PROGRESSIVE REASONING REINFORCEMENT LEARNING VIA CONFIDENCE-AWARE REWARD REWEIGHTING

Anonymous authors
Paper under double-blind review

ABSTRACT

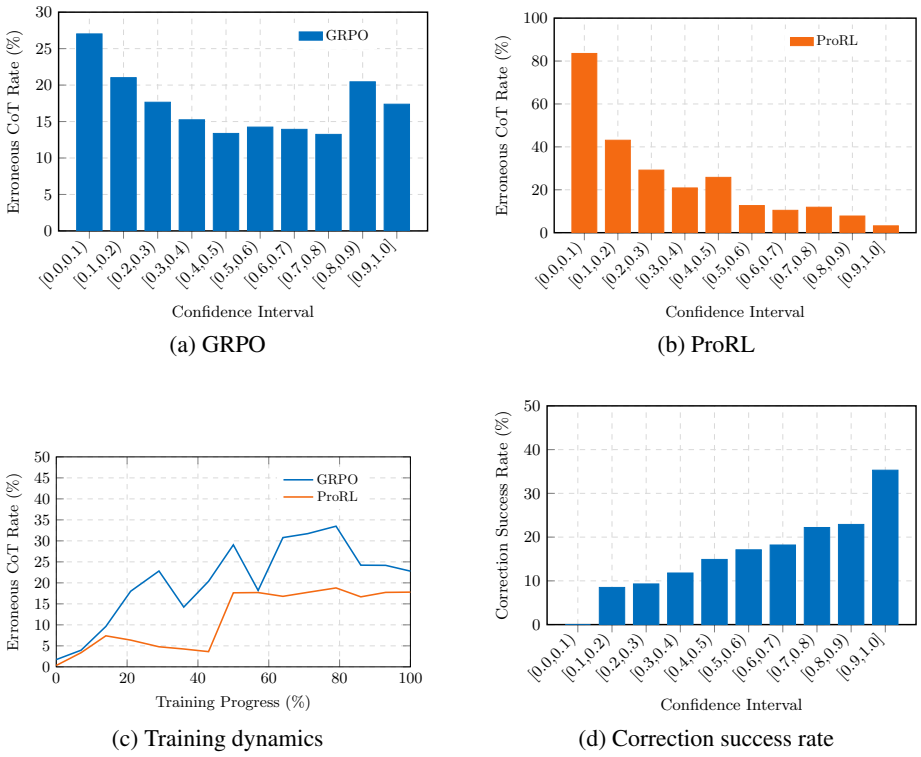
In this paper, we propose Progressive Reasoning Reinforcement Learning (ProRL) to improve the reasoning capabilities of large language models. Existing RLVR methods, such as Group Relative Policy Optimization (GRPO), rely on binary outcome rewards that treat trajectories with the same outcome indiscriminately, overlooking their varying reliability and learning value. We argue that sampled trajectories exhibit different levels of mastery and should contribute unequally to policy optimization. Based on this insight, we instantiate ProRL with a likelihood-aware reward reweighting strategy that uses the likelihood under the historical policy, detached from gradient computation, as an estimate of trajectory confidence to dynamically adjust trajectory rewards. This enables progressive consolidation of reliable reasoning, refinement of emerging capabilities, and correction of dominant errors while preserving reasoning diversity. Experiments with Qwen2.5-1.5B on MATH-lighteval and DAPO-MATH-17K, together with scaling experiments using Qwen2.5-7B on MATH-lighteval, demonstrate that ProRL consistently outperforms strong RLVR baselines on both Pass@1 and Pass@16, improving reasoning accuracy while preserving diversity in solution exploration.

1 INTRODUCTION

Reasoning has become a defining capability of modern large language models (LLMs) and multimodal large language models (MLLMs), enabling them to solve complex tasks through multi-step inference rather than direct pattern matching. Recent advances (Hu et al., 2025; Shao et al., 2024; Guo et al., 2025) have demonstrated that this capability can be substantially improved by reinforcement learning with verifiable rewards (RLVR), where models learn from trial-and-error interactions using automatically verifiable outcomes instead of human-annotated reasoning traces. By directly optimizing for task correctness, RLVR has achieved remarkable success on mathematical reasoning (Shao et al., 2024; Guo et al., 2025), coding (Wei et al., 2025; Zhao et al., 2025), and multimodal reasoning benchmarks (Liu et al., 2025b; Liang et al., 2025), and has become a cornerstone for training state-of-the-art reasoning models (Guo et al., 2025; Hu et al., 2025).

Among existing RLVR algorithms, Group Relative Policy Optimization (GRPO) (Shao et al., 2024) has become the de facto optimization framework due to its simplicity and effectiveness. Given a prompt, the standard GRPO samples multiple reasoning trajectories and assigns binary outcome rewards based solely on answer correctness, updating the policy through relative advantages within the sampled group. While effective in improving reasoning accuracy, this reward design has two fundamental limitations. First, it treats trajectories with the same outcome equally, *ignoring differences in reasoning quality and reliability*. As a result, GRPO might reinforce spurious reasoning patterns simply because they happen to produce correct answers, gradually increasing their preference by the policy despite unreliable underlying reasoning processes. Second, since advantages are computed from group-wise reward differences, *the optimization signal depends heavily on the sampled reward distribution and can become weak or even vanish when all trajectories are either correct or incorrect*. These limitations suggest that binary outcome rewards fail to fully capture the diverse learning values contained in sampled reasoning trajectories.

