INTER-ITERATION POLICY IMPROVEMENT FOR LLM REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning is central to post-training LLMs for reasoning and agentic tasks. However, existing methods optimize rewards, advantages, or distillation targets constructed from the current batch without using subsequent performance observations to inform later optimization. This open-loop structure leaves recent policy progress outside the learning signal and makes optimization sensitive to finite sampling, generation stochasticity, and noisy feedback. We introduce Policy Improvement Reinforcement Learning (PIRL), a trajectory-level formulation that makes inter-iteration policy improvement an explicit optimization signal while retaining cumulative equivalence to terminal task performance. Building on PIRL, we propose Policy Improvement Policy Optimization (PIPO), a plug-in framework that closes the loop around policy updates. At each iteration, PIPO compares current empirical performance with a sliding-window historical anchor and uses the resulting progress signal to retrospectively modulate the base algorithm's local attribution. We establish local alignment with isolated base-step improvement under explicit trend-coherence conditions and characterize signed retrospective replay for group-relative optimization. Across two model scales and mathematical reasoning, code, tool-use, and self-distillation tasks, PIPO consistently improves PPO, GRPO, GSPO, DAPO, and SDPO under matched training and evaluation settings across all reported experiments.

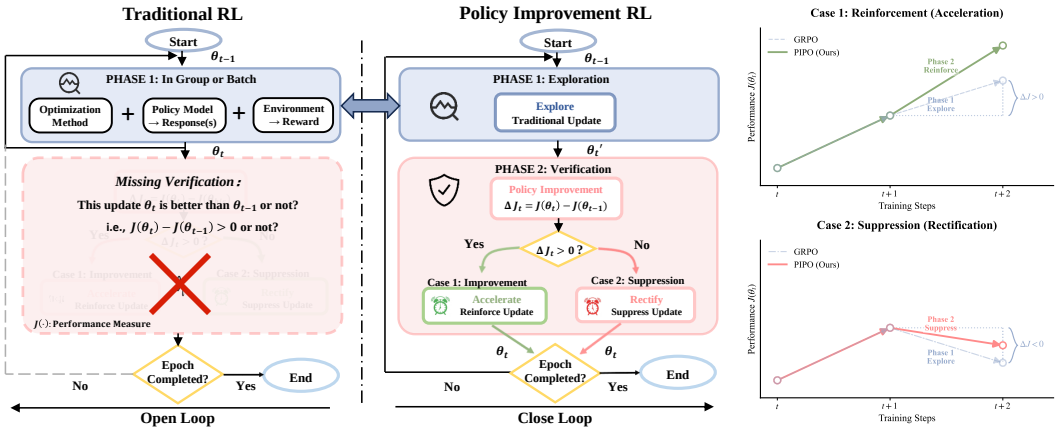


Figure 1: **Overview of Policy Improvement Reinforcement Learning (PIRL). Left**: Existing methods update from instantaneous rewards without subsequent feedback on policy progress, forming an open loop. **Middle**: PIRL adds retrospective progress feedback and closes the optimization loop. **Right**: Positive empirical feedback reinforces stored update directions, while negative feedback triggers corrective replay.

1 INTRODUCTION

Reinforcement learning is now central to post-training language models for reasoning, coding, and tool use (Guo et al., 2025a; Yang et al., 2025a; Qian et al., 2025). A prominent line of work is reinforcement learning with verifiable rewards (RLVR), where answer verifiers, program execution, and tool feedback provide scalable supervision for reasoning and executable tasks (Shao et al., 2024;

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.

He et al., 2026a). Recent progress has produced several families of policy optimization methods. PPO remains a standard proximal policy optimization algorithm (Schulman et al., 2017), while group-relative methods such as GRPO (Shao et al., 2024) and its extensions remove the critic by normalizing rewards within groups of sampled responses.

Beyond scalar rewards, reinforcement learning with rich feedback (RLRF) uses execution errors, judge comments, and tool-use traces to construct denser learning signals. A prominent direction is on-policy distillation (OPD), in which a student learns from teacher supervision on its own sampled trajectories (Song & Zheng, 2026; Li et al., 2026a). Self-distillation further specializes this idea by instantiating the teacher and student from the same model under different contexts (Hübotter et al., 2026; Zhao et al., 2026a).

Despite their different local learning signals, these methods share a common limitation: they update the policy from rewards, advantages, or distillation targets computed on the current batch without feeding subsequent performance observations back into optimization, forming an *open-loop* optimization structure. Improving a local objective on sampled trajectories does not ensure that the resulting policy outperforms its predecessor (Yue et al., 2025; Wu et al., 2025a). Finite sampling, generation stochasticity, and feedback noise further separate the local signal from realized policy progress (Yang et al., 2026; Chen et al., 2026b; Cheng et al., 2026).

We address this limitation through *policy-improvement feedback*: a signal revealed by subsequent batches that captures inter-iteration progress. This signal allows recent performance trends to inform later updates rather than treating each step as an isolated local optimization problem. We formalize this principle as **Policy Improvement Reinforcement Learning (PIRL)**, a trajectory-level formulation that treats the expected performance gain of each policy transition as an explicit learning signal. PIRL remains exactly aligned with final policy performance while exposing a temporal feedback channel absent from step-wise surrogate optimization, moving RL post-training from isolated updates toward sustained, closed-loop improvement.

Building on PIRL, we propose **Policy Improvement Policy Optimization (PIPO)**, a plug-in *closed-loop* algorithm based on inter-batch feedback. At each iteration, PIPO compares current empirical performance with a sliding-window historical anchor and uses the resulting progress signal to modulate the stored local attribution of the base algorithm: positive trends reinforce the direction, while negative trends trigger corrective replay. This construction equips PPO, group-relative optimization, and self-distillation methods with a common form of closed-loop policy-improvement feedback.

Our contributions are: (1) **New Formulation:** We introduce *Policy Improvement Reinforcement Learning*, a trajectory-level formulation that makes policy improvement an explicit optimization signal while preserving exact alignment with final performance. (2) **Plug-in Algorithm:** We propose PIPO, a closed-loop algorithm compatible with PPO, group-relative methods such as GRPO, GSPO, and DAPO, and self-distillation methods such as SDPO. (3) **Analysis and Evaluation:** We establish local base-step alignment under a trend-coherence condition, characterize group-relative retrospective replay, and show consistent gains across mathematical reasoning, code, tool-use, and self-distillation settings.

2 PRELIMINARIES

For a query $q \sim \mathcal{D}$, an autoregressive policy π_θ generates a response $y = (y_1, \ldots, y_{|y|})$ and receives reward $R(q, y)$. Let $h_\tau = (q, y_{<\tau})$ denote the token-level state. We define the per-query performance and the standard RL objective as

$$\begin{aligned} p(q;\theta) &:= \mathbb{E}_{y\sim\pi_\theta(\cdot|q)}[R(q,y)], \\ J_{\mathrm{RL}}(\theta) &:= \mathbb{E}_{q\sim\mathcal{D}}[p(q;\theta)]. \end{aligned} \tag{1}$$

At iteration t, we write the rollout batch and its token histories as

$$\mathcal{B}_t := \{(q_{t,i}, y_{t,i}, R_{t,i})\}_{i=1}^{N_t}, \qquad h_{t,i,\tau} := (q_{t,i}, y_{t,i,<\tau})$$

Here θ_t denotes the policy that generated $\mathcal{B}_t$, whereas θ denotes the candidate policy being optimized. Their token-level importance ratio is

$$\rho_{t,i,\tau}(\theta) := \frac{\pi_\theta(y_{t,i,\tau} \mid h_{t,i,\tau})}{\pi_{\theta_t}(y_{t,i,\tau} \mid h_{t,i,\tau})}. \tag{2}$$

The policy-gradient theorem expresses the objective gradient through token-level credit assignment:

$$\nabla_\theta J_{\mathrm{RL}}(\theta) = \mathbb{E}_{q,y}\left[\sum_{\tau=1}^{|y|} A^{\pi_\theta}(h_\tau, y_\tau)\nabla_\theta \log \pi_\theta(y_\tau \mid h_\tau)\right], \tag{3}$$

where A^{π_θ} is the action advantage. Practical algorithms differ mainly in how they construct this local attribution and constrain the resulting update.

PPO. PPO (Schulman et al., 2017) learns a value function and estimates token-level advantages using generalized advantage estimation (GAE):

$$A^{\mathrm{GAE}}_{t,i,\tau} = \sum_{\ell\geq 0}(\gamma\lambda_{\mathrm{GAE}})^\ell \delta_{t,i,\tau+\ell}, \qquad \delta_{t,i,\tau} = r_{t,i,\tau} + \gamma V_\psi(h_{t,i,\tau+1}) - V_\psi(h_{t,i,\tau}). \tag{4}$$

It combines this attribution with the clipped importance ratio in Eq. (2), thereby limiting how far the updated policy moves from the rollout policy.

Group-relative optimization. GRPO (Shao et al., 2024) removes the learned critic. For G responses sampled for the same query q, it constructs

$$\mu_{t,q} = \frac{1}{G}\sum_{j=1}^{G} R_{t,q,j}, \qquad \sigma^2_{t,q} = \frac{1}{G}\sum_{j=1}^{G}(R_{t,q,j} - \mu_{t,q})^2, \qquad A^{\mathrm{grp}}_{t,q,i} = \frac{R_{t,q,i} - \mu_{t,q}}{\sigma_{t,q}}, \tag{5}$$

and broadcasts this response-level advantage to all tokens in the response. GSPO and DAPO retain group-relative credit while changing the importance ratio, clipping, aggregation, or group filtering.

Self-distillation policy optimization. SDPO (Hübotter et al., 2026) forms a feedback-conditioned self-teacher $q_{\theta_t}(\cdot \mid h_{t,i,\tau}, f_{t,i})$, where $f_{t,i}$ is privileged post-rollout feedback, and minimizes the reverse KL from the policy to this stop-gradient teacher. Its dense token-level attribution is the teacher–student log-ratio

$$A^{\mathrm{SDPO}}_{t,i,\tau} := \log q_{\theta_t}(y_{t,i,\tau} \mid h_{t,i,\tau}, f_{t,i}) - \log \pi_{\theta_t}(y_{t,i,\tau} \mid h_{t,i,\tau}). \tag{6}$$

Equations (4)–(6) specify the local attribution used by each family. Their complete updates retain algorithm-specific clipping, aggregation, filtering, and reverse-KL structure, as detailed in Appendix A.3. PIRL adds inter-iteration performance feedback without replacing these local objectives.

3 Policy Improvement Reinforcement Learning

PIRL formalizes retrospective progress along a policy-learning trajectory $\theta_{0:T} := (\theta_0, \ldots, \theta_T)$. Let $J_{\mathrm{RL}}(\theta)$ denote the expected performance in Eq. (1). For transition $t \in \{0, \ldots, T-1\}$, its *policy improvement* is

$$\Delta J_t := J_{\mathrm{RL}}(\theta_{t+1}) - J_{\mathrm{RL}}(\theta_t). \tag{7}$$

Rather than viewing each update as an isolated surrogate optimization, Policy Improvement Reinforcement Learning (PIRL) evaluates the trajectory through cumulative expected progress:

$$\mathcal{J}_{\mathrm{PIRL}}(\theta_{0:T}) := \sum_{t=0}^{T-1} \mathbb{E}[\Delta J_t] = \mathbb{E}[J_{\mathrm{RL}}(\theta_T) - J_{\mathrm{RL}}(\theta_0)]\,. \tag{8}$$

For a fixed initialization and feasible trajectory class, the telescoping identity makes cumulative policy improvement exactly equivalent to expected terminal performance. Its transition-wise decomposition exposes each policy change as a unit of progress that can inform subsequent optimization, providing the objective-level basis for closing the loop around policy updates.

4 Policy Improvement Policy Optimization

PIPO instantiates the PIRL formulation by converting empirical inter-iteration progress into retrospective optimization feedback. It combines the base algorithm's local attribution with a global progress signal that modulates stored learning directions, yielding a plug-in mechanism for PPO, group-relative optimization, and self-distillation optimization.

