REVEALING THE EMERGENT LEARNING MECHANISMS OF CRITIC MODELS IN PPO FOR LLM REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning (RL) has proven effective at improving the reasoning capabilities of large language models (LLMs) and has recently been extended to complex, long-horizon agentic tasks. This shift has renewed interest in Proximal Policy Optimization (PPO) over group-sampling-based methods such as Group Relative Policy Optimization (GRPO), owing to its finer-grained credit assignment and greater compatibility with asynchronous training. While prior work has largely focused on improving the overall performance, the learning dynamics of the critic model have received relatively little attention. In this work, we systematically investigate the emergent behaviors and learning mechanisms of the critic model in PPO across three complementary scales: *task*, *group*, and *token*. At the task level, our analysis of trajectory-level advantages reveals that the critic spontaneously develops a GRPO-like group-based normalization pattern, yet without incurring the cost of group sampling. At the group and token levels, we introduce a belief-ranking analysis to demonstrate that the critic indeed provides fine-grained learning signals, while also revealing its susceptibility to potential bias. Motivated by these findings, we identify potential failure modes of PPO and propose to decouple critic and policy training samples to better exploit informative samples. Our analysis offers a new perspective on PPO and GRPO training dynamics and mechanisms, with implications for RL algorithm design for LLMs.

1 INTRODUCTION

Reinforcement learning (RL) has become a standard paradigm for enhancing the reasoning capabilities of large language models (LLMs), with particularly notable gains in domains like mathematical reasoning and code generation (Shao et al., 2024; Guo et al., 2025). As LLMs increasingly evolve from passive responders into autonomous agents (Wu et al., 2025), the role of RL is naturally extending from improving isolated reasoning capabilities to cultivating broader agentic intelligence (Wang et al., 2025; Dong et al., 2026a). Recent advances have demonstrated the promise of RL in improving LLM agents across increasingly complex interactive settings, including software engineering (Wei et al., 2025; Cao et al., 2025), terminal-based tasks (Merrill et al., 2026), and deep research (Team et al., 2025; Shao et al., 2026), where models must perform long-horizon decision making, adapt to environmental feedback, and coordinate reasoning with actions.

For LLM RL, Group Relative Policy Optimization (GRPO) has become widely adopted, which relies on group sampling to estimate advantages from relative rewards within each sampled group (Shao et al., 2024). However, the limitations of this paradigm become increasingly pronounced as the task horizon grows. At the algorithmic level, uniformly propagating a trajectory-level advantage across all tokens obscures the contribution of individual decisions, resulting in inefficient credit assignment over long horizons (Feng et al., 2025; Jiang et al., 2026). At the systems level, group sampling requires completed rollouts to wait for others within groups, limiting asynchronous execution and amplifying straggler effects, especially for long and variable trajectories (Fu et al., 2025; Hou et al., 2026). These features limit the applicability of GRPO to agentic RL settings.

In light of these limitations, Proximal Policy Optimization (PPO)-based methods (Schulman et al., 2017), which leverage a critic model to provide token-level advantage estimates, have recently re-

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.

gained attention (Yue et al., 2025b; Hou et al., 2026; Zhou et al., 2026a). An explicitly trained critic can, in principle, provide finer-grained credit assignment and support single-sample rollouts, making it more compatible with long-horizon tasks. Building on the vanilla PPO framework (Schulman et al., 2017), prior work has introduced a range of adaptations tailored to LLM RL, such as length-adaptive tuning of GAE parameters (Yuan et al., 2025; Yue et al., 2025b), trust-region designs for improved training stability (Hou et al., 2026), and architectural refinements to the critic model (Zhou et al., 2026a;b). Collectively, these modifications have improved the compatibility of PPO with LLM RL and led to better overall performance. Yet prior work has focused largely on policy training dynamics (Yue et al., 2025b; Hou et al., 2026), leaving critic behavior and learning dynamics comparatively underexplored, despite their importance for understanding PPO's learning mechanisms and limitations, and guiding principled improvements.