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



(a) GRPO (b) ProRL

(c) Training dynamics (d) Correction success rate

Figure 1: Analysis of reasoning trajectory quality with ChatGPT 5.4-assisted annotations verified by human reviewers. (a) Low-quality CoT rate among correct GRPO responses across confidence intervals, where $c_i = \exp(\ell_i)$ and ℓ_i denotes the length-normalized log-likelihood of trajectory i. Low-quality CoTs contain logical inconsistencies, invalid reasoning steps, or significant formatting errors. (b) Corresponding results for ProRL. (c) Evolution of low-quality CoT rates during training for GRPO and ProRL. (d) Correction success rate of incorrect trajectories across confidence intervals using the same length-normalized likelihood form. See Appendix D for annotation details, Appendix E for examples, and Appendix F for correction examples.

Our Motivation. We argue that reasoning optimization should follow a progressive learning process: new reasoning capabilities should be expanded from reliable knowledge that has already been mastered. Similar to human learning, stable reasoning patterns should be consolidated, while emerging capabilities should be gradually refined through further optimization. In this process, sampled trajectories naturally exhibit different levels of mastery under the current policy, and their contributions to optimization should therefore depend on their reliability rather than being treated equally.

Correct trajectories do not necessarily provide equally valuable learning signals. High-confidence correct trajectories usually reflect stable reasoning patterns that should be reinforced, whereas low-confidence correct trajectories might correspond to fragile reasoning behaviors or accidental successes caused by unreliable chains of thought (CoT). Directly reinforcing such trajectories with the same reward can cause the model to internalize unreliable reasoning patterns as valid knowledge, potentially limiting its generalization ability. As shown in Figure 1a, GRPO assigns identical rewards to all correct responses regardless of their confidence, resulting in a considerable number of low-quality CoTs being reinforced during optimization. This suggests that outcome correctness alone is insufficient to distinguish reliable reasoning from spurious success.

Incorrect trajectories also exhibit different optimization values. Although GRPO treats all incorrect responses equally, high-confidence incorrect trajectories reflect dominant reasoning patterns that the model strongly favors despite being erroneous. Such trajectories represent dominant failure modes and therefore provide more informative signals for optimization. As shown in Figure 1d, such trajectories are more likely to be corrected than low-confidence failures. Prioritizing these trajectories encourages the model to discover correct trajectories with similar structures. Reinforcing these

corrected trajectories then progressively replaces erroneous behaviors with more reliable reasoning strategies.

Progressive Reasoning Reinforcement Learning. Based on this insight, we propose *Progressive Reasoning Reinforcement Learning (ProRL)*, a confidence-aware RLVR framework that guides the policy to progressively consolidate reliable reasoning patterns, refine emerging capabilities, and correct dominant failure modes. Specifically, we instantiate *ProRL* with a likelihood-aware reward reweighting strategy, where the likelihood under the historical policy is used as a detached estimate of trajectory confidence. By normalizing confidence scores within each sampled group, *ProRL* dynamically adjusts trajectory rewards, allocating optimization emphasis according to the model's confidence. This design enables finer-grained credit assignment beyond binary correctness, allowing reliable trajectories to be reinforced while providing informative refinement signals for less mature or erroneous behaviors. Consequently, *ProRL* remains effective even when sampled trajectories share identical outcomes. As shown in Figure 2, *ProRL* provides stronger and more differentiated learning signals than GRPO, particularly during early optimization. Since likelihoods are used only for reward adjustment and are detached from gradient optimization, *ProRL* does not directly maximize high-likelihood trajectories, thereby preserving exploration capability and reasoning diversity during RLVR training.

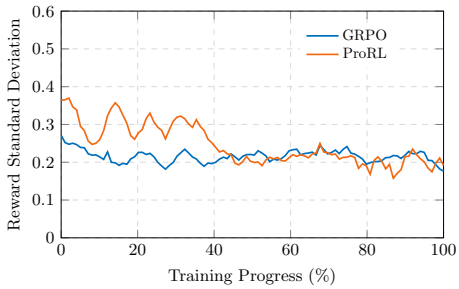


Figure 2: Comparison of reward standard deviation between GRPO and ProRL. ProRL exhibits higher reward variability in the early training stage, indicating stronger exploration, while the variance gradually decreases as training proceeds. Both methods eventually converge to a relatively stable level.

Experimental Results. We evaluate our method *ProRL* on mathematical reasoning benchmarks, including MATH-lighteval (Hendrycks et al., 2021) and DAPO-MATH-17K (Yu et al., 2025), using Qwen2.5-1.5B and Qwen2.5-7B (Yang et al., 2025). Extensive experiments demonstrate consistent improvements over strong RLVR baselines on both Pass@1 and Pass@16, indicating that our method not only improves reasoning accuracy but also preserves diverse reasoning behaviors. These results validate that confidence-aware reward optimization provides a simple yet effective mechanism for improving reasoning reinforcement learning.

Overall, our main contributions are as follows:

- We propose *Progressive Reasoning Reinforcement Learning (ProRL)*, which enables reasoning capabilities to be progressively consolidated, refined, and corrected based on the maturity of sampled trajectories.
- We instantiate *ProRL* with a likelihood-aware reward reweighting strategy, which dynamically adjusts trajectory contributions according to their confidence.
- We conduct extensive experiments on MATH-lighteval and DAPO-MATH-17K using Qwen2.5-1.5B and Qwen2.5-7B, demonstrating the effectiveness of *ProRL* in improving mathematical reasoning performance.

