Think Outside the Policy: In-Context Steered Policy Optimization

Anonymous ACL submission

Abstract

Existing Reinforcement Learning from Verifiable Rewards (RLVR) methods, such as Group Relative Policy Optimization (GRPO), have achieved remarkable progress in improving the reasoning capabilities of Large Reasoning Models (LRMs). However, they exhibit limited exploration due to reliance on on-policy rollouts which are confined to the current policy's distribution, resulting in narrow trajectory diversity. Recent approaches attempt to expand policy coverage by incorporating trajectories generated from stronger expert models, yet this reliance increases computational cost and such advanced models are often inaccessible. To address these issues, we propose In-Context Steered Policy Optimization (ICPO), a unified framework that leverages the inherent in-context learning capability of LRMs to provide expert guidance using existing datasets. ICPO introduces mixed-policy GRPO with implicit expert forcing, which expands exploration beyond the current policy distribution without requiring advanced LRM trajectories. To further stabilize optimization, ICPO integrates expert region reject sampling to filter unreliable off-policy trajectories and annealed expert-bonus reward shaping to balance early expert guidance with later autonomous improvement. Results demonstrate that ICPO consistently enhances RLVR performance and training stability on mathematical reasoning benchmarks, revealing a scalable and effective RLVR paradigm for LRMs. Our code is available at `https://anonymous.4open.science/r/ICPO`.

1 Introduction

Large Reasoning Models (LRMs) excel at solving complex mathematical problems, and Reinforcement Learning from Verifiable Rewards (RLVR) provides a scalable way to refine their reasoning through verifiable rewards (DeepSeek-AI, 2025). Yet, limited exploration under standard Group Relative Policy Optimization (GRPO) (Shao et al., 2024) often hinders robust reasoning improvement (Yue et al., 2025; Zhang et al., 2025a).

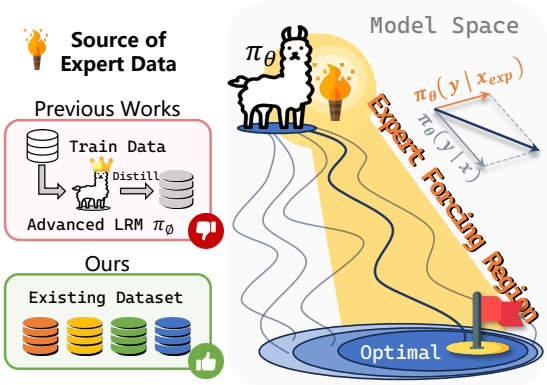


Figure 1: Illustration of optimization dynamics in parameter space. GRPO exploration is confined to the current policy's distribution, limiting trajectory diversity and often leading to suboptimal convergence. While prior methods expand exploration by incorporating expert rollouts generated by stronger LRMs on the training data, ICPO leverages existing datasets as in-context guidance, eliminating reliance on advanced LRMs.

Recent work has explored combining Supervised Fine-Tuning (SFT) with Reinforcement Learning (RL) to strengthen the exploration of LRMs. One research direction interleaves SFT and RL updates (Ma et al., 2025), enabling SFT to improve high-difficulty problem-solving while RL refines mid- and low-difficulty behaviors. However, repeated switching between paradigms introduces instability and inefficient convergence. Another line of work seeks to unify SFT and RL within a single training process through different strategies: incorporating SFT data as off-policy rollouts during RL to expand the exploration space (Yan et al., 2025); jointly optimizing SFT and RL objectives for tighter integration (Fu et al., 2025); and leveraging hints to bootstrap rollouts to improve performance on harder prompts (Liu et al., 2025a; Zhang

et al., 2025b; Fu et al., 2025).

Together, these approaches reflect a growing consensus that combining SFT and RL can effectively expand the exploration of policy search into more promising reasoning spaces. However, as shown in Figure 1, three key challenges remain: (1) Online exploration remains confined to the current policy distribution, as GRPO-based methods rely on on-policy sampling, resulting in limited trajectory diversity and might converge to local optima. (2) Expanding the exploration space with trajectories from stronger LRMs incurs high computational cost and limited accessibility, since generating additional reasoning traces from advanced models is expensive and such models are not always available. (3) External trajectories are often noisy and unstable for training, incorporating them indiscriminately as off-policy rollouts can mislead policy updates and harm convergence stability.

To address these limitations, we propose ***I**n-**C**ontext Steered **P**olicy **O**ptimization* (ICPO), a novel RLVR framework that exploits the LRM's inherent In-Context Learning (ICL) capability to provide expert guidance instead of relying on external advanced LRMs. Specifically, (1) ICPO introduces *mixed-policy GRPO with Implicit Expert Forcing* (IEF), where expert-conditioned rollouts are generated through ICL guidance, enabling exploration beyond the current policy distribution and steering the model toward expert-aligned regions of the solution space. (2) To ensure reliable guidance, ICPO employs *Expert Region Reject Sampling* (ERRS), which filters out noisy or low-quality off-policy trajectories using verifiable reward signals, preventing misleading gradients from contaminating policy updates. (3) ICPO further explores an annealed expert bonus into the *Reward Shaping* (RS) design, enforcing strong expert-guided shaping in the early stage and progressively relaxing it to facilitate autonomous optimization as LRM capabilities grow.

Our experiments show that ICL-based rollouts exhibit superior quality and diversity, making them effective sources of expert guidance (§ 3). Extensive evaluations across multiple mathematical reasoning benchmarks further demonstrate that ICPO achieves substantial performance gains over vanilla GRPO and mixed-policy GRPO (Yan et al., 2025), with maximum average improvements of up to **+4.1** and **+4.0** points, respectively (§ 6). In addition, we provide in-depth instance- and token-level analyses that offer strong evidence of distributional shifts induced by ICPO (§ 7).

Our contributions are three-fold:

- We empirically show that ICL rollouts provide diverse and high-quality expert signals for the mathematical reasoning task.
- We introduce ICPO, which includes *mixed-policy GRPO with Implicit Expert Forcing*, leveraging the model's inherent ICL ability without relying on stronger LRMs, together with *Expert Region Reject Sampling* and *Reward Shaping* for stable and efficient learning.
- We empirically validate that ICPO delivers consistent improvements on mathematical reasoning benchmarks across model scales and effectively encourages exploration, demonstrating strong potential for LRM post-training.

2 Related Work

Exploration from Within the Policy. Recent methods enhance exploration by increasing diversity within the policy itself, such as generating additional rollouts (Wang et al., 2025; Hu et al., 2025a; Tang et al., 2025), leveraging replay buffers to revisit informative prompts (Dou et al., 2025), and sampling under higher temperatures to boost stochasticity (Chen et al., 2025; Zhang et al., 2024). These approaches broaden trajectory coverage under the model's own distribution. However, their exploration remains bounded by the intrinsic capacity of the current policy.

Exploration from Outside the Policy. Other studies leverage external supervision to guide exploration beyond the model's default policy distribution. One line of work explores hybrid SFT–RL strategies to expand the reasoning space of LRMs. ReLIFT (Ma et al., 2025) alternates between RL and SFT by updating on failed rollouts. LUFFY (Yan et al., 2025) incorporates SFT trajectories as off-policy samples using importance sampling. SRFT (Fu et al., 2025) jointly optimizes SFT and RL objectives with an entropy-based weight on the SFT loss. Another line of work guides rollouts by concatenating partial SFT solutions as hints (Liu et al., 2025a; Zhang et al., 2025b; Huang et al., 2025). However, these methods rely on advanced LRMs to supply SFT traces, which may incur additional computation overhead.

In contrast, we exploit LRMs' inherent ICL ability to steer diverse rollouts with existing datasets as

demonstrations, requiring neither external LRMs' trajectories nor explicitly engineered hints.

3 Preliminary

Explicit Expert Forcing. In traditional RL and RLHF, *expert forcing* explicitly constrains the policy to align with an expert policy π_ϕ, typically through imitation or KL-based regularization (Hester et al., 2018; Haldar et al., 2023; Zhang et al., 2023; Hu et al., 2023). This explicit constraint stabilizes optimization and reduces reward variance, but it requires gradient-based imitation and access to an auxiliary expert model (e.g., a larger LRM), which can limit exploration in later stages.

ICL as Expert-Conditioned Inference. ICL provides a gradient-free way to inject expert priors through the input context, which we evaluate from three complementary perspectives. (1) *Accuracy*: the 1-shot ICL setting consistently outperforms the 0-shot baseline (Figure 2), indicating that conditioning on demonstrations improves reasoning correctness. (2) *Diversity*: compared with temperature-based sampling, introducing 1-shot demonstrations expands the sampling space (Figure 3), yielding larger inter-trajectory edit distances and enhanced exploratory diversity. (3) *Distribution quality*: under ICL-conditioned rollouts, the output distribution becomes more favorable—a higher proportion of previously incorrect generations are "flipped" to correct solutions compared with temperature perturbations (Figure 3), indicating that in-context steering provides a stronger and more targeted exploration signal. Taken together, these results support our view that ICL constitutes an effective expert-conditioned inference process.