In this work, we conduct a systematic empirical study of the emergent behaviors and underlying mechanisms of critic models in PPO around three complementary scales: *task*, *group*, and *token*. We ground this analysis in two representative settings, mathematical reasoning and agentic coding (Jimenez et al., 2024). At the task level, our analysis of trajectory-level advantages reveals that the critic spontaneously develops a group-based normalization pattern resembling GRPO, yet without the cost of group sampling. Beyond task-level similarity, we further examine the group- and token-level behaviors enabled by learned critic but absent from GRPO. Through a belief-ranking analysis, we verify that the critic provides fine-grained, richer signals, while showing that the flexibility also makes it more susceptible to potential bias.

Building on these findings, we identify two recurring failure modes in PPO training: the "soft silencing" of training samples and an undesirable "length preference". To mitigate these issues, we propose Critic-Policy Sample Decoupling (CPSD), which allows the training samples for the critic and policy to be selected separately. This approach ensures that policy updates are guided by more informative learning signals and mitigates pathological training dynamics, thereby improving PPO performance. Our contributions are summarized as follows:

- We systematically investigate the emergent behaviors of PPO critics, offering a new perspective for understanding PPO's learning dynamics in LLM RL.
- We identify two failure modes in PPO training, "soft silencing" and "length preference," and theoretically characterize the mechanisms underlying their emergence.
- We propose Critic-Policy Sample Decoupling, a principled method for better exploiting informative learning signals in policy updates and improving the stability of PPO training.

2 PRELIMINARIES

Proximal Policy Optimization. We formulate autoregressive generation as a token-level decision process over prompts $x \sim \mathcal{D}$, in which policy model π_θ induces a trajectory $\tau = (s_t, a_t)_{t=1}^{T}$ with $s_t = (x, a_{<t})$, and the resulting action sequence $y = (a_1, \ldots, a_T)$ constitutes the model response. PPO (Schulman et al., 2017) optimizes the clipped surrogate objective,

$$\mathcal{J}_{\mathrm{PPO}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_{\theta_{\mathrm{old}}}(\cdot|x)} \left[\frac{1}{T} \sum_{t=1}^{T} \min \left(\rho_t(\theta)\widehat{A}_t,\, \mathrm{clip}(\rho_t(\theta), 1-\epsilon, 1+\epsilon)\, \widehat{A}_t \right) \right], \quad (1)$$

where $\rho_t(\theta) = \pi_\theta(a_t|s_t)/\pi_{\theta_{\mathrm{old}}}(a_t|s_t)$ is the token-level importance ratio and ϵ controls the clipping range. To estimate the advantage, PPO maintains a critic $V_\phi(s_t)$ that approximates the expected discounted return from prefix s_t using generalized advantage estimation (GAE):

$$\delta_t = r_t + \gamma V_\phi(s_{t+1}) - V_\phi(s_t), \qquad \widehat{A}_t = \sum_{l=0}^{T-t} (\gamma\lambda)^l \delta_{t+l}. \quad (2)$$

In outcome-based LLM RL, the reward is typically assigned only after the complete response is generated. The critic is trained to regress toward the resulting bootstrapped return targets.

Group Relative Policy Optimization. GRPO constructs heuristic *group-wise* advantage estimates, thereby eliminating the need to explicitly maintain a critic model (Shao et al., 2024). This

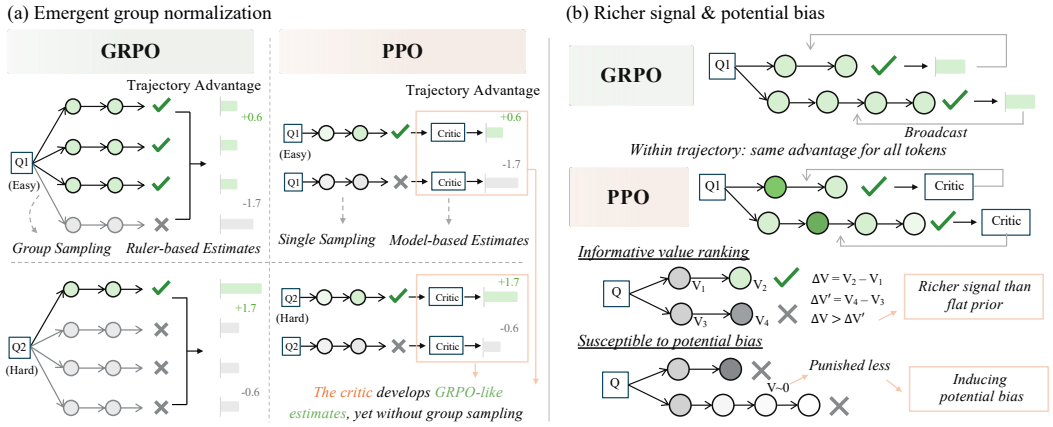


Figure 1: **Illustration of emergent behaviors and mechanisms of critic models.** (a) *Task-level*: After convergence, the critic spontaneously yields GRPO-like, reward-normalized trajectory-level advantage estimates yet without the cost of group sampling. (b) *Group- and token-level*: The critic provides finer-grained signals than GRPO's uniform estimates, but this flexibility can also introduce bias under certain conditions.

