollouts receive updates, while reflection-conditioned teacher rollouts provide both teacher-weighted answer support and a token-level contrastive reference.

On-Policy Distillation and Feedback-Conditioned Self-Improvement. On-policy distillation samples student trajectories and queries a teacher on visited states, reducing distribution mismatch relative to distillation on fixed teacher- or human-generated sequences Agarwal et al. (2024); Fu et al. (2026). Recent reasoning-oriented variants use stronger teachers, reference solutions, verifier feedback, demonstrations, auxiliary context, or privileged information Xiao et al. (2026); Zhao et al. (2026); Hübotter et al. (2026); Shenfeld et al. (2026); Ye et al. (2026); Sang et al. (2026); Penaloza et al. (2026). A related line learns from critique, reflection, revision, or failed attempts Zelikman et al. (2022); Madaan et al. (2023); Shinn et al. (2023); Shi et al. (2026); Zhang et al. (2025). These signals have been used to distill feedback-conditioned teacher policies, construct denser teacher–student discrepancies, or generate targeted test-time curricula Hübotter et al. (2026); Song et al. (2026); Zhang et al. (2026); Yang et al. (2026b); He et al. (2026); Yang et al. (2026a). However, they often rely on privileged information, verifiable feedback, or the quality of synthesized questions. RCTS instead uses reflection-conditioned behavior from the same frozen policy as a label-free contextual reference. It neither directly imitates teacher trajectories nor matches their full token distribution, but uses teacher-weighted answer supervision and CGTD for mask-selective GRPO.

3 METHOD

We present RCTS, a self-supervised test-time reinforcement learning framework for label-free reasoning adaptation. As illustrated in Fig. 1, RCTS uses a single policy snapshot in two contextual roles: query-only student rollouts provide the trajectories to be updated, while reflection-conditioned teacher rollouts provide answer-level support and a token-level contrastive reference. The resulting

update combines teacher-weighted answer supervision with CGTD-based token masking, and applies a GRPO-style objective only to selected query-only student tokens.

At the beginning of each test-time update, we set $\theta_{\text{old}} \leftarrow \theta$ and use the frozen rollout policy $\pi_{\theta_{\text{old}}}$ to generate all rollout-dependent signals. These signals are detached from the optimization graph. The current policy π_θ is then updated with the fixed teacher-supported rewards and CGTD masks, while π_{base} denotes the frozen pre-adaptation policy used only for KL regularization. This follows the standard PPO/GRPO-style protocol in which rollouts are collected before the optimization step.

3.1 Reflection-Conditioned Teacher Rollouts

Let $\mathcal{D}$ be the unlabeled test distribution. For each query $q \sim \mathcal{D}$, RCTS first samples a query-only student group $\mathcal{Y}_q^{\text{stu}} = \{y_i^{\text{stu}}\}_{i=1}^N$ from the frozen rollout policy:

$$y_i^{\text{stu}} \sim \pi_{\theta_{\text{old}}}(\cdot \mid q), \tag{1}$$

and extracts the corresponding answers $\alpha_i^{\text{stu}} = \text{Ans}(y_i^{\text{stu}})$. RCTS then queries the same frozen policy under a reflection prompt to obtain N diagnostic reflections:

$$\mathcal{R}_q = \{r_m\}_{m=1}^N = \text{Reflect}_{\theta_{\text{old}}}^{(N)}(q, \mathcal{Y}_q^{\text{stu}}). \tag{2}$$

Here, $\text{Reflect}_{\theta_{\text{old}}}^{(N)}$ denotes a batched reflection call that takes the query and the sampled student group as input and returns N reflection strings. These reflections critique the student rollouts and provide reasoning guidance rather than final answers (see appendix B). They are used only as prompt-side feedback and are not optimized by the RCTS objective.

Conditioned on these reflections, RCTS samples a reflection-conditioned teacher group $\mathcal{Y}_q^{\text{tea}} = \{y_m^{\text{tea}}\}_{m=1}^N$:

$$y_m^{\text{tea}} \sim \pi_{\theta_{\text{old}}}(\cdot \mid q, r_m), \tag{3}$$

with extracted answers $\alpha_m^{\text{tea}} = \text{Ans}(y_m^{\text{tea}})$. The student and teacher groups are generated by the same policy snapshot and differ only in their prompt contexts: student rollouts are sampled from the plain query, whereas teacher rollouts are sampled from reflection-conditioned queries. Each teacher trajectory is scored by its length-normalized generation log-probability:

$$s_m = \frac{1}{T_m^{\text{tea}}} \sum_{t=1}^{T_m^{\text{tea}}} \log \pi_{\theta_{\text{old}}}\big(y_{m,t}^{\text{tea}} \mid q, r_m, y_{m,<t}^{\text{tea}}\big). \tag{4}$$

This score provides a lightweight confidence estimate from the frozen decoder while reducing the short-answer bias of raw sequence log-probability. We convert these scores into normalized teacher weights:

$$\lambda_m = \frac{\exp(s_m)}{\sum_{j=1}^N \exp(s_j)}. \tag{5}$$

The resulting weights are used only to aggregate teacher answers into a soft pseudo-label distribution; teacher trajectories themselves are not directly optimized.

3.2 Reflection-Conditioned Weighted Pseudo-Labeling

RCTS uses the reflection-conditioned teacher group to construct a soft answer-level supervision signal for the query-only student group. The teacher group is not assumed to be an oracle. Instead, it provides a weighted answer distribution that reflects how strongly each candidate answer is supported under reflection-conditioned contexts. Given the teacher weights $\{\lambda_m\}_{m=1}^N$, we define the teacher-weighted answer distribution as

$$P_q^{\text{tea}}(\alpha) = \sum_{m=1}^N \lambda_m \mathbf{1}[\alpha_m^{\text{tea}} = \alpha], \tag{6}$$

where $\alpha_m^{\text{tea}} = \text{Ans}(y_m^{\text{tea}})$. This distribution serves as a soft pseudo-label over answers. Rather than assigning a binary label based on a single top answer, RCTS assigns each student trajectory

a teacher-supported consistency score according to the teacher-weighted support for its extracted answer. For each student trajectory y_i^{stu}, we define its teacher-supported consistency as

$$c_i = P_q^{\mathrm{tea}}(\alpha_i^{\mathrm{stu}}), \tag{7}$$

where $\alpha_i^{\mathrm{stu}} = \mathrm{Ans}(y_i^{\mathrm{stu}})$. Thus, c_i measures how strongly the teacher distribution supports the answer produced by student trajectory i, rather than agreement within the student group itself.

To estimate the reliability of the teacher distribution, we compute a query-level teacher confidence weight. Let $P_q^{\mathrm{tea},(1)} \geq P_q^{\mathrm{tea},(2)} \geq \cdots$ be the sorted nonzero values of $P_q^{\mathrm{tea}}(\alpha)$ over distinct teacher answers, with $P_q^{\mathrm{tea},(2)} = 0$ if only one distinct answer appears. We define

$$H(P_q^{\mathrm{tea}}) = -\sum_{\alpha} P_q^{\mathrm{tea}}(\alpha) \log P_q^{\mathrm{tea}}(\alpha), \tag{8}$$

$$\omega_q^{\mathrm{tea}} = \frac{1}{2}\left[P_q^{\mathrm{tea},(1)} - P_q^{\mathrm{tea},(2)} + 1 - \frac{H(P_q^{\mathrm{tea}})}{\log N}\right]. \tag{9}$$

The first term measures the margin between the two most supported teacher answers, while the second term measures the concentration of the teacher answer distribution. The confidence ω_q^{tea} is therefore high when the teacher group provides a concentrated and well-separated answer distribution, and low when teacher answers remain diffuse. We refer to the pair $(c_i, \omega_q^{\mathrm{tea}})$ as the teacher-weighted student supervision signal.

