STRUCTURED SELF-FEEDBACK IN-CONTEXT REINFORCEMENT LEARNING FOR TEST-TIME ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Unsupervised test-time self-improvement remains challenging because LLMs must identify and reuse reliable experience from their own generations without access to ground-truth or external verifiers. In-context reinforcement learning (ICRL) provides a training-free mechanism for test-time adaptation by accumulating self-generated responses and feedback in context to improve reasoning performance. However, existing ICRL methods typically derive pseudo-labels through majority voting, which treats generated responses as an unordered set and overlooks the structured evidence provided by different reasoning strategies. As a result, answers supported by only a narrow subset of reasoning processes may be indistinguishable from those consistently supported across diverse strategies. In this paper, we introduce Structured Self-feedback In-Context Reinforcement Learning (SS-ICRL), which constructs structured self-feedback from multi-strategy generations for unsupervised test-time self-improvement. Specifically, we propose a Reliability-Hill (RH) score that jointly models within-strategy agreement and cross-strategy support, enabling effective pseudo-label selection based on structured reasoning evidence. To gudide self-improvement, SS-ICRL further integrates the self-feedback into strategy-specific contextual experience across optimization rounds. Experiments across multiple mathematical reasoning benchmarks show that SS-ICRL consistently improves reasoning performance in unsupervised test-time settings without updating model parameters.

1 INTRODUCTION

Recent advances in large language models (LLMs) have demonstrated strong reasoning capabilities across a wide range of complex tasks. However, real-world deployment continually exposes models to new and previously unseen problems, making on-the-fly adaptation increasingly important. Recent work on test-time scaling (TTS) (OpenAI, 2024; DeepSeek-AI, 2025) shows that LLMs can further enhance their reasoning performance by allocating additional inference-time computation to repeated sampling and structured search, such as Best-of-N and tree search (Wang et al., 2023; Huang et al., 2025; Prasad et al., 2025; Zhang et al., 2024). For tasks with verifiable outcomes, existing methods can leverage rule-based rewards or external verifiers to guide inference-time search and optimization. However, in many real-world applications, ground-truth or external verifiers are unavailable during inference. Without access to these external correctness signals, test-time self-improvement lacks a direct criterion for determining which generated outcomes should guide further optimization. Achieving effective self-improvement at inference time without external supervision remains a crucial challenge for LLM reasoning.

To enable test-time self-improvement without external supervision, recent work has explored constructing self-generated feedback for policy optimization (Zhao et al., 2025; Prabhudesai et al., 2026; Zhang et al., 2025b). Test-time reinforcement learning (TTRL) (Zuo et al., 2025; Xu et al., 2026) methods typically infer pseudo-labels through majority voting and use them to construct rewards for policy optimization. However, existing TTRL frameworks generally introduce additional optimization overhead through test-time parameter updates. More recently, in-context reinforcement learning (ICRL) (Yu et al., 2026; Ye et al., 2026; Wang et al., 2025b; Song et al., 2026) has enabled iterative reasoning improvement without parameter updates by accumulating previous responses and rewards in the context. Despite this change in the optimization mechanism, existing unsupervised ICRL methods still largely infer pseudo-labels from aggregate final-answer frequencies. This

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.

frequency-based feedback has two limitations. First, systematic reasoning errors can repeatedly generate the same incorrect answer, causing majority voting to select an erroneous pseudo-label. Such pseudo-labels may bias subsequent rounds when used for context updates. Second, majority voting treats every occurrence of the same answer as equally informative, overlooking the structure behind repeated answers.

To better leverage self-generated reasoning experience, we observe that the same problem can be approached through different reasoning strategies, such as problem decomposition and symbolic reasoning. These strategies may induce different answer distributions, making repeated occurrences of the same answer unequally informative. This motivates a simple intuition: *a reliable answer should be reproducible within a reasoning strategy and corroborated across different strategies*. Based on this intuition, we propose Structured Self-feedback In-Context Reinforcement Learning (SS-ICRL), which exploits the structure within and across reasoning strategies to construct more reliable self-feedback for test-time self-improvement. For each problem, SS-ICRL generates responses under multiple reasoning strategies and preserves the strategy identity of each response. To exploit this structured information, we introduce a Reliability–Hill (RH) score to evaluate candidate answers using both within-strategy stability and cross-strategy support. RH first aggregates answer frequencies weighted by the concentration of each strategy's answer distribution. It then applies a bounded adjustment based on a Hill-number measure of effective cross-strategy support. The resulting pseudo-label provides feedback for selecting reasoning traces to retain in strategy-specific contexts. These contexts guide the next round of generation, allowing online experience to inform subsequent reasoning.

We evaluate SS-ICRL on three mathematical reasoning benchmarks using open-source models of different scales. Experimental results show that SS-ICRL consistently improves reasoning accuracy across models and problem difficulty levels. Notably, on AIME 2024 with Qwen3-8B, SS-ICRL improves performance from 9.90% to 33.44% without any parameter updates. On cases where RH and majority voting disagree, RH achieves 21.79% pseudo-label accuracy, compared with 1.28% for majority voting. Our main contributions are as follows:

- We propose SS-ICRL, a framework for unsupervised test-time adaptation without parameter updates. It organizes self-generated reasoning experience by strategy and uses feedback to update strategy-specific contexts across multiple rounds.
- We introduce the reliability–hill score for pseudo-label selection. RH combines stability-weighted answer support with a bounded adjustment based on effective cross-strategy support, providing distribution-based feedback for selecting reasoning experience.
- We conduct extensive experiments across three backbone models and seven mathematical reasoning settings. SS-ICRL consistently improves pseudo-label quality and final reasoning performance, outperforming the baselines across most model-task configurations.

2 Related Work

2.1 Test-Time Self-Improvement for LLM Reasoning

Test-time scaling has emerged as an effective paradigm for improving LLM reasoning by allocating additional computation during inference. Typical approaches increase inference-time computation through parallel sampling or iterative refinement (Snell et al., 2024; Huang et al., 2025; Welleck et al., 2024), enabling models to explore and improve candidate solutions without model training. Moving beyond output-space search, test-time reinforcement learning (Zuo et al., 2025) enables models to adapt directly on unlabeled test problems by constructing rewards from pseudo-labels. A main challenge of this paradigm is the reliability of self-generated supervision. Recent studies have explored richer internal signals beyond raw answer frequency, such as model confidence and consistency patterns across multiple generations (Prabhudesai et al., 2026; Zhao et al., 2025; Zhang et al., 2025b; Wang et al., 2025c; 2026). However, these signals do not characterize how answer support is distributed across heterogeneous reasoning strategies, potentially limiting the reliability of pseudo-label estimation when repeated errors are concentrated under particular reasoning patterns. In contrast, we actively induce structured generation through multiple reasoning strategies and leverage both within-strategy agreement and cross-strategy support for self-evaluation. In addition,

existing TTRL approaches require parameter updates during inference, which may be impractical when test-time parameter updates are constrained.

2.2 In-Context Reinforcement Learning

In-context reinforcement learning provides a distinct mechanism for policy improvement by allowing pretrained models to adapt from interaction histories provided in the context. Early studies investigate and characterize in-context adaptation capabilities in sequential decision-making settings (Moeini et al., 2025; Wang et al., 2025a). Algorithm distillation (Laskin et al., 2023) shows that a sequence model trained on reinforcement-learning histories can improve its policy entirely through context. Subsequent studies investigate whether pretrained models can perform in-context reinforcement learning from interaction histories (Lee et al., 2023; Krishnamurthy et al., 2024; Wang et al., 2025a). Recent work extends ICRL to general LLM self-improvement, where generated experiences are increasingly used to improve LLM reasoning. Song et al. (2026) demonstrate that LLMs can perform iterative self-improvement by conditioning on previous responses and scalar rewards. ME-ICPO (Yu et al., 2026) develops in-context policy optimization with self-assessed rewards and minimum-entropy response selection for iterative refinement. These studies establish reward specification, response selection, and context construction as key components of practical ICRL. However, the feedback signals in these methods are typically obtained from external verification or majority-voting pseudo-labels. Our work further studies structured self-feedback construction within ICRL under unsupervised reasoning settings.