From ICL to Implicit Expert Forcing. Given expert demonstrations $\mathcal{D}$ and a query q, the model generates trajectories conditioned on:

$$x_{\text{exp}} = [\mathcal{D}; q], \quad \tau_{\text{exp}} \sim \pi_\theta(\tau \mid x_{\text{exp}}). \tag{1}$$

Following the *hypothesis-class* view of ICL (Hendel et al., 2023), the forward process of a Transformer T can be decomposed into two functions:

$$T([\mathcal{D}, q]) = \mathcal{F}(q; A(\mathcal{D})), \tag{2}$$

where $A(\cdot)$ maps demonstrations $\mathcal{D}$ to a task vector $\vartheta = A(\mathcal{D})$ that encodes the expert behavior specific to that task (Hendel et al., 2023; Li et al., 2023; Todd et al., 2024; Huang et al., 2024; Liu et al., 2024; Hojel et al., 2024), and $\mathcal{F}(\cdot\,; \vartheta)$ represents the task-specific reasoning function that applies the ϑ to generate the prediction for query q. This leads to the parametric representation:

$$\pi_\theta^{\text{IEF}}(\tau \mid q) := \pi_\theta(\tau \mid [\mathcal{D}; q]) = \pi_{\mathcal{F}}(\tau \mid q; \vartheta), \tag{3}$$

indicating that ICL implicitly introduces an *expert-induced prior* ϑ that steers the rollout distribution toward expert-like regions—without any explicit optimization on π_θ. While ICL itself is an inference-time mechanism that does not update model parameters, we incorporate its induced trajectories into GRPO training to form IEF.

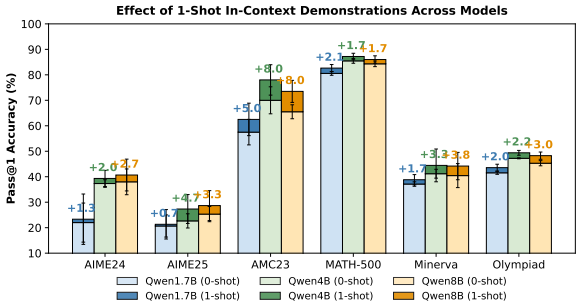


Figure 2: Comparison between 0-shot and 1-shot ICL on reasoning accuracy across benchmark datasets.

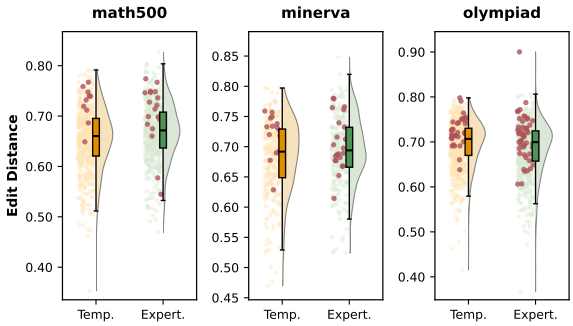


Figure 3: Effect of in-context steering on exploration and diversity. *Temp.* increases the default decoding temperature (0.6→1.2), and *Expert.* introduces in-context guidance (0-shot→1-shot). Red dots indicate instances flipped from incorrect to correct.

Group Relative Policy Optimization (GRPO). GRPO is an efficient *On-Policy* optimization algorithm tailored for RL in LLMs, where the advantage for each token is computed in a group-relative manner without requiring an additional critic model to estimate token values. Given a set of rollouts $\{\tau_i\}_{i=1}^N$ sampled from the old policy $\pi_{\theta_{\text{old}}}$, the normalized advantage is computed by:

$$A_i = \frac{R(\tau_i) - \text{mean}(G)}{\text{std}(G)}, \quad G = \{R(\tau_i)\}_{i=1}^N. \tag{4}$$

Analogous to PPO (Schulman et al., 2017), the GRPO objective is formulated as:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \frac{1}{\sum_{i=1}^{N}|\tau_i|}\sum_{i=1}^{N}\sum_{t=1}^{|\tau_i|}\text{CLIP}(r_{i,t}(\theta), A_i, \epsilon), \quad (5)$$

where $r_{i,t}(\theta) = \frac{\pi_\theta(\tau_{i,t}|\tau_{i,<t})}{\pi_{\theta_{\text{old}}}(\tau_{i,t}|\tau_{i,<t})}$ is the importance ratio, and $\text{CLIP}(r, A, \epsilon) = \min(r \cdot A, \text{clip}(r; 1-\epsilon, 1+\epsilon) \cdot A)$ is the clipping function for variance reduction. To prevent the learned policy from drifting too far from the reference model, we retain the KL regularization term $\beta \cdot D_{\text{KL}}[\pi_\theta||\pi_{\text{ref}}]$ in GRPO, which is jointly optimized to ensure training stability and maintain controllable policy updates.

By leveraging ICL-conditioned rollouts within a mixed-policy GRPO framework, our approach enables expert-guided exploration to directly participate in policy optimization, effectively realizing an *In-Context Steered Policy Optimization* process.

4 Method

Figure 4 illustrates the overall ICPO training framework and the ICPO training process is detailed in Algorithm 1.

4.1 Mixed-Policy GRPO with Implicit Expert Forcing

To incorporate expert-conditioned exploration into group rollouts, we follow Yan et al. (2025) and extend GRPO into a *mixed-policy* setting, where each group consists of N_{on} on-policy trajectories $\tau_i \sim \pi_{\theta_{\text{old}}}$ and N_{off} trajectories generate under IEF $\tau_j \sim \pi^{\text{IEF}}_{\theta_{\text{old}}}$, such that $N_{\text{on}} + N_{\text{off}} = N$. We can recompute the group-normalized advantage (as in Eq. 4) over the mixed rollout set as:

$$\hat{A}_i = \frac{R(\tau_i) - \text{mean}(G_{\text{on}} \cup G_{\text{off}})}{\text{std}(G_{\text{on}} \cup G_{\text{off}})}, \quad (6)$$

where $G_{\text{on}} = \{R(\tau_i)\}_{i=1}^{N_{\text{on}}}$ and $G_{\text{off}} = \{R(\tau_j)\}_{j=1}^{N_{\text{off}}}$.

The objective of mixed-policy GRPO with IEF balances exploitation of the current policy with exploration toward expert-aligned regions, and can be written as follows:

$$\mathcal{J}_{\text{Mixed}}(\theta) = \underbrace{\mathbb{E}_{\tau\sim\pi_{\theta_{\text{old}}}}}_{\text{on-policy}}\left[\frac{1}{|\tau|}\sum_{t=1a}^{|\tau|}\text{CLIP}\left(r_t(\theta), \hat{A}(\tau), \epsilon\right)\right] + \underbrace{\mathbb{E}_{\tau\sim\pi^{\text{IEF}}_{\theta_{\text{old}}}}}_{\text{off-policy}}\left[\frac{1}{|\tau|}\sum_{t=1}^{|\tau|}\text{CLIP}\left(\hat{r}_t(\theta), \hat{A}(\tau), \epsilon\right)\right], \quad (7)$$

where $\hat{r}_{j,t}(\theta) = \frac{\pi_\theta(\tau_{j,t}|\tau_{j,<t})}{\pi^{\text{IEF}}_\theta(\tau_{j,t}|\tau_{j,<t})}$ is the expert-conditioned importance weight.

Unlike prior work (Yan et al., 2025), which adopts a *model-based off-policy* scheme by relying on an additional advanced LRM π_ϕ to provide expert trajectories for the same training prompts, our mixed-policy GRPO with IEF operates as an *input-conditioned off-policy* method. Specifically, while all rollouts are sampled from the same policy π_θ, in-context demonstrations alter the input conditioning and steer the policy away from its default output distribution. This conditioning discrepancy induces a behavior mismatch, under which expert-conditioned rollouts $\tau_j \sim \pi_\theta(x_{\text{exp}})$ are regarded as off-policy relative to standard on-policy samples $\tau_i \sim \pi_\theta(x)$.

4.2 Expert Region Reject Sampling

Building upon the expert-conditioned off-policy branch above, we further restrict updates to those trajectories that demonstrably improve model performance. We define an *expert region* as the subset of states where expert conditioning yields superior guidance, steering the policy beyond its native distribution. A rollout τ_j generated under expert conditioning is accepted into this region if its reward exceeds a predefined threshold δ:

$$\mathcal{E}_{\text{exp}} = \left\{(x_{\text{exp}}, \tau_j) \;\middle|\; R(\tau_j) \geq \delta\right\}, \quad (8)$$

where δ is set to 1.0 by default.

To prevent low-quality expert-conditioned traces from biasing training, we define a reject sampling operator ρ that selectively retains trajectories within the expert region. Formally, ρ performs reject sampling by restricting the expectation to trajectories that fall within the expert region:

$$\rho(\cdot) = \mathbb{E}_{\tau\sim\pi_\theta(\tau|\tau\in\mathcal{E}_{\text{exp}})}[g(\tau)], \quad (9)$$

where $g(\tau)$ denotes the per-trajectory contribution to the objective. This filtering ensures that only high-reward expert-conditioned rollouts contribute to policy updates. The final objective of ICPO then becomes:

$$\mathcal{J}_{\text{ICPO}}(\theta) = \frac{1}{Z}\Big[\underbrace{\sum_{i=1}^{N_{\text{on}}}\sum_{t=1}^{|\tau_i|}\text{CLIP}(r_{i,t}(\theta), A_i, \epsilon)}_{\text{on-policy objective}} + \underbrace{\rho\left(\sum_{j=1}^{N_{\text{off}}}\sum_{t=1}^{|\tau_j|}\text{CLIP}(f(\hat{r}_{j,t}(\theta)), \hat{A}_j, \epsilon)\right)}_{\text{off-policy objective}}\Big], \quad (10)$$

where Z normalizes over all valid tokens. The shaping function $f(\cdot)$ follows prior work and is defined

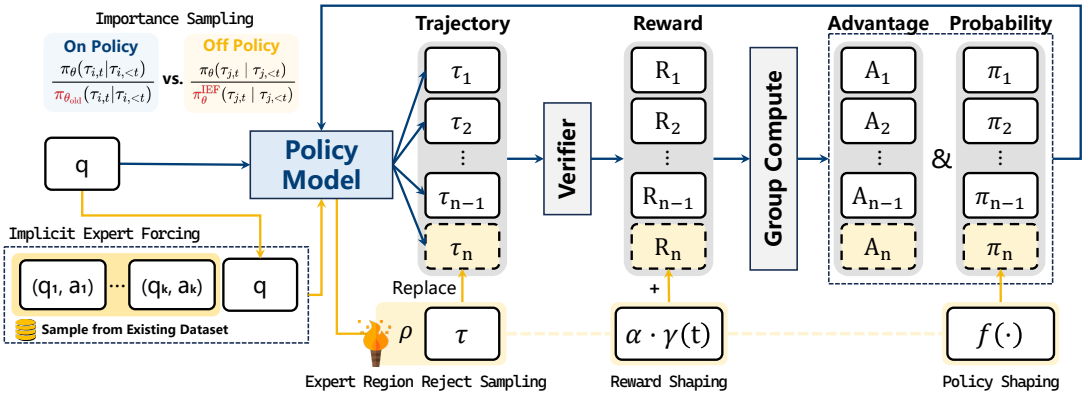


Figure 4: Overall Framework of ICPO. ICPO performs mixed-policy GRPO using off-policy trajectories generated by the policy model itself via implicit expert forcing, with reject sampling and reward shaping to stabilize training.

as $f(x) = \frac{x}{x+\lambda}$, where $\lambda = 0.01$ by default (Yan et al., 2025). This shaping biases learning toward expert-induced improvements while encouraging exploration.