2 Related Work

2.1 Group Relative Policy Optimization

Group Relative Policy Optimization (GRPO) is a reinforcement learning algorithm widely used to train reasoning models (Shao et al., 2024; Guo et al., 2025). Unlike PPO, which relies on a learned value function for advantage estimation (Schulman et al., 2017), GRPO samples multiple responses per prompt and estimates advantages through within-group reward normalization. This eliminates the need for a critic and improves scalability in large-scale reinforcement learning with verifiable rewards. However, GRPO is highly sensitive to the diversity and composition of the sampled groups. Existing studies have improved GRPO by developing more robust advantage-estimation methods

for groups with low reward variance (Shao et al., 2024; Liu et al., 2025a; Le et al., 2026), designing sampling and group-construction strategies that produce more informative reward variation (Yu et al., 2025), mitigating length and normalization biases in long-form reasoning (Liu et al., 2025a; Yu et al., 2025), and modifying the optimization objective and clipping mechanism to promote exploration and improve sequence-level alignment (Yu et al., 2025; Zheng et al., 2025).

2.2 Confidence Estimation in Language Models

Confidence estimation in large language models has been studied from several perspectives, including sequence likelihood, token-level entropy, natural-language self-assessment, and sampling-based consistency. Sequence likelihood is widely used as an uncertainty signal due to its empirical correlation with correctness in reasoning tasks (Kadavath et al., 2022; Lin et al., 2024). Token-level entropy provides a finer-grained measure of uncertainty during generation, where higher entropy indicates lower confidence (Zhang et al., 2026; Geng et al., 2024; Damani et al., 2026). Another line of work elicits confidence via prompting, asking models to express uncertainty in natural language or scalar scores; however, such estimates are often sensitive to prompt design and prone to overconfidence (Lin et al., 2022; Tian et al., 2023). In contrast, sampling-based methods estimate confidence from agreement among multiple generations, with self-consistency showing that aggregating reasoning paths can improve accuracy by leveraging stability in sampled outputs (Wang et al., 2023; Chen et al., 2023). Beyond confidence estimation itself, recent work has begun incorporating confidence signals directly into model optimization. Confidence-aware reward modeling augments correctness-based supervision with model confidence, enabling the reward model to distinguish responses with similar correctness outcomes but different levels of reliability and thereby improve reasoning performance (He et al., 2025).

2.3 Process Supervision and Reward Shaping

Process supervision improves reasoning in LLMs by providing finer-grained feedback than outcome-based RLVR, which only evaluates final answers. It assigns rewards to intermediate steps or full reasoning trajectories, enabling more precise guidance toward correct solutions and reducing error propagation (Uesato et al., 2022; Lightman et al., 2023; Yao et al., 2026; Wang et al., 2026). However, it is expensive to scale, as it requires human step annotations, process reward models (PRMs), or step verifiers (Lightman et al., 2023). While PRM-based methods support reward-guided search or best-of-N selection (Lightman et al., 2023; Zhang et al., 2025), recent work explores more data-efficient or automated alternatives, though these still introduce additional modeling complexity and potential bias (Luo et al., 2024; Khalifa et al., 2025; Yao et al., 2026; Wang et al., 2026). As a lightweight alternative, reward shaping (Xie et al., 2024; Rezaei et al., 2026; Wang et al., 2026; Yao et al., 2026; Ng et al., 1999; Damani et al., 2026) incorporates trajectory-level or intermediate information into sparse outcome rewards without requiring explicit step supervision, providing a way to inject auxiliary signals, such as trajectory reliability or uncertainty, into outcome rewards to produce more informative learning signals. Along this direction, TIPS introduces a turn-level information-potential reward-shaping approach for search-augmented LLMs, assigning fine-grained intermediate rewards to individual interaction turns rather than relying solely on sparse final-task outcomes (Xie et al., 2026).

3 Method

3.1 Preliminaries: Group Relative Policy Optimization

Group Relative Policy Optimization (GRPO) (Shao et al., 2024) is a representative algorithm for reinforcement learning with verifiable rewards (RLVR) that optimizes a policy without learning an additional value function. Given a training set of question-answer pairs $D = \{(q, a)\}$, GRPO samples a group of G reasoning trajectories $\mathcal{G} = \{o_i\}_{i=1}^{G}$ from the historical policy $\pi_{\theta_{\text{old}}}$. In our experimental instantiation, each trajectory is evaluated by a reward function according to its final answer and format constraints, producing a trajectory-level reward R_i.

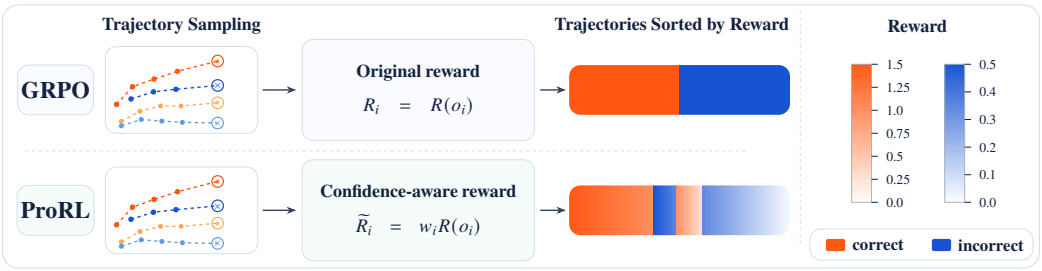


Figure 3: Conceptual illustration of confidence-aware reward redistribution with *ProRL*. Unlike GRPO, which treats trajectories with the same outcome equally, *ProRL* adjusts reward contributions according to trajectory confidence, thereby modifying the resulting advantages. It consolidates reliable reasoning through high-confidence correct trajectories, refines emerging capabilities through moderately confident trajectories, and prioritizes high-confidence incorrect trajectories for progressive error correction, while suppressing unreliable low-confidence trajectories. Note that the existence of format rewards leads to the applicability of *ProRL* to incorrect trajectories.