3.3 Cross-Group Token Divergence

The teacher-supported consistency score in Eq. 7 provides answer-level supervision for each student trajectory, but it does not specify which tokens within the trajectory should receive updates. RCTS therefore introduces CGTD to select realized student tokens whose empirical usage differs most from the reflection-conditioned teacher group. The teacher group serves as a contrastive reference for measuring token-level deviation, rather than as a token-level imitation target. The policy update is still applied only to query-only student tokens through a GRPO-style objective.

For each response position t, let $\mathcal{V}_t$ be the union of observed student and teacher tokens at that position, with tokens after EOS ignored. For $g \in \{\mathrm{stu}, \mathrm{tea}\}$, let $n_t^g(v)$ be the number of trajectories in group g whose observed token at position t equals v. We estimate the Laplace-smoothed empirical token distribution as

$$p_t^g(v) = \frac{n_t^g(v) + \eta_{\mathrm{sm}}}{\sum_{u \in \mathcal{V}_t} n_t^g(u) + \eta_{\mathrm{sm}}|\mathcal{V}_t|}, \qquad g \in \{\mathrm{stu}, \mathrm{tea}\},\ v \in \mathcal{V}_t, \tag{10}$$

where $\eta_{\mathrm{sm}} > 0$ is the smoothing constant. This position-wise statistic does not require matching semantic reasoning steps across trajectories; it only uses absolute response positions as a lightweight coordinate system for identifying where reflection conditioning induces systematic distributional shifts.

We measure the position-level student–teacher gap by total variation:

$$\delta_t = \frac{1}{2}\sum_{v \in \mathcal{V}_t} \left|p_t^{\mathrm{tea}}(v) - p_t^{\mathrm{stu}}(v)\right|. \tag{11}$$

A large δ_t indicates that the query-only student group and the reflection-conditioned teacher group use substantially different tokens at position t. For each realized student token, CGTD combines this position-level gap with the token's student-specificity:

$$S_{i,t}^{\mathrm{stu}} = \delta_t \left[\log p_t^{\mathrm{stu}}(y_{i,t}^{\mathrm{stu}}) - \log p_t^{\mathrm{tea}}(y_{i,t}^{\mathrm{stu}})\right]_+, \tag{12}$$

where $[x]_+ = \max(x, 0)$. This score is large when a realized student token appears at a position with strong student–teacher divergence and is more characteristic of the query-only student group than of the teacher group. Thus, CGTD prioritizes student-specific cross-context deviation tokens; whether these selected tokens are reinforced or suppressed is determined later by the trajectory advantage in Eq. 14.

For each student trajectory, RCTS selects only high-scoring positive CGTD positions. Let

$$\mathcal{P}_i = \{t \in \{1, \dots, T_i^{\text{stu}}\} : S_{i,t}^{\text{stu}} > 0\}. \tag{13}$$

We set $K_i = \lceil k|\mathcal{P}_i| \rceil$ and define $\mathcal{M}_i^{\text{stu}}$ as the K_i positions in $\mathcal{P}_i$ with the largest CGTD scores. If $\mathcal{P}_i = \emptyset$, then $\mathcal{M}_i^{\text{stu}} = \emptyset$. We use $k = 0.2$ following Wang et al. (2025).

3.4 Mask-Selective GRPO Objective

Given the teacher-supported consistency scores and CGTD masks, RCTS updates only the query-only student trajectories. For each query q, we first normalize the student consistency scores $\{c_i\}_{i=1}^N$ within the student group and then apply the query-level teacher confidence as an outer reliability weight:

$$A_i^{\text{stu}} = \omega_q^{\text{tea}} \frac{c_i - \mu_q^c}{\sigma_q^c + \epsilon_{\text{adv}}}, \qquad \mu_q^c = \frac{1}{N} \sum_{j=1}^{N} c_j, \tag{14}$$

where σ_q^c is the standard deviation of $\{c_j\}_{j=1}^N$ and ϵ_{adv} is a small constant for numerical stability. Applying ω_q^{tea} after group normalization prevents the query-level confidence from being canceled by GRPO standardization, allowing diffuse teacher distributions to attenuate the test-time update. For each selected student token, we define the old-policy likelihood ratio under the query-only context:

$$\rho_{i,t}^{\text{stu}}(\theta) = \frac{\pi_\theta(y_{i,t}^{\text{stu}} \mid q, y_{i,<t}^{\text{stu}})}{\pi_{\theta_{\text{old}}}(y_{i,t}^{\text{stu}} \mid q, y_{i,<t}^{\text{stu}})}. \tag{15}$$

We first define the token-level clipped surrogate as

$$\begin{aligned} \ell_{i,t}^{\text{PG}}(\theta) = \min\Big\{ & \rho_{i,t}^{\text{stu}}(\theta) A_i^{\text{stu}}, \\ & \text{clip}\big(\rho_{i,t}^{\text{stu}}(\theta), 1 - \varepsilon_{\text{low}}, 1 + \varepsilon_{\text{up}}\big) A_i^{\text{stu}} \Big\}. \end{aligned} \tag{16}$$

The selective policy-gradient loss is

$$\mathcal{L}_{\text{PG}}(\theta) = -\mathbb{E}_{q \sim \mathcal{D}} \left[\frac{1}{N} \sum_{i=1}^{N} \frac{\sum_{t \in \mathcal{M}_i^{\text{stu}}} \ell_{i,t}^{\text{PG}}(\theta)}{|\mathcal{M}_i^{\text{stu}}| + \epsilon_{\text{mask}}} \right]. \tag{17}$$

where ϵ_{mask} is a small constant used only to avoid division by zero. The mask-size normalization first averages over selected tokens within each trajectory and then averages over trajectories, preventing trajectories with more selected tokens from dominating the update. If $\mathcal{M}_i^{\text{stu}}$ is empty, the trajectory contributes no policy-gradient term.

To constrain test-time updates, RCTS applies a non-negative sampled KL penalty to the selected student positions. Let $s_{i,t} = (q, y_{i,<t}^{\text{stu}})$ denote the token-level generation state. We define

$$\begin{aligned} \widehat{d}_{i,t}^{\text{KL}}(\theta) &= r_{i,t}^{\text{base}} - \log r_{i,t}^{\text{base}} - 1, \\ r_{i,t}^{\text{base}} &= \frac{\pi_{\text{base}}(y_{i,t}^{\text{stu}} \mid s_{i,t})}{\pi_\theta(y_{i,t}^{\text{stu}} \mid s_{i,t})}. \end{aligned} \tag{18}$$

The masked KL regularization loss is

$$\mathcal{L}_{\text{KL}}(\theta) = \mathbb{E}_{q \sim \mathcal{D}} \left[\frac{1}{N} \sum_{i=1}^{N} \frac{\sum_{t \in \mathcal{M}_i^{\text{stu}}} \widehat{d}_{i,t}^{\text{KL}}(\theta)}{|\mathcal{M}_i^{\text{stu}}| + \epsilon_{\text{mask}}} \right]. \tag{19}$$