estimation scheme is built upon *group sampling*: for each prompt $x \sim \mathcal{D}$, the policy model samples a group of G responses $\{y_i\}_{i=1}^{G}$ with rewards $\{R_i\}_{i=1}^{G}$. The advantage is estimated from group-normalized rewards and broadcast to every token in y_i,

$$\widehat{A}_i = \frac{R_i - \text{mean}\big(\{R_j\}_{j=1}^{G}\big)}{\text{std}\big(\{R_j\}_{j=1}^{G}\big)}, \qquad \widehat{A}_{i,t} = \widehat{A}_i, \tag{3}$$

Subsequent variants have refined this formulation through modifications such as alternative trust-region constraints (Yu et al., 2025) and loss aggregation strategies (Zheng et al., 2025a).

3 Learning Mechanisms of Critic Models in PPO

The central algorithmic distinction between PPO and GRPO lies in *how their advantage estimates are constructed* (Schulman et al., 2017; Shao et al., 2024). Studying what behaviors emerge in the critic and how they shape policy updates therefore provides a distinct perspective on PPO's training dynamics and its relationship to GRPO. In this section, we systematically characterize the *emergent behaviors and learning mechanisms of critic models* across three complementary scales: *task*, *group*, and *token* (schematic overview see Figure 1).

3.1 Setup

Algorithmic Configurations. To account for recent advances in adapting PPO to LLM RL, we augment vanilla PPO with several empirically validated modifications, including critic pretraining (Yuan et al., 2025; Hou et al., 2026), decoupled-GAE (Yuan et al., 2025),length-adaptive GAE hyperparameters (Yue et al., 2025b), skip-observation token-level GAE (Hou et al., 2026). We likewise incorporate empirically validated improvements into GRPO, such as clip-higher (Yu et al., 2025). For brevity, we still refer to these enhanced variants simply as PPO and GRPO.

Models and Benchmarks. For mathematical reasoning, we use Qwen2.5-Math-7B (Yang et al., 2024a) and Qwen3-8B-Base (Yang et al., 2025) as the base models, with DAPO-Math-17K (Yu et al., 2025) as the training dataset, and report performance on AIME 2024–2026 (Dekoninck et al., 2026), AMC23, BeyondAIME (Seed et al., 2025), Brumo, HMMT February 2025, GPQA (Rein et al., 2024), and LiveCodeBench (Jain et al., 2025). For agentic coding, we use Qwen3-8B (Yang et al., 2025) with thinking mode enabled, train on SWE-smith (Yang et al., 2026), and report performance on SWE-bench Verified (Jimenez et al., 2024). All experiments are implemented with veRL (Sheng et al., 2024). Detailed configurations are provided in Appendix B.

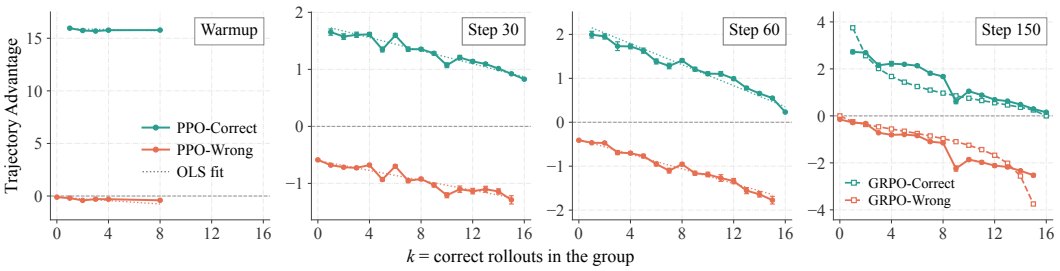


Figure 2: **Emergent group normalization in critic models.** Trajectory-level advantages versus the number of successful samples k out of 16 per task. The four panels show the critic after warmup and at training steps 30, 60, and 150, respectively. The final panel additionally includes GRPO's analytical curves for comparison.