4.3 Reward Shaping with Annealed Expert Bonus

The verifiable reward function evaluates the model output by extracting the final answer and comparing it against the predefined ground-truth answer. It assigns a binary score based on whether the extracted answer matches the correct solution under a task-specific verifier. Formally,

$$R(\tau) = \begin{cases} 1 & \text{if } \tau \text{ is correct} \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

This verifiable reward has been shown to reliably lead to successful scaling of RL training.

To encourage early imitation of expert-conditioned behavior while avoiding long-term over-reliance, we design a variant of ICPO (namely ICPO†) and add a step-annealed bonus only to trajectories that have the correct answer and within the expert region $\mathcal{E}_{\text{exp}}$:

$$R_{\text{shaped}}(\tau) = R(\tau) + \alpha \cdot \gamma(t), \tag{12}$$

where $\gamma(t) = 1 - \frac{t}{T}$ denotes a linear decay scheduler over the training step t, and α denotes the bonus weight (set to 1.0 in our experiments).

5 Experimental Setup

Dataset. We follow Yan et al. (2025) and adopt the *OpenR1-Math-220k* dataset as our main training corpus. Specifically, we use the filtered subset[1], which excludes generations exceeding 8192 tokens as well as those identified as incorrect by *Math-Verify*[2]. The resulting dataset contains approximately 45k verified reasoning prompts. For IEF, instead of using trajectories from advanced LRMs, we randomly sample demonstrations for each prompt from the *MATH* (Hendrycks et al., 2021) training set, which contains 7.5k mathematical problems paired with high-quality solutions.

Implementation Details. The complete set of hyperparameters used in our training is listed in Appendix B.1. For the main experiments, we use *Qwen3-1.7B* and *Qwen3-8B* (Yang et al., 2025) as the base models and employ GRPO (Shao et al.,

Algorithm 1 ICPO Training Procedure

Require: Policy π_θ, old policy $\pi_{\theta_{\text{old}}}$, expert data $\mathcal{D}$, batch size B, rollout size N, few-shot count k, RS threshold δ, step t, annealed bonus $\alpha \cdot \gamma(t)$

```
for each step do
    Sample prompts {x_i}_{i=1}^B
    for i = 1 to B do
        for j = 1 to N do
            τ_i^j ~ π_θold(·|x_i)
            Compute R(τ_i^j)
        end for
        Sample k expert (q, a) pairs from D and form x_i^exp
        Generate τ_i^IEF ~ π_θold^IEF(·|x_i^exp)
        if R(τ_i^IEF) ≥ δ and correct(τ_i^IEF) then
            Pick random j
            Replace τ_i^j ← τ_i^IEF
            if Enable reward shaping then
                R(τ_i^j) ← R(τ_i^IEF) + α · γ(t)
            end if
        end if
        Compute Â_i using Eq. 6
    end for
    Compute mixed rollout loss L according to J_ICPO(θ)
    θ ← θ − η∇_θ L
    π_θold ← π_θ
end for
```

[1] https://huggingface.co/datasets/Elliott/Openr1-Math-46k

[2] https://github.com/huggingface/Math-Verify

2024) as our RL algorithm. We also include *Qwen2.5-Math-7B* to facilitate comparison with other related methods. We additionally evaluate ICPO on *Qwen3-8B-Base* and *LLaMA-3.1-8B* to examine its cross-model generalization. We generate a total of 8 rollout trajectories per prompt. For the on-policy baseline, we use 8 on-policy rollouts. For our mixed-policy GRPO, we follow previous work (Yan et al., 2025) and use 1 off-policy rollout and 7 on-policy rollouts to ensure comparability. Evaluation details are provided in Appendix B.2.

Baseline Methods. For direct comparison, we evaluate our ICPO against the vanilla **GRPO** baseline trained on the same subset of *OpenR1-Math-220k*. We further include $\mathbf{GRPO_{ExtraRollouts}}$, which extends GRPO by using 16 on-policy rollouts to enhance exploration. To examine the effect of training data source, we also train a GRPO variant on expert-domain data, identical to our IEF demonstrations (*MATH*), denoted as $\mathbf{GRPO_{ExpertDomain}}$. In addition, we compare with **LUFFY** (Yan et al., 2025), which leverages trajectories generated by advanced LRMs as off-policy rollouts, along with other baselines detailed in Appendix B.3.

6 Experimental Results

6.1 Main Results

The main experiments include two variants of our proposed ICPO framework: **ICPO**, which operates without reward shaping (RS), and **ICPO**†, which incorporates RS to further enhance expert-domain alignment. To better understand their optimization behavior against GRPO, we visualize the reward dynamics over training steps across different datasets, as shown in Figure 5 and 12, where both ICPO variants consistently achieve higher rewards throughout training. We also visualize the training dynamics in Appendix D.1.

In-Distribution Evaluation. Table 1 shows both the in- and out-of-distribution performance, where *MATH-500* serves as the expert domain. Across both model scales, ICPO consistently outperforms the vanilla GRPO baseline, especially on in-distribution benchmarks. For the smaller Qwen3-1.7B model, ICPO† achieves an average improvement of **+3.0** points over GRPO, while ICPO further stabilizes optimization with a **+4.1** point overall gain. A similar trend is observed for the larger Qwen3-8B model, where ICPO and ICPO† yield **+2.2** and **+1.5** average improvements, respectively.

These consistent gains across scales indicate that ICPO effectively steers the policy toward a more optimal output distribution. By leveraging both the challenging training set and high-quality expert demonstrations through IEF, ICPO forms a more informative training signal than either source alone—a synergistic effect that yields superior performance over both $\text{GRPO}_{\text{ExpertDomain}}$ and vanilla GRPO. Moreover, unlike ICPO, simply increasing the number of on-policy rollouts in $\text{GRPO}_{\text{ExtraRollouts}}$ offers only marginal improvement, revealing the inherent exploration limitations of vanilla GRPO and underscoring the importance of effective steering.

Out-of-Distribution Evaluation. ICPO† introduces RS to explicitly amplify the advantage of trajectories falling within the expert domain $\mathcal{E}_{\text{exp}}$. Our hypothesis is that steering optimization toward these expert-aligned trajectories induces more coherent reasoning structures and stabilizes policy updates, which in turn enhance generalization both in- and out-of-distribution. Empirically, ICPO† achieves average gains of **+0.7** and **+2.4** over GRPO on Qwen3-1.7B and 8B, respectively, providing evidence supporting this hypothesis.

Comparison with Other Baselines. Table 2 presents in-distribution results compared to existing baselines. ICPO† achieves the strongest performance among all comparable methods, surpassing ReLIFT (Ma et al., 2025), LUFFY (Yan et al., 2025), and Prefix-RFT (Huang et al., 2025) by **+4.6**, **+3.3**, and **+1.6** points, respectively. For brevity, the results and analysis on OOD benchmarks are provided in Appendix D.3.

6.2 Ablation Study

Effect of Each Component. Table 3 summarizes ablations by progressively removing components of ICPO. On Qwen3-8B, all components contribute positively: *IEF* yields the largest gain (**+1.2**) by injecting expert-conditioned guidance and enhancing exploration, while *ERRS* improves accuracy by filtering invalid expert-region trajectories (**+0.8**). Across both model sizes, removing any component consistently harms performance, demonstrating that all the modules are complementary and jointly essential for the effectiveness of ICPO.