4.1 POLICY IMPROVEMENT REWARD

At iteration t, PIPO estimates the current performance $J_{\mathrm{RL}}(\theta_t)$ using the mean verifier reward μ_t over a freshly sampled batch $\mathcal{B}_t$:

$$\mu_t := \frac{1}{|\mathcal{B}_t|} \sum_{(q,y)\in\mathcal{B}_t} R(q,y), \qquad y \sim \pi_{\theta_t}(\cdot \mid q). \tag{9}$$

Here, μ_t measures the empirical performance of the current raw policy. PIPO compares it with a sliding-window anchor that summarizes recent policy performance. We compute the historical mean μ_{his} and standard deviation σ_{his} over the past K iterations as

$$\mu_{\mathrm{his}} = \frac{1}{K}\sum_{k=1}^{K} \mu_{t-k}, \qquad \sigma_{\mathrm{his}} = \sqrt{\frac{1}{K}\sum_{k=1}^{K}(\mu_{t-k} - \mu_{\mathrm{his}})^2}. \tag{10}$$

The difference $\mu_t - \mu_{\mathrm{his}}$ measures current empirical performance relative to its recent historical level. Following the implementation, we use

$$\widetilde{\sigma}_{\mathrm{his}} := \begin{cases} \sigma_{\mathrm{his}}, & \sigma_{\mathrm{his}} \geq \varepsilon_\sigma, \\ 1, & \sigma_{\mathrm{his}} < \varepsilon_\sigma, \end{cases} \qquad \varepsilon_\sigma > 0,$$

and define the *standardized improvement feedback* as

$$\xi_t := \frac{\mu_t - \mu_{\mathrm{his}}}{\widetilde{\sigma}_{\mathrm{his}}}. \tag{11}$$

When the historical variance is numerically uninformative, the unit fallback uses the unstandardized performance difference instead of amplifying it by a near-zero denominator. Specifically, $\xi_t > 0$ provides positive feedback for the stored direction, whereas $\xi_t < 0$ triggers a corrective retrospective update. Since this scalar feedback does not specify sample- or token-level credit, PIPO uses the base algorithm's attribution scores as the local basis for modulation.

Definition 4.1 (Policy Improvement Reward)**.** For each historical learning unit $u_{t-1,i}$ generated during the previous iteration $t-1$ (e.g., a sampled response for PPO or group-relative methods, or a token/step for self-distillation), let $a_{t-1,i}$ denote its local attribution score. The *Policy Improvement (PI) Reward* evaluated at iteration t is constructed as:

$$\hat{r}^{\mathrm{PI}}_{t,i} := \underbrace{a_{t-1,i}}_{\text{Local Attribution}} \cdot \underbrace{\phi_\lambda(\xi_t)}_{\text{Improvement Feedback}}, \tag{12}$$

where the choice of $a_{t-1,i}$ depends on the base RL algorithm. For $\lambda \geq 0$, we map ξ_t to a modulation coefficient using the following rectification function $\phi_\lambda : \mathbb{R} \to \mathbb{R}$:

$$\phi_\lambda(x) = \begin{cases} x & \text{if } x \geq 0, \\ \lambda \cdot x & \text{if } x < 0. \end{cases}$$

This design treats positive and negative feedback asymmetrically: positive feedback keeps its original magnitude, while λ controls the strength of correction under negative feedback. Section 5.4 evaluates this choice empirically.

This construction separates two roles: $a_{t-1,i}$ preserves the base algorithm's local credit assignment, while $\phi_\lambda(\xi_t)$ modulates the stored direction according to the empirical trend.

For PPO and SDPO, PIPO directly uses the token-level GAE and self-teacher log-ratio scores defined in Section 2. For group-relative policy optimization, we use GRPO as the representative instantiation and further normalize its response-level attribution within each group of size G. Following the group-relative implementation, define

$$D_{t-1,q} := \sum_{j=1}^{G} |A^{\mathrm{grp}}_{t-1,q,j}| + \varepsilon_D, \qquad \varepsilon_D > 0.$$

We use

$$a^{\mathrm{grp}}_{t-1,q,i} = \frac{GA^{\mathrm{grp}}_{t-1,q,i}}{D_{t-1,q}}.$$

Here $A^{\mathrm{grp}}_{t-1,q,i}$ is the group-relative score defined in Section 2. This normalization controls the scale of the PI attribution, as characterized in Section 6. For GSPO, DAPO, and other GRPO variants, we apply the same normalization to their corresponding group-level attribution scores. Thus, the same closed-loop feedback operates on PPO advantages, group-relative advantages, and dense SDPO token attributions.

4.2 Optimization Procedure

Figure 1 summarizes the temporal relation between PIPO's two phases. Phase 1 updates the feedback-adjusted policy $\tilde{\theta}_t$ on $\mathcal{B}_t$ to obtain θ_{t+1}. After θ_{t+1} generates $\mathcal{B}_{t+1}$, Phase 2 evaluates the preceding transition and produces $\tilde{\theta}_{t+1}$, from which the next base update starts. Full pseudocode, including historical-memory initialization and the K-step warm-up, is provided in Appendix A.1. At iteration t, $\mathcal{B}_t$ is sampled by the raw policy θ_t before retrospective adjustment. It measures current empirical progress, supplies attribution for the subsequent base update, and is retained for retrospective replay at iteration $t+1$.

Phase 1: Exploration. Starting from the feedback-adjusted policy $\tilde{\theta}_t$, PIPO applies the selected base RL algorithm on the current batch to obtain the next policy:

$$\theta_{t+1} \leftarrow \tilde{\theta}_t + \alpha_{\mathrm{std}} \cdot \nabla_\theta \mathcal{J}_{\mathrm{base}}(\tilde{\theta}_t; \mathcal{B}_t),$$

where α_{std} is the learning rate and $\mathcal{J}_{\mathrm{base}}$ denotes the base policy-learning objective, instantiated by PPO, GRPO, SDPO, or related variants as detailed in Appendix A.3. This phase applies the standard update of the selected base optimizer.

Phase 2: Retrospective feedback. After θ_{t+1} samples $\mathcal{B}_{t+1}$, its batch mean μ_{t+1} is compared with the historical anchor to obtain ξ_{t+1}. PIPO then constructs PI rewards for the previous batch $\mathcal{B}_t$ and performs a retrospective update:

$$\begin{aligned} \tilde{\theta}_{t+1} &\leftarrow \theta_{t+1} + \alpha_{\mathrm{PI}} \cdot \nabla_\theta \mathcal{J}_{\mathrm{PI}}(\theta_{t+1}; \mathcal{B}_t), \\ \mathcal{J}_{\mathrm{PI}}(\theta; \mathcal{B}_t) &= \frac{1}{N} \sum_{i=1}^{N} \min\Big(\rho^{\mathrm{PI}}_i(\theta) \hat{r}^{\mathrm{PI}}_{t+1,i},\ \mathrm{clip}(\rho^{\mathrm{PI}}_i(\theta), 1-\epsilon, 1+\epsilon) \hat{r}^{\mathrm{PI}}_{t+1,i} \Big), \end{aligned} \tag{13}$$

where α_{PI} is the learning rate, and ρ^{PI}_i is the retrospective importance ratio against the previous policy π_{θ_t} that generated batch $\mathcal{B}_t$. Its token-level form is

$$\rho^{\mathrm{PI}}_{t,i,\tau}(\theta) = \frac{\pi_\theta(y_{t,i,\tau} \mid q_{t,i}, y_{t,i,<\tau})}{\pi_{\theta_t}(y_{t,i,\tau} \mid q_{t,i}, y_{t,i,<\tau})},$$

which is evaluated at $\theta = \theta_{t+1}$ in Eq. (13). Sequence-level objectives use the analogous sequence ratio. The importance ratio transports the stored trajectories from their behavior policy to the current evaluation point, while clipping retains a proximal update. The scalar feedback changes the direction and magnitude of replay, whereas the stored attribution preserves sample- and token-level credit from the base algorithm.

This update modulates the stored local attributions according to the observed performance trend: positive feedback reinforces their direction, while negative feedback scales the corrective PI signal through $\phi_\lambda(\cdot)$. The resulting policy $\tilde{\theta}_{t+1}$ is then used as the starting point for the next base RL update.

5 Experiments

Our experiments address four questions: *(i)* Does PIPO yield consistent gains across policy optimization families, model scales, and reasoning benchmarks? *(ii)* How does retrospective policy-

Table 1: Pass@1 accuracy (%) on five mathematical reasoning benchmarks. Bold marks improvements over the corresponding baseline.

Method	*Qwen3-4B-Base*						*Qwen3-8B-Base*					
	MATH500	AIME25	AMC23	Minerva	Olympiad	**AVG**	MATH500	AIME25	AMC23	Minerva	Olympiad	**AVG**
Base Model	58.2	7.4	45.0	14.0	28.6	30.6	65.0	11.1	45.0	17.3	31.1	33.9
PPO	80.3	11.1	62.5	25.0	41.2	44.0	81.3	18.5	67.5	25.4	43.0	47.1
+PIPO	80.1	**18.5**	**65.0**	**25.7**	39.7	**45.8**	**83.7**	18.5	**70.0**	**28.3**	**43.3**	**48.8**
GRPO	79.3	18.5	60.0	21.0	40.5	43.9	80.1	22.2	67.5	27.6	42.4	48.0
+PIPO	**80.9**	18.5	60.0	**26.8**	**44.4**	**46.1**	**82.3**	22.2	67.5	**29.0**	**43.6**	**48.9**
GSPO	78.5	18.5	62.5	23.2	39.8	44.5	81.7	22.2	67.5	26.8	45.5	48.7
+PIPO	**80.3**	**22.2**	60.0	**24.6**	**41.8**	**45.8**	**83.5**	**25.9**	**72.5**	**28.3**	**47.5**	**51.5**
DAPO	81.3	22.2	65.0	21.7	41.8	46.4	85.3	25.9	75.0	27.9	48.7	52.6
+PIPO	**82.7**	22.2	**70.0**	**25.0**	**46.3**	**49.2**	**86.3**	**29.6**	75.0	**30.9**	**52.2**	**54.8**

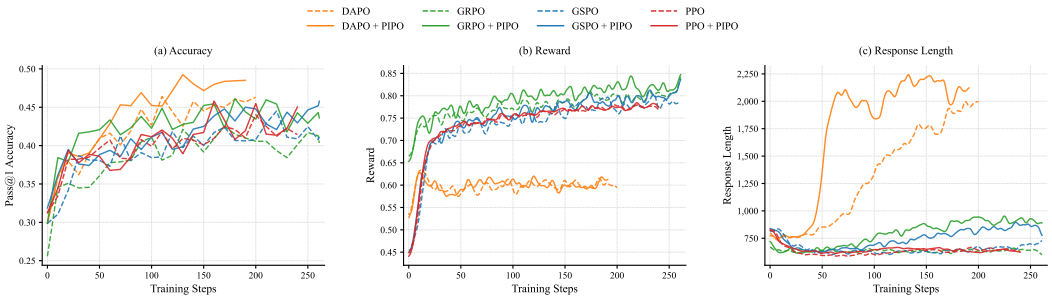


Figure 2: Training dynamics on Qwen3-4B-Base: mean five-benchmark Pass@1, training reward, and response length. Solid and dashed lines denote methods with and without PIPO.

improvement feedback affect training dynamics, including optimization stability and response-length behavior? *(iii)* Does PIPO extend to code and tool-use RL tasks? *(iv)* How sensitive is PIPO to the rectification strategy and historical window size? We evaluate PIPO as a plug-in module for PPO, group-relative, and self-distillation baselines across model scales, task domains, and ablation settings.

Setup Details. We train Qwen3-4B-Base and Qwen3-8B-Base (Yang et al., 2025a) on MATH (Hendrycks et al., 2021) and evaluate on MATH500, AIME 2025, AMC 2023, Minerva, and OlympiadBench (He et al., 2024). Code and tool-use tasks evaluate domain transfer, while SciKnowEval (Feng et al., 2024) evaluates integration with self-distillation. All comparisons share the same evaluation pipeline; Appendix A provides full details.