Instead of estimating value functions as in PPO, GRPO computes group-relative advantages by normalizing rewards within each sampled group,

$$\hat{A}_i = \frac{R_i - \mu_R}{\sigma_R}, \tag{1}$$

where $\mu_R = \frac{1}{G}\sum_{i=1}^{G} R_i$ and $\sigma_R = \mathrm{std}(\{R_i\}_{i=1}^{G})$. The trajectory-level advantage is broadcast to every token within the trajectory, $\hat{A}_{i,t} = \hat{A}_i$, and the policy is optimized using the clipped PPO objective,

$$\mathcal{J}_{\mathrm{GRPO}} = \frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\min\left(r_{i,t}(\theta)\hat{A}_{i,t}, \mathrm{clip}(r_{i,t}(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_{i,t}\right) - \beta D_{KL}\left[\pi_\theta||\pi_{\theta_{\mathrm{ref}}}\right], \tag{2}$$

where $r_{i,t}(\theta) = \dfrac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\mathrm{old}}}(o_{i,t} \mid q, o_{i,<t})}$ is the importance-sampling ratio between the current and behavior policies, which mitigates policy mismatch, $\pi_{\theta_{\mathrm{ref}}}$ is the reference model which is usually the initial model before reinforcement learning; KL divergence is $\mathbb{D}_{KL}[\pi_\theta||\pi_{\theta_{\mathrm{ref}}}] = \dfrac{\pi_{\theta_{\mathrm{ref}}}(o_{i,t}|q,o_{i,<t})}{\pi_\theta(o_{i,t}|q,o_{i,<t})} - \log\dfrac{\pi_{\theta_{\mathrm{ref}}}(o_{i,t}|q,o_{i,<t})}{\pi_\theta(o_{i,t}|q,o_{i,<t})} - 1.$

3.2 Progressive Reasoning Reinforcement Learning

Although effective, GRPO relies solely on outcome rewards and treats trajectories with identical outcomes indiscriminately. This coarse reward assignment overlooks the fact that sampled trajectories exhibit different levels of knowledge mastery and learning value under the current policy, resulting in suboptimal credit assignment during optimization.

Progressive Learning Perspective. We argue that reasoning capability should evolve progressively: new reasoning patterns should be developed from behaviors that the model has already learned reliably. Under this perspective, trajectories should contribute differently according to their confidence. High-confidence correct trajectories typically represent stable reasoning patterns that should be consolidated, whereas low-confidence correct trajectories may reflect fragile or accidental successes whose indiscriminate reinforcement can introduce unreliable reasoning behaviors. Meanwhile, high-confidence incorrect trajectories reveal dominant but erroneous reasoning patterns that the model strongly favors, making them valuable targets for refinement and correction.

Motivated by this observation, we propose *Progressive Reasoning Reinforcement Learning (ProRL)*, which incorporates trajectory confidence into reward optimization. Instead of assigning identical supervision to trajectories with the same outcome, *ProRL* dynamically adjusts their optimization contributions according to confidence, enabling the model to progressively consolidate reliable reasoning, refine emerging capabilities, and correct dominant errors.

Likelihood-aware Reward Reweighting. To instantiate the above principle, we estimate the confidence of each sampled trajectory using its likelihood under the historical policy. For each trajectory o_i, we compute its average log-likelihood,

$$\ell_i = \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \log \pi_{\theta_{\text{old}}}(o_{i,t}|q, o_{i,<t}). \tag{3}$$

The trajectory confidence is obtained by normalizing likelihoods within the sampled group,

$$w_i = \frac{\exp(\ell_i/\tau)}{\sum_{j=1}^{G} \exp(\ell_j/\tau)}, \tag{4}$$

where τ is a temperature parameter controlling the sharpness of the confidence distribution. We then reweight the original reward as:

$$\tilde{R}_i = w_i R_i. \tag{5}$$

Compared with binary rewards, Eq. (5) differentiates trajectories even when they share identical outcomes, providing finer-grained credit assignment according to the policy's confidence.

Progressive Policy Optimization. The reweighted rewards are used to compute the group-relative advantages,

$$\hat{A}_i = \frac{\tilde{R}_i - \mu_{\tilde{R}}}{\sigma_{\tilde{R}}}, \tag{6}$$

where $\mu_{\tilde{R}} = \frac{1}{G}\sum_{i=1}^{G} \tilde{R}_i$, and $\sigma_{\tilde{R}} = \text{std}(\{\tilde{R}_i\}_{i=1}^{G})$. The remaining optimization objective is identical to GRPO except that the original rewards are replaced by the confidence-aware rewards,

$$\mathcal{J}_{\text{ProRL}} = \mathcal{J}_{\text{GRPO}}(\tilde{R}_i). \tag{7}$$

Note that *trajectory likelihoods are computed using the historical policy with detached gradients and are used solely for reward reweighting.* Consequently, confidence estimation does not directly participate in policy optimization, avoiding a self-reinforcing feedback loop that continually increases the likelihood of already preferred trajectories. As a result, *ProRL* reallocates optimization effort among sampled trajectories while preserving the exploration capability and reasoning diversity of the original RLVR framework.