3.2 Learning dynamics of critic models

Evaluation on critic models. Existing analyses of critic models in LLM RL largely operate at an *aggregate level*. Metrics such as explained variance and critic loss only summarize the overall progress of value learning (Yue et al., 2025b; Hou et al., 2026), while another line of work probes value calibration by repeatedly sampling continuations from a fixed prefix and comparing the predicted value with the empirical success rate (Kazemnejad et al., 2025; Zhou et al., 2026b; Chen et al., 2026). These diagnostics, however, do not reveal *how the critic differentiates among trajectories*. Such a comparative view is particularly important, since GRPO constructs its advantage estimates just through within-group comparisons (Shao et al., 2024). Motivated by this, we conduct *trajectory-level comparative analysis* to characterize how critic estimates vary across trajectories at the *task* (advantages of successful trajectories across tasks), *group* (advantages of successful trajectories within a task), and *token* (advantages across tokens within a trajectory) levels.

***Task level*: emergent group normalization.** At the task level, GRPO implicitly incorporates *task difficulty* into advantage estimation through Eq. (3): tasks with higher group success rates receive smaller advantages for correct responses and larger penalties for incorrect ones (Shao et al., 2024). A natural question is how PPO's critic responds to such task difficulty and how its learned behavior compares with GRPO's rule-based mechanism.

To answer this question, we use trajectory-level advantages as a probe. At certain training step, we sample 16 trajectories per task and partition tasks by the number of successful samples, $k \in \{0, \ldots, 16\}$. For each k, we average trajectory-level advantages separately over successful and failed samples. GRPO directly assigns a trajectory-level advantage (Guo et al., 2025), whereas for PPO we compute it by averaging token-level advantages within each trajectory.

Figure 2 shows how trajectory-level advantages vary with k of mathematical reasoning experiments with Qwen2.5-Math-7B (Yang et al., 2024a). For GRPO, this relationship follows analytically from Eq. (3); for PPO, it is learned spontaneously through critic training. Early in training, the critic exhibits sensitivity to task difficulty, assigning larger advantages to successful trajectories and smaller penalties to failed ones on tasks with lower k. As training progresses, these curves gradually approach those of GRPO and closely match its trajectory-level advantage estimates at convergence. Similar observations have also been reported in prior work (Chen et al., 2026). Our trajectory-level analysis further traces how this similarity develops throughout training and highlights an implicit benefit: *PPO exhibits trajectory-level advantage estimation behavior similar to GRPO, yet without incurring the cost of group sampling.*

***Group and token level*: richer signals and potential bias** Beyond the task level, GRPO provides no further signals: *at the group level*, advantage estimates do not distinguish among correct/failed samples within a group; *at the token level*, each trajectory's advantage is broadcast uniformly to all its tokens (Shao et al., 2024). PPO, by contrast, allows greater flexibility at both the group and token levels. We next examine how the critic's estimates at these two levels differ from GRPO's uniform advantage assignments.

At the token level, we first ask *whether the critic provides genuinely informative fine-grained signals*. While its flexibility makes this possible in principle, the uncertainty inherent in LLM reasoning and the complexity of the decision space make the answer far from trivial in practice. One approach is to compare the value estimate at a given prefix with a Monte Carlo (MC) return estimate obtained from repeated rollouts conditioned on that prefix (Kazemnejad et al., 2025). However, this evaluation is computationally expensive, and the MC estimates can have high variance. These limitations have led to mixed findings on whether the critic provides informative fine-grained signals (Kazemnejad et al., 2025; Chen et al., 2026; Zhou et al., 2026b).