3 Methodology

In this section, we present structured self-feedback in-context reinforcement learning (SS-ICRL) for unsupervised test-time self-improvement of LLM reasoning. The key idea is to exploit the structure of strategy-conditioned answer distributions to construct reliable self-feedback for iterative in-context optimization.

3.1 Problem Formulation and Motivation

We consider an unlabeled test-time setting with a set of reasoning problems $\mathcal{D}_{\text{test}} = \{x_i\}_{i=1}^{N}$, where the ground-truth answer for each problem is unavailable during optimization. Our goal is to improve the reasoning accuracy of a pretrained LLM π_θ while keeping its parameters θ fixed. Therefore, self-improvement must rely on pseudo-labels inferred from the model's own generations.

A common approach is to aggregate sampled answers by majority voting. However, answer frequency alone does not distinguish different sources of support. As illustrated in Figure 1, a_1 and a_2 occur equally often, yet the support for a_1 is concentrated within one reasoning strategy, whereas a_2 is supported across multiple strategies. Although indistinguishable under aggregate frequency, the two answers exhibit different strategy-level support structures. This motivates us to preserve strategy-conditioned answer distributions and construct self-feedback from how candidate answers are supported across reasoning strategies.

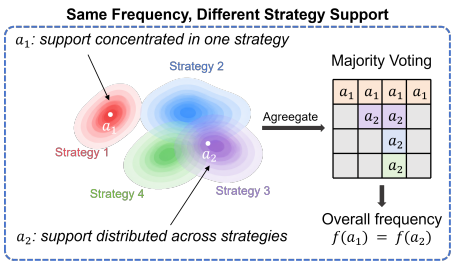

Figure 1: Equal answer frequencies can conceal different strategy-level support: the support for a_1 is concentrated in one reasoning strategy, whereas a_2 is supported across multiple strategies.

3.2 Structured Self-Feedback In-Context Reinforcement Learning

SS-ICRL leverages the structure of multi-strategy generations to construct self-feedback and guide iterative in-context optimization. At each round, the model generates responses under multiple reasoning strategies, from which SS-ICRL infers a pseudo-label and uses the feedback to update strategy-specific contexts for subsequent generation. The overall framework is shown in Figure 2.

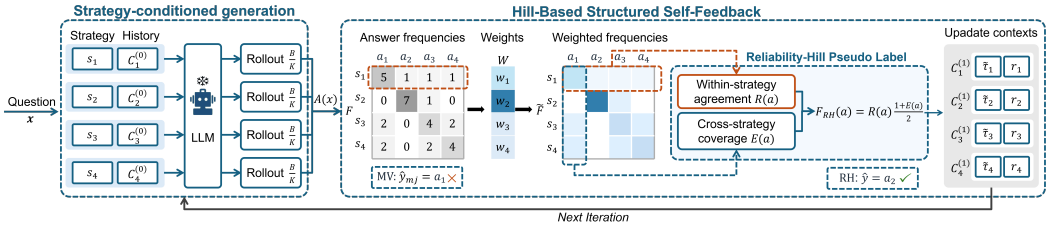

Figure 2: The illustration of SS-ICRL framework.

3.2.1 HILL-BASED STRUCTURED SELF-FEEDBACK

To construct self-feedback from different reasoning approaches, we use a set of reasoning strategies $\mathcal{S} = \{s_i\}_{i=1}^{K}$ to solve the same problem, where each strategy defines a distinct general approach to reasoning. In our implementation, we instantiate these strategies with a fixed set of prompts shared across all test problems. For a problem x, we allocate a total sampling budget of B evenly across the K strategies, yielding $n = B/K$ responses per strategy. We extract their final answers to form the candidate set $\mathcal{A}(x) = \{a_1, \ldots, a_A\}$, where $A = |\mathcal{A}(x)|$ is the number of distinct valid answers. The multi-strategy sampling procedure then yields an answer-frequency matrix

$$\mathbf{F} = \begin{bmatrix} f_1(a_1) & \cdots & f_1(a_A) \\ \vdots & \ddots & \vdots \\ f_K(a_1) & \cdots & f_K(a_A) \end{bmatrix}, \tag{1}$$

where $f_i(a_j)$ denotes the empirical frequency of answer a_j among the n responses generated under strategy s_i. Each row of $\mathbf{F}$ records the answer frequencies under one strategy, while each column records the frequencies of one answer across strategies. Majority voting collapses $\mathbf{F}$ into aggregate answer frequencies, treating all strategies equally and discarding how each answer is distributed across strategies. Instead, we exploit the two axes of $\mathbf{F}$ separately: row-wise statistics characterize agreement within each strategy, while column-wise statistics characterize the cross-strategy distribution of each answer.

Within-Strategy Agreement. We first use the rows of $\mathbf{F}$ to determine how much each reasoning strategy contributes to answer aggregation. The same frequency for an answer can arise from strategies with different internal answer distributions. When a strategy's responses are concentrated on a small number of answers, its answer distribution indicates stronger internal agreement. Conversely, a more dispersed answer distribution indicates weaker agreement. Therefore, we use within-strategy agreement to assign strategy-specific weights when aggregating answer frequencies.

Motivated by the effective-number interpretation of Hill diversity Liu et al. (2025), we quantify this agreement through the effective number of answers under each strategy. Specifically, for strategy s_i, let $r_i(a) = \frac{f_i(a)}{\sum_{b \in \mathcal{A}(x)} f_i(b)}$ denote its normalized answer distribution. The corresponding order-2 Hill number is

$$H_i = \frac{1}{\sum_{a \in \mathcal{A}(x)} r_i(a)^2}. \tag{2}$$

A larger H_i indicates a more diverse answer distribution and weaker internal agreement. Therefore, we define the strategy weight as its reciprocal, $w_i = \frac{1}{H_i} = \sum_a r_i(a)^2$. This weight also equals the probability that two independent draws from r_i yield the same answer, providing a direct probabilistic interpretation of within-strategy agreement. Using these strategy weights, we obtain an agreement-weighted frequency matrix $\widetilde{\mathbf{F}} = \mathbf{W}\mathbf{F}$, where $\mathbf{W} = \mathrm{diag}(w_1, \ldots, w_K)$ assigns the corresponding agreement weight to each strategy. We compute the weighted average frequency of each answer as

$$R(a) = \frac{\sum_{i=1}^{K} w_i f_i(a)}{\sum_{i=1}^{K} w_i}. \tag{3}$$

$R(a)$ assigns greater weight to answer frequencies from strategies with stronger internal agreement. When all strategy weights are equal, $R(a)$ reduces to the aggregate frequency $\bar{f}(a) = \frac{1}{K}\sum_{i=1}^{K} f_i(a)$. More generally, the effect of strategy weighting relative to the aggregate frequency can be characterized as follows.

Proposition 1 (Agreement-Aware Frequency Adjustment). *Let* $\bar{w} = \frac{1}{K}\sum_{i=1}^{K} w_i$. *Then*

$$R(a) = \bar{f}(a) + \frac{\mathrm{Cov}_i(w_i, f_i(a))}{\bar{w}}, \tag{4}$$

where the covariance is taken under the uniform distribution over the K *strategies.*

Proposition 1 indicates that Hill weighting increases the score relative to $\bar{f}(a)$ when an answer occurs more frequently under strategies with stronger within-strategy agreement, and decreases it when the opposite holds.