The final objective is

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{PG}}(\theta) + \beta_{\text{KL}} \mathcal{L}_{\text{KL}}(\theta), \tag{20}$$

Name	MedQA			MMLU-Hard			AMC23			AIME24		
	P@1	P@8	Maj@8	P@1	P@8	Maj@8	P@1	P@8	Maj@8	P@1	P@8	Maj@8
Base Model	**Qwen3-4B-Instruct**											
No adaptation	72.40	86.40	74.00	24.66	45.21	21.92	65.00	80.00	80.00	23.33	33.33	30.00
w/ TTRL	75.80	87.60	76.40	28.76	52.05	27.40	70.00	85.00	85.00	33.33	50.00	46.67
w/ SPINE	76.60	87.80	76.40	30.14	49.31	24.66	80.00	90.00	90.00	33.33	50.00	46.67
w/ DARE	75.00	87.80	76.60	31.50	47.94	27.40	80.00	90.00	90.00	36.67	50.00	46.67
w/ **RCTS**	77.20	87.80	77.20	32.88	53.42	30.14	82.50	**95.00**	92.50	**40.00**	53.33	46.67
w/ **RCTS**†	**77.60**	**88.20**	**79.00**	**35.62**	**54.79**	**34.25**	**85.00**	**95.00**	**95.00**	**40.00**	**60.00**	**56.67**
Base Model	**Llama-3.2-3B-Instruct**											
No adaptation	54.60	88.40	61.40	15.06	38.35	10.95	22.50	57.50	35.00	6.67	23.33	13.33
w/ TTRL	61.60	90.20	60.60	21.92	58.90	13.69	32.50	67.50	40.00	13.33	30.00	**20.00**
w/ SPINE	62.60	90.60	65.20	23.28	61.64	15.07	32.50	67.50	45.00	13.33	**33.33**	16.67
w/ DARE	61.40	90.60	66.20	24.66	**63.01**	13.69	32.50	**70.00**	40.00	**16.67**	30.00	16.67
w/ **RCTS**	**65.80**	**91.20**	**69.00**	26.02	**63.01**	17.80	**37.50**	**70.00**	**47.50**	**16.67**	30.00	**20.00**
w/ **RCTS**†	**65.80**	**91.20**	66.40	**27.40**	**63.01**	**20.54**	**37.50**	**70.00**	**47.50**	**16.67**	**33.33**	**20.00**

Table 1: Comparison with label-free test-time adaptation methods. Within each base-model block, best and second-best results are marked in **bold** and underline. For **RCTS**, † indicates reflection-conditioned teacher-side inference.

where β_{KL} controls the strength of the KL regularization.

Overall, the teacher-supported consistency score determines the relative preference among student trajectories, the teacher confidence weight controls the query-level update magnitude, and CGTD determines where the update is applied. Thus, RCTS performs clipped, mask-selective GRPO on query-only student trajectories without using teacher tokens as supervised targets. Algorithm 1 (See Appendix A) summarizes the complete RCTS test-time update.

4 Experiments

4.1 Setup

Models. We evaluate RCTS on two instruction-tuned language models from different families: **Qwen3-4B-Instruct** Yang et al. (2025), using the Qwen3-4B-Instruct-2507 checkpoint, and **Llama-3.2-3B-Instruct** Grattafiori et al. (2024). Both are publicly released 3–4B instruction-tuned checkpoints used without additional fine-tuning.

Benchmarks. We evaluate across four benchmarks spanning mathematical reasoning and knowledge-intensive QA: **AIME24** Hendrycks et al. (2021), a competition-level mathematics benchmark; **AMC23** Hendrycks et al. (2021), a high-school competition mathematics benchmark; **MedQA** Jin et al. (2021), a medical licensing examination QA dataset; and **MMLU-Hard**, the hard medical MMLU Hendrycks et al. (2020) subset curated by MedAgentsBench Tang et al. (2025).

Baselines and Metrics. We compare against the frozen model, TTRL Zuo et al. (2025) with majority-vote pseudo-rewards, SPINE Wu et al. (2025) with entropy-based token selection, and DARE Du et al. (2026) with distribution-aware rewards. We report **P@1**, first-sample accuracy; **P@8**, the fraction of problems solved by at least one of eight samples; and **Maj@8**, majority-vote accuracy over eight samples.

Implementation details. Unless otherwise specified, we sample $N = 8$ trajectories per query for both student and teacher rollouts, use a token-selection ratio of $k = 20\%$, set $\beta_{\mathrm{KL}} = 0.001$, use a learning rate of 1×10^{-6}, and run 200 adaptation steps. After adaptation, we report two inference modes: **RCTS** uses the query-only student-side context, while **RCTS**† uses the same adapted policy under the reflection-conditioned teacher-side context. All hyperparameters are fixed across benchmarks and models unless otherwise noted.

We extract final answers from the last boxed span. Math answers are graded with MathD-style normalization and symbolic equivalence checks via `sympy`, while multiple-choice tasks require a single boxed option letter (see Appendix D). Majority-based pseudo-rewards use exact agreement

Name	MMLU-Hard			AIME24		
	P@1	P@8	Maj@8	P@1	P@8	Maj@8
Zero-shot	24.66	45.21	21.92	23.33	33.33	30.00
w/ Reflection	26.02	52.05	28.76	20.00	36.67	33.33
w/ TWS	28.76	53.42	26.02	36.67	50.00	46.67
w/ TWS + TopK	30.14	52.05	26.02	36.67	50.00	46.67
w/ TWS + CGTD	32.88	53.42	30.14	40.00	53.33	46.67
w/ TWS + CGTD†	**35.62**	**54.79**	**34.25**	**40.00**	**60.00**	**56.67**

Table 2: Ablation on MMLU-Hard and AIME24 using Qwen3-4B-Instruct. TWS: teacher-weighted supervision; TopK: entropy-based selection. Full results are in Appendix F.

among extracted answers; normalization and equivalence checks are used only for final evaluation. For all tasks, we set the maximum output length to 4096 tokens and keep sampling parameters and stopping criteria identical across methods. For efficiency, the reflection prompt contains the query and only the last 1024 tokens of each student trajectory.

All runs are conducted on 8×NVIDIA H200 141GB GPUs. For fair comparison, all experiments are conducted using the verl framework Sheng et al. (2024) under a unified configuration. Except for the hyperparameters specific to our method, all other training settings are kept identical across experiments. Detailed configurations are provided in Appendix C.

4.2 Results

Main Results. Table 1 compares RCTS with representative label-free test-time adaptation methods across four benchmarks and two model families. We first consider query-only RCTS, which uses the same inference context as the baselines. Across both models, RCTS improves over the frozen policy and generally matches or exceeds the strongest non-RCTS method. On Qwen3-4B-Instruct, the clearest gains appear on MMLU-Hard and AMC23, while improvements on MedQA are smaller. RCTS also achieves the highest observed P@1 and P@8 on AIME24, although the P@1 gain corresponds to only one additional correct problem and should be interpreted cautiously. A similar pattern holds for Llama-3.2-3B-Instruct, with consistent P@1 gains on MedQA, MMLU-Hard, and AMC23. RCTS† adds reflection-conditioned context at inference and is therefore not a prompt-matched comparison. It often improves majority-vote accuracy, particularly for Qwen3-4B-Instruct, but not in every setting; we report it as a complementary inference mode. Appendix E presents an example in which reflection corrects an initially misinterpreted objective.