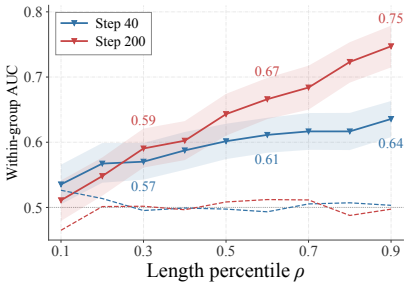


Figure 3: **Informative value rankings from the critic.**

We introduce a computationally lightweight *comparative analysis* that offers a different perspective on this question. Rather than assessing the absolute accuracy of value estimates, we instead examine whether they provide *informative rankings*. For each trajectory, we define the *belief update*

$$\Delta V_\rho = V_\rho - V_0, \tag{4}$$

where V_0 and V_ρ denote value estimates at the start and at fraction ρ of the response length, respectively. For each group g containing both correct (g^+) and incorrect (g^-) responses, we quantify ranking quality by the area under the ROC curve (AUC):

$$\mathrm{AUC}_g(\rho) = \mathbb{E}_{i\in g^+,\, j\in g^-}\left[\mathbb{I}(\Delta V_{i,\rho} > \Delta V_{j,\rho}) + \tfrac{1}{2}\mathbb{I}(\Delta V_{i,\rho} = \Delta V_{j,\rho})\right]. \tag{5}$$

An AUC of 0.5 indicates chance-level ranking. Figure 3 plots the AUC against the relative response position ρ at different training steps, with uncertainty estimated by bootstrapping. Intuitively, the critic generally assigns higher belief-update rankings to successful samples, and its ranking accuracy improves both as trajectories progress and as critic training advances. This provides evidence that the critic could supply richer signals at the token level than a non-informative prior from GRPO.

At the group level, we observe that the critic assigns different learning signals to trajectories that all fail within the same group during early stages of training. Figure 4 shows PPO's advantage estimates within all-failure groups during agentic coding training. In these groups, GRPO assigns zero advantage to every response. In PPO, however, we observe a length preference. We partition failed trajectories into five length bins (Q1–Q5), and compute the trajectory-level advantages within each bin. The critic exhibits a bias toward longer responses, assigning them smaller advantage penalties. We analyze the origins of this bias and its effects on training in Section 4.

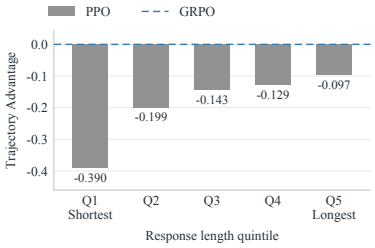


Figure 4: **Length preference in all-failure tasks from the critic.**

4 Failure Modes of PPO for LLM Reinforcement Learning

Building on analysis of critic learning mechanisms, in this section, we identify two failure modes in PPO training, *soft silencing* and *length preference*, and provide a theoretical account of their origins.

4.1 Soft Silencing

At the task level, PPO develops the similar group-normalized advantage pattern as GRPO. This connection suggests that PPO may inherit not only the benefits of group-based normalization, but also its *associated limitations*. A well-known limitation of GRPO arises when all responses within a group are correct or all are failed: Eq. (3) assigns zero advantage to every response, eliminating their contribution to policy update. Such groups are commonly termed *silent groups* or *degenerate groups*. Existing approaches primarily address this issue either by filtering these groups through dynamic sampling (Yu et al., 2025; Zheng et al., 2025b) or by introducing heuristic learning signals to recover their utility (Le et al., 2026).

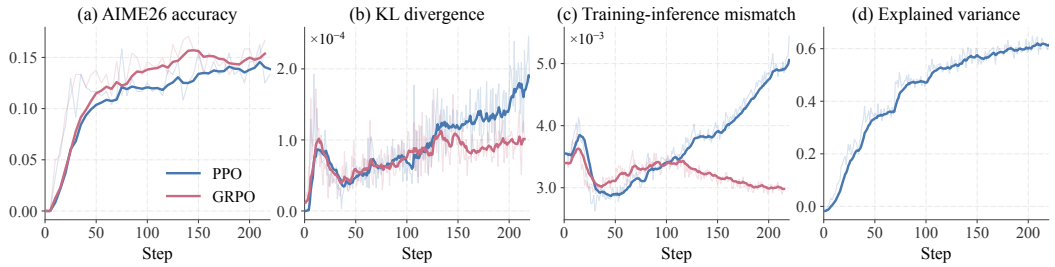


Figure 5: **Training dynamics of Qwen2.5-Math-7B for mathematical reasoning.** Panels show (a) validation score, (b) KL divergence from the old policy model, (c) training–inference mismatch, and (d) critic explained variance. GRPO filters silent groups through dynamic sampling.

