ICPO: Intrinsic Confidence-Driven Group Relative Preference Optimization for Efficient Reinforcement Learning

Anonymous ACL submission

Abstract

Reinforcement Learning with Verifiable Rewards (RLVR) demonstrates significant potential in enhancing the reasoning capabilities of Large Language Models (LLMs). However, existing RLVR methods are often constrained by issues such as sparse rewards, reward noise, and inefficient exploration, which lead to unstable training and entropy collapse. To address this challenge, we propose the **I**ntrinsic **C**onfidence-Driven Group Relative **P**reference **O**ptimization method (ICPO). The intuition behind it lies in the fact that the probabilities of an LLM generating different responses can inherently and directly reflect its self-assessment of the reasoning process. Inspired by the idea of preference modeling, ICPO calculates a preference advantage score for each response by comparing the relative generation probabilities of multiple responses under the same input prompt, and integrates this score with verifiable rewards to guide the exploration process. We have discovered that the preference advantage score not only alleviates the issues of sparse rewards and reward noise but also effectively curbs overconfident errors, enhances the relative superiority of undervalued high-quality responses, and prevents the model from overfitting to specific strategies. Comprehensive experiments across four general-domain benchmarks and three mathematical benchmarks demonstrate that ICPO steadily boosts reasoning compared to GRPO.

1 Introduction

Large-scale reinforcement learning with verifiable rewards (RLVR) has emerged as a prevailing paradigm for enhancing the reasoning capabilities of LLMs (Hu et al., 2025; Luo et al., 2025a; DeepSeek-AI et al., 2025). Unlike reinforcement learning from human feedback (RLHF), RLVR eliminates reliance on subjective human judgments or complex learned reward models by directly employing rule-based reward functions to provide explicit feedback signals for model optimization. This paradigm has given rise to a series of efficient and scalable RLVR training algorithms, such as GRPO (Guo et al., 2024) and DAPO (Yu et al., 2025a). Despite these significant advances, such methods still face several challenging issues.

Firstly, relying solely on final answer to construct binary coarse-grained rewards (DeepSeek-AI et al., 2025) results in locally sparse reward, which fails to effectively distinguish the quality of behaviors and may leads to zero advantage, thereby impeding the policy from achieving a stable optimization direction. Secondly, most RLVR methods are confined to domains such as mathematical problem-solving (Liu et al., 2025; Zeng et al., 2025) and code generation (Luo et al., 2025a; He et al., 2025; Cui et al., 2025a), rendering them incapable of designing rule-based verifiers for general-domain reasoning with free-form answers. Recent studies have attempted to address this issue by employing LLMs as verifiers (Ma et al., 2025). However, the unreliability and high variance of reward signals (reward noise) further undermine training stability. Finally, the model's policy distribution may rapidly skew toward a few high-reward output patterns due to the lack of fine-grained feedback, leading the policy to collapse into a repetitive action selection state—a phenomenon known as entropy collapse.

To address these challenges, we propose the **Intrinsic Confidence-Driven Group Relative Preference Optimization (ICPO)**, which leverages the model's inherent self-assessment capability to compensate for the deficiencies of verifiable rewards. The key intuition lies in the fact that the probability distribution generated by LLMs when producing different reasoning responses essentially represents an implicit self-assessment of the model's confidence in its own reasoning. A higher generation probability means the model deems the reasoning path as highly correct and possesses absolute confidence; however, this often corresponds to the

model's path dependency on familiar patterns, potentially leading to inertial outputs for simple scenarios. Conversely, a lower generation probability may actually come from the model's attempts to reason through complex or rare samples, even when it lacks confidence. This probabilistic preference constitutes a fine-grained signal that can also reflect the effectiveness of policy optimization. Therefore, we can leverage this **intrinsic preference as an auxiliary signal** to guide policy learning.

Specifically, ICPO draws on the modeling approach of pairwise preferences in direct preference optimization algorithms. For multiple in-group responses generated in response to the same input prompt, it first sorts them in ascending order based on their generation probabilities. Subsequently, it forms pairwise response sets by combining the responses two by two within the group. Then, by comparing the probabilities of these pairwise responses, it calculates a **preference advantage score** for each response that reflects its relative superiority or inferiority within the group. During the optimization process, ICPO deeply *integrates the preference advantage score with traditional verifiable external rewards*. This design effectively addresses two typical failure modes in reasoning-based reinforcement learning: (1) It can *precisely identify and suppress overconfident errors* in the model—that is, the model generates seemingly plausible yet actually erroneous reasoning paths with high probability, while simultaneously *enhancing the relative advantage of undervalued high-quality responses*; (2) By *continuously providing fine-grained comparative signals regarding relative merits within groups*, it sustains the exploratory drive of the policy, thereby avoiding training instability and entropy collapse caused by ambiguous or extremely sparse external rewards.

Our core contributions are threefold:

- We innovatively propose a preference advantage score calculation mechanism that transforms the intrinsic probabilities of in-group responses into relative preference signals. This breakthrough overcomes the limitations of traditional verifiable rewards' singularity, providing stable guidance in scenarios where rewards are sparse or noisy. Additionally, it can suppress overconfident errors while enhancing the relative advantage of undervalued high-quality responses, thereby laying a reliable foundation for policy updates.
- We propose the ICPO method, a simple yet highly extensible solution that deeply integrates preference advantage scores with external rewards through multi-stage weight adjustment, effectively balancing the complementary values of intrinsic self-assessment and extrinsic objective verification.
- Comprehensive experiments on seven benchmarks demonstrate that ICPO consistently outperforms baselines (e.g., GRPO) across various model architectures like Qwen, Llama, and Gemma, demonstrating superior effectiveness and cross-model generalizability.

2 Related Works

2.1 RLVR

Reinforcement Learning with Verifiable Rewards (RLVR), as a robust alternative to Reinforcement Learning from Human Feedback (RLHF), has been demonstrated to effectively enhance model reasoning capabilities (Cui et al., 2025a; Lambert et al., 2024; Luo et al., 2025b). While RLHF (Ouyang et al., 2022; Cohen et al., 2022; Gao et al., 2023) leverages human preference data to train reward models—yielding notable improvements—it incurs substantial resource costs due to heavy reliance on manual annotation (Touvron et al., 2023). In contrast, RLVR employs rule-based verification functions (e.g., exact answer matching (Team, 2025; Jaech et al., 2024)) to provide reward signals, circumventing the complexity and potential pitfalls of learned reward models. However, this approach remains limited to domains with precise verifiers and suffers from sparse rewards. Recent studies have explored LLM-based verifiers (LLM-as-a-Judge) (Li et al., 2024; Wang et al., 2024a; Zhu et al., 2025) to extend RLVR to open-ended question-answering scenarios. Yet, due to uncertainty in LLM outputs and hallucination issues, the reward signals exhibit low reliability, potentially misleading policy optimization.

2.2 Self-Reward Optimization

Self-reward optimization via intrinsic signals has emerged as an effective approach to mitigate reliance on extrinsic rewards derived from manual annotation or specialized verification tools. These methods directly extract optimization guidance from the generative process or output features of policy models (Yuan et al., 2024; Zuo et al., 2025; Zhao et al., 2025), inspired by observations that

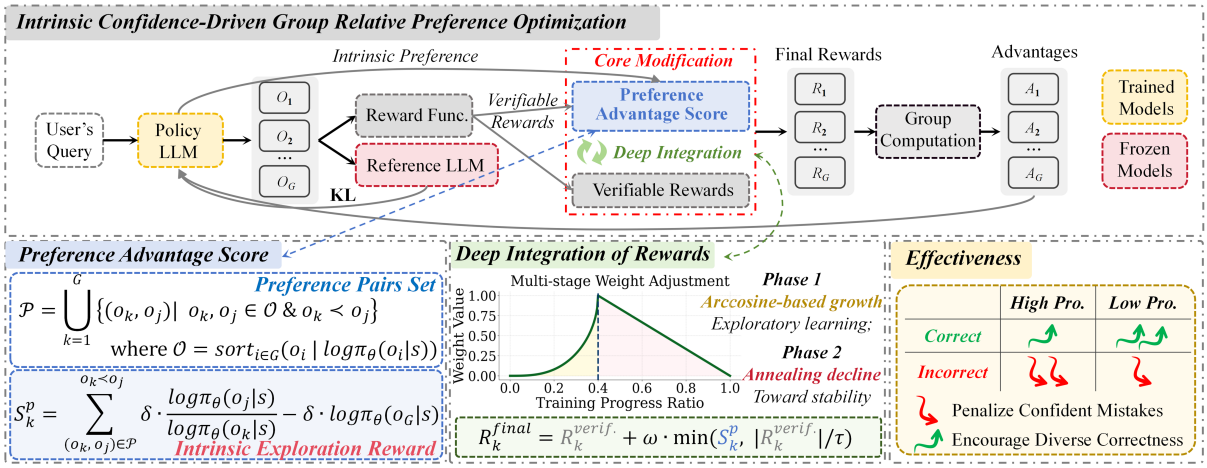


Figure 1: Illustration of ICPO. ICPO computes preference advantage scores for responses within a sampled group, integrates them with verifiable rewards, and stabilizes training via multi-stage weight adjustment.

LLMs exhibit lower confidence when handling complex problems—optimizing such confidence effectively enhances reasoning capabilities (Farquhar et al., 2024; Kuhn et al., 2023; Kang et al., 2024, 2025). Early studies, such as SPIN (Chen et al., 2024) and self-rewarding LLMs (Yuan et al., 2024), leveraged model-generated feedback for subsequent training iterations, while INTUITOR (Zhao et al., 2025) employed self-confidence as an intrinsic confidence-based reward. However, such methods risk limiting exploration (Cui et al., 2025b; Hochlehnert et al., 2025), as excessive reliance on model's own epistemic state may trap the policy in self-reinforcing local optima. PLPR (Yu et al., 2025b) introduced a stable probability-to-reward conversion method. CDE (Dai et al., 2025) employs the Perplexity of responses as an auxiliary signal to guide policy optimization. However, relying on the absolute perplexity of individual responses rather than their relative quality within a group may lead to suboptimal performance.

3 ICPO

In this section, we first introduce the fundamental principles of ICPO. Next, we describe the method for calculating preference advantage scores for each response, and the multi-stage weight adjustment to ensure the stability of training process.