3.3 Analysis of ProRL

The key difference between GRPO and ProRL lies in how trajectory-level advantages are assigned within a sampled group. In GRPO, outcome rewards are determined only by correctness. Specifically, a correct trajectory receives a positive outcome reward (e.g., 1.0) together with a format reward (e.g., 0.5), whereas an incorrect trajectory receives only the format reward (e.g., 0.5). After group-wise reward normalization, correct trajectories typically obtain positive advantages, while incorrect trajectories obtain negative advantages. Consequently, GRPO reinforces all correct trajectories and suppresses all incorrect trajectories regardless of their reasoning reliability.

In contrast, *ProRL* incorporates trajectory confidence into reward assignment before group normalization, thereby reshaping the relative ranking of sampled trajectories. Unlike GRPO, where ranking is determined solely by outcome correctness, *ProRL* allows trajectories with the same outcome to receive different optimization signals according to their confidence. Consequently, as shown in Figure 3, high-confidence incorrect trajectories might move upward in the ranking and obtain positive advantages, enabling their dominant reasoning patterns to be further refined, while extremely low-confidence correct trajectories might move downward and receive negative advantages, preventing unreliable reasoning behaviors from being reinforced.

Although high-confidence incorrect trajectories might receive positive advantages in a given optimization step, they are not treated as final solutions. Instead, preserving their optimization signals allows the model to further explore similar underlying reasoning patterns and discover correct variants in subsequent generations. Once these corrected trajectories emerge, their reinforcement gradually replaces the erroneous behaviors. As shown in Figure 1d, high-confidence incorrect trajectories can be effectively corrected through regeneration from uncertain tokens, supporting this progressive refinement process.

Table 1: Main results of Qwen2.5-1.5B-Instruct fine-tuned on MATH-lighteval (%).

Method	OlympiadBench		MATH500		AMC23		AIME24		Avg.	
	Pass@1	Pass@16	Pass@1	Pass@16	Pass@1	Pass@16	Pass@1	Pass@16	Pass@1	Pass@16
Base Model	17.51	55.19	33.00	83.20	17.50	60.00	3.33	10.00	17.84	52.10
GRPO	19.88	58.16	54.40	84.60	17.50	65.00	1.67	16.67	23.36	56.11
GRPO-VPS	19.14	58.31	41.40	85.60	18.13	67.50	3.33	16.67	20.50	57.02
RLCR	19.44	58.01	40.00	75.60	15.00	65.00	3.33	10.00	19.44	52.15
RL-ZVP	19.58	59.35	54.40	84.40	20.67	70.00	5.00	16.67	24.91	57.61
ProRL (Ours)	**20.47**	**60.83**	**58.80**	**86.80**	**24.38**	**75.00**	**6.67**	**20.00**	**27.58**	**60.66**

Table 2: Main results of Qwen2.5-7B-Instruct fine-tuned on MATH-lighteval (%).

Method	OlympiadBench		MATH500		AMC23		AIME24		Avg.	
	Pass@1	Pass@16	Pass@1	Pass@16	Pass@1	Pass@16	Pass@1	Pass@16	Pass@1	Pass@16
Base Model	37.98	73.89	57.40	93.00	26.25	82.50	8.33	28.33	32.49	69.43
GRPO	37.83	73.59	75.20	93.00	45.63	80.00	**9.17**	31.67	41.96	69.57
GRPO-VPS	37.69	73.29	73.80	**93.60**	48.75	**85.00**	8.33	23.33	42.14	68.81
RLCR	37.09	73.00	75.20	92.80	48.13	80.00	8.33	23.33	42.19	67.28
RL-ZVP	38.13	73.74	76.20	93.40	47.50	80.00	8.33	23.33	42.54	67.62
ProRL (Ours)	**40.65**	**74.33**	**77.00**	**93.60**	**50.00**	82.50	**9.17**	**33.33**	**44.21**	**70.94**

4 EXPERIMENTS

Training Settings. We implement *ProRL* on top of the Open-R1 framework (Hugging Face, 2025), using GRPO as the underlying RLVR algorithm and vLLM (Kwon et al., 2023) as the rollout generation backend. Experiments are conducted on MATH-lighteval (Hendrycks et al., 2021) and DAPO-MATH-17K (Yu et al., 2025) using Qwen2.5-1.5B-Instruct and Qwen2.5-7B-Instruct (Yang et al., 2025). For each question, we sample 16 trajectories. The maximum input and generation lengths are set to 512 and 1024 tokens, respectively. All models are trained for two epochs with a learning rate of 1×10^{-6} and a rollout temperature of 1.0. We set the random seed to 42 for all experiments. For *ProRL*, the confidence temperature is set to $\tau = 0.7$. Further details are in Appendices A and B.

Evaluation Settings. We compare *ProRL* with the base model, GRPO, GRPO-VPS (Wang et al., 2026), RLCR (Damani et al., 2026), and RL-ZVP (Le et al., 2026). All methods are trained on MATH-lighteval and DAPO-MATH-17K under identical settings, with overlapping training examples removed, and evaluated using the best validation checkpoint. For Pass@1, we generate one response per problem on OlympiadBench and MATH500. For AMC2023 and AIME2024, each problem is independently sampled four times at temperature 0.6, and Pass@1 is computed over all responses from the four repetitions. For Pass@16, we sample 16 responses per problem at temperature 1.0 and count a problem as solved if at least one response is correct. Results are averaged over three runs, with correctness verified by `math_verify`.