Cross-Strategy Coverage. Answers with similar values of $R(a)$ can still exhibit different cross-strategy coverage. Therefore, we further characterize how broadly each answer is distributed across reasoning strategies. Following the same effective-number interpretation, we apply the order-2 Hill number to each column of the agreement-weighted matrix $\widetilde{\mathbf{F}}$. For answer a, we normalize its weighted frequencies across strategies as $q_i(a) = \frac{w_i f_i(a)}{\sum_{j=1}^{K} w_j f_j(a)}$, which forms a probability distribution over the K reasoning strategies. We define the normalized cross-strategy coverage as

$$E(a) = \frac{1}{K} H^{\mathrm{cross}}(a) = \frac{1}{K} \frac{1}{\sum_{i=1}^{K} q_i(a)^2}, \tag{5}$$

where $H^{\mathrm{cross}}(a)$ represents the effective number of reasoning strategies contributing to answer a, and the normalization by K bounds $E(a)$ within $[\frac{1}{K}, 1]$.

Pseudo-Label Selection. $R(a)$ and $E(a)$ capture complementary information from the strategy-level answer statistics. $R(a)$ serves as the primary frequency signal after accounting for within-strategy agreement, while $E(a)$ provides additional information about cross-strategy coverage. We incorporate coverage through a unit baseline so that broader coverage increases the score without allowing low coverage to overly suppress the frequency signal. Normalizing the resulting factor gives the Reliability-Hill (RH) score

$$F_{\mathrm{RH}}(a) = R(a)\frac{1 + E(a)}{2}. \tag{6}$$

Among candidates with equal weighted frequency, RH assigns a higher score to the answer with a larger effective number of contributing strategies. This distinguishes candidates that weighted frequency alone leaves unresolved. The calibration factor remains within $\left[\frac{K+1}{2K}, 1\right]$, limiting its influence on the frequency-based ranking. Finally, we select the answer with the highest RH score as the pseudo-label: $\hat{y} = \arg\max_{a \in \mathcal{A}(x)} F_{\mathrm{RH}}(a)$. The selected pseudo-label is then used to identify the reasoning trajectories retained for the strategy-specific context update.

3.2.2 Multi-Strategy Iterative Optimization

To enable iterative reasoning improvement across multiple rounds, we use in-context reinforcement learning. At each round, previous trajectories and their rewards are added to the context to guide subsequent reasoning. We use $F_{\mathrm{RH}}(\hat{y}^{(t)})$ as the reward. For each reasoning strategy s_i, we select one trajectory whose final answer is $\hat{y}^{(t)}$. To preserve the independence of different reasoning strategies, we maintain a separate context history for each strategy and add the selected trajectory only to its corresponding history. Following ME-ICPO, we summarize each selected trajectory with the LLM before adding it to the context to reduce context length.

4 Experiment

4.1 Experimental Setup

Benchmarks and Models. We evaluate SS-ICRL on three mathematical reasoning benchmarks: AIME 2024, AMC, and MATH-500 Li et al.; Hendrycks et al. (2021). Following Zuo et al. (2025), we further divide MATH-500 into five difficulty levels. We evaluate three backbone models of different scales: Qwen2.5-Math-7B Yang et al. (2024), Qwen3-4B and Qwen3-8B Yang et al. (2025).

Algorithm 1 Structured Self-Feedback In-Context Reinforcement Learning (SS-ICRL)

Require: Problem x; LLM π_θ; reasoning strategies $\mathcal{S} = \{s_i\}_{i=1}^{K}$; sampling budget B; number of rounds T

Ensure: Final answer $\hat{y}$

1: Initialize strategy-specific contexts $\mathcal{C}_i^{(0)} \leftarrow \emptyset$ for all $i \in [K]$
2: **for** $t = 1, \ldots, T$ **do**
3: Sample $n = B/K$ trajectories $\{y_{i,j}^{(t)}\}_{j=1}^{n}$ from $\pi_\theta(\cdot \mid x, s_i, \mathcal{C}_i^{(t-1)})$ and extract answers $\{a_{i,j}^{(t)}\}_{j=1}^{n}$ for all $i \in [K]$
4: Compute $F_{\mathrm{RH}}^{(t)}(a)$ and select $\hat{y}^{(t)} = \arg\max_a F_{\mathrm{RH}}^{(t)}(a)$
5: For each $i \in [K]$, select one trajectory $\tau_i^{(t)}$ with final answer $\hat{y}^{(t)}$ and summarize it as $\tilde{\tau}_i^{(t)}$
6: Update $\mathcal{C}_i^{(t)} \leftarrow \mathcal{C}_i^{(t-1)} \oplus \big(\tilde{y}_i^{(t)}, F_{\mathrm{RH}}^{(t)}(\hat{y}^{(t)})\big)$ for all $i \in [K]$
7: **end for**
8: Sample final trajectories from $\pi_\theta(\cdot \mid x, s_i, \mathcal{C}_i^{(T)})$ for all $i \in [K]$
9: **return** $\hat{a} = \arg\max_a F_{\mathrm{RH}}^{\mathrm{final}}(a)$

Reasoning Strategies. We use $K = 4$ fixed reasoning strategies shared across all models and benchmarks: forward reasoning, backward reasoning, representation transformation, and solution verification. The strategy set is designed to provide heterogeneous reasoning conditions while retaining sufficient repeated samples within each strategy for estimating strategy-level agreement. Under a fixed sampling budget, using more strategies increases cross-strategy information but leaves fewer samples for estimating each strategy-specific answer distribution. We therefore use four procedurally distinct strategies to balance cross-strategy diversity and within-strategy statistical support. The strategies are obtained by prompting GPT-5.6 once to propose general and procedurally distinct approaches to mathematical reasoning and are kept fixed throughout all experiments. Complete strategy prompts are provided in the appendix.

Baselines. We mainly compare SS-ICRL with ME-ICPO Yu et al. (2026) and two controlled reference settings:

- **Base** performs single-round inference using the same forward-reasoning prompt as Single-ICRL, without any in-context update. Its results are obtained from the initial responses of Single-ICRL, avoiding differences caused by using a separate Base prompt.
- **Single-ICRL** is a single-strategy majority-voting variant of SS-ICRL. It allocates the full sampling budget of $B = 32$ to the forward reasoning strategy, constructs pseudo-labels through majority voting, and uses the same multi-round context-update procedure as SS-ICRL. It serves as a controlled reference for evaluating the benefit of structured multi-strategy self-feedback under the same sampling budget and optimization settings.
- **ME-ICPO** constructs self-assessed rewards from majority-voting pseudo-labels and uses minimum-entropy to select a summarized response-reward pair for each context update. We follow its original settings while setting the same budget with our method.

Implementation Details. We use a total sampling budget of $B = 32$ responses per problem in each optimization round. With $K = 4$, SS-ICRL uniformly allocates $n = B/K = 8$ responses to each strategy. Keeping the total budget fixed ensures that gains from multi-strategy reasoning are not attributable to additional generation calls. We perform $T = 5$ rounds of in-context optimization. We use a temperature of 0.6, top-p of 0.95, a maximum context length of 8192 tokens, and a maximum generation length of 5000 tokens. All methods use the same decoding hyperparameters and final evaluation budget, while method-specific auxiliary procedures for ME-ICPO follow its original implementation. Following TTRL Zuo et al. (2025), we estimate pass@1 by averaging correctness over the $B = 32$ non-zero-temperature responses generated for each problem.