3.1 RL with Intrinsic Confidence-Driven

The core of RLVR lies in replacing traditional RLHF reward models with rule-based reward functions (or general LLM scoring). thereby eliminating the need to construct large-scale human preference datasets. Specifically, RLVR optimizes the policy by maximizing the following objective:

$$\max_{\pi_\theta} \mathbb{E}_{o\sim\pi_\theta(q)} \left[v(q,o) - \beta \mathrm{KL}[\pi_\theta(o|q) \| \pi_{\text{ref}}(o|q)]\right] \tag{1}$$

where q denotes the input query, o represents the generated output, π_{ref} is the initial reference policy, and β controls the KL divergence coefficient to prevent excessive deviation from π_{ref}, $v(q,o)$ constitutes a verifiable reward function. Common RLVR algorithms include REINFORCE (Williams, 1992), PPO (Schulman et al., 2017), and GRPO (Shao et al., 2024). GRPO significantly reduces GPU memory consumption and training computational costs by not needing the value model. Its efficacy has been validated in DeepSeekMath (Shao et al., 2024) and DeepSeek-R1 (DeepSeek-AI et al., 2025), establishing its prominent position in reinforcement learning. Specifically, for each problem q, GRPO samples a set of outputs $\{o_1, o_2, \ldots, o_G\}$ from the old policy $\pi_{\theta_{old}}$, then optimizes the policy model by maximizing the following objective:

$$\mathcal{J}(\theta) = \mathbb{E}_{q\sim D, \{o_k\}_{k=1}^{G}\sim\pi_{\theta_{old}}(O|q)} \Big[\frac{1}{G}\sum_{k=1}^{G}\frac{1}{|o_k|}\sum_{t=1}^{|o_k|} \min\left(r_k^t A_k^t, \mathrm{clip}(r_k^t, 1-\epsilon, 1+\epsilon) A_k^t\right) - \mathbb{KL}\Big] \tag{2}$$

$$r_k^t = \frac{\pi_\theta(o_k^t|q, o_k^{<t})}{\pi_{\theta_{old}}(o_k^t|q, o_k^{<t})}, \quad A_k^t = \frac{R_k - \mathrm{mean}(\{R_k\})}{\mathrm{std}(\{R_k\})} \tag{3}$$

where ϵ and β are hyperparameters, and A_i^t represents the relative advantage estimation based on within-group rewards. However, GRPO suffers from training instability due to sparse rewards or reward noise, and may encounter entropy collapse, leading to suboptimal performance.

To address the above challenges, we propose Intrinsic Confidence-Driven Group Relative Preference Optimization (ICPO), as illustrated in Figure 1. This method integrates external rewards with the model's self-evaluation of its reasoning processes. ICPO offers three key benefits: (1) it provides finer optimization guidance under sparse reward conditions; (2) it mitigates random noise in verifiable rewards and accentuates relative value differences between high- and low-quality responses; (3) it reduces prevalent overconfidence errors in reasoning while enhancing the relative superiority of undervalued high-quality responses. The optimization objective of ICPO is:

$$\mathcal{J}(\theta) = \mathbb{E}_{q\sim D,\{o_k\}_{k=1}^{G}\sim\pi_{\theta_{old}}(O|q)}[\frac{1}{G}\sum_{k=1}^{G}\frac{1}{|o_k|}\sum_{t=1}^{|o_k|} \min\left(r_k^t\tilde{A}_k^t, \text{clip}(r_k^t, 1-\epsilon, 1+\epsilon)\tilde{A}_k^t\right) - \mathbb{KL}] \tag{4}$$

$$\mathbb{KL} = \beta\text{KL}(\pi_\theta||\pi_{\theta_{ref}}), \quad \tilde{A}_k^t = \frac{\tilde{R}_k - \text{mean}(\{\tilde{R}_k\})}{\text{std}(\{\tilde{R}_k\})} \tag{5}$$

where $\tilde{R}_k$ denotes the verifiable reward incorporating normalized preference advantage scores, which is the R_k^{final} depicted in Figure 1.

3.2 Preference Advantage Scores

Preference advantage scores essentially reflect the model's self-assessment of the relative merits of different responses within a group. This intuition stems from the observation that LLMs often exhibit lower probabilities when encountering unfamiliar tasks or lacking sufficient knowledge reserves (Kang et al., 2024). By incorporating self-confidence as an additional reward, ICPO guides the policy to proactively learn from undervalued responses, thereby uncovering underutilized knowledge. Specifically, we first sort responses within a sampled group in ascending order based on their intrinsic preferences (i.e., responses with lower intrinsic probabilities are assigned earlier ranking positions). The sorting process is represented as:

$$\mathcal{O} = \text{sort}_{i\in G}\left(o_i \mid (log\pi_\theta(o_i|s))\right) \tag{6}$$

$$\pi_\theta(o_i|s) = \frac{1}{L_i}\sum_{t=1}^{L_i}\pi_\theta(o_t \mid o_{<t}, s) \tag{7}$$

where L_i represents the effective length of o_i (i.e., the number of non-padding tokens), $\pi_\theta(o_t \mid o_{<t}, s)$ denotes the probability of model generating token o_t at position t, and $\pi_\theta(o_i|s)$ denotes the sequence-level probability of response o_i. We normalize probabilities based on effective length to ensure fair comparison across responses and prevent bias toward shorter responses. This ranking structure, which is solely dominated by generation probabilities, is decoupled from external rewards, thereby avoiding noise interference from external rewards.

Based on the sorting results, we construct all valid preference pairs (o_i, o_j) satisfying the partial order relation where o_i is strictly ranked ahead of o_j. The preference pair set is defined as:

$$\mathcal{P} = \bigcup_{k=1}^{G}\{(o_k, o_j) \mid o_k, o_j \in \mathcal{O}\ \&\ o_k \prec o_j\} \tag{8}$$

Subsequently, we calculate a preference advantage score for each response. By effectively modeling the partial order relations, we quantify the relative superiority or inferiority of each response compared to other candidate responses within the same group. Formally, the preference advantage score for the k-th response is defined as:

$$S_k^p = \sum_{(o_k,o_j)\in\mathcal{P}}^{o_k\prec o_j}\delta\cdot\frac{log\pi_\theta(o_j|s)}{log\pi_\theta(o_k|s)} - \delta\cdot log\pi_\theta(o_G|s) \tag{9}$$

where δ represents the temperature scaling factor. The first term in the formula accumulates the potential learning value of the current response for subsequent responses, while the second term ensures that high-confidence responses are not entirely overlooked. The advantage score S_k^p computed via this method emphasizes guiding the model to learn high-quality responses with low confidence, steering clear of the safe response pattern that merely pursues high probabilities but lacks substantive content. Moreover, it effectively avoids excessive encouragement of extreme low-probability responses, which are of no practical learning value (as the first term in formula of these extremely low-probability responses is relatively small). Consequently, it facilitates the creation of fine-grained intrinsic reward signals, enabling precise capture and learning of high-quality responses underestimated by models.

3.3 Reward Design

To more effectively guide the policy model in exploring and learning underutilized knowledge, we have devised a composite reward function that integrates verifiable rewards derived from external scoring strategies with advantage signals based on

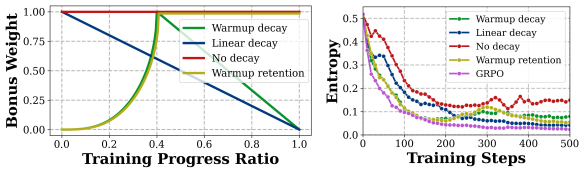


Figure 2: An illustration of different weight adjustment.

intrinsic generation preferences (i.e., preference advantage scores). This reward mechanism not only reflects the absolute performance of responses in the target task but also captures their relative learning value among candidate responses within the same group. The final reward is modeled as:

$$R_k^{verif.} = 0.9 \cdot R_k^{answer} + 0.1 \cdot R_k^{format} \quad (10)$$

$$R_k^{final} = R_k^{verif.} + \omega \cdot \min\left(S_k^p,\ |R_k^{verif.}|/\tau\right) \quad (11)$$

where the weighting parameter ω is utilized to regulate the strength of intrinsic advantage signal injection. We adopt a multi-stage weight adjustment strategy to dynamically modify ω across different training stages, with specific settings detailed in Section 3.4. Additionally, $\tilde{S}_k^p$ is clipped based on the verifiable reward of each individual response, thereby preventing intrinsic preferences from excessively dominating the composite reward. This reward mechanism ensures that encouragement is effectively applied to correct, low-probability responses that possess learning value (while incorrect low-probability responses receive minimal intrinsic rewards due to the truncation mechanism), and it also suppresses the model's overconfident errors.

3.4 Multi-stage Weight Adjustment

To effectively balance the internal and external reward signals, we have designed a multi-stage weight adjustment strategy to dynamically modulate the value of ω, as illustrated by the curve in Figure 1. Specifically, during the initial training phase, we employ an inverse cosine growth pattern to alter the weight value, enabling the model to gradually assimilate fine-grained preference information driven by its own probability distribution. Once ω reaches its maximum value, we adopt a linear annealing to reduce the weight value to its minimum, thereby preventing the training process from being excessively guided by intrinsic rewards.

This design originates from the following insight: In early training, high entropy makes it challenging to determine whether low-probability responses are worth learning. As training continues and entropy drops, correct and novel low-probability responses emerge, at which point we can incrementally enhance the learning motivation for these responses. In later training, when most novel samples are learned, remaining low-probability responses may be noise or valueless signals; so reducing intrinsic signals' influence stabilizes policy learning. Figure 2 (left) displays various weight adjustment strategies, with a detailed analysis of their effects in the experimental section. The warmup turning point is set at 0.4 because, in GRPO training, the 20%-40% phase shows suitable entropy reduction with active exploration, making it ideal for introducing strong intrinsic preference rewards. An ablation study of this setting is in Appendix C.3.

4 Experiments

This section empirically validates the effectiveness of ICPO and explores its potential applications in both verifiable reward and noisy reward domains.

4.1 Baselines and Setup

We conduct experiments on Gemma2 (Riviere et al., 2024), Llama3.1 (Grattafiori et al., 2024) and Qwen2.5 (Yang et al., 2024) series models for fair comparison with existing methods. Unless otherwise specified, experiments are conducted on Qwen2.5-7B-Base. A comprehensive performance evaluation of ICPO method was carried out using three mathematical reasoning benchmarks, namely MATH-500 (Cobbe et al., 2021), Minerva (Lewkowycz et al., 2022), and AIME24, as well as four general-domain benchmarks, including MMLU-Pro (Wang et al., 2024b), GPQA (Rein et al., 2023), TheoremQA (Chen et al., 2023), and WebInstruct. The baseline models include: (1) Base and Instruct models; (2) PRIME (Cui et al., 2025a); (3) SimpleRL-Zoo (Zeng et al., 2025); (4) Oat-Zero (Liu et al., 2025); (5) TTRL (Zuo et al., 2025); (6) General Reasoner (Ma et al., 2025); (7) RLPR (Yu et al., 2025b); (8) INTUITOR (Zhao et al., 2025); and (9) CDE (Dai et al., 2025). Detailed descriptions of datasets and experimental setups are elaborated upon in Appendix A and B.