4.1 MAIN RESULTS

Results on Qwen2.5-1.5B-Instruct Fine-tuned on MATH-lighteval. Table 1 shows that *ProRL* achieves the best overall performance at the 1.5B scale on both Pass@1 and Pass@16. Compared with standard GRPO, it improves the average Pass@1 and Pass@16 scores by 4.22 and 4.55 percentage points, respectively. Compared with RL-ZVP, the strongest competing baseline, ProRL gains 2.67 and 3.05 points on Pass@1 and Pass@16. The improvements are particularly pronounced on challenging out-of-domain benchmarks. Relative to GRPO, *ProRL* substantially improves performance on OlympiadBench, AMC 2023, and AIME 2024, suggesting that confidence-aware reward reweighting enhances credit assignment and out-of-domain generalization.

Results on Qwen2.5-7B-Instruct Fine-tuned on MATH-lighteval. Table 2 shows that *ProRL* remains effective when scaling to Qwen2.5-7B-Instruct. It achieves the highest average Pass@1 and Pass@16 scores, improving over GRPO by 2.25 and 1.37 percentage points, respectively. Compared with the strongest additional non-GRPO baselines, ProRL gains 1.67 points over RL-ZVP on Pass@1 and 2.13 points over GRPO-VPS on Pass@16. Across the four benchmarks, *ProRL* achieves the best or tied-best Pass@1 results and strong Pass@16 performance, ranking first or tied for first on OlympiadBench, MATH500, and AIME 2024 while remaining competitive on AMC 2023. These results demonstrate the scalability of confidence-aware reward reweighting across model sizes.

Table 3: Main results of Qwen2.5-1.5B-Instruct fine-tuned on DAPO-MATH-17K (%).

Method	OlympiadBench		MATH500		AMC23		AIME24		Avg.	
	Pass@1	Pass@16	Pass@1	Pass@16	Pass@1	Pass@16	Pass@1	Pass@16	Pass@1	Pass@16
Base Model	17.51	55.19	33.00	83.20	17.50	60.00	**3.33**	10.00	17.84	52.10
GRPO	19.73	59.79	48.80	87.00	30.00	65.00	**3.33**	26.67	25.47	59.62
GRPO-VPS	20.47	60.09	49.40	85.60	30.00	70.00	1.67	**28.33**	25.39	61.01
RLCR	19.29	57.86	48.00	87.00	27.50	67.50	2.50	23.33	24.32	58.92
RL-ZVP	21.51	59.79	49.00	87.40	27.50	70.00	**3.33**	26.67	25.34	60.97
ProRL (Ours)	**22.26**	**60.83**	**49.60**	**88.40**	**32.50**	**72.50**	**3.33**	26.67	**26.92**	**62.10**

Table 4: Ablation study of $\pi_{\theta_{\text{old}}}$ and π_θ on MATH-lighteval. All results are Pass@1 (%) using Qwen2.5-1.5B-Instruct.

Method	**OlympiadBench**	**MATH500**	**AMC23**	**AIME24**	**Avg.**
$\pi_{\theta_{\text{old}}}$ (Ours)	**20.47**	**58.80**	**24.38**	**6.67**	**27.58**
π_θ	19.44	45.40	11.88	1.67	19.60

Results on Qwen2.5-1.5B-Instruct Fine-tuned on DAPO-MATH-17K. To evaluate generalization under different training distributions, we further train Qwen2.5-1.5B-Instruct on DAPO-MATH-17K and compare it with GRPO and other RLVR baselines. As shown in Table 3, *ProRL* achieves the highest average Pass@1 and Pass@16 scores, outperforming GRPO by 1.45 and 2.48 percentage points, respectively. Compared with the strongest additional non-GRPO baseline, GRPO-VPS, *ProRL* further improves Pass@1 and Pass@16 by 1.53 and 1.09 points, respectively. Across the four benchmarks, *ProRL* achieves the best or tied-best Pass@1 results on all benchmarks. For Pass@16, it ranks first on OlympiadBench, MATH500, and AMC 2023 while remaining competitive on AIME 2024. These results demonstrate the robustness of confidence-aware reward reweighting across different training distributions.

Exploitation and Exploration. The consistent improvements in both Pass@1 and Pass@16 indicate that *ProRL* achieves a better balance between exploitation and exploration during reinforcement learning. By incorporating trajectory confidence into reward assignment, *ProRL* strengthens the optimization signals of trajectories that reflect stable or dominant reasoning patterns, while reducing the influence of unreliable behaviors. This confidence-aware redistribution enables the policy to consolidate reliable reasoning, refine emerging capabilities, and correct dominant failure modes without treating all sampled trajectories equally.

Importantly, trajectory confidence is estimated using the historical policy with detached gradients rather than the current policy being optimized. Therefore, confidence serves only as an auxiliary signal for reward reweighting and does not directly drive likelihood maximization. This design avoids a self-reinforcing feedback loop that could excessively amplify already preferred behaviors and reduce reasoning diversity. By decoupling confidence estimation from policy optimization, *ProRL* preserves effective exploration while improving the exploitation of valuable reasoning patterns, leading to higher reasoning accuracy without sacrificing solution diversity.