4.2 Main Results

We evaluate SS-ICRL against Base, Single-ICRL, and ME-ICPO on AIME 2024, AMC, and five MATH-500 difficulty levels using three backbone models. Table 1 summarizes the results.

Table 1: Main Pass@1 (%) results on mathematical reasoning benchmarks. The **best** and second-best results within each backbone are highlighted in bold and underlined, respectively. If a method's performance degrades significantly after its peak, we report the highest score observed across optimization steps and mark it with a dagger (†). Δ denotes the absolute improvement of SS-ICRL over the Base model.

Method	AIME24	AMC	MATH-L1	MATH-L2	MATH-L3	MATH-L4	MATH-L5	Avg.
Qwen2.5-Math-7B								
Base	18.85	48.83	85.95	72.78	69.23	60.42	41.91	56.85
ME-ICPO	25.84	51.34	86.16†	75.92†	**76.73**†	**68.47**	**50.79**	**62.18**
Single-ICRL	23.70	52.26	85.95†	73.47†	72.71†	64.04	48.62	60.11
SS-ICRL (ours)	**30.78**	**52.77**	**86.63**†	**76.39**†	74.40†	64.72	49.16	62.12
Δ	+11.93	+3.94	+0.68	+3.61	+5.17	+4.30	+7.25	+5.27
Qwen3-4B								
Base	10.31	37.54	89.03	83.23	77.89	72.41	46.76	59.60
ME-ICPO	16.67	41.57	90.12	86.04	78.57	73.64	51.10	62.53
Single-ICRL	20.99	47.25	90.62	85.49	83.63	80.32	57.88	66.60
SS-ICRL (ours)	**27.08**	**55.12**	**91.06**	**86.39**	**85.80**	**80.93**	**65.25**	**70.23**
Δ	+16.77	+17.58	+2.03	+3.16	+7.91	+8.52	+18.49	+10.64
Qwen3-8B								
Base	9.90	39.34	91.28	84.48	78.04	67.38	44.54	59.28
ME-ICPO	16.67	52.37	**95.28**	85.38	82.44	71.46	54.91	65.50
Single-ICRL	21.25	52.11	94.19	87.60	82.86	76.93	59.75	67.81
SS-ICRL (ours)	**33.44**	**58.73**	**95.28**	**90.07**	**86.55**	**80.27**	**61.94**	**72.33**
Δ	+23.54	+19.39	+4.00	+5.59	+8.51	+12.89	+17.40	+13.05

Consistent Gains Across Backbones and Benchmarks. SS-ICRL outperforms the Base model in all 21 model–task configurations. The gains are particularly pronounced on the more challenging AIME 2024 and MATH-L5 benchmarks. The largest improvement occurs on AIME 2024 with Qwen3-8B, where accuracy increases from 9.90% to 33.44%. These results demonstrate that strategy-structured self-feedback provides substantial gains across model scales and problem difficulty levels.

Superior Overall Performance Against ICRL Baselines. Compared with Single-ICRL, which uses a single reasoning strategy and majority voting, SS-ICRL achieves higher accuracy in all 21 configurations, demonstrating the effectiveness of constructing self-feedback from strategy-structured reasoning evidence. SS-ICRL also surpasses ME-ICPO in 17 of the 21 configurations. The two methods perform comparably on Qwen2.5-Math-7B, while SS-ICRL achieves relative average gains of 12.3% and 10.4% on Qwen3-4B and Qwen3-8B, respectively.

4.3 Does Structured Self-Feedback Improve Pseudo-Label Selection?

To decouple pseudo-label quality from subsequent context updates, we evaluate pseudo-label accuracy using responses from the initial sampling round on MATH-L5. We compare four variants under the same total sampling budget. Single-Strategy-MV uses only the forward reasoning strategy and selects the majority answer. Multi-Strategy-MV uniformly distributes the budget across four reasoning strategies and applies majority voting to the pooled responses. Multi-Strategy-R uses the same multi-strategy responses but selects pseudo-labels according to $R(a)$, which incorporates within-strategy agreement. Multi-Strategy-RH further incorporates cross-strategy coverage $E(a)$ and corresponds to the pseudo-label selection rule used by SS-ICRL.

As shown in Table 2, Multi-Strategy-MV improves the average pseudo-label accuracy from 58.88% to 62.46%, showing that reasoning under multiple strategies provides useful complementary evidence for pseudo-label selection. Incorporating within-strategy agreement through $R(a)$ further increases the average accuracy, with consistent improvements across all three models. The full RH score achieves the best performance on every backbone and raises the average accuracy to 63.52%, indicating that cross-strategy coverage provides additional information beyond within-strategy agreement. The advantage of RH becomes substantially more pronounced when it disagrees with majority voting. Among the 78 disagreement cases, RH correctly selects the pseudo-label in

Table 2: Pseudo-label selection accuracy (%) on MATH-L5 after the initial sampling round. We report the average results over three runs. Disagreement accuracy is computed on instances where Multi-Strategy-MV and the corresponding selector choose different pseudo-labels, with 81 disagreement cases for R and 78 for RH.

Method	Multi.	Within-Strategy Agreement	Cross-Strategy Coverage	Qwen2.5-Math-7B	Qwen3-4B	Qwen3-8B	Avg.	Disagreement Acc. MV → Method
Single-Strategy-MV	–	–	–	61.20 ± 1.29	56.97 ± 1.14	58.46 ± 1.55	58.88	–
Multi-Strategy-MV	✓	–	–	60.75 ± 0.85	63.93 ± 1.55	62.69 ± 0.75	62.46	–
Multi-Strategy-R	✓	✓	–	60.90 ± 0.85	64.68 ± 1.14	63.18 ± 1.14	62.92	$2.47 \rightarrow 9.88$
Multi-Strategy-RH	✓	✓	✓	$\mathbf{61.94 \pm 0.91}$	$\mathbf{64.93 \pm 1.50}$	$\mathbf{63.68 \pm 0.86}$	**63.52**	$\mathbf{1.28 \rightarrow 21.79}$

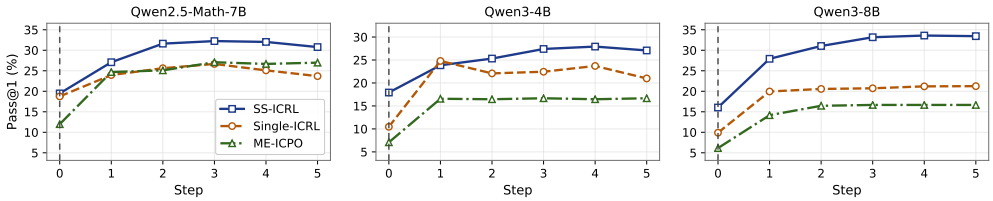

Figure 3: Pass@1 (%) at different optimization steps on AIME 2024. Step 0 denotes the initial generation without historical feedback, as marked by the gray dashed line.

21.79% of the instances, compared with only 1.28% for Multi-Strategy-MV. This suggests that strategy-structured feedback is particularly useful when aggregate answer frequency alone leads to an unreliable selection, allowing RH to recover better-supported pseudo-labels from the strategy-conditioned evidence.

4.4 Multi-Round Optimization Analysis

4.4.1 Performance Across Optimization Rounds

We further investigate how reasoning performance evolves across optimization steps. Figure 3 presents pass@1 results on AIME 2024, where step 0 corresponds to the initial generation stage without any historical feedback incorporated. All methods improve substantially after context updates. SS-ICRL maintains the strongest performance throughout the process. At step 0, SS-ICRL attains the highest accuracy for all three backbone models, which reflects the benefit of reasoning with multiple strategies. As optimization continues, the performance gap between SS-ICRL and other methods grows further, particularly for Qwen3-4B and Qwen3-8B. These steady performance gains demonstrate that self-feedback stored within the context can effectively guide subsequent reasoning. Single-ICRL achieves competitive results relative to ME-ICPO and outperforms it consistently on the Qwen3 backbones. This may stem from the explicit forward-reasoning strategy adopted in Single-ICRL, which itself enhances the quality of mathematical reasoning. We also observe that the improvement is not strictly monotonic. Several curves saturate or fluctuate at later optimization steps, indicating diminishing benefits from additional context updates. A more pronounced pattern is observed for Qwen2.5-Math-7B on some MATH-500 subsets, where extended optimization does not consistently improve upon the initial performance.