5.1 Main Results

Overall Performance. Table 1 shows that PIPO improves the average score of every base algorithm at both model scales. Gains range from 1.3 to 2.8 points on Qwen3-4B-Base and from 0.9 to 2.8 points on Qwen3-8B-Base, with the largest gains obtained by DAPO at 4B and GSPO at 8B. The improvements span critic-based PPO and three critic-free group-relative objectives, showing that inter-iteration feedback complements different forms of local attribution. In particular, DAPO rises from 46.4 to 49.2 at 4B and from 52.6 to 54.8 at 8B. By retaining each optimizer's attribution rule and adding a shared temporal signal, PIPO applies the same closed-loop mechanism across otherwise distinct update families.

Training Dynamics. Figure 2 tracks mean five-benchmark Pass@1, training reward, and response length on Qwen3-4B-Base. Accuracy and reward rise together during training, with PIPO variants generally reaching higher, stable plateaus. Response-length trajectories remain characteristic of each base optimizer, while accuracy and reward improve across these different reasoning profiles. Appendix B reports additional multi-seed training dynamics.

5.2 CODE AND TOOL-USE RL TASKS

Table 2: Code and Tool-Use RL Results for Qwen3-4B-Base.

	Code					Tool-Use			
Method	TACO	LCBv6	HumanEval	MBPP	Code AVG	RLLA	BFCL	APIBank	Tool AVG
PPO	23.4	31.2	85.4	41.4	45.4	87.1	66.6	90.4	81.4
+PIPO	22.1	**34.9**	**86.0**	**45.0**	**47.0**	**89.0**	**71.2**	90.4	**83.5**
GRPO	23.8	31.5	83.5	44.2	45.8	85.0	58.2	90.6	77.9
+PIPO	**25.4**	**32.5**	**86.0**	43.6	**46.9**	**85.6**	**61.7**	**91.7**	**79.7**
GSPO	23.3	32.6	81.7	45.0	45.7	85.6	59.7	89.6	78.3
+PIPO	**23.4**	32.1	**84.2**	**45.8**	**46.4**	84.0	**62.8**	**90.6**	**79.1**
DAPO	25.4	33.5	86.0	43.8	47.2	87.4	62.2	90.2	79.9
+PIPO	**27.3**	**36.4**	**87.2**	**44.4**	**48.8**	**87.7**	**67.7**	**90.4**	**81.9**

Across PPO, GRPO, GSPO, and DAPO, PIPO improves Code AVG by 0.7–1.6 points and Tool AVG by 0.8–2.1 points. Unlike mathematical answer matching, these domains use executable tests and structured tool feedback. All eight method–domain averages improve, extending the same inter-iteration controller across verifier types and base optimizers.

5.3 PIPO WITH SELF-DISTILLATION

Table 3: Self-Distillation Study on SciKnowEval.

Method	Biology	Chemistry	Material	Physics	AVG
Qwen3-8B	4.0	13.3	39.4	30.0	21.7
GRPO	40.0	72.9	71.3	47.5	57.9
+PIPO	**54.0**	60.5	**73.4**	**53.8**	**60.4**
SDPO	52.0	76.7	73.4	77.5	69.9
+PIPO	**60.0**	**79.1**	**79.8**	76.3	**73.8**

Table 3 evaluates PIPO with dense token-level self-teacher feedback in the SDPO SciKnowEval setting (Hübotter et al., 2026). PIPO raises the GRPO average from 57.9 to 60.4 ($+2.5$) and the SDPO average from 69.9 to 73.8 ($+3.9$). SDPO supplies fine-grained attribution from a privileged self-teacher, while PIPO contributes a global temporal signal across policy iterations; their combination shows that the two feedback levels provide complementary information.

5.4 ABLATION STUDIES

Table 4: Ablation on Rectification Coefficient λ (Qwen3-4B-Base).

λ	MATH500	AIME25	AMC23	Minerva	Olympiad	AVG
GRPO Baseline	79.3	18.5	60.0	21.0	40.5	43.9
0	80.5	18.5	62.5	22.1	39.7	44.7
0.05	78.5	**22.2**	62.5	26.1	39.1	45.7
0.1	80.9	18.5	60.0	**26.8**	**44.4**	**46.1**
0.2	**81.9**	18.5	60.0	25.0	42.6	45.6
0.5	81.3	18.5	**65.0**	22.8	40.9	45.7
1	78.3	**22.2**	57.5	23.5	41.2	44.5

Rectification Coefficient λ. Table 4 varies the scale applied to negative feedback. Intermediate attenuation performs best: $\lambda \in [0.05, 0.5]$ yields similar averages of 45.6–46.1%, with the highest score at $\lambda = 0.1$. Performance remains stable across this range of moderate coefficient values.

Window Size K**.** A moderate window ($K = 8$) balances historical smoothing and responsiveness. The full K ablation, multi-seed stability, wall-clock analysis, and a matched fixed-positive replay control are in Appendix B.

6 Theoretical Analysis

We analyze local retrospective alignment and its group-relative instantiation; proofs are in Appendix C.

6.1 Local Retrospective Alignment

Algorithm 1 separates the retrospective update from the subsequent base transition $\widetilde{\theta}_{t-1} \to \theta_t$. Let

$$\Delta J_t^{\text{base}} := J_{\text{RL}}(\theta_t) - J_{\text{RL}}(\widetilde{\theta}_{t-1}), \quad d_t := \theta_t - \widetilde{\theta}_{t-1} = \alpha_{\text{base}} g_{t-1}^{\text{base}}.$$

In the attribution-matched regime, replay with scalar feedback c_t has the local form

$$\Delta\theta_{\text{PI}}(c_t) = \alpha_{\text{PI}} c_t \left(g_{t-1}^{\text{base}} + e_t\right), \qquad \|e_t\| = \mathcal{O}(\|d_t\|).$$

Proposition 6.1 (Local Retrospective Alignment)**.** *Under the smoothness, bounded-gradient, attribution-matching, and local trust-region conditions in Appendix C.2,*

$$\langle \nabla J_{\text{RL}}(\theta_t), \Delta\theta_{\text{PI}}(c_t) \rangle = \frac{\alpha_{\text{PI}}}{\alpha_{\text{base}}} c_t \Delta J_t^{\text{base}} + \mathcal{O}(\alpha_{\text{PI}} |c_t| \|d_t\|).$$

Thus any feedback satisfying $c_t \Delta J_t^{\text{base}} \geq 0$ has a non-negative leading alignment term. In particular, ideal feedback $c_t = \phi_\lambda(\Delta J_t^{\text{base}})$ satisfies this condition directly.

This result gives PIPO a signed control interpretation: progress-aligned feedback reinforces improving base directions and corrects degrading ones while preserving the base algorithm's local attribution.

Practical PIPO uses $c_t = \phi_\lambda(\xi_t)$. Let $\delta_t^{(K)} := J_{\text{RL}}(\theta_t) - K^{-1} \sum_{k=1}^{K} J_{\text{RL}}(\theta_{t-k})$ denote the population window trend. For n conditionally independent prompt groups with bounded verifier rewards, Lemma C.1 gives, for any $m > 0$,

$$\mathbb{P}\Big(\text{sign}(\xi_t) \neq \text{sign}(\delta_t^{(K)}),\ |\delta_t^{(K)}| \geq m\Big) \leq 2 \exp\left(-\frac{2nm^2}{1 + 1/K}\right).$$

When the raw-policy trend agrees with the intervening base step, $\delta_t^{(K)} \Delta J_t^{\text{base}} > 0$ (*local trend coherence*), the empirical feedback yields a non-negative leading alignment term with high probability. The bound quantifies the reliability of batch-level feedback: its sign becomes more reliable with more prompt groups, larger trend margins, and historical averaging. Together with Proposition 6.1, it connects observed policy progress to the direction of retrospective optimization.

6.2 Conditional Group-Relative Sensitivity and Control

Theorem 6.2 (Conditional Group-Relative Scaling Sensitivity)**.** *For a fixed query with binary rewards, let S count the successes among G responses, $p = p(q;\theta)$, and $\mathcal{E} := \{1 \leq S \leq G-1\}$. Under the i.i.d.-rollout and local trust-region conditions in Appendix C.4,*

$$g_{\text{GRPO}}(\theta) = \eta(p) \nabla_\theta p, \quad \eta(p) = \frac{\sum_{k=1}^{G-1} \sqrt{k(G-k)} \binom{G}{k} p^k (1-p)^{G-k}}{G\, p(1-p)\, (1 - p^G - (1-p)^G)}.$$

Thus conditional GRPO is a state-dependent rescaling of $\nabla_\theta p$, with $\eta(p)$ diverging toward the probability boundaries.

The theorem identifies the scale of group-relative learning as a natural control target: the update follows the success-probability gradient, while its state-dependent multiplier grows for near-solved and near-impossible queries.

Proposition 6.3 (Signed Groupwise Retrospective Replay)**.** *Let D_q, g_q^{base}, and g_q^{PI} be the stabilized attribution denominator, transported base direction, and normalized replay direction defined in Appendix C.5. At any common locally unclipped evaluation point,*

$$g_q^{\mathrm{PI}}(\theta; c_t) = c_t \frac{G}{D_q} g_q^{\mathrm{base}}(\theta).$$

For $g_q^{\mathrm{base}}(\theta) \neq 0$, the corresponding inner product has the sign of c_t; if $0 \leq \rho_{q,i} \leq \bar{\rho}$ and $\|s_{q,i}\| \leq M$, then $\|g_q^{\mathrm{PI}}\| \leq \bar{\rho} M |c_t|$. Appendix C.5 further extends this signed relation across the consecutive base and replay evaluation points under a local Lipschitz transport condition.

Proposition 6.3 shows that PIPO supplies structured groupwise control: the feedback sign selects reinforcement or correction, attribution normalization provides composition-uniform scale control, and the cross-point bound carries this relation across consecutive policies.

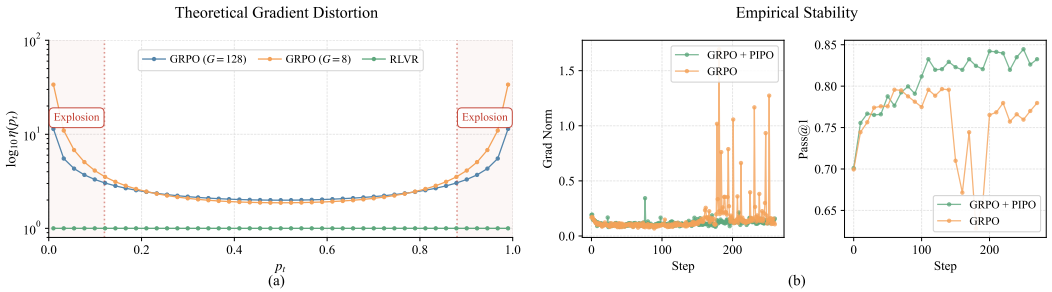


Figure 3: Conditional group-relative sensitivity and observed training dynamics. (a) Scaling in the stylized non-degenerate model. (b) PIPO exhibits a smaller gradient spike while maintaining Pass@1.

7 Related Work

RL post-training spans verifiable rewards for reasoning, code, and tool use (Yang et al., 2025a; Guo et al., 2025a; Zou et al., 2025; Chen et al., 2025b; Xie et al., 2026), critic-free group-relative optimization (Shao et al., 2024; Ahmadian et al., 2024), and on-policy or self-distillation feedback (Song & Zheng, 2026; Hübotter et al., 2026; Zhao et al., 2026a). Stability methods mainly modify credit assignment, clipping, sampling, or exploration (Liu et al., 2025; Yang et al., 2026; Hao et al., 2025; Cheng et al., 2026; Chen et al., 2026b), while related work uses moving averages or historical statistics to correct local estimation (Xu & Ding, 2026; Wang et al., 2020; 2021). PIPO instead preserves the base attribution and applies observed inter-iteration performance as retrospective feedback. Appendix D provides further comparison.