4.2 Main Results

Main results and training dynamic are presented in Table 1 and Figure 3. Key findings are as follows:

(1) Intrinsic confidence-driven mechanism has substantially enhanced reasoning performance in general domains. Compared to RLVR trained using

Model	Verifier	MMLUPro Avg@2	GPQA Avg@4	TheoremQA Avg@2	WebInst. Avg@2	MATH500 Avg@2	Minerva Avg@2	AIME24 Avg@16	General -	Math -
Gemma Models										
Gemma2-2B-It	–	27.9	19.3	16.4	33.5	26.6	15.9	0.0	24.3	14.2
RLVR(GRPO)†	Rule	31.6	25.8	20.1	52.3	**30.7**	16.5	**0.2**	32.5	15.8
ICPO (Ours)	Rule	**33.0**	**28.9**	**20.4**	**53.8**	30.6	**17.0**	**0.2**	**34.0**	**15.9**
Llama Models										
Llama3.1-8B-Inst	–	46.4	31.6	31.3	54.7	50.1	32.7	4.2	40.5	29.0
RLVR(GRPO)†	Rule	50.3	36.0	33.0	63.2	51.9	35.2	6.5	45.6	31.2
ICPO (Ours)	Rule	**52.7**	**37.2**	**34.7**	**68.9**	**53.8**	**39.0**	**8.8**	**48.4**	**33.9**
Qwen Models										
Qwen2.5-7B	–	45.3	32.4	41.4	60.4	51.0	37.6	6.5	44.9	31.7
Qwen2.5-7B-Inst	–	54.5	34.2	47.3	72.6	75.4	49.4	9.4	52.2	44.7
Orat-Zero(M)	Rule	45.8	38.8	53.3	71.5	80.8	52.1	**29.8**	52.4	**54.2**
PRIME(M)	Rule	39.5	32.1	47.7	54.5	76.4	45.5	20.4	43.4	47.4
SimpleRL-Zoo(M)	Rule	46.9	38.4	51.1	70.3	77.1	51.0	26.5	51.7	51.5
TTRL	Rule	51.1	34.1	48.8	68.0	**82.1**	52.8	15.8	50.5	50.2
SimpleRL-Zoo	Rule	54.1	36.2	49.5	70.7	76.3	49.2	14.8	52.6	46.8
General Reasoner	Model	55.4	37.4	52.1	74.5	77.0	51.7	16.0	54.8	48.2
RLPR	✗	56.0	37.6	**55.4**	75.5	78.0	**56.5**	16.3	56.1	50.3
INTUITOR†	✗	54.9	37.3	53.0	76.1	76.5	53.6	16.0	55.3	48.7
CDE†	Rule	57.4	**40.1**	54.0	72.3	76.0	53.7	23.3	55.9	51.0
RLVR(GRPO)†	Rule	55.1	36.2	52.2	75.3	76.5	54.9	17.7	54.7	49.7
ICPO (Ours)	Rule	**57.6**	39.4	54.0	**77.8**	76.2	56.2	23.3	**57.2**	51.9

Table 1: Overall performance on seven reasoning benchmarks. General: Average of MMLU-Pro, GPQA, TheoremQA and WebInst. Math: Average of MATH-500, Minerva and AIME24. The best and second results are marked in **bold** and underlined, respectively. ✗: Method does not require verifiers. (M): Method is trained based on Qwen2.5-Math-7B. †: Official code/methods-based reproduced results.

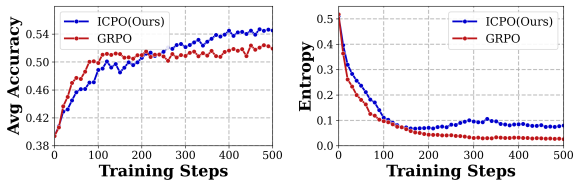


Figure 3: Comparison of Avg accuracy on seven benchmarks and entropy of GRPO (Baseline) and ICPO (Our).

Methods	MMLUPro Avg@2	GPQA Avg@4	MATH500 Avg@2	Minerve Avg@16
Baseline	45.3	32.4	51.0	37.6
+ GRPO	46.9	31.8	56.2	38.9
+ ICPO	**52.3**	**35.0**	**68.5**	**45.5**

Table 2: Performance on sparse reward settings.

the vanilla GRPO, we observed greater improvements in general reasoning performance across the Gemma, Llama, and Qwen models, with average increases of 1.5, 2.8, and 2.5 points, respectively. Furthermore, ICPO also demonstrated a significant advantage over other baseline methods.

(2) ICPO demonstrates exceptional performance in mathematical reasoning scenarios, surpassing both PRIME and SimpleRL-Zoo across three mathematical benchmarks. Notably, even without training on mathematical models, ICPO still demonstrates a significant enhancement in mathematical reasoning capability, comparable to methods specifically trained on mathematical models.

(3) During the training process, ICPO exhibits a slower rate of improvement in test set accuracy compared to the vanilla GRPO in the early training stages. However, it gradually catches up and ultimately surpasses the performance of GRPO. Moreover, the policy entropy is maintained at a stable exploratory state throughout the entire training process. This aligns with intrinsic confidence-driven exploration: ICPO prevents premature convergence to false high-reward paths, encouraging thorough exploration of high-quality actions in low-confidence regions. This helps the model shift smoothly toward goal-directed exploration, reducing entropy collapse and boosting accuracy.

4.3 Analysis of ICPO

(1) Beyond Binary Rewards: Providing Continuous Optimization Signals. We filtered the training data, keeping only samples where all instances were entirely correct or entirely incorrect, and used them as extremely sparse reward training set. As

Model	Verifier	MMLUPro Avg@2	GPQA Avg@4	TheoremQA Avg@2	WebInst. Avg@2	MATH500 Avg@2	Minerva Avg@2	AIME24 Avg@16	General -	Math -
Qwen2.5-Base Models										
Qwen2.5-3B	–	34.6	26.3	27.4	44.5	42.6	19.4	0.0	33.2	21.3
+ GRPO	Rule	50.1	30.4	37.6	61.1	**60.8**	35.4	**10.0**	44.8	**35.4**
+ ICPO	Rule	**51.7**	**32.9**	**39.1**	**63.2**	60.4	**36.5**	9.4	**46.7**	**35.4**
Qwen2.5-7B	–	45.3	32.4	41.4	60.4	51.0	37.6	6.5	44.9	31.7
+ GRPO	Rule	55.1	36.2	52.2	75.3	**76.5**	54.9	17.7	54.7	49.7
+ ICPO	Rule	**57.6**	**39.4**	**54.0**	**77.8**	76.2	**56.2**	**23.3**	**57.2**	**51.9**
Qwen2.5-14B	–	51.2	32.8	43.0	66.1	55.6	41.7	10.0	48.3	35.8
+ GRPO	Rule	**63.2**	43.6	**58.8**	79.0	80.4	59.2	**20.0**	61.1	53.2
+ ICPO	Rule	63.0	**45.1**	**58.8**	**80.3**	**81.2**	**60.5**	**20.0**	**61.8**	**53.9**
Qwen2.5-Instruct Models										
Qwen2.5-7B-Inst	–	54.5	34.2	47.3	72.6	75.4	49.4	9.4	52.2	44.7
+ GRPO	Rule	52.0	40.1	**56.3**	63.2	77.0	**55.8**	17.7	52.9	50.2
+ ICPO	Rule	**58.1**	**42.6**	55.4	**74.9**	**77.8**	**55.8**	**26.7**	**57.8**	**53.4**
Qwen3-Base Models										
Qwen3-4B	–	50.5	36.8	33.7	55.7	54.1	29.4	3.33	44.2	28.9
+ GRPO	Rule	57.3	41.5	53.3	65.8	78.5	54.7	16.7	54.5	49.9
+ ICPO	Rule	**59.5**	**43.9**	**56.8**	**73.1**	**81.7**	**57.8**	**20.0**	**58.3**	**53.2**

Table 3: Overall performance comparison between ICPO (Our Method) and vanilla GRPO (Baseline) across models of different scales (3B, 7B and 14B), types (Base and Instruct) and versions (Qwen2.5 and Qwen3).

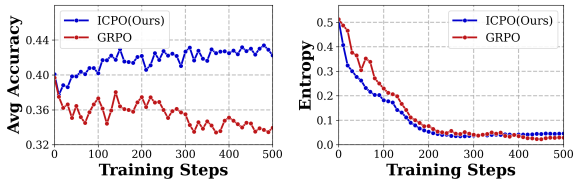


Figure 4: Comparison of Avg accuracy and entropy between vanilla GRPO (baseline) and ICPO (Our method) on seven benchmarks under noisy reward scenarios.

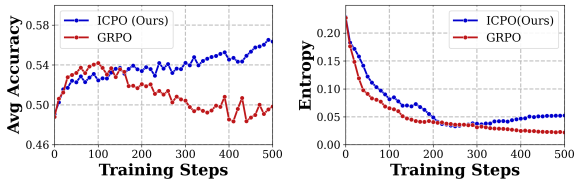


Figure 5: Comparison of Avg accuracy and entropy between vanilla GRPO (baseline) and ICPO (Our method) across seven benchmarks using Qwen2.5-7B-Instruct.

shown in Table 2, compared to GRPO, ICPO still effectively explores under complete reward sparsity: it progressively identifies high-potential response paths and ultimately achieves stable performance gains. In contrast, GRPO, lacking fine-grained feedback, fails to pinpoint accurate policy update directions, resulting in stagnant or even deteriorating performance. These findings confirm that ICPO, through implicit contrastive modeling, effectively mitigates the exploration-exploitation dilemma in extremely sparse reward settings, offering a viable optimization approach for scenarios without explicit preference labels.