4.4.2 Effect of Feedback-Guided Context Organization

We analyze how different feedback-guided context organization strategies affect multi-round optimization. In SS-ICRL, self-generated feedback serves two key roles: it produces pseudo-labels at each optimization step, and governs which reasoning trajectories are preserved and integrated into contexts. Therefore, we compare three variants: (1) **w/o RH (MV)**, which replaces RH-based pseudo-label selection with majority voting; (2) **w/o Reward**, which removes the reward value from stored contexts and only keeps the summarized reasoning trajectories; and (3) **Random Trajectory**, which randomly selects trajectories for context updates instead of using feedback-based selection.

As shown in Figure 4, replacing RH with majority voting consistently degrades performance across different models, which demonstrates that the proposed RH feedback provides more effective guidance for multi-round optimization. Removing reward signals leads to performance degradation com-

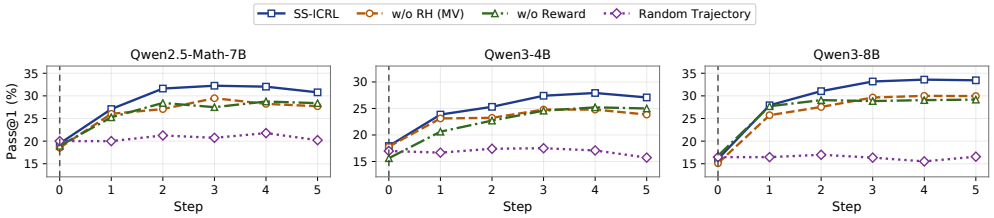

Figure 4: Ablation results of feedback-guided context organization in SS-ICRL on AIME 2024.

Table 3: Generalization of SS-ICRL to dynamically constructed reasoning strategies.

	AIME 2024			AMC		
Backbone	**Single-ICRL**	**SS-ICRL-Dy**	**SS-ICRL-Fixed**	**Single-ICRL**	**SS-ICRL-Dy**	**SS-ICRL-Fixed**
Qwen2.5-Math-7B	23.70	26.35	**30.78**	52.26	**55.35**	52.77
Qwen3-4B	20.99	22.71	**27.08**	47.25	46.91	**55.12**
Qwen3-8B	21.25	25.68	**33.44**	52.11	55.50	**58.73**
Avg.	21.98	24.91	**30.43**	50.54	52.59	**55.54**

pared with the full SS-ICRL, indicating that directly incorporating previously generated trajectories without feedback is insufficient for effective context optimization. Instead, associating trajectories with reward signals enables reward-guided experience refinement through context updates, leading to improved subsequent reasoning. The Random Trajectory variant further reveals the importance of experience selection in context optimization. Without feedback-guided selection, randomly incorporated trajectories provide less effective guidance for future reasoning. Overall, these results suggest that effective multi-round optimization comes from progressively refining accumulated reasoning experiences through feedback-guided context updates. By leveraging strategy-structured self-feedback, SS-ICRL continuously improve future reasoning without external verification or parameter updates.

4.5 How Does Strategy Construction Affect SS-ICRL?

Our main experiments use a predefined set of four reasoning strategies, which leaves open whether the gains of SS-ICRL depend on this particular strategy configuration. To examine this question, we construct a dynamic variant termed SS-ICRL-Dy, where the model automatically generates four distinct reasoning strategies for each problem before trajectory sampling.

As shown in Table 3, SS-ICRL-Dy outperforms the Single-ICRL baseline across five out of six experimental settings, improving the average accuracy on both AIME 2024 ($21.98 \rightarrow 24.91$) and AMC ($50.54 \rightarrow 52.59$). These results show that the gains of SS-ICRL are not limited to the predefined reasoning prompts and can also extend to model-generated strategies. Nevertheless, the proposed SS-ICRL-Fixed still achieves stronger overall performance than SS-ICRL-Dy. One possible explanation is that model-generated strategies may partially overlap and provide insufficiently differentiated reasoning directions, resulting in weaker multi-strategy information for pseudo-label construction. Notably, SS-ICRL-Dy surpasses SS-ICRL-Fixed on the AMC dataset with Qwen2.5-Math-7B, highlighting the potential of automatic strategy construction. Automatically mining high-quality, differentiated reasoning strategies can further unleash the potential of SS-ICRL, which we regard as a valuable future work.

5 Conclusion

In this paper, we introduce structured self-feedback in-context reinforcement learning (SS-ICRL), a parameter-free framework for unsupervised inference-time self-improvement of LLM reasoning. Rather than relying on aggregate answer frequency, SS-ICRL constructs self-feedback from both within-strategy agreement and cross-strategy support, providing a more reliable signal for iterative in-context optimization. Experiments across mathematical reasoning benchmarks and different backbone models consistently demonstrate its effectiveness. Our analyses further show that reliable self-improvement depends not only on generating additional reasoning experience, but also on exploiting strategy-level structure to identify and progressively refine useful experience across optimization rounds. These findings highlight structured self-generated experience as a promising basis for parameter-free adaptation at inference time.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, LaTeX formatting. All AI-assisted outputs were reviewed and verified by the authors. All LLM-assisted text was reviewed by the authors, and all generated code was inspected and tested for correctness. The authors take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We affirm that all authors have read and adhere to the ICLR Code of Ethics. Our work uses only publicly available benchmarks and pretrained models and does not involve human or animal subjects, sensitive personal data, or identifiable privacy risks.

REPRODUCIBILITY STATEMENT

We provide detailed experimental settings, implementation details, and prompts in Appendix C. All datasets and model checkpoints used in this work are publicly available. The source code, configurations, and experiment scripts are available in the anonymous repository: `https://anonymous.4open.science/r/SS-ICRL/ss_icrl_anonymous`.

REFERENCES

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948, 2025. doi: 10.48550/ARXIV.2501.12948.

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. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/be83ab3ecd0db773eb2dc1b0a17836a1-Abstract-round2.html.

Audrey Huang, Adam Block, Qinghua Liu, Nan Jiang, Akshay Krishnamurthy, and Dylan J. Foster. Is best-of-n the best of them? coverage, scaling, and optimality in inference-time alignment. In *International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025.

Wenxuan Jiang, Yuxin Zuo, Zijian Zhang, Xuecheng Wu, Zining Fan, Wenxuan Liu, Li Chen, Xiaoyu Li, Xuezhi Cao, Xiaolong Jin, and Ninghao Liu. TR-ICRL: test-time rethinking for in-context reinforcement learning. *CoRR*, abs/2604.00438, 2026. doi: 10.48550/ARXIV.2604.00438. URL https://doi.org/10.48550/arXiv.2604.00438.

Akshay Krishnamurthy, Keegan Harris, Dylan J. Foster, Cyril Zhang, and Aleksandrs Slivkins. Can large language models explore in-context? In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/d951f73c521d069fefbb73396df01424-Abstract-Conference.html.