Ablation Study. Table 2 evaluates the main components of RCTS on MMLU-Hard and AIME24. Reflection-only prompting improves most metrics, but decreases AIME24 P@1, suggesting that reflection alone is not uniformly beneficial. Adding teacher-weighted supervision (TWS) produces clear gains over the zero-shot model, supporting the use of reflection-conditioned teacher answers as an answer-level pseudo-supervision signal. Compared with entropy-based TopK selection, CGTD improves all three MMLU-Hard metrics and AIME24 P@1 and P@8, while matching AIME24 Maj@8. Reflection-conditioned inference further improves most results. Overall, the ablation supports the complementary roles of TWS for trajectory supervision and CGTD for selective token updates.

Model / Method	P@1	P@8	Maj@8
Qwen3-1.7B	13.69	36.98	15.06
w/ TTRL	20.54	43.83	16.43
w/ RCTS	23.28	45.21	19.17
w/ RCTS†	24.66	49.31	23.28
Qwen3-8B	26.02	46.57	21.92
w/ TTRL	28.76	49.31	27.40
w/ RCTS	32.88	53.42	30.14
w/ RCTS†	32.88	56.16	30.14

Table 3: Qwen3 variants on MMLU-Hard.

4.3 Robustness and Sampling Controls

Model robustness. Table 3 extends MMLU-Hard evaluation to Qwen3-1.7B and Qwen3-8B. We use these variants as a checkpoint robustness test rather than a scaling study. RCTS improves over the

Dataset	Method	P@1	P@8	Maj@8
MMLU-Hard	DARE	31.04 ± 2.09	49.77 ± 2.09	26.94 ± 2.09
	RCTS	$\mathbf{32.88 \pm 1.37}$	$\mathbf{53.42 \pm 1.37}$	$\mathbf{29.68 \pm 0.80}$
AIME24	DARE	36.67 ± 3.34	50.00 ± 3.33	46.67 ± 3.34
	RCTS	$\mathbf{38.89 \pm 1.92}$	$\mathbf{54.44 \pm 1.93}$	$\mathbf{48.89 \pm 1.92}$

Table 4: Run-to-run robustness on MMLU-Hard and AIME24 using Qwen3-4B-Instruct. Results are mean $\pm$ standard deviation over three independent test-time adaptation runs.

Dataset	Method	Sampling Configuration	P@1	Maj@8
MMLU-Hard	DARE	$N = 8$	31.50	27.40
	DARE	$N = 24$	30.14	27.40
	RCTS	$8_{\mathrm{stu}} + 1_{\mathrm{refl}} + 8_{\mathrm{tea}}$	**32.88**	**30.14**
AMC23	DARE	$N = 8$	80.00	90.00
	DARE	$N = 24$	80.00	**92.50**
	RCTS	$8_{\mathrm{stu}} + 1_{\mathrm{refl}} + 8_{\mathrm{tea}}$	**82.50**	**92.50**

Table 5: Large-rollout comparison on MMLU-Hard and AMC23 using Qwen3-4B-Instruct. DARE uses 8 or 24 query-only rollouts, while RCTS uses 8 student rollouts, one batched reflection generation, and 8 teacher rollouts.

frozen models and TTRL, with further gains under reflection-conditioned inference, suggesting that the effect is not specific to the 4B checkpoint.

Run-to-run robustness. We repeat DARE and RCTS over three independent test-time adaptation runs on MMLU-Hard and AIME24 using Qwen3-4B-Instruct. As shown in Table 4, RCTS achieves higher mean performance and lower standard deviation than DARE on all reported metrics. The AIME24 differences should nevertheless be interpreted cautiously given its small test set.

Large-rollout control. To test whether the gains of RCTS can be attributed to additional sampling, we increase the DARE rollout group from $N = 8$ to $N = 24$. As shown in Table 5, the larger query-only group does not improve either reported MMLU-Hard metric and improves only Maj@8 on AMC23. RCTS remains stronger in P@1 on both benchmarks and in Maj@8 on MMLU-Hard, suggesting that simply increasing query-only rollouts is insufficient to reproduce its gains. **Qualitative Token Analysis.** Appendix G compares Random, entropy TopK, and CGTD on the same AMC23 trajectory. Entropy TopK emphasizes uncertain verification and transition spans, whereas CGTD selects lower-entropy, problem-specific relations and transformations.

5 Limitations and Conclusion

RCTS has several limitations. Its adaptation quality depends on the quality and diversity of self-generated rollouts. When both query-only student rollouts and reflection-conditioned teacher rollouts collapse to the same incorrect reasoning pattern, the teacher-weighted answer distribution may become confidently wrong rather than corrective. In addition, CGTD provides a cross-context distributional signal rather than explicit semantic-step alignment, and therefore does not guarantee causal localization of the reasoning steps that determine the final answer. Future work may improve reliability and efficiency through adaptive sampling, stronger reflection mechanisms, explicit semantic-step alignment, and cost-aware adaptation schedules.

Despite these limitations, RCTS provides a practical framework for label-free test-time RL by combining reflection-conditioned answer supervision with cross-context token selection in a single policy. Across mathematical reasoning and knowledge-intensive QA tasks, RCTS improves frozen models and remains competitive with representative label-free adaptation baselines. Our qualitative analysis further suggests that CGTD captures information distinct from local predictive uncertainty.

AI USE STATEMENT

In this work, we used generative AI tools to assist with language editing and polishing, including improving grammar, clarity, and presentation of author-written text. We also used generative AI tools to support literature search and discovery of potentially relevant prior work. All suggested references were independently checked against the original sources before being included in the paper.

We did not use generative AI tools to generate or conduct the research ideas or experiments, write substantive sections of the paper, generate synthetic datasets, or prove mathematical claims. All AI-assisted text was reviewed and edited by the authors, and all technical claims, experimental results, and citations were independently verified. We take responsibility for the final content of this work, including any text or other artifacts produced with the aid of generative AI.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The twelfth international conference on learning representations*, 2024.

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. The unreasonable effectiveness of entropy minimization in llm reasoning. *arXiv preprint arXiv:2505.15134*, 2025.

Lingjiao Chen, Jared Davis, Boris Hanin, Peter Bailis, Ion Stoica, Matei Zaharia, and James Zou. Are more llm calls all you need? towards the scaling properties of compound ai systems. *Advances in Neural Information Processing Systems*, 37:45767–45790, 2024.

Bodong Du, Xuanqi Huang, and Xiaomeng Li. Distribution-aware reward estimation for test-time reinforcement learning. *arXiv preprint arXiv:2601.21804*, 2026.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *arXiv preprint arXiv:2603.25562*, 2026.

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

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

Haoyang He, Zihua Rong, Liangjie Zhao, Yunjia Zhao, Lan Yang, and Honggang Zhang. Ttsr: Test-time self-reflection for continual reasoning improvement. *arXiv preprint arXiv:2603.03297*, 2026.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

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

Peyman Hosseini, Ondrej Bohdal, Taha Ceritli, Ignacio Castro, Matthew Purver, Mete Ozay, and Umberto Michieli. Cg-ttrl: Context-guided test-time reinforcement learning for on-device large language models. *arXiv preprint arXiv:2511.06430*, 2025.