(2) Correcting Reward Bias: Guiding Toward the Right Optimization Direction. Figure 4 compares the performance of ICPO and vanilla GRPO on noisy reward scenarios. In scenarios without precise verification tools, LLMs automatically generate reward signals. However, due to LLMs' inherent uncertainty or hallucination issues, these rewards are often noisy and may even unfairly assign low scores to high-quality outputs due to assessment model bias. To validate the effectiveness of ICPO in such contexts, we randomly selected 40% of the training data in each update round and injected random noise into their rewards to simulate this scenario. The experimental results demonstrate that ICPO, leveraging its intrinsic confidence-driven mechanism, effectively decouples the strong dependency between policy confidence and external rewards, maintaining positive reinforcement for high-potential responses and enabling effective learning even when external rewards are distorted. In contrast, GRPO, lacking accurate advantage signals, experiences a sharp increase in estimation variance, leading to rapid misalignment of policy update directions and severe performance degradation. These findings further corroborate the effectiveness of ICPO in mitigating reward noise.

4.4 Natural Scalability

ICPO exhibits scalability and can effectively adapt to models of varying scales. As shown in Table 3,

Model	MMLUPro Avg@2	GPQA Avg@4	TheoremQA Avg@2	WebInst. Avg@2	MATH500 Avg@2	Minerva Avg@2	AIME24 Avg@16	General -	Math -
Qwen2.5-7B-GRPO	55.1	36.2	52.2	75.3	**76.5**	54.9	17.7	54.7	49.7
$\Rightarrow \omega$ No decay	54.8	35.8	**54.0**	77.0	76.0	55.9	16.7	55.4	49.5
$\Rightarrow \omega$ Linear decay	56.0	39.0	53.5	75.8	76.2	55.4	20.0	56.1	50.5
$\Rightarrow \omega$ Warmup retention	57.0	38.7	53.5	76.2	**76.5**	55.4	20.0	56.4	50.6
$\Rightarrow \omega$ **Warmup decay**	**57.6**	**39.4**	**54.0**	**77.8**	76.2	**56.2**	**23.3**	**57.2**	**51.9**

Table 4: Performance comparison of ICPO under different weight adjustment schemes for intrinsic confidence reward. The weight adjustment schemes adhere to those illustrated in Figure 2.

with the growth of model parameters (from 3B to 7B, then to 14B), its performance keeps improving, fully reflecting ICPO's inherent scalability. An interesting finding is that ICPO shows the greatest performance gain on 7B-scale model, with smaller improvements on 3B and 14B-scale models. We posit that this may stem from the non-linear relationship between model scale and confidence calibration: small models lack confidence, providing uncertain probability signals for many questions, while large models tend to be overconfident (Huang et al., 2025), assigning high confidence even to wrong answers. Consequently, both types of models exhibit relatively flat probability distributions, resulting in small in-group confidence differences.

Furthermore, ICPO continues to demonstrate its versatility and robustness on both the Qwen2.5-7B-Instruct model and the Qwen3-4B-base model. Whether applied to interactive models emphasizing instruction-following capabilities or foundational reasoning models focusing on thinking abilities, ICPO can seamlessly adapt and effectively enhance their performance in core tasks.

Notably, GRPO training on Qwen2.5-7B-Inst. shows sustained performance decline in later stages (as shown in Figure 5), whereas ICPO maintains stable exploration. We attribute this to the substantial performance improvement of the instruction-tuned model, which results in smaller or even zero relative advantages among responses within groups. This further validating the robustness and anti-degradation benefits of intrinsic confidence-driven mechanism in extremely sparse reward scenarios.

4.5 Investigate the Effect of Intrinsic Bonus

Reward weight adjustment is crucial. We compared four adjustment schemes of ω: No decay, linear decay, warmup retention, and warmup decay (as shown in Figure 2 (left)). Table 4 presents the performance under each scheme and uncovers two key findings: (1) Weight decay is essential since all decay schemes outperform the no decay baseline, facilitating a smooth shift from exploration to exploitation. (2) Early-stage warmup exploration matters. Warmup decay scheme expands the state-action coverage by gradually increasing the exploration intensity in early phase and ensures stable convergence through linear decay in later phase. In contrast, schemes without warmup exhibit an excessively high exploration intensity at the initial stage, leading to an imbalanced policy update direction and a consequent reduction in performance.

Additionally, as previously indicated by research, entropy offers a crucial perspective for understanding exploration capabilities (Cui et al., 2025b). A sharp decline or high-level fluctuation in entropy typically indicates insufficient model exploration or policy failure. Figure 2 (right) presents a comparative analysis of entropy dynamics between vanilla GRPO and ICPO. Firstly, the intrinsic confidence-driven mechanism effectively mitigates the phenomenon of entropy collapse, fully demonstrating its role in promoting exploration. Secondly, when comparing different decay schemes, the warmup decay approach yields a more stable entropy trajectory. This finding aligns with our understanding: prioritizing warmup and appropriately decaying reward weights ensures stable convergence while effectively supporting the exploration process.

5 Conclusion

We propose an Intrinsic Confidence-Driven Group Relative Preference Optimization (ICPO), an efficient technique that enhances LLMs learning by modeling preferences based on the generation probabilities of responses within a group. ICPO is lightweight, requiring only minor adjustments to the original GRPO training framework. Its efficacy is demonstrated by consistent accuracy improvements across general and mathematical benchmarks, outperforming strong baselines. These results strongly support the intuition that intrinsic confidence contains rich preference information to effectively guide policy optimization.

Limitations

Although ICPO can achieve effective learning in sparse reward settings and mitigate the problem of noisy rewards, it still encounters limitations and challenges when rewards are completely inaccurate. The fundamental limitation is that ICPO inherently promotes in-depth learning of correct but low-probability samples while curbing overconfidence in incorrect yet high-probability ones. When rewards completely deviate from the true task objectives, this **encouragement-suppression** mechanism experiences a total reversal. Specifically, genuinely valid low-probability behaviors are incorrectly marked as erroneous samples by the inaccurate rewards, while truly ineffective or even detrimental behaviors are falsely portrayed as correct but low-probability high-quality samples and become the focal point of ICPO's optimization, ultimately causing policy failure. We are currently actively exploring solutions to these issues and will present them in our future research.

References

Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. 2023. Theoremqa: A theorem-driven question answering dataset. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023*, pages 7889–7901.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024. Self-play fine-tuning converts weak language models to strong language models. *CoRR*, abs/2401.01335.

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. *CoRR*, abs/2110.14168.

Deborah Cohen, Moonkyung Ryu, Yinlam Chow, Orgad Keller, Ido Greenberg, Avinatan Hassidim, Michael Fink, Yossi Matias, Idan Szpektor, Craig Boutilier, and Gal Elidan. 2022. Dynamic planning in open-ended dialogue using reinforcement learning. *CoRR*, abs/2208.02294.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, and et al. 2025a. Process reinforcement through implicit rewards. *CoRR*, abs/2502.01456.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, Zhiyuan Liu, Hao Peng, Lei Bai, Wanli Ouyang, Yu Cheng, Bowen Zhou, and Ning Ding. 2025b. The entropy mechanism of reinforcement learning for reasoning language models. *CoRR*, abs/2505.22617.

Runpeng Dai, Linfeng Song, Haolin Liu, Zhenwen Liang, Dian Yu, Haitao Mi, Zhaopeng Tu, Rui Liu, Tong Zheng, Hongtu Zhu, and Dong Yu. 2025. Cde: Curiosity-driven exploration for efficient reinforcement learning in large language models. *CoRR*, abs/2509.09675.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Leo Gao, John Schulman, and Jacob Hilton. 2023. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning, ICML 2023*, pages 10835–10866.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, and et al. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Y. Wu, Y. K. Li, Fuli Luo, Yingfei Xiong, and Wenfeng Liang. 2024. Deepseek-coder: When the large language model meets programming - the rise of code intelligence. *CoRR*, abs/2401.14196.

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, and et al. 2025. Skywork open reasoner series. Notion Blog.

Andreas Hochlehnert, Hardik Bhatnagar, Vishaal Udandarao, Samuel Albanie, Ameya Prabhu, and Matthias Bethge. 2025. A sober look at progress in language model reasoning: Pitfalls and paths to reproducibility. *CoRR*, abs/2504.07086.

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

Yin Huang, Yifan Ethan Xu, Kai Sun, Vera Yan, Alicia Sun, Haidar Khan, Jimmy Nguyen, Jingxiang Chen, Mohammad Kachuee, Zhaojiang Lin, Yue Liu, Aaron Colak, Anuj Kumar, Wen tau Yih, and Xin Luna Dong. 2025. Confrag: Confidence-guided retrieval-augmenting generation. *CoRR*, abs/2506.07309.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, and et al. 2024. Openai o1 system card. *CoRR*, abs/2412.16720.

Katie Kang, Eric Wallace, Claire Tomlin, Aviral Kumar, and Sergey Levine. 2024. Unfamiliar finetuning examples control how language models hallucinate. *CoRR*, abs/2403.05612.

Zhewei Kang, Xuandong Zhao, and Dawn Song. 2025. Scalable best-of-n selection for large language models via self-certainty. *CoRR*, abs/2502.18581.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *CoRR*, abs/2302.09664.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and et al. 2024. TÜlu 3: Pushing frontiers in open language model post-training. *CoRR*, abs/2411.15124.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, and 1 others. 2022. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, Kai Shu, Lu Cheng, and Huan Liu. 2024. From generation to judgment: Opportunities and challenges of llm-as-a-judge. *CoRR*, abs/2411.16594.

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

Michael Luo, Sijun Tan, Roy HuangandAmeen Patel, Alpay Ariyak, Qingyang Wu, Xiaoxiang Shi, Rachel Xin, Colin Cai, Maurice Weber, Ce Zhang, Li Erran Li, Raluca Ada Popa, and Ion Stoica. 2025a. Deepcoder: A fully open-source 14b coder at o3-mini level. Notion Blog.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, and et al. 2025b. Deepscaler: Surpassing o1-preview with a 1.5 b model by scaling rl. Notion Blog.

Xueguang Ma, Qian Liu, Dongfu Jiang, Ge Zhang, Zejun Ma, and Wenhu Chen. 2025. General-reasoner: Advancing llm reasoning across all domains. *CoRR*, abs/2505.14652.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, and et al. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022*.

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 qa benchmark. *CoRR*, abs/2311.12022.

Gemma Team: Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, and et al. 2024. Gemma 2: Improving open language models at a practical size. *CoRR*, abs/2408.00118.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *CoRR*, abs/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. *CoRR*, abs/2402.03300.

Xiaomi LLM-Core Team. 2025. Mimo: Unlocking the reasoning potential of language model – from pretraining to posttraining. Https://github.com/XiaomiMiMo/MiMo.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Yidong Wang, Zhuohao Yu, Wenjin Yao, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, and et al. 2024a. Pandalm: An automatic evaluation benchmark for llm instruction tuning optimization. In *The Twelfth International Conference on Learning Representations, ICLR 2024*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, and 1 others. 2024b. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290.

Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256.

Qwen: An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, and et al. 2024. Qwen2.5 technical report. *CoRR*, abs/2412.15115.

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

Tianyu Yu, Bo Ji, Shouli Wang, Shu Yao, Zefan Wang, Ganqu Cui, Lifan Yuan, Ning Ding, Yuan Yao, Zhiyuan Liu, Maosong Sun, and Tat-Seng Chua. 2025b. Rlpr: Extrapolating rlvr to general domains without verifiers. *CoRR*, abs/2506.18254.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. In *Forty-first International Conference on Machine Learning, ICML 2024*.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. 2025. Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *CoRR*, abs/2503.18892.

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

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2025. Judgelm: Fine-tuned large language models are scalable judges. In *The Thirteenth International Conference on Learning Representations, ICLR 2025*.

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. 2025. Ttrl: Test-time reinforcement learning. *CoRR*, abs/2504.16084.

A Experimental data

A.1 Training data

Our training data is identical to that employed in RLPR (Yu et al., 2025b), utilizing the high-quality reasoning question bank released by Ma et al. (Ma et al., 2025), which encompasses high-caliber reasoning problems across multiple complex domains. This training dataset excludes mathematics-related prompts to focus on general-domain reasoning and adopts a multi-stage filtering strategy to ensure its difficulty level: initially, history-related questions and those at primary/junior high school levels are removed to avoid overly simplistic or commonsense content; subsequently, only high-difficulty samples are retained based on reasoning scores assigned by GPT-4.1-mini. Ultimately, this process yields 77,687 focused and high-quality complex non-mathematical reasoning data entries.

A.2 Evaluation data

We evaluate reasoning capabilities through multiple general reasoning benchmarks and mathematical benchmarks. For mathematical reasoning, we have selected the following three benchmarks:

- MATH-500 (Cobbe et al., 2021) is a challenging benchmark specifically designed for evaluating the mathematical reasoning capabilities of LLMs. It comprises 500 high-quality, challenging mathematical problems, aiming to comprehensively assess a model's ability to solve complex mathematical issues.

- Minerva (Lewkowycz et al., 2022), developed by Google DeepMind, is a high-quality dataset for training mathematical reasoning models. It aims to narrow the capability gap of LLMs in quantitative reasoning, empowering models with mathematical thinking and symbolic reasoning skills from basic calculations to advanced scientific research levels.

- AIME24 centers on intricate problems at the AIME (American Invitational Mathematics Examination) level, covering key areas like algebra, geometry, and combinatorics. It offers structured data, including problem descriptions, solution steps, and standard answers, setting a new standard for assessing AI models' logical reasoning skills.

For general domains, we adopt four benchmarks:

- MMLU-Pro (Wang et al., 2024b) is a widely-adopted multi-task language understanding benchmark, encompassing challenging reasoning-intensive questions across multiple domains. To strike a balance between evaluation efficiency and data diversity, we randomly sampled 1,000 prompts from this benchmark for our experiments.

- GPQA (Rein et al., 2023) encompasses graduate-level questions across multiple disciplines such as physics and chemistry. We employ the highest-quality GPQA-diamond subset for our evaluation.

- TheoremQA (Chen et al., 2023) is designed to evaluate a model's ability to solve complex scientific problems by applying theorems. This benchmark comprises 800 high-quality questions, covering 350 theorems across fields

ICPO training prompt

```
<|im_start|>system
A conversation between User and Assistant. The user asks a question, and the
Assistant solves it. The assistant first thinks about the reasoning process in the
mind and then provides the user with the answer. The reasoning process and answer
are enclosed within <think> </think> and <answer> </answer> tags, respectively,
i.e., <think> reasoning process here </think> <answer> answer here </answer>.
<|im_end|>
<|im_start|>user
{{question}}<|im_end|>
<|im_start|>assistant
```

Table 5: We adopt the training prompt of R1 (DeepSeek-AI et al., 2025) for ICPO.

such as mathematics and physics. We remove the 53 multimodal instructions.

- WebInstruct: We derived a validation subset from WebInstruct (Ma et al., 2025) to serve as an easily evaluable benchmark tailored for medium-sized models. Despite its relatively lower difficulty, the dataset can still evaluate multidisciplinary reasoning skills. Adhering to the filtering settings introduced in RLPR (Yu et al., 2025b), we conducted our evaluation using 638 unique questions.

B Experimental Setup

B.1 Training Setup

Unless otherwise specified, all our experiments were conducted on the Qwen2.5-7B model. Following the majority of RLVR practices, we omitted the supervised fine-tuning process and directly performed post-training on the base model. In the main experiment, we controlled the output structure to extract parsable reasoning chains and answers by adjusting the prompt templates during both the training and validation phases. The specific prompt templates are detailed in Table 5. All experimental results presented in this paper are averaged over multiple trials.

In the experiments targeting the Gemma and Llama models, we adjusted the temperature parameters for both training and evaluation to 0.6. Following RLPR (Yu et al., 2025b) guidelines, we removed `<think>` segments from templates to prevent a decline in generation quality. We observed that rule-based scoring scripts introduced errors in benchmark tests containing non-multiple-choice formatted questions. To address this issue, we deployed a Qwen2.5-7B-Instruct model server for evaluation and additionally utilized Qwen2.5-72B-Instruct to handle more complex benchmarks, such as TheoremQA and Minerva.

In the experiment focusing on sparse reward scenarios, we employed Qwen2.5-7B model to conduct a single-epoch training session on the training set. We retained the original data where all responses within each group were either uniformly correct or uniformly incorrect, while filtering out those data instances where responses within the same group exhibited discrepancies in rewards. Subsequently, we utilized the retained data as extremely sparse reward training set to further train Qwen2.5-7B model, thereby validating the efficacy of ICPO in extremely sparse reward settings.

In the experiment concerning noisy reward scenarios, we randomly selected 40% of the responses generated at each training step and injected random noise into their rewards. Specifically, we randomly added or subtracted 0.3 points from the original reward values to simulate the hallucinations and unstable fluctuations that may occur when LLMs assign scores. Subsequently, we utilized the noise-injected rewards to calculate the intra-group advantage for guiding policy updates, thereby validating the comparative performance of ICPO and vanilla GRPO in noisy reward environments.

B.2 Parameters Setup

We employed Verl as our training framework[1]. Table 6 presents the configuration schemes adopted for training both ICPO and vanilla GRPO based on Qwen2.5 series models. All experiments were conducted on a single node, with each node equipped with 8 NVIDIA H200 141GB GPUs.

[1] https://github.com/volcengine/verl

Config	GRPO	ICPO
actor-lr	1e-6	1e-6
kl_coef	0.001	0.001
max_prompt_length	2K	2K
max_response_length	3K	3K
train_batch_size	768	768
ppo_mini_batch_size	192	192
clip_ratio	0.20	0.20
sample_temperature	1.0	1.0
rollout.n	5	5
total_training_steps	500	500

(a)

Config	3B	7B	14B
δ	0.5	0.4	0.3
τ	2.0	2.0	2.0
ω	Warmup decay	Warmup decay	Warmup decay

(b)

Table 6: (a) General Training Configurations for the Qwen2.5 Series Models: Training parameters shared by both ICPO and GRPO. These settings are uniformly applied across all methods based on GRPO and ICPO (e.g., "Qwen2.5-3B (7B, 14B)-GRPO" and "Qwen2.5-3B (7B, 14B)-ICPO" in Table 3). (b) ICPO-Specific Training Configurations for the Qwen2.5 Series Models: Unique training parameters exclusive to ICPO.

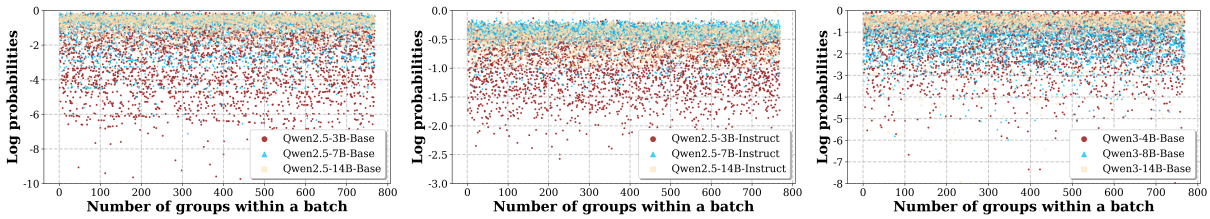


Figure 6: Log-probability distribution of output responses from models of different scales. From left to right are the Qwen2.5-Base series, Qwen2.5-Instruct series, and Qwen3-Base series.

C Hyperparameter Search Experiment

We conducted additional hyperparameter search experiments focusing on the following hyperparameters: (1) Selection method of parameter δ; (2) Impact of τ on model performance; and (3) Analysis of the weight adjustment approach for ω.

C.1 Selection method of parameter δ

Table 7 presents a performance comparison of δ across models of different types and scales. Through analysis of the experimental results, we observe that the optimal value of δ varies across models of different scales. However, for models of different types (Qwen2.5-Base, Qwen2.5-Instruct, and Qwen3-Base), the optimal values of δ are approximately the same under the same scale.

We analyze that the possible reason lies in the fact that the probability distributions of model outputs under the same scale are roughly similar, even when the model types differ. To verify this hypothesis, we conducted a statistical analysis of the logarithmic probability distributions output by different models and plotted scatter diagrams, as illustrated in Figure 6. The experimental results robustly validate our hypothesis and reveal a key insight: Smaller models exhibit low confidence, providing uncertain probabilistic signals for many queries, whereas larger models tend toward overconfidence, assigning high probabilities even to incorrect answers. Models of moderate scale (4B-8B parameters) demonstrate optimal consistency in response probability distributions. Moreover, for models of comparable scales, their output probability distributions exhibit substantial similarity. Consequently, the configuration of parameter δ demonstrates general applicability across different model types. Specifically, we recommend setting $\delta = 0.5$ for models with fewer than 4B parameters, $\delta = 0.4$ for models ranging from 4B to 8B parameters, and $\delta = 0.3$ for models exceeding 8B parameters.