Michael Laskin, Luyu Wang, Junhyuk Oh, Emilio Parisotto, Stephen Spencer, Richie Steigerwald, DJ Strouse, Steven Stenberg Hansen, Angelos Filos, Ethan Brooks, Maxime Gazeau, Himanshu Sahni, Satinder Singh, and Volodymyr Mnih. In-context reinforcement learning with algorithm distillation. In *International Conference on Learning Representations*. OpenReview.net, 2023. URL https://openreview.net/forum?id=hy0a5MMPUv.

Jonathan Lee, Annie Xie, Aldo Pacchiano, Yash Chandak, Chelsea Finn, Ofir Nachum, and Emma Brunskill. Supervised pretraining can learn in-context reinforcement learning. In *Neural Information Processing Systems*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/8644b61a9bc87bf7844750a015feb600-Abstract-Conference.html.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Long Yu, Albert Qiaochu Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, Stanislas Polu, Hugging Face, and A. Mistral. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. URL https://api.semanticscholar.org/CorpusID:286088415.

Zilong Liu, Krzysztof Janowicz, Ivan Majic, Meilin Shi, Alexandra Fortacz, Mina Karimi, Gengchen Mai, and Kitty Currier. Operationalizing geographic diversity for the evaluation of ai-generated content. *Trans. GIS*, 29(3), 2025. doi: 10.1111/TGIS.70057.

Amir Moeini, Jiuqi Wang, Jacob Beck, Ethan Blaser, Shimon Whiteson, Rohan Chandra, and Shangtong Zhang. A survey of in-context reinforcement learning. *CoRR*, abs/2502.07978, 2025. doi: 10.48550/ARXIV.2502.07978. URL https://doi.org/10.48550/arXiv.2502.07978.

OpenAI. Openai o1 system card. *CoRR*, abs/2412.16720, 2024. doi: 10.48550/ARXIV.2412.16720.

Mihir Prabhudesai, Lili Chen, Alex Ippoliti, Katerina Fragkiadaki, Hao Liu, and Deepak Pathak. Maximizing confidence alone improves reasoning. *Trans. Mach. Learn. Res.*, 2026, 2026. URL https://openreview.net/forum?id=gInznr8EsQ.

Archiki Prasad, Weizhe Yuan, Richard Yuanzhe Pang, Jing Xu, Maryam Fazel-Zarandi, Mohit Bansal, Sainbayar Sukhbaatar, Jason E. Weston, and Jane Yu. Self-consistency preference optimization. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/prasad25a.html.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *CoRR*, abs/2408.03314, 2024. doi: 10.48550/ARXIV.2408.03314. URL https://doi.org/10.48550/arXiv.2408.03314.

Kefan Song, Amir Moeini, Peng Wang, Lei Gong, Rohan Chandra, Shangtong Zhang, and Yanjun Qi. Reward is enough: Llms are in-context reinforcement learners. In *International Conference on Learning Representations*, volume 2026, pp. 112747–112770, 2026.

Jiuqi Wang, Ethan Blaser, Hadi Daneshmand, and Shangtong Zhang. Transformers can learn temporal difference methods for in-context reinforcement learning. In *International Conference on Learning Representations*. OpenReview.net, 2025a. URL https://openreview.net/forum?id=Pj06mxCXPl.

Jiuqi Wang, Rohan Chandra, and Shangtong Zhang. Towards provable emergence of in-context reinforcement learning. In *Neural Information Processing Systems*, 2025b.

Ru Wang, Wei Huang, Qi Cao, Yusuke Iwasawa, Yutaka Matsuo, and Jiaxian Guo. Self-harmony: Learning to harmonize self-supervision and self-play in test-time reinforcement learning. *CoRR*, abs/2511.01191, 2025c. doi: 10.48550/ARXIV.2511.01191. URL https://doi.org/10.48550/arXiv.2511.01191.

Weiqin Wang, Yile Wang, Kehao Chen, and Hui Huang. Beyond majority voting: Towards fine-grained and more reliable reward signal for test-time reinforcement learning. In *Association for Computational Linguistics*, pp. 37251–37265. Association for Computational Linguistics, 2026. doi: 10.18653/V1/2026.ACL-LONG.1727. URL https://doi.org/10.18653/v1/2026.acl-long.1727.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*. OpenReview.net, 2023.

Sean Welleck, Amanda Bertsch, Matthew Finlayson, Hailey Schoelkopf, Alex Xie, Graham Neubig, Ilia Kulikov, and Zaïd Harchaoui. From decoding to meta-generation: Inference-time algorithms for large language models. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://openreview.net/forum?id=eskQMcIbMS.

Kunbin Xu, Xingzuo Li, Xuefeng Bai, and Kehai Chen. Hi-ttrl: Regulating consensus with hints for test-time reinforcement learning. *CoRR*, abs/2608.03545, 2026. doi: 10.48550/ARXIV.2608.03545. URL https://doi.org/10.48550/arXiv.2608.03545.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *CoRR*, abs/2409.12122, 2024. doi: 10.48550/ARXIV.2409.12122. URL https://doi.org/10.48550/arXiv.2409.12122.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Yaoqi Ye, Yiran Zhao, Keyu Duan, Zeyu Zheng, Kenji Kawaguchi, Cihang Xie, and Michael Qizhe Shieh. In-context reinforcement learning for tool use in large language models. *CoRR*, abs/2603.08068, 2026. doi: 10.48550/ARXIV.2603.08068.

Tianrun Yu, Yuxiao Yang, Zhaoyang Wang, Kaixiang Zhao, Porter Jenkins, Xuchao Zhang, Chetan Bansal, Huaxiu Yao, and Weitong Zhang. Provable and practical in-context policy optimization for self-improvement. *CoRR*, abs/2603.01335, 2026. doi: 10.48550/ARXIV.2603.01335.

Di Zhang, Xiaoshui Huang, Dongzhan Zhou, Yuqiang Li, and Wanli Ouyang. Accessing GPT-4 level mathematical olympiad solutions via monte carlo tree self-refine with llama-3 8b. *CoRR*, abs/2406.07394, 2024. doi: 10.48550/ARXIV.2406.07394.

Qingyang Zhang, Haitao Wu, Changqing Zhang, Peilin Zhao, and Yatao Bian. Right question is already half the answer: Fully unsupervised LLM reasoning incentivization. In *Neural Information Processing Systems*, 2025a. URL http://papers.nips.cc/paper_files/paper/2025/hash/614f8eba720cfc7ff00274bd64fb0a3f-Abstract-Conference.html.

Zizhuo Zhang, Jianing Zhu, Xinmu Ge, Zihua Zhao, Zhanke Zhou, Xuan Li, Xiao Feng, Jiangchao Yao, and Bo Han. Co-reward: Self-supervised reinforcement learning for large language model reasoning via contrastive agreement. *CoRR*, abs/2508.00410, 2025b. doi: 10.48550/ARXIV.2508.00410. URL https://doi.org/10.48550/arXiv.2508.00410.

Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards. *CoRR*, abs/2505.19590, 2025. doi: 10.48550/ARXIV.2505.19590.

Yuxin Zuo, Kaiyan Zhang, Li Sheng, Shang Qu, Ganqu Cui, Xuekai Zhu, Haozhan Li, Yuchen Zhang, Xinwei Long, Ermo Hua, Biqing Qi, Youbang Sun, Zhiyuan Ma, Lifan Yuan, Ning Ding, and Bowen Zhou. TTRL: test-time reinforcement learning. In *Neural Information Processing Systems*, 2025.

A Further Comparison with Related Test-Time Self-Feedback Methods

Confidence- and Uncertainty-Based Self-Feedback. Recent test-time optimization methods have explored alternatives to majority voting by deriving supervision from the model's own predictive uncertainty. Intuitor Zhao et al. (2025) uses self-certainty as an intrinsic reward, while RENT Prabhudesai et al. (2026) constructs rewards from generation entropy. EMPO Zhang et al. (2025a) further measures uncertainty over semantic response clusters rather than individual token distributions. These methods derive feedback from the uncertainty of the model's prediction distribution, either at the token level or semantic response level. In contrast, SS-ICRL derives feedback from how candidate answers are supported across different reasoning strategies.