Sources of Expert Guidance. We compare ICPO with LUFFY (Yan et al., 2025), which incorporates trajectories generated by advanced LRMs

Method	In-Distribution Benchmarks							Out-of-Distribution Benchmarks				
	AIME 24/25	AMC23	MATH	Minerva	Olympiad	Avg.(Impr.)		ARC	GPQA	MMLU	Avg.(Impr.)	
Qwen3-1.7B	21.7 / 20.6	56.8	79.4	37.1	40.6	42.7	–	88.3	22.2	52.3	54.3	–
GRPO	28.4 / 22.5	66.7	83.6	40.8	48.2	48.4	–	88.3	34.3	54.4	59.0	–
$\text{GRPO}_{\text{ExtraRollouts}}$	28.3 / 24.7	69.8	84.4	43.0	53.8	50.7	(+2.3)	**88.9**	30.8	55.0	58.2	(-0.8)
$\text{GRPO}_{\text{ExpertDomain}}$	26.8 / 24.8	66.3	83.8	**45.6**	50.5	49.6	(+1.2)	88.8	32.8	**55.5**	59.0	(0.0)
ICPO (Ours)	**31.3** / 26.3	**70.4**	86.8	44.1	**56.4**	**52.5**	**(+4.1)**	88.1	27.8	**55.5**	57.1	(-1.9)
ICPO† (Ours)	29.0 / **26.6**	70.0	**87.2**	42.7	52.7	51.4	(+3.0)	87.7	**36.4**	55.0	**59.7**	(+0.7)
Qwen3-8B	33.7 / 21.4	65.4	86.2	40.4	43.3	48.4	–	96.2	37.9	68.0	67.4	–
GRPO	54.8 / 38.5	83.8	91.0	50.7	62.4	63.5	–	95.8	51.0	72.0	72.9	–
$\text{GRPO}_{\text{ExtraRollouts}}$	53.0 / 40.0	85.4	**93.0**	**52.9**	61.6	64.3	(+0.8)	95.7	53.0	72.9	73.9	(+1.0)
$\text{GRPO}_{\text{ExpertDomain}}$	**58.3** / 40.4	86.2	91.8	50.7	60.0	64.6	(+1.1)	92.2	51.0	70.2	71.1	(-1.8)
ICPO (Ours)	55.2 / **43.7**	**87.0**	92.0	51.1	**65.2**	**65.7**	**(+2.2)**	95.5	**55.1**	72.3	74.3	(+1.4)
ICPO† (Ours)	56.2 / 40.9	85.4	92.0	51.5	64.3	65.0	(+1.5)	**95.6**	**55.1**	**75.3**	**75.3**	**(+2.4)**

Table 1: Evaluation results for Qwen3. ICPO shows clear and consistent improvements on in-distribution data, while ICPO† provides a more balanced trade-off between in-distribution and OOD performance.

Method	AIME24	AIME25	AMC23	MATH	Minerva	Olympiad	Average
Qwen2.5-Math-7B (Yang et al., 2025)	11.5	4.9	31.3	43.6	7.4	15.6	19.0
Previous RLVR Methods							
SimpleRL-Zero* (Zeng et al., 2025)	27.0	6.8	54.9	76.0	25.0	34.7	37.4
PRIME-Zero* (Cui et al., 2025)	17.0	12.8	54.0	81.4	39.0	40.3	40.8
OpenReasoner-Zero* (Hu et al., 2025b)	16.5	15.0	52.1	82.4	33.1	47.1	41.0
Oat-Zero* (Liu et al., 2025b)	33.4	11.9	61.2	78.0	34.6	43.4	43.8
SFT and RL							
SFT (Yan et al., 2025)	23.8	22.9	61.8	82.2	37.9	42.1	45.1
RL_{GRPO} (Shao et al., 2024)	22.9	13.3	63.0	81.2	37.1	43.1	43.4
$\text{SFT+RL}_{\text{GRPO}}$ (Shao et al., 2024)	29.3	22.3	67.1	85.8	44.1	50.8	49.9
Comparable Baseline Methods							
ReLIFT* (Ma et al., 2025)	28.2	22.9	64.8	85.0	37.1	54.9	48.8
LUFFY* (Yan et al., 2025)	29.4	23.1	65.6	87.6	37.5	**57.2**	50.1
Prefix-RFT* (Huang et al., 2025)	31.8	26.4	68.2	**88.4**	40.3	55.7	51.8
ICPO (Ours)	29.3	**28.9**	74.9	**88.4**	**45.6**	52.4	53.2
ICPO† (Ours)	**32.8**	26.7	**75.9**	86.6	44.9	53.6	**53.4**

Table 2: Evaluation results for Qwen2.5-Math-7B. "*" indicate results reported from their original papers.

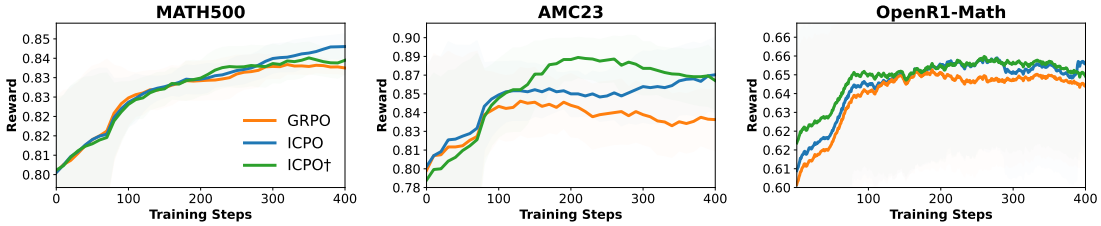
into off-policy GRPO, as shown in Table 4. ICPO*, which removes ERRS and is thus directly comparable to LUFFY in its source of expert guidance, already surpasses LUFFY. This demonstrates that IEF can steer the model toward a better policy distribution by leveraging existing datasets as contextual guidance, eliminating the need for costly external LRM computation. Please refer to Appendix E for more ablation studies.

7 Analysis of Distribution Shift

To examine how ICPO alters the model's predictive distribution, we conduct an in-depth analysis on the expert-domain dataset *MATH-500* from both the instance- and token-level perspectives.

Instance-Level. Figure 6 presents per-instance perplexity, where each point corresponds to an individual test case. Points lying below the diagonal $y = x$ indicate that the target method yields lower perplexity than GRPO on the same instance, suggesting a shift of the predictive distribution toward the expert domain. Simply increasing the number of on-policy rollouts yields no deviation from the GRPO baseline, indicating that rollout amplification alone does not alter the underlying policy distribution. In contrast, ICPO induces a clear shift toward expert-like reasoning patterns, and incorporating expert-bonus RS further strengthens it.

Token-Level. Figure 7 presents token-rank shift (Lin et al., 2023) based on percentile positions (de-



(a) Expert Domain Reward (b) In-Distribution Reward (c) Training Reward

Figure 5: Reward curves of Qwen3-8B over training steps across test (*Mean@2*) and train sets.

Variant	MATH	AIME24/25	AMC	Mnrv.	Avg.
Qwen3-1.7B					
ICPO	**86.8**	31.3 / 26.3	**70.4**	**44.1**	**51.8**
- ERRS	85.6	**32.2** / 25.9	66.8	42.3	50.6
- IEF (GRPO)	83.6	28.4 / 22.5	66.7	40.8	48.4
Qwen3-8B					
ICPO	**92.0**	55.2 / **43.6**	**87.0**	51.1	**65.8**
- ERRS	89.6	55.2 / 41.7	85.2	**53.3**	65.0
- IEF (GRPO)	91.0	54.8 / 38.5	83.8	50.7	63.8

RS = *Reward Shaping*, ERRS = *Expert Region Reject Sampling*, IEF = *Implicit Expert Forcing*.

Table 3: Ablation analysis by progressively removing components from ICPO.

Variant	A.E.R.	MATH	AIME24/25	AMC	Mnrv.	Avg.
Qwen3-1.7B						
LUFFY	✓✗✗	83.6	26.8 / 23.5	63.3	41.9	47.8
ICPO*	✗✓✗	85.6	**32.2** / 25.9	66.8	42.3	50.6
ICPO	✗✓✓	**86.8**	31.3 / **26.3**	**70.4**	**44.1**	**51.8**
Qwen3-8B						
LUFFY	✓✗✗	91.0	53.1 / 37.0	85.3	52.2	63.7
ICPO*	✗✓✗	89.6	**55.2** / 41.7	85.2	**53.3**	65.0
ICPO	✗✓✓	**92.0**	**55.2 / 43.6**	**87.0**	51.1	**65.8**

A. = *Advanced LRM Trajectory*, E. = *Existing Dataset*, R. = *Expert Region Reject Sampling (ERRS)*. ICPO* = *ICPO w/o ERRS*.

Table 4: Comparison across expert guidance sources.

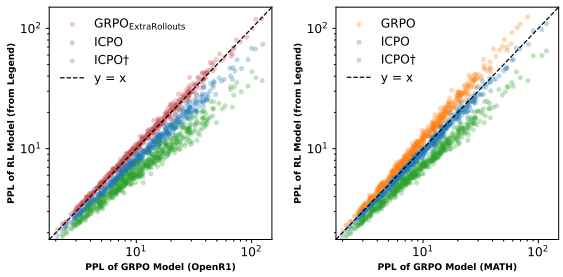


Figure 6: Perplexity as an instance-level measure of expert-domain distribution shift.

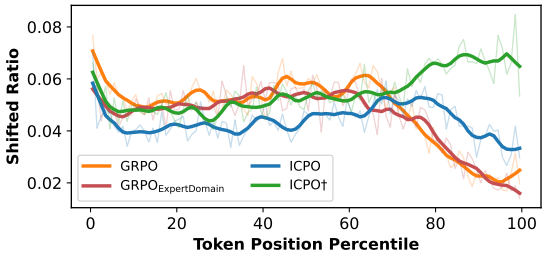


Figure 7: Token-rank shift as a token-level measure of expert-domain distribution shift.

tails are in Appendix F). For 0%–60%, all methods exhibit similar stable trends. Beyond the 60% percentile, the differences become pronounced: GRPO shows a steep decline, with the model reverting to high-probability outputs of the base policy in later generations, indicating exploration collapse. In contrast, ICPO exhibits a slower drop in token shift ratio during later stages, as IEF continuously provides directional guidance toward the expert domain. Notably, ICPO† benefits from the expert-bonus RS, which serves as an effective regularizer, resulting in the largest shift ratio in later generations. As a result, it is more likely to explore solutions that are different from the base policy.

Overall, the results demonstrate that ICPO effectively enables LRM to "***think outside the policy***."

8 Conclusion

We have presented ICPO, a unified RLVR framework that enhances reasoning without relying on external expert models. Leveraging the inherent ICL capability of LRMs, ICPO introduces mixed-policy GRPO with IEF, which constructs expert-conditioned rollouts from existing datasets, improving data utilization and expanding exploration beyond the current policy distribution. To ensure stable optimization, ICPO further integrates ERRS to eliminate noisy off-policy trajectories and adopts RS to facilitate a smooth transition from expert-guided imitation to autonomous optimization. Experiments show that ICPO consistently improves RL performance, highlighting its promise as a scalable and general post-training paradigm for LRMs.