C.2 Impact of parameter τ on performance

The core function of parameter τ is to regulate the weight allocation of intrinsic confidence rewards within the overall reward, thereby determining its proportion of influence on the policy update process. If τ is set too large, the policy update will be excessively dominated by intrinsic preferences,

Model	GRPO	ICPO				
		$\delta = 0.2$	$\delta = 0.3$	$\delta = 0.4$	$\delta = 0.5$	$\delta = 0.6$
Qwen2.5-Base Models						
Qwen2.5-3B-Base	40.10	40.30	40.88	40.70	**41.10**	40.85
Qwen2.5-7B-Base	52.20	52.03	54.23	**54.90**	53.65	52.90
Qwen2.5-14B-Base	57.05	57.63	**57.85**	57.24	57.30	56.45
Qwen2.5-Instruct Models						
Qwen2.5-3B-Instruct	42.40	42.15	43.33	43.67	**44.10**	43.90
Qwen2.5-7B-Instruct	51.55	53.65	55.47	**55.60**	54.70	51.90
Qwen2.5-14B-Instruct	60.01	61.05	**61.40**	60.38	59.70	60.13
Qwen3-Base Models						
Qwen3-4B-Base	49.55	52.20	53.35	**54.05**	53.90	52.67
Qwen3-8B-Base	57.30	59.27	59.15	**59.93**	58.80	57.33
Qwen3-14B-Base	58.93	59.70	**60.57**	59.48	59.20	58.75

Table 7: The impact of parameter δ on performance across models of different scales (3B, 7B and 14B), types (Base and Instruct) and versions (Qwen2.5 and Qwen3), where the performance is the average across seven benchmarks.

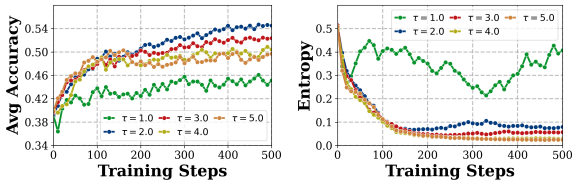


Figure 7: Impact of parameter τ on performance.

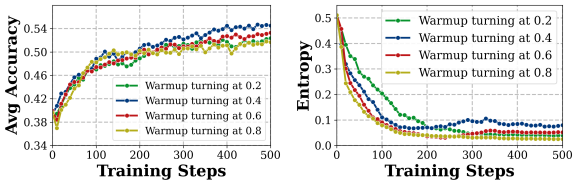


Figure 8: Parameter search for warm-up turning point.

making it difficult to ensure the performance effectiveness of the policy in real-world task scenarios. Conversely, if τ is set too small, the guiding role of intrinsic preferences in the optimization process will be weakened, failing to effectively calibrate the direction of policy updates. We set the values of parameter τ to 1.0, 2.0, 3.0, 4.0, and 5.0 respectively, and observed the changes in the performance of ICPO under different settings. The experimental results are shown in Figure 7.

From the results, we observe that the model achieves optimal performance when the parameter $\tau = 2.0$, with $\tau = 3.0$ yielding the second-best yet still competitive performance. In contrast, setting $\tau = 1.0$ leads to a significant degradation in both policy entropy and overall performance, indicating that over-reliance on intrinsic preferences for guiding policy updates destabilizes training. When τ is further increased to 4.0 or 5.0, the model exhibits rapid performance gains in the early stages of training but subsequently plateaus, accompanied by a sharp decline in policy entropy. This suggests that excessively downweighting the influence of intrinsic preferences during policy updates undermines the agent's exploratory capacity. Collectively, these findings demonstrate that appropriately leveraging intrinsic preferences to guide policy updates helps mitigate entropy collapse and effectively preserves policy exploration.

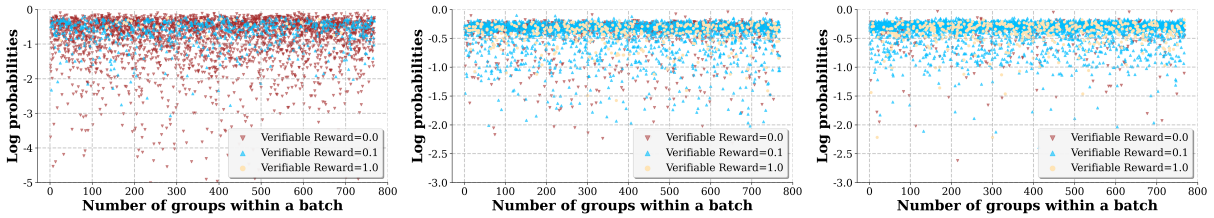
C.3 Analysis of Weight Adjustment Strategies

Table 8 presents the hyperparameter search experiments for the weight adjustment strategies across different models. Figure 8 illustrates the parameter search experiment for the warm-up turning point. From these, we observe that: (1) Weight decay mechanism is of critical importance. For Qwen2.5-7B-Base, Qwen2.5-7B-Instruct, and Qwen3-4B-Base, all decay schemes outperform the no-decay baseline, enabling a smooth transition from exploration to exploitation. (2) For the Qwen2.5-7B-Base and Qwen3-4B-Base models, warm-up exploration in the early stages is crucial. However, for the Qwen2.5-7B-Instruct model, the impact of warm-up exploration on performance improvement is not significant. (3) For the Qwen2.5-7B-Base model, the warm-up decay weight adjustment method achieves optimal performance when the warm-up turning point is set to 0.4. Both arriving at the warm-up turning time point earlier or later will exert a certain impact on the exploration-exploitation effectiveness of the strategy.

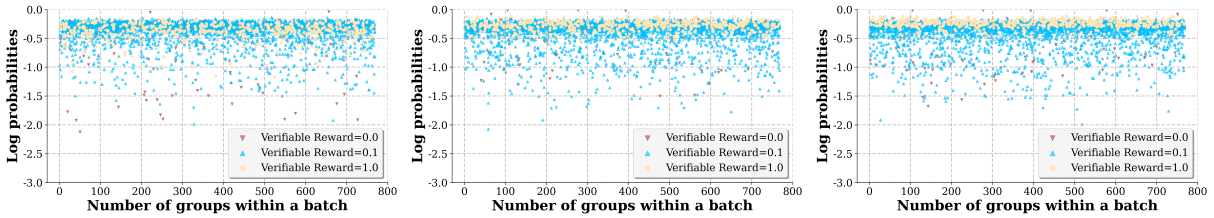
These experimental results are precisely consistent with the key insights we proposed in Section 3.4: (1) In the early stages of training, a high-entropy environment makes it difficult to determine

Model	MMLUPro Avg@2	GPQA Avg@4	TheoremQA Avg@2	WebInst. Avg@2	MATH500 Avg@2	Minerva Avg@2	AIME24 Avg@16	All -
Bonus Weight Adjustment Schedules								
Qwen2.5-7B-Base-GRPO	55.1	36.2	52.2	75.3	**76.5**	54.9	17.7	52.20
⇒ ω No decay	54.8	35.8	**54.0**	77.0	76.0	55.9	16.7	52.89
⇒ ω Linear decay	56.0	39.0	53.5	75.8	76.2	55.4	20.0	53.70
⇒ ω Warmup retention	57.0	38.7	53.5	76.2	**76.5**	55.4	20.0	53.90
⇒ ω **Warmup decay**	**57.6**	**39.4**	**54.0**	**77.8**	76.2	**56.2**	**23.3**	**54.90**
Qwen2.5-7B-Inst.-GRPO	52.0	40.1	**56.3**	63.2	77.0	**55.8**	17.7	51.73
⇒ ω No decay	57.7	42.2	52.1	76.8	74.5	52.6	20.0	53.70
⇒ ω Linear decay	**58.3**	**43.8**	54.7	**77.3**	77.0	54.2	23.3	55.51
⇒ ω Warmup retention	57.4	42.0	54.3	75.2	75.7	53.5	20.0	54.01
⇒ ω **Warmup decay**	58.1	42.6	55.4	74.9	**77.8**	**55.8**	**26.7**	**55.90**
Qwen3-4B-Base-GRPO	57.3	41.5	53.3	65.8	78.5	54.7	16.7	52.54
⇒ ω No decay	53.6	40.8	54.5	65.6	79.2	54.8	16.7	52.17
⇒ ω Linear decay	58.1	43.2	56.1	71.4	**82.2**	55.7	**20.0**	55.24
⇒ ω Warmup retention	57.5	43.5	55.7	68.2	80.8	56.9	**20.0**	54.66
⇒ ω **Warmup decay**	**59.5**	**43.9**	**56.8**	**73.1**	81.7	**57.8**	**20.0**	**56.11**

Table 8: Performance comparison of ICPO under different weight adjustment schemes for intrinsic confidence reward across different models. The weight adjustment schemes adhere to those illustrated in Figure 2.



(a) 0% of entire training process (b) 20% of entire training process (c) 40% of entire training process



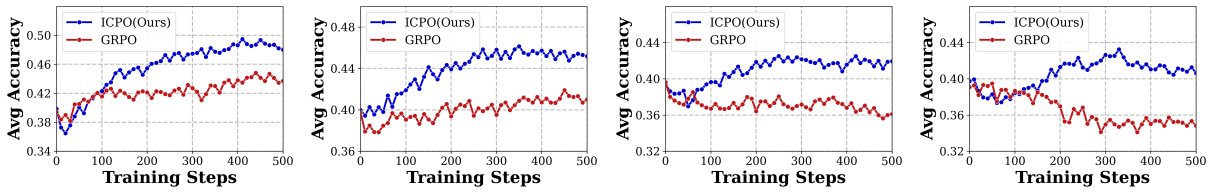
(d) 60% of entire training process (e) 80% of entire training process (f) 100% of entire training process

Figure 9: The distribution of generation probabilities for correct responses and incorrect responses across different training phases throughout the entire training process on the Qwen2.5-7B-Base model.

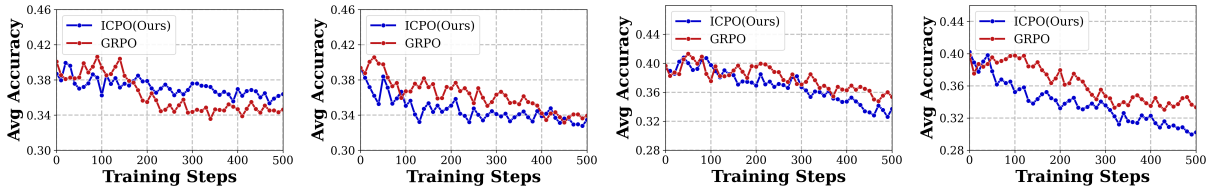
whether low-probability responses are worth learning. As training progresses, the entropy gradually decreases, and correct yet novel low-probability responses begin to emerge, thereby gradually enhancing the motivation to learn these responses. (2) In the later stages of training, when most novel samples have already been mastered, the remaining low-probability responses may merely constitute noise or valueless signals; thus, reducing the influence of intrinsic signals helps stabilize policy learning. (3) The period during which the warm-up phase sharply rises and reaches its peak turning point should align with the period when the entropy gradually decreases to an appropriate level while still retaining active exploration behavior.