8 Conclusion

We introduce PIRL, a trajectory-level formulation that makes inter-iteration policy progress an explicit optimization signal, and PIPO, its practical closed-loop realization. PIPO measures recent empirical progress, converts it into signed feedback, and replays stored local attribution while preserving the credit-assignment structure of the base optimizer.

Across two model scales, five optimizers, and four task domains, PIPO consistently improves aggregate performance. Multi-seed dynamics and the fixed-positive replay control further highlight the value of adaptive signed modulation. Our analysis connects cumulative improvement to terminal performance and characterizes local alignment, finite-sample feedback reliability, and signed groupwise replay. Together, these results show that inter-iteration feedback complements both sparse verifier rewards and dense self-teacher signals.

AI USE STATEMENT

In this work, generative AI tools were used to assist with language editing and with checking and refining mathematical proofs. All mathematical claims were independently verified by the authors, and all AI-assisted textual revisions were reviewed before inclusion in the manuscript. The authors take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We provide our implementation in the supplementary materials and at the following anonymous code repository: `https://anonymous.4open.science/r/PIPO-44F3/README.md`. Appendix A reports the models, datasets, baselines, training and evaluation configurations, and implementation settings; Algorithm 1 gives the full PIPO training procedure.

REFERENCES

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting REINFORCE-style optimization for learning from human feedback in LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 12248–12267. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.662.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021. URL https://arxiv.org/abs/2108.07732.

Yikun Ban, Fengkai Yang, Fangzheng Chen, Yibo Wang, Zhijun Chen, Zhongyi Li, Gongxun Li, Huaiyang Wang, Xiaodong Lu, Zixuan Huang, et al. Agent exploration toward artificial general intelligence: A survey. *SSRN Electronic Journal*, 2026. doi: 10.2139/ssrn.6748619. URL https://ssrn.com/abstract=6748619.

Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, et al. MiniMax-M1: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*, 2025a.

Liang Chen, Xueting Han, Qizhou Wang, Bo Han, Jing Bai, Hinrich Schuetze, and Kam-Fai Wong. EEPO: Exploration-enhanced policy optimization via sample-then-forget. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=ObF4WIMkY6.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

Peter Chen, Xiaopeng Li, Ziniu Li, Wotao Yin, Xi Chen, and Tianyi Lin. Exploration vs exploitation: Rethinking RLVR through clipping, entropy, and spurious reward. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=sE8DCSJTzd.

Xiaoyin Chen, Jiarui Lu, Minsu Kim, Dinghuai Zhang, Jian Tang, Alexandre Piché, Nicolas Gontier, Yoshua Bengio, and Ehsan Kamalloo. Self-evolving curriculum for LLM reasoning. In *Conference on Language Modeling (COLM)*, October 2026c. URL https://www.microsoft.com/en-us/research/publication/self-evolving-curriculum-for-llm-reasoning/.

Zehao Chen, Gongxun Li, Tianxiang Ai, Zixuan Huang, Xiaodong Liu, Yifei Li, Wang Zhou, Fuzhen Zhuang, Xianglong Liu, Jianxin Li, Deqing Wang, and Yikun Ban. Weak-driven learning: How weak agents make strong agents stronger. *arXiv preprint arXiv:2602.08222*, 2026d. URL https://arxiv.org/abs/2602.08222.

Zhijun Chen, Zeyu Ji, Qianren Mao, Hao Wu, Jinhuan Song, Junhang Cheng, Bangjie Qin, Zhuoran Li, Jingzheng Li, Kai Sun, Zizhe Wang, Yikun Ban, Zhu Sun, Xiangyang Ji, Hailong Sun, and Xiao Huang. Scoring, reasoning, and selecting the best! ensembling large language models via a peer-review process. *arXiv preprint arXiv:2512.23213*, 2025b. URL https://arxiv.org/abs/2512.23213.

Zhipeng Chen, Xiaobo Qin, Youbin Wu, Yue Ling, Qinghao Ye, Wayne Xin Zhao, and Guang Shi. Pass@k training for adaptively balancing exploration and exploitation of large reasoning models. *arXiv preprint arXiv:2508.10751*, 2025c. URL https://arxiv.org/abs/2508.10751.

Daixuan Cheng, Shaohan Huang, Xuekai Zhu, Bo Dai, Xin Zhao, Zhenliang Zhang, and Furu Wei. Reasoning with exploration: An entropy perspective. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(36):30377–30385, 2026. doi: 10.1609/aaai.v40i36.40290. URL https://ojs.aaai.org/index.php/AAAI/article/view/40290.

Runpeng Dai, Linfeng Song, Haolin Liu, Zhenwen Liang, Dian Yu, Haitao Mi, Zhaopeng Tu, Rui Liu, Tong Zheng, Hongtu Zhu, and Dong Yu. CDE: Curiosity-driven exploration for efficient reinforcement learning in large language models. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=5rXN5knHKW.

Yanqi Dai, Yuxiang Ji, Xiao Zhang, Yong Wang, Xiangxiang Chu, and Zhiwu Lu. Harder is better: Boosting mathematical reasoning via difficulty-aware GRPO and multi-aspect question reformulation. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=nfURupkdRJ.

Jia Deng, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, and Ji-Rong Wen. Decomposing the entropy-performance exchange: The missing keys to unlocking effective reinforcement learning. *arXiv preprint arXiv:2508.02260*, 2025.

Kehua Feng, Xinyi Shen, Weijie Wang, Xiang Zhuang, Yuqi Tang, Qiang Zhang, and Keyan Ding. SciKnowEval: Evaluating multi-level scientific knowledge of large language models. *arXiv preprint arXiv:2406.09098*, 2024. URL https://arxiv.org/abs/2406.09098.

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, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633638, September 2025a. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Yiran Guo, Lijie Xu, Jie Liu, Dan Ye, and Shuang Qiu. Segment Policy Optimization: Effective segment-level credit assignment in RL for large language models. In *Proceedings of the 39th Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025b. URL https://openreview.net/forum?id=9osvTOYbT4.

Anthony GX-Chen, Jatin Prakash, Jeff Guo, Rob Fergus, and Rajesh Ranganath. Kl-regularized reinforcement learning for generative modelling is designed to mode collapse. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 50665–50693, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/5306f6ce1082d29e4968acdacab84fd7-Paper-Conference.pdf.

Yaru Hao, Li Dong, Xun Wu, Shaohan Huang, Zewen Chi, and Furu Wei. On-policy rl with optimal reward baseline. *arXiv preprint arXiv:2505.23585*, 2025.

Bingxiang He, Yuxin Zuo, Zeyuan Liu, Shangziqi Zhao, Zixuan Fu, Junlin Yang, Cheng Qian, Kaiyan Zhang, Yuchen Fan, Ganqu Cui, Xiusi Chen, Youbang Sun, Xingtai Lv, Xuekai Zhu, Li Sheng, Ran Li, Huan-ang Gao, Yuchen Zhang, Lifan Yuan, Bowen Zhou, Zhiyuan Liu, and Ning Ding. How far can unsupervised rlvr scale llm training? In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 14823–14865, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/18929f3253e4d4e28086a89ab88d1276-Paper-Conference.pdf.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, 2024.

Haoran He, Yuxiao Ye, Qingpeng Cai, Chen Hu, Binxing Jiao, Daxin Jiang, and Ling Pan. Random policy valuation is enough for LLM reasoning with verifiable rewards. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=ujLgLz6QQa.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021.

Jian Hu, Mingjie Liu, Ximing Lu, Fang Wu, Zaid Harchaoui, Shizhe Diao, Yejin Choi, Pavlo Molchanov, Jun Yang, Jan Kautz, and Yi Dong. BroRL: Scaling reinforcement learning via broadened exploration. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=KmS7pdFBEh.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=QkfkxyRizZ.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, et al. Livecodebench: Holistic and contamination-free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=chfJJYC3iL.

Aitor Lewkowycz, Anders Johan Andreassen, David Dohan, Ethan Dyer, et al. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 3843–3857, 2022. URL https://openreview.net/forum?id=IFXTZERXdM7.

Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Pall Song, et al. API-bank: A comprehensive benchmark for tool-augmented LLMs. In *The 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023a. URL https://openreview.net/forum?id=o2HBfgY20b.

Rongao Li, Jie Fu, Bo-Wen Zhang, Tao Huang, et al. TACO: Topics in algorithmic COde generation dataset. *arXiv preprint arXiv:2312.14852*, 2023b. URL https://arxiv.org/abs/2312.14852.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. In *ICML 2026 Workshop on Foundations of Deep Generative Models: Understanding Memorization, Generalization, and Reasoning*, 2026a. URL https://openreview.net/forum?id=aaDDXXDGXB.

Yingru Li, Jiawei Xu, Ziniu Li, Jiacai Liu, Wei Liu, Yuxuan Tong, Longtao Zheng, Zhenghai Xue, Yaxiang Zhang, Tianle Cai, Ge Zhang, Qian Liu, and Baoxiang Wang. The optimal token baseline: Variance reduction for long-horizon LLM-RL. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=hpxI4Mndt4.

Zhongyi Li, Wan Tian, Jingyu Chen, Kangyao Huang, Huiming Zhang, Hui Yang, Tao Ren, Jinyang Jiang, Yijie Peng, Yikun Ban, and Fuzhen Zhuang. Adaptive Robust Estimator for Multi-Agent Reinforcement Learning. *arXiv preprint arXiv:2603.21574*, 2026c.

Zhongyi Li, Wan Tian, Jinju Chen, Huiming Zhang, Yang Liu, Yikun Ban, and Fuzhen Zhuang. Counterfactual credit policy optimization for multi-agent collaboration. *arXiv preprint arXiv:2603.21563*, 2026d. URL https://arxiv.org/abs/2603.21563.

Dengcan Liu, Fengkai Yang, Xiaohan Wang, Shurui Yan, Jiajun Chai, Jiahao Li, Yikun Ban, Zhendong Mao, Wei Lin, and Guojun Yin. CDRRM: Contrast-Driven Rubric Generation for Reliable and Interpretable Reward Modeling. *arXiv preprint arXiv:2603.08035*, 2026.

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. In *2nd AI for Math Workshop @ ICML 2025*, 2025. URL https://openreview.net/forum?id=jLpC1zavzn.

Xiaodong Lu, Xiaohan Wang, Jiajun Chai, Guojun Yin, Wei Lin, Zhijun Chen, Yu Luo, Fuzhen Zhuang, Yikun Ban, and Deqing Wang. Contextual rollout bandits for reinforcement learning with verifiable rewards. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=weMYE1B16x.

Shishir G Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48371–48392. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/patil25a.html.

Daniil Plyusov, Alexey Gorbatovski, Boris Shaposhnikov, Viacheslav Sinii, Alexey Malakhov, and Daniil Gavrilov. F-GRPO: Don't let your policy learn the obvious and forget the rare. In *The 1st Workshop on Scaling Post-training for LLMs*, 2026. URL https://openreview.net/forum?id=DWS8VtJt6a.

Penghui Qi, Xiangxin Zhou, Zichen Liu, Tianyu Pang, Chao Du, Min Lin, and Wee Sun Lee. Rethinking the trust region in LLM reinforcement learning. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=f4g8zblbwp.

Cheng Qian, Emre Can Acikgoz, Qi He, Hongru WANG, Xiusi Chen, Dilek Hakkani-Tur, Gokhan Tur, and Heng Ji. Toolrl: Reward is all tool learning needs. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 105523–105553. Curran Associates, Inc., 2025. doi: 10.52202/085713-3524. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/97c5b2707228e7e3fb67e4ecc2e0e607-Paper-Conference.pdf.

Deng Qiyuan, Kehai Chen, Min Zhang, and Zhongwen Xu. HiPO: Self-hint policy optimization for RLVR. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=rcb20pHmT1.

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

Amrith Setlur, Chirag Nagpal, Adam Fisch, Xinyang Geng, Jacob Eisenstein, Rishabh Agarwal, Alekh Agarwal, Jonathan Berant, and Aviral Kumar. Rewarding progress: Scaling automated process verifiers for LLM reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=A6Y7AqlzLW.