Limitations

Despite the demonstrated improvements, our approach has several limitations that open avenues for future investigation.

Task Scope. Our experiments are currently limited to mathematical reasoning tasks under the RLVR framework. Extending ICPO to other domains such as code generation would require additional task-specific adaptation.

Model Capability Assumption. Our method assumes that the underlying model possesses basic ICL capability; without it, the implicit expert-conditioned rollouts may not yield meaningful supervision. Due to time constraints, we did not conduct experiments on models lacking ICL ability, such as training them to acquire this capability before applying GRPO.

Demonstration Sensitivity. The model may be sensitive to the choice of ICL demonstrations. We experimented with various heuristic demonstration selection methods, as detailed in the Appendix C. The results show that randomly sampling demonstrations—both across batches and per prompt—yields the best performance. Consequently, we adopt random demonstration selection as the default strategy in our main experiments, without any deliberate curation. We hypothesize that carefully selecting demonstrations tailored to each problem could further improve performance, which we plan to explore in future work.

Expert Data Selection. Our method eliminates the reliance on stronger external LRMs by leveraging the model's own ICL capability to treat existing datasets as implicit experts, steering rollouts beyond the default GRPO distribution and enhancing exploration. Although ICPO improves the utility of available high-quality datasets through IEF, it still imposes several constraints on the expert data: (1) expert data must provide complete solutions; (2) the solutions must be of sufficiently high quality to offer effective guidance; and (3) using domain-specific expert data as demonstrations may bias the model toward that domain, potentially degrading its general-purpose capabilities, as detailed in Appendix D.3.

Despite these limitations, ICPO also opens promising directions. First, identifying high-quality, general-purpose expert datasets could further enhance model performance on both in-distribution and out-of-distribution tasks. Second, because IEF encourages the model to adopt the stylistic patterns of expert demonstrations, ICPO may naturally support controllable style transfer in reasoning tasks. We leave these directions to future work.

Ethics Statement

Use of AI Assistants. We have employed ChatGPT as a writing assistant, primarily for polishing the text after the initial composition. We certify that any use of AI tools, including ChatGPT, was strictly limited to linguistic refinement such as improving grammar, clarity, and style. All substantive ideas, analyses, and arguments presented in this work originate from the authors or from properly cited prior research.

Computational Budget. All our experiments are conducted on a machine with CentOS 8, 384 AMD® EPYC™ 9K84 96-Core Processor CPUs and 2.2TiB memory. We use 8× NVIDIA H20 GPUs for all the experiments. The training of ICPO/ICPO† takes around 3 and 7 days for the 1.5B and 8B models, respectively.

Reproducibility. Our work is reproducible because we have provided our source code and implementation details.

Potential Risks. To the best of our knowledge, there are no potential risks concerning our work.

Licenses. The licenses of the scientific artifacts we use are shown in Table 5.

Category	Artifact	License
Model	Qwen3 Models	Apache-2.0
	Qwen2.5 Models	Apache-2.0
Framework	LIMO	MIT
	VERL	Apache-2.0
	Math-Verify	Apache-2.0
Dataset	Skywork-OR1-RL-Data	Apache-2.0
	Openr1-Math-46k-8192	MIT
	MathInstruct	MIT
	MATH	MIT
	MATH-500	MIT
	Minerva	MIT
	OlympiadBench	MIT
	AMC23	N/A
	AIME24	N/A
	AIME25	N/A
Methods	LUFFY	N/A

Table 5: Licenses of scientific artifacts used in this work.

References

Weizhe Chen, Zhicheng Zhang, Guanlin Liu, Renjie Zheng, Wenlei Shi, Chen Dun, Zheng Wu, Xing Jin, and Lin Yan. 2025. Flaming-hot initiation with regular execution sampling for large language models. *Preprint*, arXiv:2410.21236.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *Preprint*, arXiv:1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *Preprint*, arXiv:2110.14168.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, Jiarui Yuan, Huayu Chen, Kaiyan Zhang, Xingtai Lv, Shuo Wang, Yuan Yao, Xu Han, and 6 others. 2025. Process reinforcement through implicit rewards. *Preprint*, arXiv:2502.01456.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Shihan Dou, Muling Wu, Jingwen Xu, Rui Zheng, Tao Gui, Qi Zhang, and Xuanjing Huang. 2025. Improving rl exploration for llm reasoning through retrospective replay. *Preprint*, arXiv:2504.14363.

Yuqian Fu, Tinghong Chen, Jiajun Chai, Xihuai Wang, Songjun Tu, Guojun Yin, Wei Lin, Qichao Zhang, Yuanheng Zhu, and Dongbin Zhao. 2025. Srft: A single-stage method with supervised and reinforcement fine-tuning for reasoning. *Preprint*, arXiv:2506.19767.

Aaron Grattafiori, Abhimanyu Dubey, and Abhinav Jauhri et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Siddhant Haldar, Vaibhav Mathur, Denis Yarats, and Lerrel Pinto. 2023. Watch and match: Supercharging imitation with regularized optimal transport. In *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 32–43. PMLR.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *Preprint*, arXiv:2402.14008.

Jujie He, Jiacai Liu, Chris Yuhao Liu, Rui Yan, Chaojie Wang, Peng Cheng, Xiaoyu Zhang, Fuxiang Zhang, Jiacheng Xu, Wei Shen, Siyuan Li, Liang Zeng, Tianwen Wei, Cheng Cheng, Bo An, Yang Liu, and Yahui Zhou. 2025. Skywork open reasoner 1 technical report. *arXiv preprint arXiv:2505.22312*.

Roee Hendel, Mor Geva, and Amir Globerson. 2023. In-context learning creates task vectors. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9318–9333, Singapore. Association for Computational Linguistics.

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

Todd Hester, Matej Vecerik, Olivier Pietquin, Marc Lanctot, Tom Schaul, Bilal Piot, Dan Horgan, John Quan, Andrew Sendonaris, Ian Osband, Gabriel Dulac-Arnold, John Agapiou, Joel Leibo, and Audrunas Gruslys. 2018. Deep q-learning from demonstrations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).

Alberto Hojel, Yutong Bai, Trevor Darrell, Amir Globerson, and Amir Bar. 2024. Finding visual task vectors. *Preprint*, arXiv:2404.05729.

Hengyuan Hu, Suvir Mirchandani, and Dorsa Sadigh. 2023. Imitation bootstrapped reinforcement learning. *arXiv preprint arXiv:2311.02198*.

Jian Hu, Mingjie Liu, Ximing Lu, Fang Wu, Zaid Harchaoui, Shizhe Diao, Yejin Choi, Pavlo Molchanov, Jun Yang, Jan Kautz, and Yi Dong. 2025a. Brorl: Scaling reinforcement learning via broadened exploration. *Preprint*, arXiv:2510.01180.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. 2025b. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *Preprint*, arXiv:2503.24290.

Brandon Huang, Chancharik Mitra, Leonid Karlinsky, Assaf Arbelle, Trevor Darrell, and Roei Herzig. 2024. Multimodal task vectors enable many-shot multimodal in-context learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zeyu Huang, Tianhao Cheng, Zihan Qiu, Zili Wang, Yinghui Xu, Edoardo M. Ponti, and Ivan Titov. 2025. Blending supervised and reinforcement fine-tuning with prefix sampling. *Preprint*, arXiv:2507.01679.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. 2022. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 3843–3857. Curran Associates, Inc.

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

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, volume 36, pages 41451–41530. Curran Associates, Inc.

Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. 2023. The unlocking spell on base llms: Rethinking alignment via in-context learning. *Preprint*, arXiv:2312.01552.

Mingyang Liu, Gabriele Farina, and Asuman Ozdaglar. 2025a. Uft: Unifying supervised and reinforcement fine-tuning. *Preprint*, arXiv:2505.16984.

Sheng Liu, Haotian Ye, Lei Xing, and James Zou. 2024. In-context vectors: Making in context learning more effective and controllable through latent space steering. *Preprint*, arXiv:2311.06668.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025b. Understanding r1-zero-like training: A critical perspective. *Preprint*, arXiv:2503.20783.

Lu Ma, Hao Liang, Meiyi Qiang, Lexiang Tang, Xiaochen Ma, Zhen Hao Wong, Junbo Niu, Chengyu Shen, Runming He, Yanhao Li, Bin Cui, and Wentao Zhang. 2025. Learning what reinforcement learning can't: Interleaved online fine-tuning for hardest questions. *Preprint*, arXiv:2506.07527.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2023. Gpqa: A graduate-level google-proof q&a benchmark. *Preprint*, arXiv:2311.12022.

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

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

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2025. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, EuroSys '25, page 1279–1297, New York, NY, USA. Association for Computing Machinery.

Chenming Tang, Hsiu-Yuan Huang, Weijie Liu, Saiyong Yang, and Yunfang Wu. 2025. Do not step into the same river twice: Learning to reason from trial and error. *Preprint*, arXiv:2510.26109.

Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. 2024. Function vectors in large language models. *Preprint*, arXiv:2310.15213.

Chen Wang, Lai Wei, Yanzhi Zhang, Chenyang Shao, Zedong Dan, Weiran Huang, Yuzhi Zhang, and Yue Wang. 2025. Eframe: Deeper reasoning via exploration-filter-replay reinforcement learning framework. *Preprint*, arXiv:2506.22200.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, volume 37, pages 95266–95290. Curran Associates, Inc.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. 2025. Learning to reason under off-policy guidance. *Preprint*, arXiv:2504.14945.