4.2 Ablation Study

Ablation on Likelihood with π_θ and $\pi_{\theta_{\text{old}}}$. We compare two choices for trajectory confidence estimation: the historical rollout policy $\pi_{\theta_{\text{old}}}$ and the current policy π_θ. By default, *ProRL* computes likelihoods using the frozen policy that generated the sampled trajectories; for comparison, we recompute them with the current policy while keeping the remaining procedures unchanged. Table 4 shows that $\pi_{\theta_{\text{old}}}$ consistently outperforms π_θ across all benchmarks.

This result highlights the importance of a stable confidence signal. Since trajectories are sampled from $\pi_{\theta_{\text{old}}}$, their likelihoods provide consistent confidence estimates at sampling time. In contrast, the evolving π_θ assigns changing confidence scores to the same trajectories, introducing instability into reward assignment. Although stop-gradient blocks direct gradient propagation through confidence estimation, it cannot eliminate this moving-target effect. Therefore, $\pi_{\theta_{\text{old}}}$ enables more stable and effective confidence-aware training.

Ablation on τ for Likelihood Normalization. We investigate the effect of the temperature τ used in likelihood-based weight normalization on Qwen2.5-1.5B-Instruct trained on MATH-lighteval. As

Table 5: Ablation study of the temperature parameter τ for likelihood normalization. All results are Pass@1 (%) using Qwen2.5-1.5B-Instruct trained on MATH-lighteval.

Temperature τ	OlympiadBench	MATH500	AMC23	AIME24	Avg.
0.5	18.25	55.80	23.75	3.33	25.28
0.7	20.47	**58.80**	**24.38**	**6.67**	**27.58**
0.8	**21.51**	56.00	22.50	3.33	25.84
1.0	19.88	56.00	21.25	1.67	24.70

shown in Table 5, $\tau = 0.7$ achieves the highest average Pass@1 accuracy of 27.58%, demonstrating the importance of proper temperature for *ProRL*.

A smaller τ produces a sharper confidence distribution, causing the reward reweighting to overemphasize a small number of high-confidence trajectories and weaken the contribution of other potentially valuable reasoning behaviors. In contrast, a larger τ smooths confidence differences, reducing the ability of *ProRL* to distinguish trajectories with different learning values and making the method behave more like standard GRPO. These results suggest that an appropriate temperature is crucial for balancing confidence-guided trajectory prioritization and diverse reasoning exploration.

Ablation of Reweighting Correct and Incorrect Trajectories. To validate the necessity of applying the confidence-aware reward reweighting strategy to both correct and incorrect trajectories, we compare *ProRL* with its two variants that reweight only correct or incorrect trajectories. Specifically, we preserve the total reward magnitude of the selected subset while only adjusting the relative contribution among trajectories within that subset.

Table 6: Ablation study of confidence-aware reweighting applied exclusively to correct or incorrect trajectories. All results are Pass@1 (%) using Qwen2.5-1.5B-Instruct trained on MATH-lighteval.

Method	OlympiadBench	MATH500	AMC23	AIME24	Avg.
Both (Ours)	**20.47**	**58.80**	**24.38**	6.67	**27.58**
Correct only	18.69	52.40	21.88	7.50	25.18
Incorrect only	17.06	53.20	21.25	**9.17**	25.17

As shown in Table 6, *ProRL* achieves the best performance, demonstrating that confidence-aware reweighting is most effective when applied to the sampled group. Among the subset-level variants, reweighting correct trajectories achieves slightly better performance than reweighting incorrect trajectories, highlighting the importance of distinguishing the reliability of successful reasoning patterns. However, incorrect trajectories provide valuable signals for refining dominant failure modes. Reweighting either correct or incorrect trajectories alone cannot capture the full range of trajectory-level differences. These results confirm that *ProRL* benefits from jointly modeling confidence across both correct and incorrect trajectories, enabling effective progressive reasoning optimization.

5 Conclusion

In this paper, we propose *Progressive Reasoning Reinforcement Learning (ProRL)*, a confidence-aware RLVR framework that improves reasoning optimization beyond binary outcome supervision. We show that sampled reasoning trajectories exhibit varying levels of reliability and learning value, and that treating them indiscriminately can reinforce unreliable behaviors and lead to weak or even vanishing learning signals while overlooking informative refinement signals. By incorporating historical policy likelihood as a detached confidence estimate, *ProRL* dynamically reweights trajectory rewards to enable progressive consolidation of reliable reasoning, refinement of emerging capabilities, and correction of dominant errors. Extensive experiments on MATH-lighteval and DAPO-MATH-17K demonstrate that *ProRL* consistently improves reasoning performance across different model scales while preserving solution diversity. Our work highlights the importance of confidence-aware credit assignment for advancing reinforcement learning-based reasoning optimization.

AI USE STATEMENT

In this work, we used generative AI tools to polish the writing of the manuscript and to assist with initial annotations for the reasoning trajectory analysis. All AI-assisted edits and annotations were reviewed and verified by the authors. We take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide the key implementation and evaluation details in the main paper and appendix. Sections 3.2 and 3.3 describe the formulation of *ProRL*, including trajectory confidence estimation, reward reweighting, and policy optimization. The experimental setup, including the training data, backbone models, baselines, sampling strategy, evaluation metrics, and answer verification procedure, is described in Section 4. The exact prompt template is provided in Appendix A, and the detailed training configuration is reported in Appendix B. Additional ablation details and results are provided in Appendix C. We also document the protocol used for the Chain-of-Thought quality analysis in Appendix D, together with representative annotated examples in Appendix E. The trajectory-correction analysis and its experimental procedure are described in Appendix F. These materials collectively provide the information needed to reproduce the main experiments and analyses reported in this work.