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

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A flexible and efficient RLHF framework. In *Proceedings of the Twentieth European Conference on Computer Systems (EuroSys)*, pp. 1279–1297, Rotterdam, The Netherlands, 2025. ACM. doi: 10.1145/3689031.3696075.

Mingyang Song and Mao Zheng. A survey of on-policy distillation for large language models. *arXiv preprint arXiv:2604.00626*, 2026.

Wei Sun, Wen Yang, Pu Jian, Qianlong Du, Fuwei Cui, Shuo Ren, and Jiajun Zhang. KTAE: A model-free algorithm to key-tokens advantage estimation in mathematical reasoning. In *Proceedings of the 39th Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025. URL https://openreview.net/forum?id=yqQVRNdmKJ.

Shobhita Sundaram, John Quan, Ariel Kwiatkowski, Kartik Ahuja, Yann Ollivier, and Julia Kempe. Teaching models to teach themselves: Reasoning at the edge of learnability. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=GnqHK8Ww98.

Fahim Tajwar, Guanning Zeng, Yueer Zhou, Yuda Song, Daman Arora, Yiding Jiang, Jeff Schneider, Ruslan Salakhutdinov, Haiwen Feng, and Andrea Zanette. Maximum likelihood reinforcement learning. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=EeuLO2BjFN.

Jinpeng Wang, Jieming Zhu, and Xiuqiang He. Cross-batch negative sampling for training two-tower recommenders. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1632–1636, 2021.

Xun Wang, Haozhi Zhang, Weilin Huang, and Matthew R Scott. Cross-batch memory for embedding learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6388–6397, 2020.

Fang Wu, Weihao Xuan, Ximing Lu, Mingjie Liu, Yi Dong, Zaid Harchaoui, and Yejin Choi. The Invisible Leash: Why RLVR may or may not escape its origin. *arXiv preprint arXiv:2507.14843*, 2025a.

Jiajun Wu, Jian Yang, Wei Zhang, Lin Jing, Yuqing Ma, Ensheng Shi, Yuchi Ma, Zhoujun Li, and Xianglong Liu. UCoder: Unsupervised Code Generation by Internal Probing of Large Language Models. *arXiv preprint arXiv:2512.17385*, 2025b.

Hongyan Xie, Yikun Ban, Ruiyu Fang, Zixuan Huang, Deqing Wang, Jianxin Li, Yitong Yao, Chao Wang, and Shuangyong Song. UniARM: Towards a Unified Autoregressive Reward Model for Multi-Objective Test-Time Alignment. *arXiv preprint arXiv:2602.09538*, 2026.

Zhongwen Xu and Zihan Ding. Single-stream Policy Optimization. In *Proceedings of the 14th International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=b61UW62K7W.

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

Fengkai Yang, Zherui Chen, Xiaohan Wang, Xiaodong Lu, Jiajun Chai, Guojun Yin, Wei Lin, Shuai Ma, Fuzhen Zhuang, Deqing Wang, et al. Your group-relative advantage is biased. *arXiv preprint arXiv:2601.08521*, 2026.

Shihui Yang, Chengfeng Dou, Peidong Guo, Kai Lu, Qiang Ju, Fei Deng, and Rihui Xin. DCPO: Dynamic clipping policy optimization. *arXiv preprint arXiv:2509.02333*, 2025b.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru 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, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *Proceedings of the 39th Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025. URL https://openreview.net/forum?id=2a36EMSSTp.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Proceedings of the 39th Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025. URL https://openreview.net/forum?id=4OsgYD7em5.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=Jpxfof0EaS.

Yuzhong Zhao, Yue Liu, Junpeng Liu, Jingye Chen, Xun Wu, Yaru Hao, Tengchao Lv, Shaohan Huang, Lei Cui, Qixiang Ye, Fang Wan, and Furu Wei. Geometric-mean policy optimization. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=nCEs0tSwc2.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group Sequence Policy Optimization. *arXiv preprint arXiv:2507.18071*, 2025.

Jiaru Zou, Yikun Ban, Zihao Li, Yunzhe Qi, Ruizhong Qiu, Ling Yang, and Jingrui He. Transformer copilot: Learning from the mistake log in LLM fine-tuning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=MRvxlTlkNQ.

A Algorithm and Experimental Details

A.1 Full PIPO Optimization Procedure

Algorithm 1 gives the complete execution order of retrospective feedback and the subsequent base-policy update.

Algorithm 1 PIPO Training Procedure

Input: Base objective $\mathcal{J}_{\text{base}}$, attribution rule $a(\cdot)$, window size K, learning rates $\alpha_{\text{PI}}, \alpha_{\text{std}}$
Initialize memory $\mathcal{M} \leftarrow \emptyset$ and policy θ_1
for $t = 1, 2, \ldots, T$ **do**
  Sample $\mathcal{B}_t \sim \pi_{\theta_t}$ and compute μ_t using Eq. (9)
  Compute local attribution scores $a_t = a(\mathcal{B}_t)$ for the base RL algorithm
  Initialize the feedback-adjusted policy $\tilde{\theta}_t \leftarrow \theta_t$
  # Phase 2: Retrospective feedback on stored attribution
  if $|\mathcal{M}| \geq K$ **then**
    Compute μ_{his} and σ_{his} from the latest K records in $\mathcal{M}$
    Compute ξ_t using Eq. (11)
    Retrieve $(\mathcal{B}_{t-1}, a_{t-1})$ from $\mathcal{M}$
    Construct $\hat{r}^{\text{PI}}_{t,i}$ for $\mathcal{B}_{t-1}$ using Eq. (12)
    $\tilde{\theta}_t \leftarrow \theta_t + \alpha_{\text{PI}} \nabla_\theta \mathcal{J}_{\text{PI}}(\theta_t; \mathcal{B}_{t-1}, \hat{r}^{\text{PI}}_t)$
  end if
  # Phase 1: Exploration with the base RL algorithm
  $\theta_{t+1} \leftarrow \tilde{\theta}_t + \alpha_{\text{std}} \nabla_\theta \mathcal{J}_{\text{base}}(\tilde{\theta}_t; \mathcal{B}_t, a_t)$
  Update $\mathcal{M}$ with $(\mu_t, \mathcal{B}_t, a_t)$
end for
Output: Optimized policy θ_{T+1}

A.2 Models and Datasets

Models For the main RL post-training experiments, we evaluate all methods on Qwen3-4B-Base and Qwen3-8B-Base (Yang et al., 2025a), covering two representative model scales. We use base (non-instruction-tuned) models to assess how RL methods elicit reasoning behaviors from the same pretrained initialization. Following the SDPO setting (Hübotter et al., 2026), the self-distillation study uses Qwen3-8B with instruction-following capability to construct the privileged self-teacher.

Datasets Mathematical reasoning experiments are trained on the MATH dataset (Hendrycks et al., 2021), which contains approximately 7.5k multi-step mathematical reasoning problems. Evaluation is performed on MATH500, a representative subset of the MATH test set; AIME 2025 and AMC 2023, which measure competition-level mathematical reasoning; Minerva (Lewkowycz et al., 2022), which emphasizes scientific and quantitative reasoning; and OlympiadBench (He et al., 2024), which contains Olympiad-level mathematical and scientific problems. All methods use the same answer extraction and verification pipeline, yielding directly comparable scores within each benchmark setting. For code RL, models are trained on TACO (Li et al., 2023b) and evaluated on TACO, LiveCodeBench v6 (LCBv6) (Jain et al., 2025), HumanEval (Chen et al., 2021), and MBPP (Austin et al., 2021) using executable tests or task-specific correctness checks. For tool-use RL, models are trained on ToolRL/RLLA (Qian et al., 2025) and evaluated on RLLA (Qian et al., 2025), BFCL (Patil et al., 2025), and APIBank (Li et al., 2023a), where policies must invoke tools or external functions and receive verifiable task-level feedback. In addition, we use SciKnowEval (Feng et al., 2024) for the focused self-distillation study in Section 5.3. This benchmark evaluates large language models across biology, chemistry, physics, and materials science.

A.3 Baselines

We instantiate PIPO on top of PPO (Schulman et al., 2017), GRPO (Shao et al., 2024), GSPO (Zheng et al., 2025), DAPO (Yu et al., 2025), and SDPO (Hübotter et al., 2026). Let x_t denote a prompt at

training step t, and let $o_{t,i} = (y_{t,i,1}, \ldots, y_{t,i,|o_{t,i}|})$ be the i-th sampled response with verifier reward $R_{t,i}$. For token-level objectives, we define the importance ratio

$$\rho_{t,i,\tau}(\theta) = \frac{\pi_\theta(y_{t,i,\tau} \mid x_t, o_{t,i,<\tau})}{\pi_{\theta_{\text{old}}}(y_{t,i,\tau} \mid x_t, o_{t,i,<\tau})}.$$

For PPO, a common sequence-level shaped reward is

$$\widetilde{R}(x_t, o_t) = R(x_t, o_t) - \beta \log \frac{\pi_\theta(o_t \mid x_t)}{\pi_{\text{ref}}(o_t \mid x_t)},$$

where β controls reference-policy regularization. A learned value function provides the GAE attribution

$$A^{\text{PPO}}_{t,\tau} = \sum_{\ell \geq 0} (\gamma \lambda_{\text{GAE}})^\ell \delta_{t,\tau+\ell}, \qquad \delta_{t,\tau} = r_{t,\tau} + \gamma V_\psi(s_{t,\tau+1}) - V_\psi(s_{t,\tau}).$$

Since PPO uses a single rollout per prompt, we omit the response index. Its clipped surrogate is

$$\mathcal{J}_{\text{PPO}}(\theta) = \mathbb{E}\left[\frac{1}{|o_t|}\sum_{\tau=1}^{|o_t|} \min\big(\rho_{t,\tau}(\theta)A^{\text{PPO}}_{t,\tau}, \text{clip}(\rho_{t,\tau}(\theta), 1-\epsilon, 1+\epsilon)A^{\text{PPO}}_{t,\tau}\big)\right].$$

GRPO removes the value network by assigning each response a group-normalized advantage. For the current query $q = x_t$, let

$$\mu_{t,q} = \frac{1}{G}\sum_{j=1}^{G} R_{t,j}, \qquad \sigma^2_{t,q} = \frac{1}{G}\sum_{j=1}^{G}(R_{t,j} - \mu_{t,q})^2, \qquad A^{\text{GRPO}}_{t,i} = \frac{R_{t,i} - \mu_{t,q}}{\sigma_{t,q}},$$

and broadcasts it to all tokens in the response:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_{t,i}|}\sum_{\tau=1}^{|o_{t,i}|} \min\big(\rho_{t,i,\tau}(\theta)A^{\text{GRPO}}_{t,i}, \text{clip}(\rho_{t,i,\tau}(\theta), 1-\epsilon, 1+\epsilon)A^{\text{GRPO}}_{t,i}\big)\right].$$

GSPO keeps the group-relative advantage but replaces the token-wise ratio with a sequence-level ratio

$$\rho^{\text{seq}}_{t,i}(\theta) = \exp\left(\frac{1}{|o_{t,i}|}\sum_{\tau=1}^{|o_{t,i}|} \log \rho_{t,i,\tau}(\theta)\right),$$

yielding

$$\mathcal{J}_{\text{GSPO}}(\theta) = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G} \min\big(\rho^{\text{seq}}_{t,i}(\theta)A_{t,i}, \text{clip}(\rho^{\text{seq}}_{t,i}(\theta), 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}})A_{t,i}\big)\right].$$

DAPO further uses asymmetric clipping and filters degenerate groups. Denoting the retained prompt groups by $\mathcal{F}_t$, its objective is

$$\mathcal{J}_{\text{DAPO}}(\theta) = \mathbb{E}_{x_t \in \mathcal{F}_t}\Bigg[\frac{1}{\sum_i |o_{t,i}|}\sum_{i=1}^{G}\sum_{\tau=1}^{|o_{t,i}|} \min(\rho_{t,i,\tau}(\theta)A_{t,i}, \text{clip}(\rho_{t,i,\tau}(\theta), 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}})A_{t,i})\Bigg].$$

For self-distillation, SDPO uses privileged information f available after the rollout, such as execution feedback, a verified solution, or the ground-truth option, to construct a feedback-conditioned self-teacher distribution $q_\theta(\cdot \mid x_t, f, o_{t,<\tau})$. It then minimizes a token-level reverse KL objective:

$$\mathcal{L}_{\text{SDPO}}(\theta) = \mathbb{E}\left[\sum_\tau D_{\text{KL}}\left(\pi_\theta(\cdot \mid x_t, o_{t,<\tau}) \,\|\, \text{stopgrad}(q_\theta(\cdot \mid x_t, f, o_{t,<\tau}))\right)\right],$$

which provides dense token-level attribution through the self-teacher/student log-probability difference $A^{\text{SDPO}}_{t,i,\tau} = \log q_\theta(y_{t,i,\tau} \mid x_t, f, o_{t,i,<\tau}) - \log \pi_\theta(y_{t,i,\tau} \mid x_t, o_{t,i,<\tau})$.