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, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

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. 2024. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *Preprint*, arXiv:2409.12122.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. 2025. Limo: Less is more for reasoning. *Preprint*, arXiv:2502.03387.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2023. Mammoth: Building math generalist models through hybrid instruction tuning. *Preprint*, arXiv:2309.05653.

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

Weihao Zeng, Yuzhen Huang, Wei Liu, Keqing He, Qian Liu, Zejun Ma, and Junxian He. 2025.

7b model and 8k examples: Emerging reasoning with reinforcement learning is both effective and efficient. https://hkust-nlp.notion.site/simplerl-reason. Notion Blog.

Haichao Zhang, We Xu, and Haonan Yu. 2023. Policy expansion for bridging offline-to-online reinforcement learning. *arXiv preprint arXiv:2302.00935*.

Kaiyi Zhang, Ang Lv, Jinpeng Li, Yongbo Wang, Feng Wang, Haoyuan Hu, and Rui Yan. 2025a. Stephint: Multi-level stepwise hints enhance reinforcement learning to reason. *Preprint*, arXiv:2507.02841.

Shimao Zhang, Yu Bao, and Shujian Huang. 2024. Edt: Improving large language models' generation by entropy-based dynamic temperature sampling. *Preprint*, arXiv:2403.14541.

Xuechen Zhang, Zijian Huang, Yingcong Li, Chenshun Ni, Jiasi Chen, and Samet Oymak. 2025b. Bread: Branched rollouts from expert anchors bridge sft & rl for reasoning. *Preprint*, arXiv:2506.17211.

A Prompt Template

Here we provide the detailed prompt formats used for our experiments.

Prompt for Qwen3

 # Prompt for RL and 0-shot Inference
<|im_start|>user
{QUESTION} Let's think step by step and output the final answer within \boxed{}
<|im_end|>
<|im_start|>assistant

 # Prompt for few-shot Inference
<|im_start|>user
Question: {QUESTION} Let's think step by step and output the final answer within \boxed{}
Answer: {ANSWER} \n \n \n
...
Question: {QUESTION} Let's think step by step and output the final answer within \boxed{}
Answer: <|im_end|>
<|im_start|>assistant

Prompt for Qwen2.5

SYSTEM_PROMPT=Your task is to follow a systematic, thorough reasoning process before providing the final solution. This involves analyzing, summarizing, exploring, reassessing, and refining your thought process through multiple iterations. Structure your response into two sections: *Thought* and *Solution*. In the Thought section, present your reasoning using the format: "<think>\nthoughts </think>\n". Each thought should include detailed analysis, brainstorming, verification, and refinement of ideas. After "</think>\n" in the Solution section, provide the final, logical, and accurate answer, clearly derived from the exploration in the Thought section. If applicable, include the answer in \boxed{} for closed-form results such as multiple-choice or mathematical solutions.

 # Prompt for RL and 0-shot Inference
{SYSTEM_PROMPT}
User:
Question: {QUESTION}
Answer:
Assistant: <think>

 # Prompt for few-shot Inference
{SYSTEM_PROMPT}
User:
Question: {QUESTION}
Answer: {ANSWER} \n \n \n
...
Question: {QUESTION}
Answer:
Assistant: <think>

Figure 8: Prompt formats used for RL training, zero-shot inference, and few-shot inference.

B Experimental Details

B.1 Training Settings

For RL finetuning, we use the widely adopted GRPO algorithm built on *VERL* (Sheng et al., 2025) framework. The full hyperparameters used in our training are listed in Table 6. Specifically, for rollout generation, we use a temperature of 1.0, and rewards are computed using *Math-Verify*. All models are trained for $T = 400$ optimization steps, and we report results using the final checkpoint.

Hyper-parameter	Value
Learning Rate	1e-6
Totel Steps	400
Batch Size	128
Mini Batch Size	64
KL Loss Coefficient	0.0
Clip Ratio	0.2
Temperature	1.0
Total Number of Rollouts	8
Maximum Prompt Length	4096
Maximum Response Length	8192

Table 6: Full hyper-parameters for training.

B.2 Evaluation Settings

We evaluate on six widely used mathematical reasoning benchmarks: **AIME24**, **AIME25**, **AMC23** (Li et al., 2024), **Minerva** (Lewkowycz et al., 2022), **Olympiad** (He et al., 2024), and **MATH-500** (Hendrycks et al., 2021). For main experiments, the AIME24, AIME25, and AMC23 benchmark, which have relatively small test sets, we report ***Avg@32*** (for ***Pass@1***), while for the other benchmarks we report ***Pass@1***. As a complementary analysis, we also report ***Pass@32*** results in Appendix D.2.

To assess generalization beyond in-domain reasoning, we further test on three out-of-distribution (OOD) benchmarks: **ARC-C** (Clark et al., 2018), **GPQA-Diamond** (Rein et al., 2023), and **MMLU-Pro** (Wang et al., 2024), with multiple-choice options shuffled to prevent contamination. All evaluations are conducted using the LIMO framework (Ye et al., 2025). During inference, we follow Yan et al. (2025) and set the generation temperature to 0.6. For ICL evaluation, we randomly sample demonstrations from the **MATH** (Hendrycks et al., 2021) training set using 5 different random seeds, and report the average performance across them.

B.3 Baseline Methods

We benchmark ICPO against the following baselines using *Qwen2.5-Math-7B* (Yang et al., 2024) as the base model. We compare our method with three lines of work, which is previous RLVR methods, SFT and RL baseline, and methods combining SFT and RL that is comparable with our ICPO.

Previous RLVR Methods. (1) **SimpleRL-Zero** (Zeng et al., 2025), which applies GRPO to approximately 24k mathematical samples from *GSM8K* (Cobbe et al., 2021) and *MATH* dataset (Hendrycks et al., 2021); (2) **PRIME-Zero** (Cui et al., 2025), which conducts policy rollouts on 150k *Numina-Math* queries with implicit process rewards; (3) **OpenReasoner-Zero** (Hu et al., 2025b), a PPO-based approach trained on 129k multi-source samples; and (4) **Oat-Zero** (Liu et al., 2025b), which removes the standard deviation in GRPO advantage calculation, and is trained on the *MATH* dataset.

SFT and RL. Here we consider three kind of methods: (1) $\mathbf{RL_{GRPO}}$, which is train on-policy within RLVR paradigm using GRPO with the same reward and data as ICPO; (2) **SFT** , where the model is supervised on the same prompts and reasoning traces as LUFFY (Yan et al., 2025); (3) **SFT+RL**, which is a two-stage training that continues RL training after SFT.

Comparable Baseline Methods. For methods that integrate supervised fine-tuning (SFT) with reinforcement learning (RL), we evaluate three representative approaches: (1) **LUFFY** (Yan et al., 2025), a mixed-policy GRPO method that incorporates advanced LRM reasoning trajectories as off-policy rollouts; (2) **ReLIFT** (Ma et al., 2025), which alternates standard RLVR updates with targeted supervised fine-tuning on the hardest online-collected questions, enabling the model to acquire new reasoning skills that pure RL cannot provide; (3) **Prefix-RFT** (Huang et al., 2025), which samples a prefix from advanced LRM reasoning trajectories and reinforces the model's continuation, mixing these hybrid sequences with online rollouts under the RFT objective to unify SFT-style imitation and RFT-style exploration.

All these methods rely on advanced LRM-generated reasoning trajectories from the training set as demonstrations, whereas ICPO leverages few-shot demonstrations retrieved from an existing external dataset—independent of the training

set—and uses them purely as in-context prompts to steer the rollout distribution on-policy.

B.4 Data Statistics

Figure 9 presents the length distribution of the demonstrations (question + answer) and the training prompts. We observe that the demonstrations are relatively short, ensuring that they do not cause input truncation or the loss of important information. Compared to approaches that append the model's generated reasoning traces as additional context, our overall input length remains much shorter and thus more efficient.

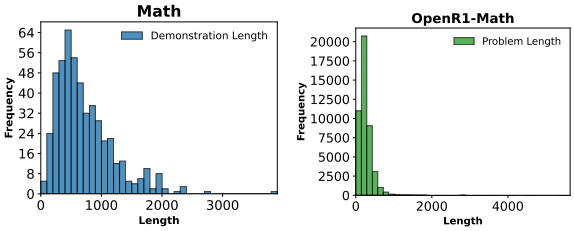


Figure 9: Length distribution of expert demonstrations (left) and training prompts (right).

C Demonstration Selection Strategies

Effects of Heuristic Demonstration Selection Strategies. We investigate several heuristic strategies for selecting demonstrations within ICPO. Experiments are conducted on the *Skywork-OR1-RL-Data* (He et al., 2025) dataset using *Qwen3-1.7B*, with all demonstrations sourced from the training split of *MATH* (Hendrycks et al., 2021) as our main setup. We evaluate the following heuristic selection methods:

- **Difficulty-based selection.** We sample each MATH training instance three times: examples solved correctly in all three attempts are labeled *Easy*, those answered incorrectly three times are labeled *Hard*, and the remaining examples are categorized as *Medium*. During training, each prompt is matched with a randomly selected demonstration from its corresponding difficulty group.

- **Length-based selection.** We partition both the demonstration pool and the training set into two equal-sized buckets based on response length. For each training sample, a demonstration is randomly selected from the bucket with the same length range.

- **Subject-based selection.** We group both the demonstration data and the training data by their problem category (subject). Each training instance is paired with a randomly selected example from the same subject as its 1-shot demonstration.

- **Random.** We randomly sample 32 examples from the entire demonstration pool and, for each training instance at every step, randomly select one of these 32 as the demonstration.

The results are shown in Figure 10. Among all heuristic strategies, the ***Hard***, ***Short***, and ***Random32*** variants achieve comparable performance, and all surpass the GRPO baseline. Given that difficulty-based and length-based methods require additional sampling or preprocessing cost, we adopt the simplest and most cost-efficient option, namely ***Random*** demonstration selection, as our default strategy.