PPO *would be expected* to avoid this issue, as its critic should provide differentiated advantages even within all-success or all-failure groups. However, in Figure 2, we could find that trajectory-level advantages of these groups also gradually approach zero during PPO training. In contrast to GRPO's hard silencing, which yields exactly zero advantages for degenerate groups, we refer to this implicit attenuation of learning signals as *soft silencing*.

Such guaranteed successes and failures should theoretically yield vanishing critic-derived advantages, as formalized in the Lemma 1. Intuitively, *the critic does not penalize inevitable failures*.

Lemma 1. *Consider a prefix x such that every possible rollout from x yields the same terminal reward, with no intermediate rewards and $\gamma = 1$. Then, for every state-action pair (s, a) reachable from x, $A^{\pi}(s, a) = 0$.*

As shown in Figure 5, soft-silenced groups introduce instability in the later stages of PPO training, manifested in lower validation scores, drifting Kullback-Leibler (KL) divergence, and an abnormal increase in the training–inference mismatch (Yao et al., 2025; Dong et al., 2026b) (Fig. 5c), despite the critic itself having largely converged (see Appendix D).

4.2 Length Preference

The length preference observed in Figure 4 effectively provides an *implicit boost* to longer action outputs. Figure 6 shows the corresponding training dynamics, where this preference manifests as increasing action length (Fig. 6a). The sharp increase in action length is accompanied by performance degradation: by training step 40, performance on SWE-bench Verified (Jimenez et al., 2024) drops from the base model's 14.8% to 10.4%, with the additional failures primarily attributable to excessively long trajectories and task timeouts (see Appendix D). By contrast, under comparable conditions, action length decreases during GRPO training rather than increasing sharply. GRPO learns to complete tasks within the context length limit, with a larger distribution shift from the base model (Figure 6b), whereas PPO does not.

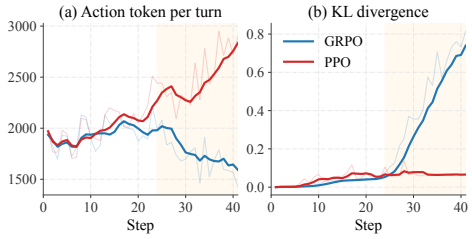


Figure 6: **Training dynamics of Qwen3-8B for agentic coding.** (a) Mean action tokens per turn, (b) KL divergence from reference model.

We investigate the origin of this phenomenon. Table 2 in Appendix D reports success rates across action-token bins. Longer trajectories with more action tokens have substantially lower success rates: they tend to involve harder tasks, while length limits and timeouts further suppress the already scarce positive signals. Consequently, as in Lemma 1, the critic increasingly treats trajectories with more action tokens as *inevitable failures*, assigning them lower value estimates and thus smaller advantage penalties (Schulman et al., 2017). When the policy *struggles to obtain sufficient positive signals by improving its ability* to solve tasks, it may instead exploit this implicit incentive by producing longer actions (further analysis see Appendix D).

This failure mode likewise stems from the critic's tendency not to penalize inevitable failures. The same tendency extends beyond the task level, shaping how the critic assigns advantages to individual trajectories within a group. GRPO, by contrast, assigns the same advantage to all failed trajectories within a group (Shao et al., 2024), regardless of their length, thereby *avoiding the bias at least in advantage level*.

5 CRITIC-POLICY SAMPLE DECOUPLING FOR PPO

To address the PPO failure modes identified in Section 4, we propose Critic-Policy Sample Decoupling (CPSD), a principled method for improving training stability of PPO. The method rests on the insight that critic training and policy updates serve different objectives and need not rely on the same samples. Decoupling their sample selection allows the critic to learn from a broader set of trajectories while focusing policy updates on informative learning signals.

5.1 METHOD

Decoupled sample selection. Let $\mathcal{B}$ denote a batch of trajectories collected by $\pi_{\theta_{\text{old}}}$. CPSD constructs separate training sets $\mathcal{B}_V \subseteq \mathcal{B}$ and $\mathcal{B}_\pi \subseteq \mathcal{B}$ for critic learning and policy optimization, respectively. This separation reflects their different objectives: a trajectory with a weak or unreliable policy learning signal may still provide a useful target for value estimation. In particular, we retain the full rollout batch for critic training, $\mathcal{B}_V = \mathcal{B}$, while selecting $\mathcal{B}_\pi$ according to the informativeness of its policy learning signals.