PIPO preserves each baseline objective and appends a retrospective policy-improvement update. Let $\mathcal{J}_{\text{base}}$ denote the maximization form of the corresponding baseline objective above, and let $a_{t,i,\tau}$ denote its local attribution score: $A^{\text{PPO}}_{t,\tau}$ for PPO, group-relative advantages for GRPO, GSPO, and DAPO, and the SDPO token attribution for self-distillation. Given the improvement signal ξ_{t+1}, PIPO forms

$$\hat{r}^{\text{PI}}_{t+1,i,\tau} = a_{t,i,\tau}\, \phi_\lambda(\xi_{t+1}),$$

and performs the additional clipped update on the previous batch:

$$\theta^{\text{base}}_{t+1} = \theta_t + \alpha_{\text{std}} \nabla_\theta \mathcal{J}_{\text{base}}(\theta_t; \mathcal{B}_t),$$
$$\theta^{\text{PIPO}}_{t+1} = \theta^{\text{base}}_{t+1} + \alpha_{\text{PI}} \nabla_\theta \mathcal{J}_{\text{PI}}(\theta^{\text{base}}_{t+1}; \mathcal{B}_t),$$

where

$$\mathcal{J}_{\text{PI}}(\theta; \mathcal{B}_t) = \frac{1}{Z_t} \sum_{i,\tau} \min\left(\rho_{i,\tau}(\theta)\hat{r}^{\text{PI}}_{t+1,i,\tau}, \text{clip}(\rho_{i,\tau}(\theta), 1-\epsilon, 1+\epsilon)\hat{r}^{\text{PI}}_{t+1,i,\tau}\right),$$

Z_t is the number of historical learning units, and $\rho_{i,\tau}(\theta)$ is the importance ratio between the retrospective policy and the policy that generated $\mathcal{B}_t$. This construction makes PIPO a direct plug-in: it leaves the baseline objective intact and modulates stored attribution according to the empirical performance trend.

A.4 Training Hyperparameters

Table 5: Hyperparameter settings for PPO, group-relative methods (GRPO, GSPO, DAPO), and their PIPO-enhanced versions

	Hyperparameter	PPO	PPO+PIPO	GRPO	GRPO+PIPO	GSPO	GSPO+PIPO	DAPO	DAPO+PIPO
Compute	Nodes	1	1	1	1	1	1	1	1
	GPUs per Node	8	8	8	8	8	8	8	8
General	Use KL in Reward	True	True	False	False	False	False	False	False
	Use KL Loss	False	False	True	True	True	True	True	True
	KL Coef	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW
Training	Epochs	24	24	3	3	3	3	9	9
	Batch Size	1024	1024	128	128	128	128	128	128
	Mini Batch Size	512	512	64	64	64	64	64	64
	Micro Batch Size	4	4	4	4	4	4	4	4
	Learning Rate	1×10^{-6}	1×10^{-6}	1×10^{-6}	1×10^{-6}	1×10^{-6}	1×10^{-6}	1×10^{-6}	1×10^{-6}
Clipping	Clip High (ϵ_{high})	0.2	0.2	0.2	0.2	0.0004	0.0004	0.28	0.28
	Clip Low (ϵ_{low})	0.2	0.2	0.2	0.2	0.0003	0.0003	0.2	0.2
Rollout	Rollout (G)	1	1	8	8	8	8	8	8
	Temperature	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	Max Prompt Len	512	512	512	512	512	512	512	512
	Max Response Len	4096	4096	4096	4096	4096	4096	4096	4096
	Filter Groups	No	No	No	No	No	No	Yes	Yes
PIPO Specific	Window Size (K)	-	8	-	8	-	8	-	8
	PI LR (α_{PI})	-	1×10^{-6}	-	1×10^{-6}	-	1×10^{-6}	-	1×10^{-6}
	Rectification Coefficient($\phi_\lambda(\cdot)$)	-	0.1	-	0.1	-	0.1	-	0.1

We conduct all RL training within the **VeRL** framework (Sheng et al., 2025) on a single compute node equipped with **8 × NVIDIA H20 (141GB) GPUs**. Each baseline and its PIPO-enhanced counterpart share the same core hyperparameters, including the learning rate and KL coefficient. For PPO, we use batch size 1024 and 24 epochs to match the response-level budget of group-relative methods with batch size 128 and $G = 8$. Method-specific settings, such as clipping strategies, follow the original implementations. PIPO introduces additional parameters for retrospective feedback, including the historical window size K and the policy improvement learning rate. We set $\varepsilon_\sigma = 10^{-8}$ for PPO and group-relative experiments and 10^{-6} for SDPO, and use $\varepsilon_D = 10^{-6}$ for group-attribution normalization. Detailed configurations for all methods are reported in Table 5.

B Additional Experiments

B.1 Robust Multi-Sample Reasoning

We further evaluate multi-sample reasoning with Pass@8 accuracy on Qwen3-4B-Base. Table 6 compares PPO and group-relative baselines under this evaluation setting.

Table 6: Pass@8 Results for Qwen3-4B-Base.

Method	MATH500	AIME25	AMC23	Minerva	Olympiad	AVG
PPO	87.2	20.4	79.0	32.9	54.0	54.7
+PIPO	**88.0**	**28.1**	72.6	**33.0**	53.9	**55.1**
GRPO	88.3	27.6	80.2	34.7	54.1	57.0
+PIPO	**89.4**	**30.3**	**84.0**	**35.9**	**57.1**	**59.3**
GSPO	87.2	29.1	81.3	34.8	57.2	57.9
+PIPO	**88.9**	**31.2**	**82.7**	**35.9**	**57.6**	**59.3**
DAPO	88.2	30.7	84.9	35.9	59.4	59.8
+PIPO	**89.2**	**31.4**	**86.2**	35.8	**60.0**	**60.5**

PIPO improves the average Pass@8 score for every evaluated base algorithm, with the largest gains on the group-relative baselines. The improvement under multi-sample decoding extends from single-solution accuracy to the quality of the candidate solution set.

B.2 Ablation on Window Size K

Table 7: Ablation on Window Size K on Qwen3-4B-Base.

K	MATH500	AIME25	AMC23	Minerva	Olympiad	AVG
GRPO Baseline	79.3	18.5	60.0	21.0	40.5	43.9
2	78.5	18.5	**67.5**	22.8	41.5	45.8
4	77.3	**22.2**	62.5	**26.8**	40.9	45.9
8	**80.9**	18.5	60.0	**26.8**	**44.4**	**46.1**
16	80.1	14.8	65.0	23.2	41.2	44.9
32	77.1	11.1	65.0	25.4	39.4	43.6

The sliding-window size K controls the trade-off between smoothing and staleness in the historical anchor μ_{his}. As shown in Table 7, moderate windows ($K \in \{2, 4, 8, 16\}$) improve over the GRPO baseline on average, with $K = 8$ performing best. A short window is more sensitive to variation in prompt difficulty and rollout noise, whereas an excessively long window can lag behind a rapidly changing policy. The baseline-level result at $K = 32$ further supports using a responsive, moderately sized anchor.

B.3 Fixed-Positive Replay Control

To isolate adaptive feedback from additional replay computation, we replace $\phi_\lambda(\xi_t)$ with $c = 1$ while keeping the stored batch, warm-up, replay frequency, and extra update unchanged. The unit coefficient avoids introducing a separately tuned replay-strength hyperparameter.

Table 8: Fixed-positive replay control on mathematical reasoning (AVG, %).

Method	Qwen3-4B-Base	Qwen3-8B-Base
GRPO	43.9	48.0
GRPO + fixed replay ($c = 1$)	44.3	44.7
GRPO + PIPO	**46.1**	**48.9**

Fixed-positive replay underperforms PIPO at both model scales and falls below GRPO at 8B. Thus, replaying every update with a positive coefficient is not reliably beneficial: it can reinforce weak or harmful directions, whereas PIPO conditions the replay sign and magnitude on observed policy progress.

B.4 Robustness Across Random Seeds

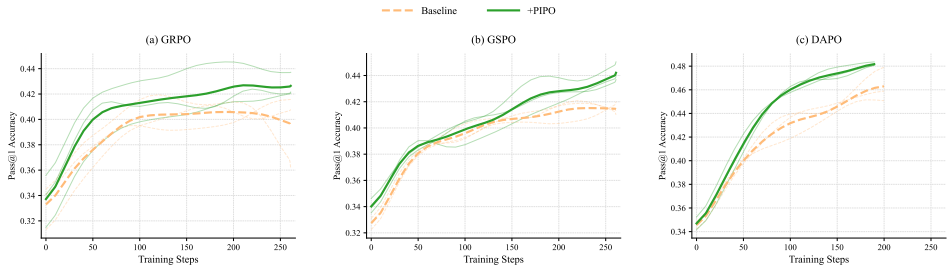


Figure 4: Training dynamics across multiple random seeds on Qwen3-4B-Base.

We further evaluate the closed-loop mechanism across multiple data seeds, covering variation in sampling order and initialization.

We evaluate GRPO, DAPO, and GSPO alongside their PIPO-augmented variants on the Qwen3-4B-Base model using three different data seeds (6, 21, and 42). As shown in Figure 4, the PIPO-enhanced algorithms consistently converge to a higher performance plateau across seeds, indicating reduced sensitivity to seed-specific sampling noise.

B.5 Wall-Clock Efficiency Analysis

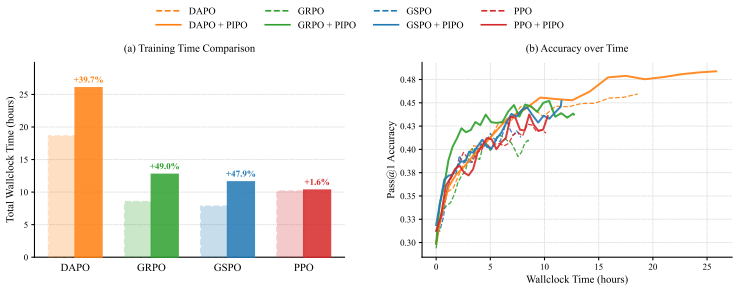


Figure 5: Computational efficiency analysis on Qwen3-4B-Base. (a) Total wall-clock time comparison. (b) Evolution of average Pass@1 accuracy with respect to wall-clock time.

We analyze computational efficiency in Figure 5. PIPO adds a retrospective forward–backward optimization pass, whose relative cost depends on the base algorithm. The cost is small for PPO, where it is largely amortized by critic training and the heavier PPO update, and larger for critic-free methods. The accuracy-over-time curves show that PIPO attains higher accuracy than the corresponding baseline at comparable elapsed times.