Jinwu Hu, Zhitian Zhang, Guohao Chen, Xutao Wen, Chao Shuai, Wei Luo, Bin Xiao, Yuanqing Li, and Mingkui Tan. Test-time learning for large language models. *arXiv preprint arXiv:2505.20633*, 2025.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

Dulhan Jayalath, Shashwat Goel, Thomas Foster, Parag Jain, Suchin Gururangan, Cheng Zhang, Anirudh Goyal, and Alan Schelten. Compute as teacher: Turning inference compute into reference-free supervision. *arXiv preprint arXiv:2509.14234*, 2025.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? A large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021.

Vanshaj Khattar, Moumita Choudhury, Jing Liu, Toshiaki Koike-Akino, Ming Jin, Ye Wang, et al. Amplification effects in test-time reinforcement learning: Safety and reasoning vulnerabilities. *arXiv preprint arXiv:2603.15417*, 2026.

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

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in neural information processing systems*, 36:46534–46594, 2023.

Srijoni Majumdar, Edith Elkind, and Evangelos Pournaras. Generative ai voting: fair collective choice is resilient to llm biases and inconsistencies. *EPJ Data Science*, 2026.

Changdae Oh, Zhen Fang, Shawn Im, Xuefeng Du, and Yixuan Li. Understanding multimodal llms under distribution shifts: An information-theoretic approach. *arXiv preprint arXiv:2502.00577*, 2025.

Emiliano Penaloza, Dheeraj Vattikonda, Nicolas Gontier, Alexandre Lacoste, Laurent Charlin, and Massimo Caccia. Privileged information distillation for language models. *arXiv preprint arXiv:2602.04942*, 2026.

Hejian Sang, Yuanda Xu, Zhengze Zhou, Ran He, Zhipeng Wang, and Jiachen Sun. On-policy self-distillation for reasoning compression. *arXiv preprint arXiv:2603.05433*, 2026.

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

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.

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

Taiwei Shi, Sihao Chen, Bowen Jiang, Linxin Song, Longqi Yang, and Jieyu Zhao. Experiential reinforcement learning. *arXiv preprint arXiv:2602.13949*, 2026.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023.

Akshit Singh, Shyam Marjit, Wei Lin, Paul Gavrikov, Serena Yeung-Levy, Hilde Kuehne, Rogerio Feris, Sivan Doveh, James Glass, and M Jehanzeb Mirza. Ttrv: Test-time reinforcement learning for vision language models. *arXiv preprint arXiv:2510.06783*, 2025.

Yuda Song, Lili Chen, Fahim Tajwar, Remi Munos, Deepak Pathak, J Andrew Bagnell, Aarti Singh, and Andrea Zanette. Expanding the capabilities of reinforcement learning via text feedback. *arXiv preprint arXiv:2602.02482*, 2026.

Xiangru Tang, Daniel Shao, Jiwoong Sohn, Jiapeng Chen, Jiayi Zhang, Jinyu Xiang, Fang Wu, Yilun Zhao, Chenglin Wu, Wenqi Shi, Arman Cohan, and Mark Gerstein. Medagentsbench: Benchmarking thinking models and agent frameworks for complex medical reasoning. 2025.

Kun Wang and Reinhard Heckel. Test-time rl alignment exposes task familiarity artifacts in llm benchmarks. *arXiv preprint arXiv:2603.12875*, 2026.

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

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

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

Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, et al. Mimo-v2-flash technical report. *arXiv preprint arXiv:2601.02780*, 2026.

Dong Yan, Jian Liang, Yanbo Wang, Shuo Lu, Ran He, and Tieniu Tan. What if consensus lies? selective-complementary reinforcement learning at test time. *arXiv preprint arXiv:2603.19880*, 2026.

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

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

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled rlvr. *arXiv preprint arXiv:2604.03128*, 2026b.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. *arXiv preprint arXiv:2602.12275*, 2026.

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

Lifan Yuan, Yangyi Chen, Ganqu Cui, Hongcheng Gao, Fangyuan Zou, Xingyi Cheng, Heng Ji, Zhiyuan Liu, and Maosong Sun. Revisiting out-of-distribution robustness in nlp: Benchmarks, analysis, and llms evaluations. *Advances in Neural Information Processing Systems*, 36:58478–58507, 2023.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.

Kai Zhang, Xiangchao Chen, Bo Liu, Tianci Xue, Zeyi Liao, Zhihan Liu, Xiyao Wang, Yuting Ning, Zhaorun Chen, Xiaohan Fu, et al. Agent learning via early experience. *arXiv preprint arXiv:2510.08558*, 2025.

Zhaoyang Zhang, Shuli Jiang, Yantao Shen, Yuting Zhang, Dhananjay Ram, Shuo Yang, Zhuowen Tu, Wei Xia, and Stefano Soatto. Reinforcement-aware knowledge distillation for llm reasoning. *arXiv preprint arXiv:2602.22495*, 2026.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

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

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

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

A ALGORITHM

Algorithm 1 summarizes one RCTS test-time update for an unlabeled query. RCTS first freezes the current policy as a rollout snapshot and uses it to generate query-only student trajectories, diagnostic reflections, and reflection-conditioned teacher trajectories. The teacher trajectories are then aggregated into a weighted answer distribution to provide trajectory-level pseudo-supervision, while CGTD identifies student tokens whose empirical usage deviates most from the reflection-conditioned teacher group. Finally, the current policy is updated by a mask-selective GRPO objective on the selected query-only student tokens, with a masked KL regularizer to the frozen base policy. Teacher tokens are never used as direct supervised targets.

Algorithm 1: RCTS test-time update for one unlabeled query

Input: Unlabeled query q; current policy π_θ; base policy π_{base}; rollout size N; token-selection ratio k; KL weight β_{KL}; clip range $(\varepsilon_{\text{low}}, \varepsilon_{\text{up}})$.
Output: Updated policy parameters θ.

$\theta_{\text{old}} \leftarrow \theta$ and freeze $\pi_{\theta_{\text{old}}}$ for rollout-dependent signal construction

Student rollouts. For $i = 1, \ldots, N$, sample $y_i^{\text{stu}} \sim \pi_{\theta_{\text{old}}}(\cdot \mid q)$; set $\mathcal{Y}_q^{\text{stu}} = \{y_i^{\text{stu}}\}_{i=1}^N$ and $\alpha_i^{\text{stu}} = \text{Ans}(y_i^{\text{stu}})$

Reflection-conditioned teacher rollouts. Generate diagnostic reflections $\mathcal{R}_q = \{r_m\}_{m=1}^N = \text{Reflect}_{\theta_{\text{old}}}^{(N)}(q, \mathcal{Y}_q^{\text{stu}})$

For $m = 1, \ldots, N$, sample $y_m^{\text{tea}} \sim \pi_{\theta_{\text{old}}}(\cdot \mid q, r_m)$; set $\mathcal{Y}_q^{\text{tea}} = \{y_m^{\text{tea}}\}_{m=1}^N$ and $\alpha_m^{\text{tea}} = \text{Ans}(y_m^{\text{tea}})$

Teacher-weighted answer support. Compute length-normalized teacher scores s_m by Eq. 4 and normalized weights λ_m by Eq. 5

Construct the teacher-weighted answer distribution $P_q^{\text{tea}}(\alpha)$ by Eq. 6

For each student trajectory, assign teacher-supported consistency $c_i = P_q^{\text{tea}}(\alpha_i^{\text{stu}})$

Compute query-level teacher confidence ω_q^{tea} by Eq. 9

CGTD token masking. Estimate position-wise empirical token distributions p_t^{stu} and p_t^{tea} from student and teacher groups by Eq. 10