For the Qwen2.5-7B-Instruct model, there is little difference in performance between warm-up exploration and directly applying weight decay. The primary reason lies in the fact that this model, having undergone instruction fine-tuning training on top of a pre-trained model, starts with a relatively low entropy level in the initial training phase (absent of a high-entropy environment), thus allowing for selective warm-up exploration. In contrast, for models like Qwen2.5-7B-Base and Qwen3-4B-Base, which are solely pre-trained, the presence of a high-entropy environment in the early training stages makes it more effective to gradually enhance their learning motivation for correct yet low-probability responses.

To validate our key insights, we plotted the probability distributions of generating correct and in-

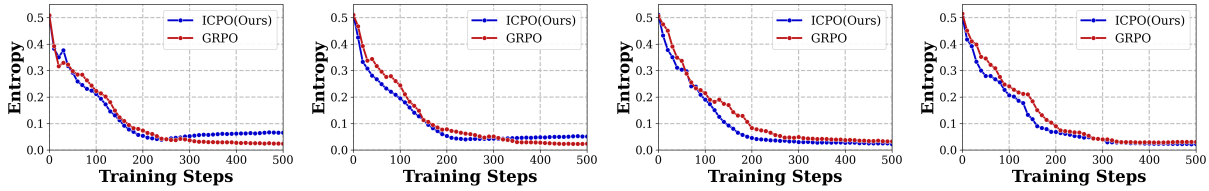


(a) 0.1 random reward noise (b) 0.2 random reward noise (c) 0.4 random reward noise (d) 0.5 random reward noise

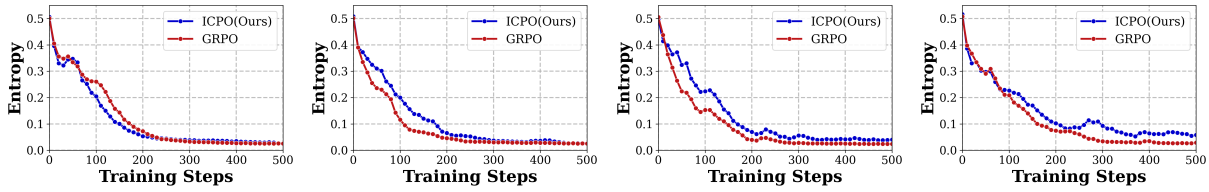


(e) 0.6 random reward noise (f) 0.7 random reward noise (g) 0.8 random reward noise (h) 0.9 random reward noise

Figure 10: Comparison of ICPO and GRPO Performance under Different Intensities of Random Noise Injection during the Training Process Based on the Qwen2.5-7B-Base Model.



(a) 0.1 random reward noise (b) 0.2 random reward noise (c) 0.4 random reward noise (d) 0.5 random reward noise



(e) 0.6 random reward noise (f) 0.7 random reward noise (g) 0.8 random reward noise (h) 0.9 random reward noise

Figure 11: Comparison of ICPO and GRPO Entropy under Different Intensities of Random Noise Injection during the Training Process Based on the Qwen2.5-7B-Base Model.

correct responses across different training periods throughout the entire training process, as shown in Figure 9. It can be observed that: (1) In the very initial stage of training, there are extremely few correct yet low-probability responses, with the majority of low-probability responses being incorrect ones, making it impossible to effectively distinguish novel responses with learning value. (2) As training progresses to the 20%-40% stage, correct yet low-probability responses increase significantly, revealing a large number of novel samples with high learning value. (3) In the later stages of training, the number of correct yet low-probability samples returns to a relatively low level, with most low-probability samples now being incorrect ones (data noise or overly complex problems), whose learning value diminishes. This is highly consistent with our proposed viewpoints and also provides an intuitive explanation for why the warm-up decay weight adjustment scheme is effective.

D Ablation Study on Noisy Rewards

To further investigate the performance of ICPO in scenarios with noisy rewards, we varied the extent of random noise injected into the rewards. In the noise reward experiment in Section 4.3, we randomly selected 40% of the data during each update iteration and randomly added or subtracted 0.3 from the rewards of this portion of data as noise. Here, we further explored the comparative performance of ICPO and GRPO when the random noise levels were set to 0.1, 0.2, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9. The training dynamics are illustrated in Figures 10 and 11. It can be observed that when the random noise ranges from 0.1 to 0.5, ICPO can still effectively learn through its intrinsic confidence-driven mechanism, maintaining positive reinforcement for correct yet low-probability

responses and effectively suppressing incorrect yet high-probability responses when external rewards are distorted, namely the two scenarios mentioned in Appendix E.3.

However, when the random noise in rewards exceeds 0.6, ICPO's noise resistance reaches a turning point, which aligns precisely with the viewpoint we mentioned in the section on limitations: when rewards completely deviate from the true objectives of the task, this "encouragement-suppression" mechanism undergoes a complete reverse shift. Originally genuine and effective low-probability behaviors are mistakenly labeled as incorrect samples by the erroneous rewards, while those genuinely ineffective or even harmful behaviors are mispackaged by the erroneous rewards as correct yet low-probability high-quality samples, becoming the primary focus of ICPO's optimization efforts and ultimately leading to policy collapse. This also represents a major direction for future optimization of ICPO.

E Theoretical Explanation

We systematically explain the effectiveness of ICPO from three aspects: (1) why ICPO can calibrate policy updates; (2) why ICPO can address the sparse reward problem; and (3) why ICPO can mitigate the interference from noisy rewards.

E.1 Aspect 1: Calibrate Policy Updates

As depicted in the "Effectiveness" section of Figure 1, we have discovered that the intrinsic confidence-driven reward mechanism inherently penalizes confident errors while encouraging novel and correct responses. Specifically, novel correct replies (with relatively lower confidence levels) receive greater positive rewards, whereas incorrect replies stemming from confident reactions (with higher confidence levels) incur greater penalties due to receiving smaller intrinsic rewards. Most notably, for correct responses with extremely low confidence, ICPO does not excessively encourage them, as such responses may lack any learning value due to reward misjudgment or data noise. The following theorem formalizes this intuition.

Theorem 1. *Let π_θ^t denote the policy at training step t. Under the influence of the preference advantage score bonus term in Equation 9, the calibration rules for updating the policy π_θ^{t+1} are specified as follows:*

(i) For correct responses, trajectories with relatively lower confidence (relatively lower generation probabilities) achieve a greater relative advantage enhancement;

(ii) For incorrect responses, trajectories with higher confidence (higher generation probabilities) experience a greater relative advantage reduction;

(iii) For trajectories with extremely low confidence (extremely low probabilities), their relative advantage enhancement is, conversely, smaller.

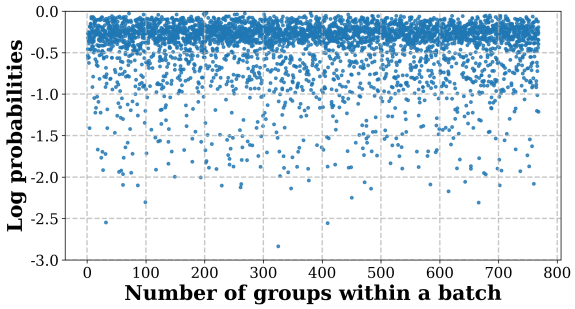


Figure 12: Scatter plot of responses log-probability distributions within a batch, where each batch contains 768 groups and each group comprises 5 responses.

We randomly selected a training batch during the overall training process and statistically analyzed the log-probability distributions of different responses within the group. Furthermore, we employed two sets of real in-group response examples to illustrate the aforementioned theorem. Figure 12 presents the log-probability distributions of responses within a training batch. It can be observed that the log-probability of most responses fluctuate between -2.0 and 0.0, while a small number of extreme responses exhibit log-probability reaching -3.0.

Table 9 illustrates the disparities in advantages of response examples within two real groups under the GRPO and ICPO methods. It can be observed that, compared to GRPO, ICPO tends to provide greater encouragement for correctly generated trajectories with lower confidence levels, thereby facilitating the model's acquisition of novel knowledge. Simultaneously, it reduces the weight assigned to erroneous trajectories with high confidence, thereby curbing the model's overconfidence. More importantly, ICPO's incentive mechanism is relatively moderate, avoiding excessive encouragement for samples that are correct but exhibit extremely low confidence levels.

E.2 Aspect 2: Tackle sparse Reward

The sparse reward problem refers to the scenario where, under the same prompt, all generated re-

sponses receive identical rewards (typically stemming from the sparse design of the reward function). In such cases, GRPO is unable to compute the relative advantages among responses within the group, resulting in a lack of effective gradient signals for policy optimization. This, in turn, triggers policy collapse and a sharp decline in model performance.

In contrast, ICPO effectively addresses this issue by introducing preference advantage scores based on the response generation probabilities. Even when the external rewards for all responses are identical, the generation probabilities of different responses under the current policy may still vary. Consequently, ICPO can assign distinct preference advantage scores to each response. When these preference advantage scores are fused with externally verifiable rewards, the relative merits among responses within the group become distinguishable, thereby providing fine-grained and meaningful optimization signals for policy updates. This significantly enhances training stability and ultimate performance. The examples presented in Table 9 also effectively substantiate this point: when responses have identical rewards, the advantage values computed by GRPO are entirely uniform, failing to accurately differentiate between the quality of responses. However, ICPO successfully distinguishes the relative merits among responses with the same rewards, offering more fine-grained guidance for policy updates.

E.3 Aspect 3: Mitigate Noisy Reward

Through empirical analysis, we have discovered that the intrinsic confidence-driven reward mechanism can mitigate two types of interference caused by noisy rewards. For responses that are inherently erroneous yet generated with high probability, when reward noise erroneously inflates their verifiable rewards, ICPO can effectively suppress such responses. Conversely, for responses that are inherently correct but generated with low probability, when reward noise erroneously reduces their verifiable rewards, ICPO can enhance their relative advantages, thereby providing effective encouragement. The following theorem formalizes this intuition:

Theorem 2. *Let π_θ^t denote the policy at training step t. Under the influence of the preference advantage score bonus term in Equation 9, the update from π_θ^t to π_θ^{t+1} follows the following specific rule for alleviating noisy rewards:*

(i) For trajectories that are correct yet have relatively low confidence (i.e., relatively lower generation probability), when noise causes their rewards to be lower, ICPO, compared to GRPO, enhances their relative advantage;

(ii) For trajectories that are erroneous yet exhibit higher confidence (i.e., higher generation probability), when noise causes their rewards to be higher, ICPO, compared to GRPO, suppresses their relative advantage.