C Theoretical Proofs

C.1 Smoothed Historical Anchor

For notational convenience, denote the expected RL performance at iteration t by $J_t := J_{\mathrm{RL}}(\theta_t)$. The sliding-window anchor used in PIPO can be viewed as a smoothed version of the one-step improvement signal. Define the idealized K-step smoothed improvement as

$$\Delta J_t^{(K)} := J_t - \frac{1}{K}\sum_{k=1}^{K} J_{t-k}.$$

Assuming the policies before optimization are fixed to the initial model, their performance contributes only a constant offset. Summing the smoothed improvements gives

$$\begin{aligned}\sum_{t=1}^{T}\Delta J_t^{(K)} &= \sum_{t=1}^{T} J_t - \frac{1}{K}\sum_{t=1}^{T}\sum_{k=1}^{K} J_{t-k} \\ &= \sum_{m=T-K+1}^{T}\left(1-\frac{T-m}{K}\right)J_m + C_{\text{init}} \\ &= \sum_{i=0}^{K-1}\frac{K-i}{K}J_{T-i} + C_{\text{init}},\end{aligned}$$

where C_{init} absorbs terms determined by the fixed initialization. Thus, maximizing cumulative smoothed improvement is equivalent to maximizing a positively weighted average of the final K raw-policy performances. The sliding-window anchor therefore targets recent raw-policy progress, while Proposition 6.1 isolates the intervening base step; Corollary C.2 connects the two under local trend coherence.

C.2 Local Retrospective Alignment

Algorithm 1 first applies retrospective feedback to obtain $\widetilde{\theta}_{t-1}$ and then takes the base step to θ_t. We therefore isolate that base transition:

$$\Delta J_t^{\text{base}} := J_{\text{RL}}(\theta_t) - J_{\text{RL}}(\widetilde{\theta}_{t-1}), \qquad d_t := \theta_t - \widetilde{\theta}_{t-1}.$$

We analyze an attribution-matched regime in which the retrospective objective reuses the first-order direction of this base update. Specifically, assume

$$d_t = \alpha_{\text{base}} g_{t-1}^{\text{base}}, \qquad \Delta\theta_{\text{PI}}(c_t) = \alpha_{\text{PI}} c_t \left(g_{t-1}^{\text{base}} + e_t\right),$$

where $\alpha_{\text{base}}, \alpha_{\text{PI}} > 0$ and

$$\|e_t\| \le C_{\text{rep}}\|d_t\|.$$

The term e_t accounts for evaluating the stored learning units at θ_t rather than at the base-step starting point $\widetilde{\theta}_{t-1}$. We assume that J_{RL} is L-smooth in this neighborhood and that $\|\nabla J_{\text{RL}}(\theta)\| \le B_J$ and $\|g_{t-1}^{\text{base}}\| \le B_g$.

Proof of Proposition 6.1. By L-smoothness,

$$\left|\Delta J_t^{\text{base}} - \left\langle \nabla J_{\text{RL}}(\widetilde{\theta}_{t-1}), d_t \right\rangle\right| \le \frac{L}{2}\|d_t\|^2.$$

Since $d_t = \alpha_{\text{base}} g_{t-1}^{\text{base}}$ and $\|\nabla J_{\text{RL}}(\theta_t) - \nabla J_{\text{RL}}(\widetilde{\theta}_{t-1})\| \le L\|d_t\|$, it follows that

$$\left\langle \nabla J_{\text{RL}}(\theta_t), g_{t-1}^{\text{base}} \right\rangle = \frac{\Delta J_t^{\text{base}}}{\alpha_{\text{base}}} + r_t, \qquad |r_t| \le \frac{3L}{2} B_g \|d_t\|.$$

Therefore,

$$\langle \nabla J_{\text{RL}}(\theta_t), \Delta\theta_{\text{PI}}(c_t)\rangle = \frac{\alpha_{\text{PI}}}{\alpha_{\text{base}}} c_t \Delta J_t^{\text{base}} + R_t,$$

where

$$|R_t| \le \alpha_{\text{PI}}|c_t|\left(\frac{3L}{2}B_g + B_J C_{\text{rep}}\right)\|d_t\|.$$

This is the claimed local expansion. Whenever $c_t \Delta J_t^{\text{base}} \ge 0$, its leading alignment term is nonnegative. For ideal feedback $c_t = \phi_\lambda(\Delta J_t^{\text{base}})$,

$$\phi_\lambda(\Delta J_t^{\text{base}})\Delta J_t^{\text{base}} = \begin{cases}(\Delta J_t^{\text{base}})^2, & \Delta J_t^{\text{base}} \ge 0, \\ \lambda(\Delta J_t^{\text{base}})^2, & \Delta J_t^{\text{base}} < 0,\end{cases}$$

which completes the proof. □

C.3 Finite-Sample Reliability of Sliding-Window Feedback

Let $\mathcal{F}_{s,0}$ denote the training history immediately before sampling batch $\mathcal{B}_s$, so that θ_s is $\mathcal{F}_{s,0}$-measurable, and let $\mathcal{F}_{s,j}$ additionally include the first j groups in that batch. Suppose each batch contains n prompt groups and let $X_{s,j} \in [0,1]$ be the mean verifier reward of group j. Given the preceding history, assume the groups are conditionally independent, queries follow a fixed distribution, and

$$\mathbb{E}[X_{s,j} \mid \mathcal{F}_{s,j-1}] = J_{\mathrm{RL}}(\theta_s), \qquad \mu_s = \frac{1}{n}\sum_{j=1}^{n} X_{s,j}.$$

Define

$$\delta_t^{(K)} := J_{\mathrm{RL}}(\theta_t) - \frac{1}{K}\sum_{k=1}^{K} J_{\mathrm{RL}}(\theta_{t-k}), \qquad \widehat{\delta}_t^{(K)} := \mu_t - \frac{1}{K}\sum_{k=1}^{K} \mu_{t-k}.$$

Lemma C.1 (Sliding-Window Feedback Concentration)**.** *For every $m > 0$,*

$$\mathbb{P}\Big(\Big|\widehat{\delta}_t^{(K)} - \delta_t^{(K)}\Big| \geq m\Big) \leq 2\exp\left(-\frac{2nm^2}{1+1/K}\right).$$

Consequently,

$$\mathbb{P}\Big(\mathrm{sign}(\xi_t) \neq \mathrm{sign}(\delta_t^{(K)}),\ |\delta_t^{(K)}| \geq m\Big) \leq 2\exp\left(-\frac{2nm^2}{1+1/K}\right).$$

Proof. The estimation error is the weighted martingale sum

$$\widehat{\delta}_t^{(K)} - \delta_t^{(K)} = \frac{1}{n}\sum_{j=1}^{n}\left(X_{t,j} - J_{\mathrm{RL}}(\theta_t)\right) - \frac{1}{Kn}\sum_{k=1}^{K}\sum_{j=1}^{n}\left(X_{t-k,j} - J_{\mathrm{RL}}(\theta_{t-k})\right).$$

The sum of the squared increment ranges is

$$n\left(\frac{1}{n}\right)^2 + Kn\left(\frac{1}{Kn}\right)^2 = \frac{1+1/K}{n}.$$

The first inequality follows from the Hoeffding–Azuma inequality. Because $\widetilde{\sigma}_{\mathrm{his}} > 0$, ξ_t and $\widehat{\delta}_t^{(K)}$ have the same sign. On the event $|\delta_t^{(K)}| \geq m$, a sign error implies $|\widehat{\delta}_t^{(K)} - \delta_t^{(K)}| \geq m$, which gives the second inequality. □

The sliding-window quantity $\delta_t^{(K)}$ concerns the raw policies $\theta_t, \theta_{t-1}, \ldots$, whereas Proposition 6.1 concerns the isolated base transition $\widetilde{\theta}_{t-1} \to \theta_t$. We say that *local trend coherence* holds at iteration t when

$$\delta_t^{(K)} \Delta J_t^{\mathrm{base}} > 0.$$

Corollary C.2 (Practical Base-Step Alignment)**.** *Suppose $|\delta_t^{(K)}| \geq m$ and local trend coherence holds. Then, except on an event of probability at most $2\exp(-2nm^2/(1+1/K))$, the practical coefficient $c_t = \phi_\lambda(\xi_t)$ satisfies $c_t \Delta J_t^{\mathrm{base}} \geq 0$ and hence has a non-negative leading term in Proposition 6.1. The inequality is strict when $\lambda > 0$.*

C.4 Group-Relative Boundary Sensitivity

For a fixed query q, let $S := \sum_{i=1}^{G} R(q, y_i)$ be the number of successful responses in a group of size G. Degenerate groups ($S \in \{0, G\}$) have zero group-relative advantage. To make the GRPO objective analytically tractable, we adopt the following premises: (i) **Non-degenerate Event:** We condition our analysis on the non-degenerate event $\mathcal{E} = \{1 \leq S \leq G-1\}$, where the gradient signal is non-zero. (ii) **Local Trust Region:** We assume the policy remains close to the behavior policy ($\pi_\theta \approx \pi_{\theta_{\mathrm{old}}}$), abstracting away the effects of the clipping mechanism. (iii) **i.i.d. Rollouts:** Following standard parallel decoding mechanisms, the G responses are sampled independently and identically given q. This guarantees that the number of successful responses S follows a binomial distribution $S \sim \mathrm{Binomial}(G, p)$, where $p = p(q;\theta)$ is the policy success probability.

Based on the simplifications above and the non-degenerate event $\mathcal{E} = \{1 \le S \le G-1\}$, we have the policy gradient of GRPO:

$$g_{\text{GRPO}} = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G}\nabla_\theta \log \pi_\theta(y_i)\frac{R_i - \hat{\mu}}{\hat{\sigma}} \;\middle|\; \mathcal{E}\right]. \tag{14}$$

Proof of Theorem 6.2. For any given state $S \in \mathcal{E}$, exactly S responses receive a reward of 1, and the remaining $G - S$ responses receive 0. The group-level reward mean is therefore $\mu = \frac{S}{G}$. The empirical variance can be explicitly computed by summing over the correct and incorrect responses:

$$\begin{aligned}\sigma^2 &= \frac{1}{G}\left[S\left(1 - \frac{S}{G}\right)^2 + (G-S)\left(0 - \frac{S}{G}\right)^2\right] \\ &= \frac{S(G-S)}{G^2}.\end{aligned} \tag{15}$$

Substituting μ and σ into the advantage $A_i = (R_i - \mu)/\sigma$, the advantage collapses into two discrete values for correct ($R_i = 1$, denoted as A^+) and incorrect ($R_i = 0$, denoted as A^-) responses:

$$A^+ = \frac{1 - S/G}{\sqrt{S(G-S)}/G} = \sqrt{\frac{G-S}{S}}, \qquad A^- = \frac{0 - S/G}{\sqrt{S(G-S)}/G} = -\sqrt{\frac{S}{G-S}}. \tag{16}$$

Let $C = \{i : R_i = 1\}$ be the indices of the successful responses. We can decompose the marginal probability of any response y by conditioning on its reward outcome R: $\pi_\theta(y) = \mathbb{P}(R) \cdot \pi_\theta(y \mid R)$. Taking the gradient of the log-probability yields:

$$\nabla_\theta \log \pi_\theta(y) = \begin{cases} \frac{\nabla_\theta p_t}{p_t} + \nabla_\theta \log \pi_\theta(y \mid R=1), & \text{if } R(y) = 1, \\ -\frac{\nabla_\theta p_t}{1-p_t} + \nabla_\theta \log \pi_\theta(y \mid R=0), & \text{if } R(y) = 0. \end{cases} \tag{17}$$

We now substitute this decomposition to aggregate the gradient contributions across the sampled group:

$$\begin{aligned}&\sum_{i=1}^{G}\nabla_\theta \log \pi_\theta(y_i)A_i \\ &= A^+\sum_{i\in C}\nabla_\theta \log \pi_\theta(y_i) + A^-\sum_{j\notin C}\nabla_\theta \log \pi_\theta(y_j) \\ &= \left(S \cdot A^+ \cdot \frac{1}{p_t} - (G-S)\cdot A^- \cdot \frac{1}{1-p_t}\right)\nabla_\theta p_t \\ &\quad + A^+\sum_{i\in C}\nabla_\theta \log \pi_\theta(y_i \mid R=1) + A^-\sum_{j\notin C}\nabla_\theta \log \pi_\theta(y_j \mid R=0).\end{aligned} \tag{18}$$