Compute the position-level gap δ_t by Eq. 11

for $i = 1, \ldots, N$ **do**

  Compute CGTD scores $S_{i,t}^{\text{stu}}$ for realized student tokens by Eq. 12

  Let $\mathcal{P}_i = \{t : S_{i,t}^{\text{stu}} > 0\}$ and $K_i = \lceil k|\mathcal{P}_i| \rceil$

  Set $\mathcal{M}_i^{\text{stu}}$ to the K_i positions in $\mathcal{P}_i$ with the largest CGTD scores; if $\mathcal{P}_i = \emptyset$, set $\mathcal{M}_i^{\text{stu}} = \emptyset$

end

Mask-selective GRPO update. Teacher trajectories are used only to construct answer-level support and CGTD masks; no teacher token is optimized as a supervised target

Compute student advantages A_i^{stu} from $\{c_i\}_{i=1}^N$ and ω_q^{tea} by Eq. 14

Update π_θ by minimizing

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{PG}}(\theta) + \beta_{\text{KL}}\mathcal{L}_{\text{KL}}(\theta),$$

where $\mathcal{L}_{\text{PG}}$ is applied only to selected query-only student tokens $\{(i,t) : t \in \mathcal{M}_i^{\text{stu}}\}$ and $\mathcal{L}_{\text{KL}}$ regularizes the same masked positions toward π_{base}

return *updated* θ

B BATCHED REFLECTION PROMPTS

The reflection prompt is designed to provide diagnostic feedback without revealing, confirming, or hinting at the final answer. For each query, the prompt receives the original question and the sampled student attempts, and returns one reflection per attempt. These reflections are used only as prompt-side feedback for reflection-conditioned teacher rollouts and are not directly optimized by the

RCTS objective. The concrete prompt templates for MedQA and mathematical reasoning are shown in Tables 6 and 7, respectively.

Table 6: Batched Reflection Prompting for MedQA.

Batched Reflection Prompting for MedQA

Batched Reflection User Prompt Template

" You are a rigorous medical reasoning reviewer. You will be shown a multiple-choice clinical question and N attempted solutions. For each attempt produce ONE focused reflection that critiques THAT SPECIFIC attempt in concrete, problem-grounded terms.
Examine: formatting (missing `\boxed{X}` *A-J letter, use of <answer>), length/structure, concrete clinical / biomedical reasoning errors, internal inconsistencies, and what to verify on retry.*
Each reflection must (a) be SPECIFIC to THIS attempt; (b) NOT reveal, name, confirm, or rank any specific answer option letter; (c) be 2–5 sentences; (d) be DISTINCT across attempts.
Output format: exactly N reflection blocks, in order: <reflection i="1">...</reflection> ... up to <reflection i="N">...</reflection> Return ONLY the reflection blocks, nothing else. "

User prompt.

" Question
{question}
8 Student Attempts
Attempt 1
attempt 1 tail excerpt
Attempt 2
attempt 2 tail excerpt
...
Attempt 8
attempt 8 tail excerpt
Now produce exactly 8 <reflection i="k"> blocks, one per attempt, in order (k from 1 to 8). Focus on format, length, concrete reasoning errors visible in that attempt, internal inconsistencies, and what to re-verify on a retry. Do NOT reveal, confirm, or hint at the correct final answer. Make each reflection distinct. "

C Experimental Configuration and Reproducibility

To facilitate reproducibility, Table 8 summarizes the core optimization and rollout configuration used in our RL experiments. Unless otherwise specified, all methods and ablations are run under the same base optimization setup, sampling budget, and sequence length limits. Method-specific components, such as reward construction, token-level masking, and reflection-conditioned teacher rollouts, are varied independently as described in the corresponding experimental sections.

D Evaluation Protocol: Answer Extraction

For math problems, we extract the final answer from the model's long-form solution by searching for the final boxed expression. If the response contains special box delimiters, we use the last span between `<|begin_of_box|>` and `<|end_of_box|>`; otherwise, we extract the content of the last LaTeX `\boxed{}` expression. Responses without a valid boxed answer are treated as missing predictions.

For multiple-choice questions, we use a stricter option extractor for voting-based evaluation: the prediction must appear as `\boxed{X}`, where X is an uppercase option letter from A to J. The extracted option letter is normalized to uppercase. If no valid boxed option is found, the response is treated as missing.

Missing predictions are counted as incorrect for P@1 and majority voting, and they do not contribute a valid answer candidate in answer-frequency based pseudo-reward construction.

Table 7: Batched Reflection Prompting for Math.

Batched Reflection Prompting for Math

Batched Reflection User Prompt Template

" You are a rigorous mathematical reasoning reviewer. You will be shown a math problem and N attempted solutions. For each attempt, produce ONE focused reflection that critiques THAT SPECIFIC attempt in concrete, problem-grounded terms.
For each reflection examine and point out where relevant: formatting (missing/extra `\boxed{}`), length and structure issues, concrete reasoning errors visible in the attempt, internal inconsistencies, and what to verify on a retry.
Each reflection must (a) be SPECIFIC — reference the step, quantity, or transition in THIS attempt; (b) NOT reveal, confirm, or hint at any specific final numeric answer; (c) be 2–5 sentences, concrete and actionable; (d) be DISTINCT across attempts.
Output format: exactly N reflection blocks, in order, using <reflection i="1">...</reflection> <reflection i="2">...</reflection> ... up to <reflection i="N">...</reflection> Return ONLY the reflection blocks, nothing else. "

User prompt.

" Question
{problem}
{N} Student Attempts
Attempt 1
attempt 1 excerpt
Attempt 2
attempt 2 excerpt
...
Attempt N
attempt N excerpt
Now produce exactly N <reflection i="k"> blocks, one per attempt, in order (k from 1 to N). Focus on format, length, concrete reasoning errors visible in that attempt, internal inconsistencies, and what to re-verify on a retry. Do NOT reveal, confirm, or hint at the correct final answer. Make each reflection distinct. "

E Qualitative Example: Reflection Corrects an Objective Misinterpretation

Table 10 provides a representative example illustrating how reflection helps recover from an initially wrong reasoning path. In the first answer, the model correctly recognizes the geometric structure of the rhombus and derives the orthogonality condition between the two diagonal directions. However, it misinterprets the problem objective: instead of finding the greatest lower bound of BD^2, it treats the task as maximizing BD^2 and even produces the erroneous boxed target $\boxed{\sup\{BD^2\} - \varepsilon}$.

The reflection does not simply provide the final answer. Instead, it diagnoses the failure mode of the initial reasoning: the parametrization and expression for BD^2 are useful, but the model has not analyzed the required minimum or infimum. This feedback redirects the subsequent reasoning toward the correct objective while preserving the valid intermediate derivations. With this reflection, the model reuses the hyperbolic parametrization, derives the feasibility condition $|\tanh\beta| > 5/6$, and analyzes the boundary case $|\tanh\beta| \to 5/6^+$. This leads to $BD^2 > 480$ for all feasible rhombi, with values arbitrarily close to 480, yielding the correct final answer $\boxed{480}$.

This example shows that reflection can act as a targeted diagnostic signal: rather than replacing the solver's reasoning, it identifies which part of the initial chain should be corrected and guides the model to complete the missing infimum analysis.

Table 8: Core experimental configuration used in our RL experiments.