Cross-View Pseudo-Label Refinement. Another line of work improves pseudo-label quality by comparing predictions across multiple views. Co-Reward Zhang et al. (2025b) constructs supervision from agreement across semantically related problems. Self-Harmony Wang et al. (2025c) instead compares the original problem with a semantically equivalent reformulation and combines answer frequencies across the two views using a harmonic mean. These methods exploit consistency across different input views. SS-ICRL keeps the problem unchanged and varies the reasoning procedure. It uses within-strategy agreement to weight answer support and cross-strategy coverage to refine the resulting score. Therefore, the structured evidence in SS-ICRL reflects how an answer is supported across reasoning procedures rather than how invariant it is to input reformulation.

In-Context Reinforcement Learning. ME-ICPO Yu et al. (2026) performs multi-round in-context optimization by accumulating selected responses and rewards in the context, with entropy-based response selection used to determine which experience is retained. TR-ICRL Jiang et al. (2026) retrieves related unlabeled problems, constructs pseudo-labels for the retrieved examples through majority voting, and incorporates their predictions and feedback into the context. SS-ICRL differs from both approaches in how self-generated experience is structured and reused. It constructs feedback from multiple reasoning strategies applied to the same test problem and maintains a separate context history for each strategy. Thus, the reasoning-strategy structure is used not only to evaluate candidate answers but also to organize the experience retained across optimization rounds.

B Proof of Proposition 1

For an answer a, let

$$R(a) = \frac{\sum_{i=1}^{K} w_i f_i(a)}{\sum_{i=1}^{K} w_i}, \qquad \bar{f}(a) = \frac{1}{K}\sum_{i=1}^{K} f_i(a), \qquad \bar{w} = \frac{1}{K}\sum_{i=1}^{K} w_i.$$

Then

$$R(a) = \bar{f}(a) + \frac{\mathrm{Cov}_i(w_i, f_i(a))}{\bar{w}}, \tag{7}$$

where the covariance is computed under the uniform distribution over the K reasoning strategies.

Proof. Under the uniform distribution over strategies,

$$\mathrm{Cov}_i(w_i, f_i(a)) = \frac{1}{K}\sum_{i=1}^{K} w_i f_i(a) - \bar{w}\,\bar{f}(a). \tag{8}$$

Since $\sum_{i=1}^{K} w_i = K\bar{w}$, we have

$$\begin{aligned} R(a) &= \frac{\frac{1}{K}\sum_{i=1}^{K} w_i f_i(a)}{\bar{w}} \\ &= \frac{\bar{w}\,\bar{f}(a) + \mathrm{Cov}_i(w_i, f_i(a))}{\bar{w}} \\ &= \bar{f}(a) + \frac{\mathrm{Cov}_i(w_i, f_i(a))}{\bar{w}}. \end{aligned} \tag{9}$$

By construction, $w_i = \sum_{a'} r_i(a')^2 > 0$, and therefore $\bar{w} > 0$. Consequently, the sign of $R(a) - \bar{f}(a)$ is determined by $\mathrm{Cov}_i(w_i, f_i(a))$. Answers receiving greater support from strategies with stronger within-strategy agreement are up-weighted relative to their unweighted frequency, whereas answers supported primarily by less consistent strategies are down-weighted. □

C Additional Experimental Details

C.1 Implementation and Evaluation

Experimental configuration. Table 4 summarizes the main configuration. We perform $T = 5$ context-update steps. At each step, SS-ICRL samples $B = 32$ responses per problem, uniformly allocating eight responses to each of four reasoning strategies. After the fifth update, an independent set of 32 responses is generated for final evaluation. Qwen2.5-Math-7B uses raw prompts. Qwen3-4B and Qwen3-8B use their chat templates with thinking mode disabled. ME-ICPO uses the same primary response budget and retains its method-specific entropy-lookahead procedure, which draws 16 additional samples.

Table 4: Hyperparameters used in the main experiments.

Hyperparameter	**Value**
Reasoning strategies K	4
Responses per step B	32
Responses per strategy	8
Context-update steps T	5
Final evaluation responses	32
Sampling temperature	0.6
Top-p	0.95
Runner context budget	8,192 tokens
Maximum response length	5,000 tokens
Maximum summary length	100 tokens
Summary temperature	0.05
Summary top-p	0.9
Random seed	42

Answer extraction and metrics. Answer extraction is restricted to the current assistant response; text following a simulated subsequent user or system turn is discarded. When a complete `boxed{...}` expression is present, we use the first one and apply the benchmark-specific answer normalizer. If no complete boxed expression is present, the benchmark-specific parser determines whether an answer can still be recovered. Invalid responses consume the allocated sampling budget.

We estimate pass@1 by averaging correctness over the 32 non-zero-temperature responses generated for each problem. Step 0 denotes generation without historical feedback, and step t denotes generation after t context updates. Main results use the independent evaluation after $T = 5$ updates.

In the initial-round pseudo-label analysis, Multi-Strategy-MV, Multi-Strategy-R, and Multi-Strategy-RH are evaluated on the same generated responses. For the reported pseudo-label accuracy, invalid answers participate in candidate selection and a tie for the highest score is counted as incorrect. During context updates, the operational selector uses valid candidates and resolves ties with its seeded random generator.

C.2 Reasoning Strategy Prompts

Fixed strategies. The main experiments use the following four strategy instructions. The set was generated once by GPT-5.6 and kept fixed across backbone models, benchmarks, and test problems. For each response, the assigned strategy instruction is placed before the problem statement.

Fixed Reasoning Strategy Instructions

forward: Start from the given conditions and construct the solution forward toward the target. Derive each necessary intermediate result from information already established.

backward: Start from the target quantity or statement and reason backward about what conditions or intermediate results would be sufficient to establish it. Then connect these requirements back to the given conditions.

transform: Before solving, transform the problem into an alternative mathematical representation, such as equations, coordinates, a graph, a recurrence, a table, or an equivalent problem. Carry out the main derivation in this transformed representation.

verify: First form a plausible candidate result or key claim. Actively test it using boundary cases, small cases, alternative possibilities, or consistency checks. Revise it if necessary, then justify the final result.

Dynamic strategies. SS-ICRL-Dy asks the evaluated backbone model to generate four problem-specific strategies before sampling responses. The strategy-generation prompt uses the system instruction "You design diverse reasoning plans. Follow the requested output format exactly and do not solve the problem." The user instruction is:

Dynamic Strategy Generation Instruction

Generate exactly four short and mutually distinct plans for solving the problem.
Requirements:

- Each plan must be one concise imperative sentence that describes a reasoning approaches only.
- Do not solve the problem, perform calculations, derive intermediate results, or reveal an answer.
- Do not rank, compare, recommend, or select among the plans.
- Avoid plans that differ only in wording.
- Give all plans comparable specificity and equal priority.
- Return exactly four `<plan>` elements inside `<plans>`, with no other text.

The problem statement and a four-plan output template are appended to this instruction. Plans are generated once per problem and reused across its optimization steps. Generation is retried up to three times if the required output format cannot be parsed. All four parsed plans receive equal sampling budgets; none is ranked or selected before response generation.

C.3 Feedback Prompt and Context Format

System prompts. AIME 2024 and AMC use the same system prompt. MATH uses a different answer-format instruction. The following text is used before any historical feedback is available.