Substituting the explicit values of A^+ and A^-, the coefficient of $\nabla_\theta p_t$ simplifies exactly to:

$$S\sqrt{\frac{G-S}{S}}\frac{1}{p_t} - (G-S)\left(-\sqrt{\frac{S}{G-S}}\right)\frac{1}{1-p_t} = \frac{\sqrt{S(G-S)}}{p_t(1-p_t)}.$$

We next take the conditional expectation for a fixed number of successes S. Conditioned on S, the advantage values (A^+, A^-) and marginal probability p_t are constants; the remaining randomness lies in the responses y_i drawn from their conditional distributions $\pi_\theta(y \mid R)$. Linearity of expectation

therefore gives

$$\begin{aligned}
&\mathbb{E}_{y_i|S}\left[\sum_{i=1}^{G}\nabla_\theta \log \pi_\theta(y_i)A_i\right] \\
&= \frac{\sqrt{S(G-S)}}{p_t(1-p_t)}\nabla_\theta p_t \\
&\quad + A^+\sum_{i\in C}\mathbb{E}_{y\sim\pi(\cdot|R=1)}\left[\nabla_\theta \log \pi_\theta(y \mid R=1)\right] \\
&\quad + A^-\sum_{j\notin C}\mathbb{E}_{y\sim\pi(\cdot|R=0)}\left[\nabla_\theta \log \pi_\theta(y \mid R=0)\right].
\end{aligned} \tag{19}$$

For any valid conditional probability distribution, the expectation of its score function is identically zero:

$$\mathbb{E}_{y\sim\pi_\theta(\cdot|R)}\left[\nabla_\theta \log \pi_\theta(y \mid R)\right] = \nabla_\theta \sum_y \pi_\theta(y \mid R) = \nabla_\theta 1 = 0.$$

Therefore, the residual expectation terms vanish, and the expected gradient contribution conditioned on S becomes

$$\mathbb{E}\left[\sum_{i=1}^{G}\nabla_\theta \log \pi_\theta(y_i)A_i \;\middle|\; S\right] = \frac{\sqrt{S(G-S)}}{p_t(1-p_t)}\nabla_\theta p_t.$$

Taking the full expectation over $S \in \mathcal{E}$, and recognizing that $\nabla_\theta p_t = g_{\text{ideal}}(q)$, we obtain:

$$g_{\text{GRPO}} = \left(\frac{\mathbb{E}[\sqrt{S(G-S)} \mid \mathcal{E}]}{G \cdot p_t(1-p_t)}\right) \cdot g_{\text{ideal}}(q). \tag{20}$$

To derive the exact analytical form of the scaling factor $\eta(p)$, we expand the conditional expectation. Given the i.i.d. premise, the unconditional probability mass function of S is $\mathbb{P}(S=k) = \binom{G}{k}p^k(1-p)^{G-k}$. The probability of the non-degenerate event $\mathcal{E}$ is the complement of the extremes ($S=0$ and $S=G$):

$$\mathbb{P}(\mathcal{E}) = 1 - p^G - (1-p)^G.$$

Thus, the conditional expectation can be explicitly written as:

$$\mathbb{E}[\sqrt{S(G-S)} \mid \mathcal{E}] = \sum_{k=1}^{G-1}\sqrt{k(G-k)}\frac{\mathbb{P}(S=k)}{\mathbb{P}(\mathcal{E})} = \frac{\sum_{k=1}^{G-1}\sqrt{k(G-k)}\binom{G}{k}p^k(1-p)^{G-k}}{1-p^G-(1-p)^G}. \tag{21}$$

Substituting Eq. (21) into Eq. (20) yields the expression for $\eta(p)$ in Theorem 6.2. □

Corollary C.3 (Boundary Scaling Sensitivity)**.** *At the probability boundaries ($p \to 0$ or $p \to 1$), the gradient scaling factor exhibits a symmetric singularity and diverges to infinity:*

$$\eta(p) \sim \frac{\sqrt{G-1}}{G \cdot p(1-p)} \longrightarrow \infty.$$

Proof of Corollary C.3. From Theorem 6.2,

$$\eta(p) = \frac{\sum_{k=1}^{G-1}\sqrt{k(G-k)}\binom{G}{k}p^k(1-p)^{G-k}}{G\,p(1-p)\,(1-p^G-(1-p)^G)}.$$

As $p \to 0$, the leading term of the numerator is $k=1$:

$$\sum_{k=1}^{G-1}\sqrt{k(G-k)}\binom{G}{k}p^k(1-p)^{G-k} = \sqrt{G-1}\,Gp(1-p)^{G-1} + \mathcal{O}(p^2).$$

For the denominator,

$$(1-p)^G = 1 - Gp + \mathcal{O}(p^2), \qquad 1 - p^G - (1-p)^G = Gp + \mathcal{O}(p^2).$$

Taking the ratio gives

$$\lim_{p\to 0} \frac{\sqrt{G-1}\,Gp(1-p)^{G-1} + \mathcal{O}(p^2)}{Gp + \mathcal{O}(p^2)} = \sqrt{G-1}.$$

Substituting back yields

$$\eta(p) = \frac{\sqrt{G-1} + o(1)}{G\,p(1-p)} \sim \frac{\sqrt{G-1}}{G\,p(1-p)}.$$

By symmetry of $\binom{G}{k}$ and $p(1-p)$, the same argument holds as $p \to 1$, implying $\eta(p) \to \infty$ at both boundaries. □

C.5 Signed Groupwise Retrospective Replay

For a stored prompt group q, let $A_{q,i}$ be the attribution of response i and define

$$Z_q := \sum_{i=1}^{G} |A_{q,i}|.$$

The group-relative implementation uses

$$D_q := Z_q + \varepsilon_D, \qquad \varepsilon_D > 0,$$

and sets

$$a_{q,i} := \frac{GA_{q,i}}{D_q}.$$

For the stored responses, define

$$\rho_{q,i}(\theta) := \frac{\pi_\theta(y_{q,i} \mid q)}{\pi_{\theta_{\text{beh}}}(y_{q,i} \mid q)}, \qquad s_{q,i}(\theta) := \nabla_\theta \log \pi_\theta(y_{q,i} \mid q).$$

In the locally unclipped regime, the transported base-attribution direction and retrospective direction are

$$g_q^{\text{base}}(\theta) := \frac{1}{G}\sum_{i=1}^{G} A_{q,i}\rho_{q,i}(\theta)s_{q,i}(\theta),$$

$$g_q^{\text{PI}}(\theta; c_t) := \frac{1}{G}\sum_{i=1}^{G} c_t a_{q,i}\rho_{q,i}(\theta)s_{q,i}(\theta).$$

Proof of Proposition 6.3. Substituting $a_{q,i} = GA_{q,i}/D_q$ gives, at any common evaluation point,

$$\begin{aligned} g_q^{\text{PI}}(\theta; c_t) &= \frac{1}{G}\sum_{i=1}^{G} c_t \frac{GA_{q,i}}{D_q}\rho_{q,i}(\theta)s_{q,i}(\theta) \\ &= c_t \frac{G}{D_q} g_q^{\text{base}}(\theta). \end{aligned}$$

Taking its inner product with $g_q^{\text{base}}(\theta)$ proves the signed-scaling identity.

For the norm bound, the triangle inequality yields

$$\begin{aligned} \left\|g_q^{\text{PI}}(\theta; c_t)\right\| &\le \frac{|c_t|}{G}\sum_{i=1}^{G} |a_{q,i}|\rho_{q,i}(\theta)\|s_{q,i}(\theta)\| \\ &\le \bar{\rho}M|c_t|\frac{Z_q}{D_q} \le \bar{\rho}M|c_t|. \end{aligned}$$

For the cross-point comparison, assume $u_{q,i}(\theta) := \rho_{q,i}(\theta)s_{q,i}(\theta)$ is L_q-Lipschitz between $\widetilde{\theta}_{t-1}$ and θ_t. Then

$$
\begin{aligned}
&\left\| g_q^{\mathrm{PI}}(\theta_t; c_t) - c_t \frac{G}{D_q} g_q^{\mathrm{base}}(\widetilde{\theta}_{t-1}) \right\| \\
&\quad \le \frac{|c_t|}{D_q} \sum_{i=1}^{G} |A_{q,i}| L_q \|\theta_t - \widetilde{\theta}_{t-1}\| \\
&\quad = |c_t| L_q \frac{Z_q}{D_q} \|\theta_t - \widetilde{\theta}_{t-1}\| \\
&\quad \le |c_t| L_q \|\theta_t - \widetilde{\theta}_{t-1}\|.
\end{aligned}
$$

Writing the preceding difference as r_q, Cauchy–Schwarz gives

$$
\left| \left\langle r_q, g_q^{\mathrm{base}}(\widetilde{\theta}_{t-1}) \right\rangle \right| \le |c_t| L_q \|\theta_t - \widetilde{\theta}_{t-1}\| \left\| g_q^{\mathrm{base}}(\widetilde{\theta}_{t-1}) \right\|.
$$

If

$$
L_q \|\theta_t - \widetilde{\theta}_{t-1}\| < \frac{G}{D_q} \left\| g_q^{\mathrm{base}}(\widetilde{\theta}_{t-1}) \right\|,
$$

this remainder is smaller than the leading signed-scaling term, so the cross-point inner product retains the sign of c_t whenever $c_t \neq 0$. □

D Additional Related Work

RL Post-Training with Verifiable and Rich Feedback. RL post-training uses verifiable rewards for reasoning and executable tasks (Yang et al., 2025a; Guo et al., 2025a; Chen et al., 2026d; He et al., 2026a; Wu et al., 2025b), with extensions to tool use and multi-agent credit assignment (Ban et al., 2026; Qian et al., 2025; Li et al., 2026d). Richer signals include mistake logs, judge comments, reward models, privileged information, and self-teacher targets (Zou et al., 2025; Chen et al., 2025b; Xie et al., 2026; Song & Zheng, 2026; Li et al., 2026a; Hübotter et al., 2026; Zhao et al., 2026a). PIPO is complementary: it retains these local signals and adds cross-iteration feedback.

Critic-Free RL and Advantage Estimation. GRPO (Shao et al., 2024) and RLOO (Ahmadian et al., 2024) remove the value network used by PPO (Schulman et al., 2017). Later work studies normalization, outliers, clipping, bias correction, baselines, and sub-trajectory credit (Liu et al., 2026; Lu et al., 2026; Li et al., 2026c; Dai et al., 2026b; Plyusov et al., 2026; Qi et al., 2026; Li et al., 2026b; Qiyuan et al., 2026; Yang et al., 2026; Liu et al., 2025; Zhao et al., 2026b; Yang et al., 2025b; Chen et al., 2025a; Hao et al., 2025; Guo et al., 2025b; Sun et al., 2025). These methods primarily refine intra-batch estimation; PIPO adds feedback across policy updates.

Stability in RL Post-Training. Prior work improves stability through exploration regularization, larger rollout budgets, curricula, and diversity-preserving objectives (Hao et al., 2025; Cheng et al., 2026; Deng et al., 2025; Dai et al., 2026a; Hu et al., 2026; He et al., 2026b; Chen et al., 2026c; Setlur et al., 2025; Sundaram et al., 2026; Tajwar et al., 2026; Wu et al., 2025a; Yue et al., 2025; GX-Chen et al., 2026; Chen et al., 2025c; 2026b;a). These approaches adjust the local objective or sampling process; PIPO instead uses historical performance feedback.

Beyond Intra-Batch Estimation. Historical information has been used beyond individual batches: cross-batch memory approximates global statistics in representation learning (Wang et al., 2020; 2021), SPO uses a moving-average baseline (Xu & Ding, 2026), and HA-DW and OPO use historical statistics for bias correction or reweighting (Yang et al., 2026; Hao et al., 2025). PIPO instead uses recent performance as a dynamic anchor for retrospective modulation.