Configuration	Value
Common optimization and rollout settings	
Base models	Qwen3-4B-Instruct-2507 / Llama-3.2-3B-Instruct
Optimizer	AdamW
Learning rate	1×10^{-6}
Total RL steps	200
Train batch size	8
Policy mini-batch size	8
Rollout size per group	$N = 8$
KL coefficient	$\beta_{\mathrm{KL}} = 0.001$
Maximum prompt length	12,288 tokens
Maximum response length	4,096 tokens
Rollout temperature	0.7
Rollout top-p	0.8
Rollout top-k	20
Precision	bfloat16
Hardware	8 × NVIDIA H200 141GB GPUs
RCTS-specific settings	
Student rollout size	$N = 8$
Teacher rollout size	$N = 8$
CGTD token-selection ratio	$k = 0.2$
Reflection context length	Last 1024 tokens per student attempt

Table 9: Answer extraction rules for long-form model responses.

Task type	Extraction rule
Math reasoning	We extract the final answer from the last boxed expression in the response. If the response contains <|begin_of_box|> and <|end_of_box|> delimiters, we take the last delimited span. Otherwise, we take the content of the last LaTeX \boxed{} expression. Responses without a valid boxed expression are marked as missing.
Multiple-choice reasoning	We extract a single option letter from a strict boxed format, \boxed{X}, where $X \in \{A, \ldots, J\}$. The option letter is normalized to uppercase. Responses without a valid boxed option are marked as missing for voting-based evaluation.

F Complete Ablation Results

For completeness, Table 11 reports the full ablation study across all four benchmarks. This extends the main-text ablation, which focuses on MMLU-Hard and AIME24, by additionally reporting MedQA and AMC23. The same notation is used as in the main paper.

G CGTD Diagnostic on a Fixed Student Trajectory

To better understand what CGTD selects beyond local token uncertainty, we perform a controlled diagnostic on a fixed AMC23 student response. Random selection, entropy TopK, and CGTD are all applied to the same 3,030-token student trajectory. CGTD uses the same reflection-conditioned teacher group used by RCTS, while the entropy baseline ranks tokens according to local predictive entropy.

Entropy TopK strongly concentrates on high-uncertainty positions and selects relatively more tokens near the end of the response, where verification and transition spans are common. By contrast, CGTD selects substantially lower-entropy tokens and places relatively more mass in the early and middle regions of the same trajectory. The Jaccard overlap between Entropy TopK and CGTD is only 7.9%, showing that CGTD does not simply recover a local uncertainty criterion.

Inspection of the selected spans further indicates that entropy-based selection often emphasizes generic transition and verification tokens, whereas CGTD more frequently highlights problem-specific algebraic relations and transformations whose usage changes between query-only and reflection-conditioned contexts. We therefore interpret CGTD as a detector of student-specific cross-context deviations rather than as a causal localization mechanism for answer-determining tokens.

This distinction is also reflected in the ablation results. Replacing entropy TopK with CGTD improves MMLU-Hard P@1/Maj@8 from 30.14/26.02 to 32.88/30.14, and improves AIME24 P@1/P@8 from 36.67/50.00 to 40.00/53.33.

H Additional Robustness and Sampling-Budget Controls

H.1 Multi-Seed Evaluation

We repeat TTRL, DARE, and RCTS over three independent adaptation runs on MMLU-Hard and AIME24. Table 13 reports mean $\pm$ standard deviation. RCTS obtains the highest mean across all reported metrics, while exhibiting equal or lower variation in most comparisons. We interpret these results as evidence of run-to-run robustness rather than as a claim of statistical significance, especially on the 30-problem AIME24 set.

H.2 Sampling-Budget Control

RCTS uses additional generations to construct reflection-conditioned teacher signals. To test whether its gains can be explained simply by additional query-only sampling, we increase DARE from $N = 8$ to $N = 24$ while keeping the same framework, optimization setup, response-length limit, and hardware. We also account for prompt and generated tokens per adaptation step.

Table 10: A representative case where reflection corrects an initially wrong answer. We show verbatim excerpts from model outputs; ellipses indicate omitted intermediate reasoning for readability.

Example of reflection correcting an initially wrong reasoning path.

Problem. *Let A, B, C, and D be points on $\frac{x^2}{20} - \frac{y^2}{24} = 1$ such that $ABCD$ is a rhombus whose diagonals intersect at the origin. Find the greatest real number that is less than BD^2 for all such rhombi.*

First answer (excerpt, incorrect).

We are to find the greatest real number that is less than BD^2 for all such rhombi. That is, we want:

$$\sup\left\{r \mid r < BD^2 \text{ for all such rhombi } ABCD\right\}.$$

So, we are looking for the least upper bound (supremum) of BD^2, and then take the greatest real number strictly less than that ...

But since it says "the greatest real number that is less than BD^2 for all such rhombi", we are to find:

$$\boxed{\sup\{BD^2\} - \varepsilon} \quad \text{for any } \varepsilon > 0.$$

So ultimately, we want the supremum of BD^2 over all such rhombi ...

$$\tanh\theta \tanh\phi = -\frac{5}{6}.$$

... We now want to maximize $BD^2 = 176c - 96$ ...

Reflection.

The attempt correctly parametrizes points on the hyperbola using hyperbolic functions and derives the condition $\tanh u \tanh v = -\frac{5}{6}$ from orthogonality. However, it then computes $BD^2 = 176\cosh^2 v - 96$ without addressing the constraint or exploring the range of v under the hyperbolic condition. The expression for BD^2 is valid, but the attempt stops short of analyzing its minimum or infimum, missing the core objective of finding the greatest lower bound for BD^2 across all such rhombi.

Answer with reflection (excerpt, correct).

We are to find the greatest real number that is less than BD^2 for all such rhombi — that is, find the greatest lower bound (infimum) of BD^2 over all such rhombi.

... We must have

$$\tanh\alpha \tanh\beta = -\frac{5}{6}.$$

Now, we want to compute $BD^2 = 4(c^2 + d^2)$, where $Q = (c, d)$, so:

$$c^2 + d^2 = 20\cosh^2\beta + 24\sinh^2\beta.$$

We want to minimize this expression ... Since $\tanh\alpha = -\frac{5}{6} \cdot \frac{1}{\tanh\beta}$ and $|\tanh\alpha| < 1$, we must have

$$|\tanh\beta| > \frac{5}{6}.$$

...

$$f(\beta) = 20\cosh^2\beta + 24\sinh^2\beta = 22\cosh 2\beta - 2.$$

Taking $|\tanh\beta| \to \frac{5}{6}^+$ gives

$$\cosh 2\beta = \frac{1 + (5/6)^2}{1 - (5/6)^2} = \frac{61}{11},$$

and hence

$$f(\beta) = 22 \cdot \frac{61}{11} - 2 = 120.$$

Therefore,

$$BD^2 = 4(c^2 + d^2) > 480.$$

So $BD^2 > 480$ for all such rhombi, and 480 is a lower bound but not attained. Thus, the greatest real number less than BD^2 for all such rhombi is:

$$\boxed{480}.$$

Name	MedQA			MMLU-Hard			AMC23			AIME24		
	P@1	**P@8**	**Maj@8**	**P@1**	**P@8**	**Maj@8**	**P@1**	**P@8**	**Maj@8**	**P@1**	**P@8**	**Maj@8**
Zero-shot	72.40	86.40	74.00	24.66	45.21	21.92	65.00	80.00	80.00	23.33	33.33	30.00
w/ Reflection	76.40	86.40	76.00	26.02	52.05	28.76	82.50	87.50	82.50	20.00	36.67	33.33
w/ TWS	76.80	87.20	76.60	28.76	53.42	26.02	82.50	95.00	92.50	36.67	50.00	46.67
w/ TWS†	76.60	87.50	78.60	30.14	53.42	31.50	82.50	95.00	95.00	36.67	56.67	56.67
w/ TWS + TopK	76.20	87.60	76.60	30.14	52.05	26.02	82.50	95.00	92.50	36.67	50.00	46.67
w/ TWS + TopK†	76.60	88.20	79.00	32.88	53.42	31.50	85.00	95.00	92.50	40.00	56.67	53.33
w/ TWS + CGTD	77.20	87.80	77.20	32.88	53.42	30.14	82.50	95.00	92.50	40.00	53.33	46.67
w/ TWS + CGTD†	77.60	88.20	79.00	35.62	54.79	34.25	85.00	95.00	95.00	40.00	60.00	56.67