Effects of Demonstration Pool Size. We further investigate how the size of the demonstration pool available to each batch affects ICPO training using *Qwen3-8B*, with the results shown in Figure 11. We compare two configurations. The first sets the demo pool size to 10 (denoted as *ICPO-10/7500*): for each batch, we randomly sample 10 demonstrations, and each instance within the batch randomly selects one of these as its 1-shot demonstration. This design aims to prevent excessive steering directions within a batch, which may destabilize training. The second configuration sets the demonstration pool size equal to the full demonstration set (denoted as *ICPO-7500/7500*), where each training instance samples its 1-shot demonstration from the entire pool of 7,500 examples, thereby increasing exploration during training.

As shown in the results, *ICPO-10/7500* yields more stable and stronger early-stage performance but restricts exploration in later stages. In contrast, *ICPO-7500/7500* demonstrates greater potential in the later phase, achieving higher rewards on the expert domain (MATH-500) and resolving more non-pass cases. Based on these observations, we adopt random selection with a full demonstration pool size of 7,500 as the default configuration for our main experiments.

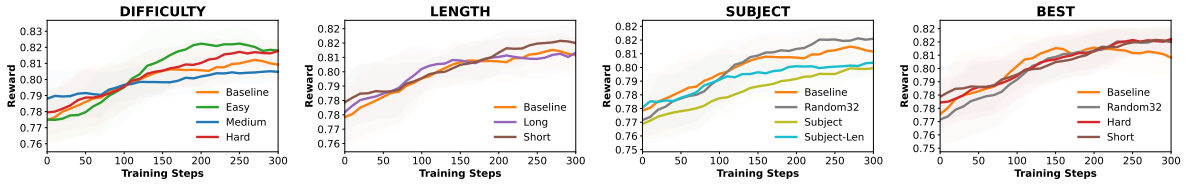


Figure 10: Rewards on the *MATH-500* dataset for different heuristic demonstration-selection strategies using *Qwen3-1.7B*. The ***Hard***, ***Short***, and ***Random32*** demonstration strategies achieve comparable performance and all outperform the GRPO baseline.

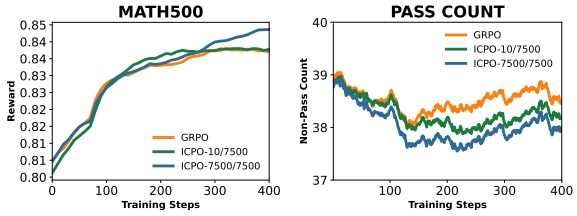


Figure 11: Comparison of ICPO training using different demonstration pool sizes on *Qwen3-8B*. The left figure reports the reward and training dynamics on the MATH-500 benchmark, while the right figure presents the number of non-pass cases on the training set.

D Results and Analysis

D.1 Training Dynamics

Figure 13 illustrates the training dynamics. For smaller models, ICPO maintains a higher policy entropy than GRPO, reflecting a broader policy support and increased exploration during training.

For larger models, their stronger ICL capability makes IEF more effective. This leads to higher policy entropy and KL divergence, while we retain the KL loss to provide a valid constraint on policy updates. As the stronger exploration, correspondingly, the generated responses become longer.

Overall, these observations indicate that ICPO facilitates stronger exploration compared to GRPO.

D.2 Pass@32 Performance

We report *Pass@32* results to further analyze the exploration behavior of our ICPO. As shown in Table 7, simply increasing the number of rollouts ($\text{GRPO}_{\text{ExtraRollouts}}$) or directly training on expert-domain data ($\text{GRPO}_{\text{ExpertDomain}}$) yields only limited or inconsistent improvements in *Pass@32* across benchmarks. In contrast, ICPO achieves higher scores across both *Qwen3-8B-Base* and *Qwen3-8B*, indicating that IEF and RS during RL training effectively guides exploration toward expert-aligned regions of the solution space, enlarging the set of reachable correct solutions rather than merely sharpening the policy around a single mode. This behavior aligns with our motivation of leveraging LRM's inherent ICL ability to enhance exploration without relying on advanced LRMs.

D.3 OOD Performance

We evaluate ICPO and the corresponding baselines on out-of-distribution (OOD) benchmarks using *Qwen2.5* models, as shown in Table 2 and Table 8. Although *Qwen2.5-Math-7B* performs reasonably well on in-distribution datasets, it fails to outperform other baselines on OOD evaluations. In contrast, stronger base models such as *Qwen3* achieve consistently superior performance on both in-distribution and OOD settings (see Table 1).

We hypothesize that this discrepancy stems primarily from differences in the base models' inherent ICL capabilities. Models with stronger in-context learning ability exhibit more robust generalization, and consequently achieve better overall performance across both in-distribution and OOD tasks.

D.4 Generalizability Across Models

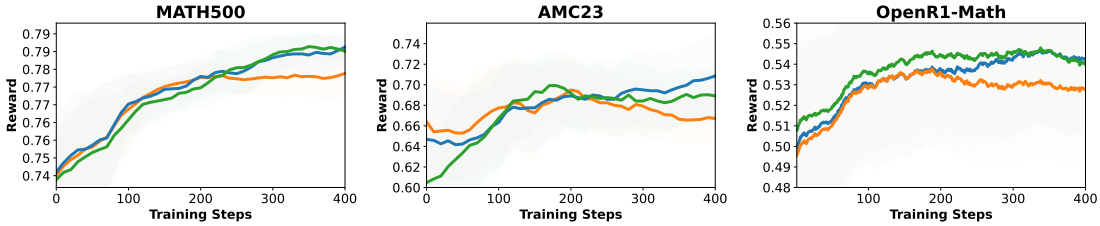
To evaluate the generalizability of our method across various model architectures, we extend ICPO to the base model *Qwen3-8B-Base* and *LLaMA-3.1-8B*. As shown in Table 9 and Table 10, ICPO variants surpass SFT, RL baseline and LUFFY (Yan et al., 2025) on six mathematical benchmarks, underscoring its reasoning performance.

E Ablation Study

E.1 Ablation on Expert Guidance

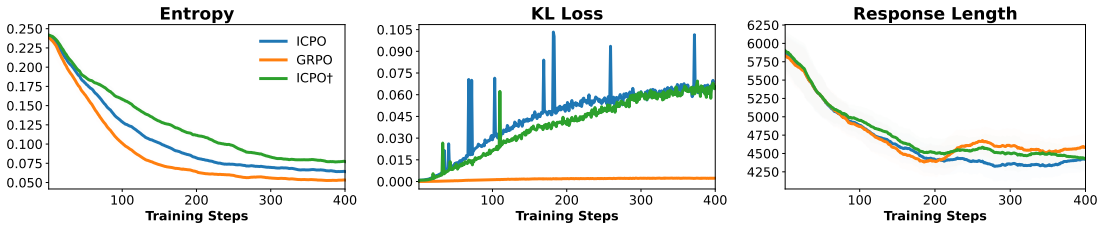
Expert-Domain Data. We study the generalizability of ICPO under different sources of expert-domain data. In addition to our default setting, we adopt *MathInstruct*[3] (Yue et al., 2023) as an alternative expert demonstration corpus for ICPO.

[3] https://huggingface.co/datasets/TIGER-Lab/MathInstruct

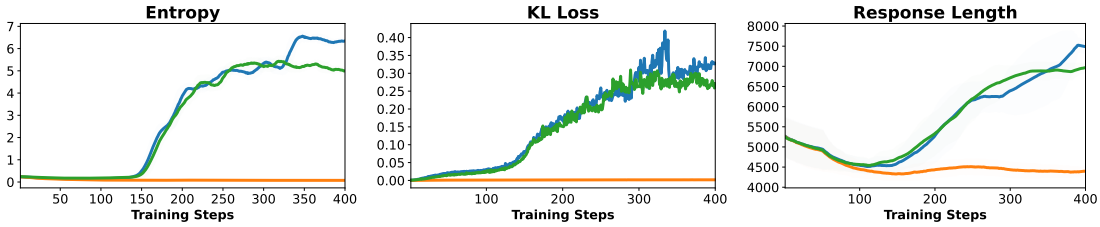


(a) Expert Domain Reward (b) In-Distribution Reward (c) Training Reward

Figure 12: Reward curves of Qwen3-1.7B over training steps across test and train sets.



(a) Training dynamics of Qwen3-1.7B.



(b) Training dynamics of Qwen3-8B.

Figure 13: Training dynamics of ICPO.

MathInstruct contains approximately 262k instruction–solution pairs. To enable reliable extraction of ground-truth answers for RLVR, we select a subset of 118k examples whose solutions explicitly include the phrase "The answer is". As shown in Table 11, ICPO and its variants guided by different expert-domain data sources consistently outperform the GRPO baseline, demonstrating the robustness of our approach to the choice of expert data. Notably, although all evaluations are conducted on in-distribution benchmarks, expert data drawn from different domains exhibits varying strengths across benchmarks. This observation suggests that more carefully curated or benchmark-aligned demonstrations may further improve generalization performance across diverse tasks.

E.2 Ablation on Difficulty Levels

Train Data. We additionally train on a simpler dataset, *Skywork-OR1-RL-Data*[4] (He et al., 2025), to verify the generalization ability of our IEF under different reasoning conditions. This dataset is annotated with difficulty levels predicted by *DeepSeek-R1-Distill-Qwen-7B* (DeepSeek-AI, 2025), and we use the subset with difficulty level = 1, which corresponds to the easiest reasoning problems, as our simplified training corpus for controlled comparison. As shown in Table 12, results verify that ICPO is consistently beneficial across training datasets of different difficulty levels. On the simpler Skywork dataset, ICPO improves GRPO by **+1.9** average points, while on the more challenging OpenR1-Math corpus, the gain further increases to **+3.4** average points. This demonstrates that ICPO not only enhances learning on complex reasoning traces but also generalizes effectively to settings with weaker supervision.