AIME 2024 and AMC System Prompt

You are an AI mathematician. All content you output MUST be in English.
You are only allowed to provide explanations in plain English. Do NOT write any code, pseudocode, or technical snippets.
Finish all your reasoning, then on a NEW line output exactly one number (the answer) and nothing else.
Your final output MUST be in the format `boxed{<number>}`, where `<number>` is the final numeric answer only (no expressions, variables, or additional text). The content inside `boxed{ }` must be a decimal number, not a fraction or any other form.

MATH System Prompt

You are an AI mathematician. All content you output MUST be in English.
You are only allowed to provide explanations in plain English. Do NOT write any code, pseudocode, or technical snippets.
Finish all your reasoning, then on a NEW line output exactly one answer (the answer) and nothing else.
Your final output MUST be in the format `boxed{<answer>}`, where `<answer>` is the final answer in any form (number, fraction, expression, or text), without extra explanation or text.

Once feedback has been accumulated, the system prompt additionally explains its meaning: prior evidence consists of compressed solution ideas from previous test-time attempts and is reference data rather than dialogue. Each idea carries a score between 0 and 1, with higher scores described as indicating stronger "multi-view support." In the implemented prompt, a view corresponds to one of our reasoning strategies.

Trajectory summaries. The same backbone model summarizes selected trajectories with temperature 0.05, top-p 0.9, and a maximum of 500 generated tokens. The summary instruction requests no more than 100 tokens. The trajectory is inserted between `[Answer start]` and `[Answer end]`.

AIME 2024 and AMC Summary Prompt

Provide a concise summary of the reasoning in the answer below. Do NOT add any introductory phrases or extra explanations. Omit all numerical calculations. The summary must be self-contained, no more than 100 tokens.
If there is a final numeric result, include it at the end in the format `boxed{<number>}` (decimal only, no fractions, variables, or extra text). If there is no numeric answer, do not output `boxed{}`.

```
[Answer start]
<selected trajectory>
[Answer end]
Summary:
```

MATH Summary Prompt

Provide a concise summary of the reasoning in the answer below. Do NOT add any introductory phrases or extra explanations. Omit all calculations unless essential to understanding. The summary must be self-contained, no more than 100 tokens.
If there is a final answer, include it at the end in the format `boxed{<answer>}` (any form is allowed: number, fraction, expression, or text). If there is no final answer, do not output `boxed{}`.

```
[Answer start]
<selected trajectory>
[Answer end]
Summary:
```

Stored feedback. For each strategy, SS-ICRL selects the first generated trajectory whose final answer matches the global pseudo-label. If that strategy has no matching trajectory, its context remains unchanged at the current step. The selected trajectory is summarized, paired with its answer and RH score, and stored in the corresponding strategy's context. Its format is:

```
<prior_evidence>
<trajectory_evidence value="a" score="s">
trajectory summary
</trajectory_evidence>
</prior_evidence>
```

Here, `a` denotes the selected answer and `s` its RH score, formatted to four decimal places. At the next step, each strategy receives its own accumulated evidence.

D More Results

D.1 Additional Analysis of Pseudo-Label Selection

We provide additional analysis of the pseudo-label selection ablations reported in Table 2. Figure 5 summarizes the average accuracy across the three backbone models. Multi-Strategy-RH denotes the complete pseudo-label selector used in SS-ICRL, while the remaining variants ablate different components of strategy-structured self-feedback.

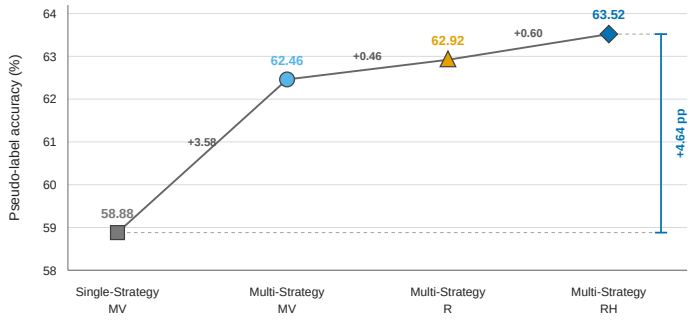

Figure 5: Average pseudo-label accuracy of SS-ICRL and its pseudo-label selection ablations on MATH-L5. Multi-Strategy-RH is the complete selector used in SS-ICRL, while the other variants ablate multi-strategy reasoning and the within- and cross-strategy information used for structured self-feedback.

As shown in Figure 5, the complete RH selector achieves the highest average pseudo-label accuracy of 63.52%. Removing cross-strategy coverage reduces the accuracy to 62.92%, while further removing within-strategy agreement and reverting to majority voting reduces it to 62.46%. The larger drop to 58.88% under Single-Strategy-MV further highlights the importance of collecting evidence from multiple reasoning strategies. These ablations show that multi-strategy reasoning provides the basis for structured self-feedback, while within-strategy agreement and cross-strategy coverage provide complementary information for improving pseudo-label selection.

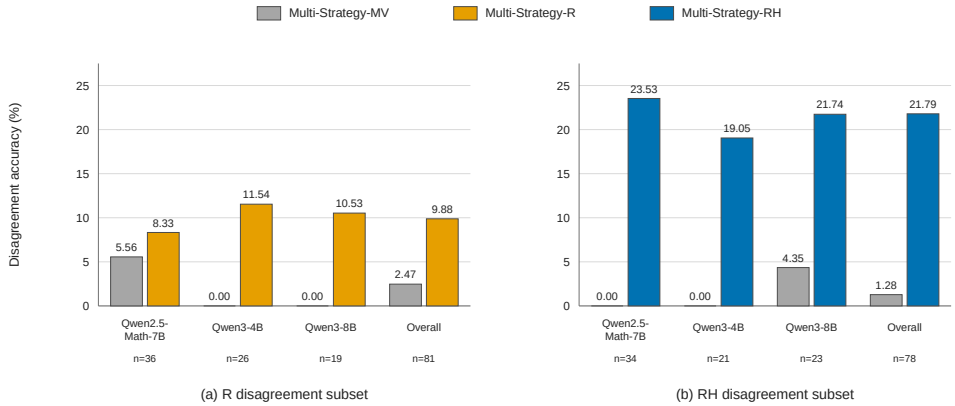

Figure 6: Pseudo-label accuracy on disagreement cases between Multi-Strategy-MV and the corresponding strategy-structured selector. The left and right panels show results for R and RH, respectively, where n denotes the number of disagreement instances.

D.2 Disagreement Analysis of Pseudo-Label Selection

Aggregate pseudo-label accuracy does not reveal whether the decisions changed by strategy-structured scoring are actually beneficial. Therefore, we further focus on instances where Multi-Strategy-MV and the corresponding selector produce different pseudo-labels. Separate disagreement subsets are constructed for R and RH, since the two selectors may overturn different majority-voting decisions.

As shown in Figure 6, the decisions changed by strategy-structured scoring are substantially more accurate than the corresponding majority-voting decisions. On the 81 disagreement cases for R, pseudo-label accuracy increases from 2.47% with Multi-Strategy-MV to 9.88% with R. The difference is substantially larger for RH: among 78 disagreement cases, RH achieves 21.79% accuracy, compared with only 1.28% for Multi-Strategy-MV. This pattern is consistent across all three backbone models, where RH obtains disagreement accuracy between 19.05% and 23.53%, whereas the corresponding majority-voting accuracy ranges from 0.00% to 4.35%.

These results provide a more direct view of where strategy-structured self-feedback is most useful. RH does not merely improve aggregate pseudo-label accuracy; when its ranking differs from raw majority voting, the resulting decisions are substantially more reliable. This indicates that within-strategy agreement and cross-strategy coverage provide useful evidence precisely in cases where aggregate answer frequency is insufficient for reliable pseudo-label selection.