Table 11: Ablation study of **RCTS** on four benchmarks using Qwen3-4B-Instruct. TWS denotes teacher-weighted supervision, and CGTD denotes Cross-Group Token Divergence. w/ Reflection denotes inference with reflection-conditioned prompts but without any update. † indicates reflection-conditioned teacher-side inference; otherwise, query-only student-side inference is used. TWS + TopK denotes an entropy-based token-selection baseline.

Table 12: Token-selection diagnostic on the same fixed 3,030-token AMC23 student response. "Early / middle / late" reports the number of selected tokens in three equal-length response regions.

Selector	**Selected**	**Early / middle / late**	**Mean entropy**	**Jaccard w/ CGTD**
Random-20%	606 (20.0%)	185 / 203 / 218	0.172	11.4%
Entropy TopK	606 (20.0%)	148 / 171 / 287	0.832	7.9%
CGTD	555 (18.3%)	189 / 211 / 155	**0.124**	–

Table 13: Three-run comparison on MMLU-Hard and AIME24 using Qwen3-4B-Instruct-2507.

Dataset	**Method**	**P@1**	**P@8**	**Maj@8**
MMLU-Hard	TTRL	28.76 ± 1.37	51.59 ± 2.09	26.48 ± 0.80
	DARE	31.04 ± 2.09	49.77 ± 2.09	26.94 ± 2.09
	RCTS	$\mathbf{32.88 \pm 1.37}$	$\mathbf{53.42 \pm 1.37}$	$\mathbf{29.68 \pm 0.80}$
AIME24	TTRL	32.22 ± 1.92	50.00 ± 3.33	47.78 ± 1.92
	DARE	36.67 ± 3.34	50.00 ± 3.33	46.67 ± 3.34
	RCTS	$\mathbf{38.89 \pm 1.92}$	$\mathbf{54.44 \pm 1.93}$	$\mathbf{48.89 \pm 1.92}$

Table 14: Sampling-budget control. RCTS token counts include student, reflection, and teacher generations. The comparison is a token-budget control, not an exact wall-clock match.

Dataset	Method / sampling configuration	Tokens / step (prompt / generated)	P@1	Maj@8
MMLU-Hard	DARE, $N = 8$ query-only	1.43k / 5.83k	31.50	27.40
	DARE, $N = 24$ query-only	4.30k / 17.50k	30.14	27.40
	RCTS, 8 student + reflection + 8 teacher	6.51k / 12.84k	**32.88**	**30.14**
AMC23	DARE, $N = 8$ query-only	1.31k / 19.81k	80.00	90.00
	DARE, $N = 24$ query-only	3.93k / 59.44k	80.00	**92.50**
	RCTS, 8 student + reflection + 8 teacher	8.19k / 41.17k	**82.50**	**92.50**

Increasing DARE to $N = 24$ does not improve either MMLU-Hard metric and improves only AMC23 Maj@8. RCTS retains higher P@1 on both datasets while generating fewer tokens than DARE with $N = 24$. Thus, additional query-only sampling alone does not reproduce the RCTS gains.

I Sensitivity Analyses

All main experiments use one shared default configuration: $N = 8$, $k = 0.2$, $\beta_{\mathrm{KL}} = 0.001$, and a 1,024-token reflection context. We perform one-factor-at-a-time sensitivity analyses on Qwen3-4B-Instruct-2507 and MMLU-Hard. No hyperparameter is tuned separately for an individual benchmark.

Table 15: One-factor-at-a-time sensitivity analysis on MMLU-Hard.

Parameter	Value	P@1
k	0.10	31.50
	0.20	**32.88**
	0.40	28.76
N	4	30.14
	8	32.88
	16	**35.62**
β_{KL}	0	30.14
	0.001	**32.88**
	0.01	31.50
Reflection context (tokens)	256	28.76
	512	30.14
	1,024	**32.88**

Performance is stable for $k = 0.1$–0.2 but decreases at $k = 0.4$. Increasing N improves P@1 over the tested range, while $N = 8$ provides a practical accuracy–sampling-cost trade-off. Removing the KL anchor reduces P@1 from 32.88 to 30.14, while a stronger $\beta_{\mathrm{KL}} = 0.01$ yields 31.50. Longer reflection context also improves P@1 over the tested range, suggesting that the default configuration is not an isolated sharp optimum.

J Additional Ablations on Reflection and Teacher Supervision

J.1 Generic Self-Check Prompt

One possible simplification is to remove trajectory-conditioned reflections and teacher rollouts and instead use the same fixed self-check instruction for every query. We evaluate this alternative by applying the following generic instruction with standard TTRL:

> *Solve the problem carefully. Before finalizing your answer, review your reasoning for possible mistakes, verify the key steps, and correct any inconsistency. Then provide the final answer in the required format.*

Table 16: Generic self-check prompting on MMLU-Hard.

Method	P@1
TTRL with default prompt	28.76
Generic Self-Check Prompt + TTRL	30.14
RCTS with default student prompt	**32.88**

The fixed self-check prompt improves TTRL by 1.37 points but remains 2.75 points below RCTS. This result indicates that generic self-verification is useful, but does not reproduce the benefit of trajectory-conditioned reflection together with teacher-weighted supervision and CGTD.

Reflection-conditioned inference without parameter updates is also non-uniform: P@1 changes from 24.66 to 26.02 on MMLU-Hard but from 23.33 to 20.00 on AIME24. Reflection context alone therefore does not explain the query-only RCTS gains.

J.2 Soft versus Hard Teacher Supervision

To isolate whether the benefit of teacher-weighted supervision comes from the soft answer distribution rather than merely using teacher-generated answers, we keep the student and teacher rollouts, reflections, teacher trajectory weights, query-level confidence, CGTD masks, and optimization unchanged.

The soft variant uses

$$c_i^{\mathrm{soft}} = P_q^{\mathrm{tea}}(\alpha_i^{\mathrm{stu}}),$$

whereas the hard variant collapses the same weighted teacher distribution to its top-1 answer:

$$c_i^{\mathrm{hard}} = \mathbf{1}\left[\alpha_i^{\mathrm{stu}} = \arg\max_{\alpha} P_q^{\mathrm{tea}}(\alpha)\right].$$

Table 17: Soft versus hard teacher supervision on MMLU-Hard.

Teacher supervision	P@1
Hard top-1 + CGTD	31.50
Soft TWS + CGTD (RCTS)	**32.88**

Soft supervision improves P@1 by 1.38 points in this controlled comparison. Although the difference is modest on the 73-question benchmark, the result supports retaining graded teacher agreement instead of collapsing the teacher distribution to a single pseudo-label.