REFERENCES

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, and Denny Zhou. Universal self-consistency for large language model generation, 2023. URL https://arxiv.org/abs/2311.17311.

Mehul Damani, Isha Puri, Stewart Slocum, Idan Shenfeld, Leshem Choshen, Yoon Kim, and Jacob Andreas. Beyond binary rewards: Training LMs to reason about their uncertainty, 2026. URL https://arxiv.org/abs/2507.16806.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. A survey of confidence estimation and calibration in large language models, 2024. URL https://arxiv.org/abs/2311.08298.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li,

Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Qianxi He, Qingyu Ren, Shanzhe Lei, Xuhong Wang, and Yingchun Wang. Beyond correctness: Confidence-aware reward modeling for enhancing large language model reasoning, 2025. URL https://arxiv.org/abs/2511.07483.

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

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

Hugging Face. Open R1: A fully open reproduction of DeepSeek-R1, January 2025. URL https://github.com/huggingface/open-r1.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022. URL https://arxiv.org/abs/2207.05221.

Muhammad Khalifa, Rishabh Agarwal, Lajanugen Logeswaran, Jaekyeom Kim, Hao Peng, Moontae Lee, Honglak Lee, and Lu Wang. Process reward models that think, 2025. URL https://arxiv.org/abs/2504.16828.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention, 2023. URL https://arxiv.org/abs/2309.06180.

Thanh-Long V. Le, Myeongho Jeon, Kim Vu, Viet Lai, and Eunho Yang. No prompt left behind: Exploiting zero-variance prompts in LLM reinforcement learning via entropy-guided advantage shaping, 2026. URL https://arxiv.org/abs/2509.21880.

Yiqing Liang, Jielin Qiu, Wenhao Ding, Zuxin Liu, James Tompkin, Mengdi Xu, Mengzhou Xia, Zhengzhong Tu, Laixi Shi, and Jiacheng Zhu. MoDoMoDo: Multi-domain data mixtures for multimodal LLM reinforcement learning, 2025. URL https://arxiv.org/abs/2505.24871.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step, 2023. URL https://arxiv.org/abs/2305.20050.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words, 2022. URL https://arxiv.org/abs/2205.14334.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Contextualized sequence likelihood: Enhanced confidence scores for natural language generation, 2024. URL https://arxiv.org/abs/2406.01806.

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

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-rft: Visual reinforcement fine-tuning, 2025b. URL https://arxiv.org/abs/2503.01785.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Meiqi Guo, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, Jiao Sun, and Abhinav Rastogi. Improve mathematical reasoning in language models by automated process supervision, 2024. URL https://arxiv.org/abs/2406.06592.

Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, volume 99, pp. 278–287. Citeseer, 1999.

Mohammad Rezaei, Jens Lehmann, and Sahar Vahdati. LLM reasoning with process rewards for outcome-guided steps, 2026. URL https://arxiv.org/abs/2604.02341.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/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. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback, 2023. URL https://arxiv.org/abs/2305.14975.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback, 2022. URL https://arxiv.org/abs/2211.14275.

Jingyi Wang, Lei Zhu, Tengjin Weng, Song-Li Wu, Haochen Tan, Jierun Chen, Chaofan Tao, Haoli Bai, Lu Hou, Lifeng Shang, and Xiao-Ping Zhang. GRPO-VPS: Enhancing group relative policy optimization with verifiable process supervision for effective reasoning, 2026. URL https://arxiv.org/abs/2604.20659.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2023. URL https://arxiv.org/abs/2203.11171.

Yuxiang Wei, Olivier Duchenne, Jade Copet, Quentin Carbonneaux, Lingming Zhang, Daniel Fried, Gabriel Synnaeve, Rishabh Singh, and Sida I. Wang. SWE-RL: Advancing LLM reasoning via reinforcement learning on open software evolution, 2025. URL https://arxiv.org/abs/2502.18449.

Tianbao Xie, Siheng Zhao, Chen Henry Wu, Yitao Liu, Qian Luo, Victor Zhong, Yanchao Yang, and Tao Yu. Text2reward: Reward shaping with language models for reinforcement learning, 2024. URL https://arxiv.org/abs/2309.11489.

Yutao Xie, Nathaniel Thomas, Nicklas Hansen, Yang Fu, Li Erran Li, and Xiaolong Wang. TIPS: Turn-level information-potential reward shaping for search-augmented LLMs, 2026. URL https://arxiv.org/abs/2603.22293.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Jiarui Yao, Ruida Wang, Hao Bai, and Tong Zhang. Future-KL regularized GRPO: Process-level credit assignment from f-divergence regularization, 2026. URL https://arxiv.org/abs/2601.10201.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476.

Tunyu Zhang, Haizhou Shi, Yibin Wang, Hengyi Wang, Xiaoxiao He, Zhuowei Li, Haoxian Chen, Ligong Han, Kai Xu, Huan Zhang, Dimitris Metaxas, and Hao Wang. Tokur: Token-level uncertainty estimation for large language model reasoning, 2026. URL https://arxiv.org/abs/2505.11737.

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. The lessons of developing process reward models in mathematical reasoning, 2025. URL https://arxiv.org/abs/2501.07301.

Andrew Zhao, Yiran Wu, Yang Yue, Tong Wu, Quentin Xu, Yang Yue, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data, 2025. URL https://arxiv.org/abs/2505.03335.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025. URL https://arxiv.org/abs/2507.18071.