Table 10 elucidates the theorem through a concrete example. However, for the other two scenarios—namely, trajectories that are correct yet possess high confidence and those that are erroneous with low confidence—when rewards are subject to noise, neither ICPO nor GRPO can achieve accurate guidance. This also constitutes a key direction for future exploration and a significant challenge for the ICPO method.

Log-probability: $P_1(-0.135)$, $P_2(-0.583)$, $P_3(-0.216)$, $P_4(-0.050)$, $P_5(-0.407)$
Verifiable reward: $R_1^{verif.}(0.100)$, $R_2^{verif.}(0.100)$, $R_3^{verif.}(1.000)$, $R_4^{verif.}(0.100)$, $R_5^{verif.}(1.000)$

1. Responses within group are ranked as O_2, O_5, O_3, O_1, O_4 by Equation 6.

2. Based on Equation 8, a set of preference pairs is constructed as: $\{(O2, O5), (O2, O3), (O2, O1), (O2, O4), (O5, O3), (O5, O1), (O5, O4), (O3, O1), (O3, O4), (O1, O4)\}$.

3. According to Equation 9, preference advantage scores are calculated for each response ($\delta = 0.4$):

- $S_2^p = 0.4 \cdot (\frac{-0.407}{-0.583} + \frac{-0.216}{-0.583} + \frac{-0.135}{-0.583} + \frac{-0.050}{-0.583}) - 0.4 \cdot (-0.050) = 0.5744$
- $S_5^p = 0.4 \cdot (\frac{-0.216}{-0.407} + \frac{-0.135}{-0.407} + \frac{-0.050}{-0.407}) - 0.4 \cdot (-0.050) = 0.4141$
- $S_3^p = 0.4 \cdot (\frac{-0.135}{-0.216} + \frac{-0.050}{-0.216}) - 0.4 \cdot (-0.050) = 0.3626$
- $S_1^p = 0.4 \cdot (\frac{-0.050}{-0.135}) - 0.4 \cdot (-0.050) = 0.1681$
- $S_4^p = -0.4 \cdot (-0.050) = 0.020$

4. Reward for each response is calculated by equation 11 ($\tau = 2.0, \omega = 1.0$):

- $R_2^{final} = 0.1000 + 1.0 \cdot \min(0.5744, \frac{0.1000}{2.0}) = 0.15$
- $R_5^{final} = 1.0000 + 1.0 \cdot \min(0.4141, \frac{1.0000}{2.0}) = 1.4141$
- $R_3^{final} = 1.0000 + 1.0 \cdot \min(0.3626, \frac{1.0000}{2.0}) = 1.3626$
- $R_1^{final} = 0.1000 + 1.0 \cdot \min(0.1681, \frac{0.1000}{2.0}) = 0.15$
- $R_4^{final} = 0.1000 + 1.0 \cdot \min(0.020, \frac{0.1000}{2.0}) = 0.120$

5. Calculate the relative advantages of different responses in vanilla GRPO and ICPO:

- **vanilla GRPO:** $A_1(-0.816)$, $A_2(-0.816)$, $A_3(1.225)$, $A_4(-0.816)$, $A_5(1.225)$
- **ICPO:** $A_1(-0.7996)$, $A_2(-0.7996)$, $A_3(1.1819)$, $A_4(-0.8986)$, $A_5(1.2660)$

Conclusion: For the correct yet low-confidence response O_5, ICPO confers a greater advantage, whereas for the incorrect yet high-confidence response O_4, ICPO imposes a more substantial penalty.

Log-probability: $P_1(-0.286)$, $P_2(-0.287)$, $P_3(-0.144)$, $P_4(-0.066)$, $P_5(-2.356)$
Verifiable reward: $R_1^{verif.}(0.100)$, $R_2^{verif.}(0.100)$, $R_3^{verif.}(1.000)$, $R_4^{verif.}(0.100)$, $R_5^{verif.}(1.000)$

Step 1.2 is analogous to the example presented above and will not be elaborated on excessively.

3. According to Equation 9, preference advantage scores are calculated for each response ($\delta = 0.4$):

- $S_5^p = 0.4 \cdot (\frac{-0.287}{-2.356} + \frac{-0.286}{-2.356} + \frac{-0.144}{-2.356} + \frac{-0.066}{-2.356}) - 0.4 \cdot (-0.066) = 0.1593$
- $S_2^p = 0.4 \cdot (\frac{-0.286}{-0.287} + \frac{-0.144}{-0.287} + \frac{-0.066}{-0.287}) - 0.4 \cdot (-0.066) = 0.7177$
- $S_1^p = 0.4 \cdot (\frac{-0.144}{-0.286} + \frac{-0.066}{-0.286}) - 0.4 \cdot (-0.066) = 0.3201$
- $S_3^p = 0.4 \cdot (\frac{-0.066}{-0.144}) - 0.4 \cdot (-0.066) = 0.2097$
- $S_4^p = -0.4 \cdot (-0.066) = 0.0264$

4. Reward for each response is calculated by equation 11 ($\tau = 2.0, \omega = 1.0$):

- $R_5^{final} = 1.0000 + 1.0 \cdot \min(0.1593, \frac{1.0000}{2.0}) = 1.1593$
- $R_2^{final} = 0.1000 + 1.0 \cdot \min(0.7177, \frac{0.1000}{2.0}) = 0.15$
- $R_1^{final} = 0.1000 + 1.0 \cdot \min(0.3201, \frac{0.1000}{2.0}) = 0.15$
- $R_3^{final} = 1.0000 + 1.0 \cdot \min(0.2097, \frac{1.0000}{2.0}) = 1.2097$
- $R_4^{final} = 0.1000 + 1.0 \cdot \min(0.0264, \frac{0.1000}{2.0}) = 0.1264$

5. Calculate the relative advantages of different responses in vanilla GRPO and ICPO:

- **vanilla GRPO:** $A_1(-0.816)$, $A_2(-0.816)$, $A_3(1.225)$, $A_4(-0.816)$, $A_5(1.225)$
- **ICPO:** $A_1(-0.8006)$, $A_2(-0.8006)$, $A_3(1.2733)$, $A_4(-0.8468)$, $A_5(1.1746)$

Conclusion: For correct yet extremely low-confidence response O_5, ICPO does not over-encourage.

Table 9: The advantage disparities of response examples in two real groups under GRPO and ICPO.

Log-probability: $P_1(-0.135)$, $P_2(-0.583)$, $P_3(-0.216)$, $P_4(-0.050)$, $P_5(-0.407)$
Verifiable reward: $R_1^{verif.}(0.100)$, $R_2^{verif.}(0.100)$, $R_3^{verif.}(1.000)$, $R_4^{verif.}(0.100)$, $R_5^{verif.}(1.000)$
Noisy reward: $R_1^{Noisy}(0.400)$, $R_2^{Noisy}(0.100)$, $R_3^{Noisy}(1.000)$, $R_4^{Noisy}(0.400)$, $R_5^{Noisy}(0.700)$

1. Responses within group are ranked as O_2, O_5, O_3, O_1, O_4 by Equation 6.

2. Based on Equation 8, a set of preference pairs is constructed as: $\{(O2, O5), (O2, O3), (O2, O1), (O2, O4), (O5, O3), (O5, O1), (O5, O4), (O3, O1), (O3, O4), (O1, O4)\}$.

3. According to Equation 9, preference advantage scores are calculated for each response ($\delta = 0.4$):
- $S_2^p = 0.4 \cdot (\frac{-0.407}{-0.583} + \frac{-0.216}{-0.583} + \frac{-0.135}{-0.583} + \frac{-0.050}{-0.583}) - 0.4 \cdot (-0.050) = 0.5744$
- $S_5^p = 0.4 \cdot (\frac{-0.216}{-0.407} + \frac{-0.135}{-0.407} + \frac{-0.050}{-0.407}) - 0.4 \cdot (-0.050) = 0.4141$
- $S_3^p = 0.4 \cdot (\frac{-0.135}{-0.216} + \frac{-0.050}{-0.216}) - 0.4 \cdot (-0.050) = 0.3626$
- $S_1^p = 0.4 \cdot (\frac{-0.050}{-0.135}) - 0.4 \cdot (-0.050) = 0.1681$
- $S_4^p = -0.4 \cdot (-0.050) = 0.020$

4. Reward for each response is calculated by equation 11 ($\tau = 2.0, \omega = 1.0$):
- $R_2^{final} = 0.1000 + 1.0 \cdot \min(0.5744, \frac{0.1000}{2.0}) = 0.15$
- $R_5^{final} = 0.7000 + 1.0 \cdot \min(0.4141, \frac{0.7000}{2.0}) = 1.05$
- $R_3^{final} = 1.0000 + 1.0 \cdot \min(0.3626, \frac{1.0000}{2.0}) = 1.3626$
- $R_1^{final} = 0.4000 + 1.0 \cdot \min(0.1681, \frac{0.4000}{2.0}) = 0.5681$
- $R_4^{final} = 0.4000 + 1.0 \cdot \min(0.020, \frac{0.4000}{2.0}) = 0.420$

5. Calculate the relative advantages of different responses in vanilla GRPO and ICPO:
- **vanilla GRPO** $w\ R^{verif.}$**:** $A_1(-0.816)$, $A_2(-0.816)$, $A_3(1.225)$, $A_4(-0.816)$, $A_5(1.225)$
- **vanilla GRPO** $w\ R^{Noisy}$**:** $A_1(-0.3371)$, $A_2(-1.3485)$, $A_3(1.6856)$, $A_4(-0.3371)$, $A_5(0.3371)$
- **ICPO** $w\ R^{Noisy}$**:** $A_1(-0.3243)$, $A_2(-1.2788)$, $A_3(1.4900)$, $A_4(-0.6624)$, $A_5(0.7759)$

Conclusion: For responses that are inherently erroneous yet generated with high probability (such as O_4), when reward noise erroneously inflates their verifiable rewards, ICPO can effectively suppress such responses. Conversely, for responses that are inherently correct but generated with low probability (such as O_5), when reward noise erroneously reduces their verifiable rewards, ICPO can amplify their relative advantages, thereby providing effective encouragement.

Table 10: The advantage disparities of response examples under GRPO and ICPO in noisy reward scenarios.