Prompt Groups. Table 13 further analyzes the effectiveness of ICPO by grouping prompts according to their initial rollout success rates. We denote prompts with a success rate of `0.0` as *Non-Pass*, and those with success rates between `0.0` and `1.0` as *Some-Pass*. ICPO applies implicit expert forcing (IEF) across all prompt groups.

[4] https://huggingface.co/datasets/Skywork/OR1-RL-Data

Method	AIME24	AIME25	AMC23	MATH-500	Minerva	Olympiad	Average
Qwen3-8B-Base	46.7	40.0	85.0	94.2	**65.4**	68.3	66.6
GRPO	53.3	43.3	90.0	**96.2**	58.8	72.7	69.1
$\text{GRPO}_{\text{ExtraRollouts}}$	50.0	46.7	92.5	95.4	64.3	**73.8**	70.4
$\text{GRPO}_{\text{ExpertDomain}}$	46.7	26.7	90.0	92.0	61.0	64.2	63.4
ICPO (Ours)	**60.0**	**60.0**	**95.0**	95.8	63.6	73.0	**74.6**
ICPO† (Ours)	56.7	40.0	**95.0**	**96.2**	62.1	72.7	70.5
Qwen3-8B	56.7	36.7	85.0	93.2	57.7	57.5	64.5
GRPO	80.0	60.0	**95.0**	97.0	65.1	79.0	79.3
$\text{GRPO}_{\text{ExtraRollouts}}$	80.0	60.0	**95.0**	97.0	62.9	79.4	79.0
$\text{GRPO}_{\text{ExpertDomain}}$	83.3	66.7	92.5	**97.8**	64.7	77.6	80.4
ICPO (Ours)	**86.7**	66.7	**95.0**	96.8	65.4	75.0	80.9
ICPO† (Ours)	80.0	**70.0**	**95.0**	**97.8**	**67.7**	**81.6**	**82.0**

Table 7: Pass@32 results for Qwen3-8B-Base and Qwen3-8B.

Method	ARC-C	GPQA-D	MMLU-Pro	Average
Qwen2.5-Math-7B (Yang et al., 2025)	18.2	11.1	16.9	15.4
Previous RLVR Methods				
SimpleRL-Zero* (Zeng et al., 2025)	30.2	23.2	34.5	29.3
PRIME-Zero* (Cui et al., 2025)	73.3	18.2	32.7	41.4
OpenReasoner-Zero* (Hu et al., 2025b)	66.2	29.8	58.7	51.6
Oat-Zero* (Liu et al., 2025b)	70.1	23.7	41.7	45.2
SFT and RL				
SFT (Yan et al., 2025)	74.7	28.3	44.4	49.1
RL (Shao et al., 2024)	75.9	32.8	42.6	50.4
SFT+RL (Shao et al., 2024)	79.2	37.9	49.6	55.5
Comparable Baseline Methods				
ReLIFT (Ma et al., 2025)	74.9	40.9	51.9	55.9
LUFFY (Yan et al., 2025)	80.5	39.9	53.0	57.8
Prefix-RFT (Huang et al., 2025)	84.0	39.1	52.1	58.4
ICPO (Ours)	74.0	34.3	46.5	51.6
ICPO† (Ours)	77.0	33.8	47.6	52.8

Table 8: Evaluation results for Qwen2.5-Math-7B. Methods marked with "*" indicate results reported from their original papers.

We observe that expert guidance consistently improves performance for both *Some-Pass* and *Non-Pass* prompts. Moreover, the impact of IEF differs across prompt types and exhibits complementary benefits. For *Some-Pass* prompts, IEF enriches solution diversity while remaining within expert-aligned regions, thereby mitigating premature policy specialization. For *Non-Pass* prompts, IEF provides reliable expert guidance that enables the policy to solve prompts that cannot be handled by vanilla GRPO.

Taken together, these results demonstrate that ICPO enhances exploration for *Some-Pass* cases while enabling effective learning on challenging *Non-Pass* prompts.

F Token Rank Shift Analysis.

To quantify how different training methods alter model behavior at a fine-grained level, we follow Lin et al. (2023) and adopt *token-rank shift* as a token-level diagnostic metric. The core idea is to measure how unlikely the generated tokens of an aligned model are under the *base model* distribution.

Given a generated trajectory produced by an aligned policy, we replay the same trajectory under the base model and examine, at each generation step, how the aligned token is ranked according to the base model's conditional distribution. Let $x_{<t}$ denote the prefix up to position $t-1$, and let y_t be the token generated by the aligned model at step t. We compute the base model logits $\ell^{\text{base}}(\cdot \mid x_{<t})$

Method	AIME24	AIME25	AMC23	MATH-500	Minerva	Olympiad	Average
Qwen3-8B-Base	10.8	10.4	47.6	67.2	32.0	34.4	33.7
GRPO	22.5	21.4	**71.8**	84.6	43.8	50.1	49.0
$\text{GRPO}_{\text{ExtraRollouts}}$	**26.9**	20.2	67.4	85.8	46.7	51.6	49.8
$\text{GRPO}_{\text{ExpertDomain}}$	22.9	12.8	60.9	80.0	47.8	42.8	44.5
LUFFY	24.6	**22.1**	68.2	87.2	47.4	52.3	50.3
ICPO	26.8	21.6	69.2	**88.6**	46.3	**53.8**	**51.0**
ICPO†	25.9	16.9	67.1	86.6	**49.3**	51.3	49.5

Table 9: Results on mathematical reasoning benchmarks for Qwen3-8B-Base.

Method	AIME24	AIME25	AMC23	MATH-500	Minerva	Olympiad	Average
LLaMA-3.1-8B-Instruct	**4.6**	0.2	**20.9**	**45.4**	**22.8**	15.7	**18.3**
SFT*	0.5	0.1	5.4	20.2	4.0	5.3	5.9
GRPO	3.1	**1.0**	10.8	28.2	17.6	**16.2**	12.8
LUFFY*	1.9	0.1	13.5	39.0	15.1	9.6	13.2
ICPO	3.1	0.2	16.6	33.4	21.0	9.0	13.9
ICPO†	0.4	0.1	13.4	38.2	22.4	12.0	14.4

Table 10: Results on mathematical reasoning benchmarks for LLaMA-3.1-8B (Grattafiori et al., 2024). Methods marked with "*" indicate results reported from Yan et al. (2025).

Variant	MATH	AIME24/25	AMC	Mnrv.	Avg.
GRPO	91.0	54.8 / 38.5	83.8	50.7	63.8
***MATH* (Hendrycks et al., 2021)**					
ICPO	**92.0**	55.2 / **43.6**	**87.0**	51.1	**65.8**
ICPO†	**92.0**	56.2 / 40.9	85.4	51.5	65.2
***MathInstruct* (Yue et al., 2023)**					
ICPO	90.8	**56.7** / 40.7	83.8	**53.7**	65.1
ICPO†	90.8	52.4 / 42.3	82.9	51.1	63.9

Table 11: Ablation analysis across different expert-domain data on Qwen3-8B.

Method	MATH	AIME24/25	AMC	Mnrv.	Avg.
OpenR1-Math-220k					
GRPO	83.6	28.4 / 22.5	66.7	40.8	48.4
ICPO	**86.8**	**31.3 / 26.3**	**70.4**	**44.1**	**51.8**
Skywork-OR1-RL-Data					
GRPO	83.0	25.1 / 22.2	66.9	42.3	47.9
ICPO	**86.0**	**26.8 / 24.1**	**69.6**	**42.6**	**49.8**

Table 12: Comparison of ICPO performance on Qwen3-1.7B under two training regimes.

and define the *base-rank* of y_t as:

$$\text{rank}(y_t) \;=\; 1+\Big|\Big\{v \;\Big|\; \ell^{\text{base}}(v \mid x_{<t}) > \ell^{\text{base}}(y_t \mid x_{<t})\Big\}\Big|. \tag{13}$$

A rank of 1 indicates that the aligned token coincides with the base model's top-1 prediction, while larger ranks indicate increasing deviation from the base policy's high-probability region.

For interpretability, we further categorize token ranks into three regimes: (1) **Unshifted**: $\text{rank}(y_t) = 1$; (2) **Marginally shifted**: $1 < \text{rank}(y_t) \leq k$; (3) **Shifted**: $\text{rank}(y_t) > k$, where k is a small threshold (e.g., $k = 3$).

We define the *shifted ratio* as the fraction of tokens whose base-rank exceeds $k = 3$, which serves as a proxy for how frequently the aligned policy samples tokens that would be unlikely under the base model distribution.

Due to the varying lengths of reasoning trajectories, analyzing token rank shift using absolute token positions can be misleading. Instead, we normalize token positions by trajectory length and compute statistics over length percentiles. In Figure 7, we discretize the normalized positions into 100 bins and aggregate token shift metrics within each bin. By aggregating these statistics across token positions, token rank shift enables a fine-grained analysis of *when* and *to what extent* the aligned policy departs from the base policy during generation.

Variant	MATH	AIME24/25	AMC	Mnrv.	Avg.
GRPO	83.0	25.1 / 22.2	66.9	42.3	47.9
+IEF on Some-Pass	82.0	**26.9 / 26.0**	67.8	**43.4**	49.2
+IEF on Non-Pass	82.4	25.42 / 23.33	67.19	42.28	48.12
ICPO	**86.0**	26.8 / 24.1	**69.6**	42.7	**49.8**

Table 13: Ablation study of IEF on Qwen3-1.7B using the *Skywork* dataset, grouped by rollout success rates.