The resulting objectives of critic training and policy update are

$$\mathcal{L}_V(\phi) = \frac{1}{\sum_{\tau \in \mathcal{B}_V} T_\tau} \sum_{\tau \in \mathcal{B}_V} \sum_{t=1}^{T_\tau} \left(V_\phi(s_t) - \widehat{G}_t \right)^2, \tag{6}$$

$$\mathcal{J}_{\text{CPSD}}(\theta) = \frac{1}{\sum_{\tau \in \mathcal{B}_\pi} T_\tau} \sum_{\tau \in \mathcal{B}_\pi} \sum_{t=1}^{T_\tau} \min \left(\rho_t(\theta) \widehat{A}_t,\ \text{clip}\left(\rho_t(\theta), 1-\epsilon, 1+\epsilon\right) \widehat{A}_t \right), \tag{7}$$

where T_τ denotes the number of action tokens in trajectory τ, $\widehat{G}_t$ is the critic regression target, and $\widehat{A}_t$ is the critic-derived GAE estimate. CPSD changes which trajectories contribute to each objective while preserving PPO's critic-based advantage estimation.

An instantiation: filtering silent groups. As a concrete instantiation, consider $G \geq 2$ trajectories sampled for each prompt x, with binary outcome rewards $R_{x,i} \in \{0, 1\}$. Define the empirical group success rate as

$$\widehat{p}_x = \frac{1}{G} \sum_{i=1}^{G} R_{x,i}. \tag{8}$$

We exclude all-success and all-failure groups from policy optimization while retaining them for critic training:

$$\mathcal{B}_V = \mathcal{B}, \qquad \mathcal{B}_\pi = \{\tau_{x,i} \in \mathcal{B} : 0 < \widehat{p}_x < 1\}. \tag{9}$$

This filtering strategy is motivated by the analysis in Section 4 and follows the same criterion as dynamic sampling in DAPO (Yu et al., 2025). The key difference is that we retain all rollout trajectories for critic training, preserving broader coverage of the training data for the critic model.

5.2 SETUP AND RESULTS

Mathematical reasoning. Detailed training settings are provided in Section 3.1 and Appendix B, with results reported in Table 1. PPO-CPSD achieves better overall performance than both GRPO and PPO. Compared with PPO, it addresses late-stage training instability and improves performance by ensuring informative samples for each policy update (see Appendix D). These results show that, although the critic can in principle provide learning signals for any sample, *the quality of data used for policy updates remains important in PPO*.

Table 1: **Performance across reasoning benchmarks.** Best performance is highlighted in bold, and the Δ row reports the differences between PPO-CPSD and GRPO with dynamic sampling.

	AIME 24	AIME 25	AIME 26	AMC 23	Beyond AIME	Brumo	HMMT 25-Feb	GPQA	LiveCode Bench	Avg.
Base model: Qwen2.5-Math-7B										
Base model	0.16	0.0	0.0	0.39	0.05	0.16	0.05	4.42	8.98	1.58
GRPO	31.25	18.54	16.30	76.41	**6.53**	13.18	2.86	37.85	12.26	23.91
PPO	28.65	16.82	13.23	72.85	6.03	13.02	2.81	38.04	10.04	22.39
PPO-CPSD	**36.46**	**18.70**	**16.61**	**77.89**	6.39	**15.73**	**3.44**	**39.08**	**12.95**	**25.25**
Δ	↑5.21	↑0.16	↑0.31	↑1.48	↓0.14	↑2.55	↑0.58	↑1.23	↑0.69	↑1.34
Base model: Qwen3-8B-Base										
Base model	2.97	2.55	2.97	21.56	1.55	5.62	0.42	24.46	13.04	8.35
GRPO	26.04	22.97	19.12	75.43	12.03	16.15	9.06	**45.80**	17.73	27.15
PPO	23.91	22.86	19.06	74.65	11.67	16.15	8.33	37.91	17.01	25.73
PPO-CPSD	**26.09**	**23.02**	**21.62**	**76.37**	**12.08**	**17.40**	**9.74**	44.43	**19.80**	**27.84**
Δ	↑0.05	↑0.05	↑2.50	↑0.94	↑0.05	↑1.25	↑0.68	↓1.37	↑2.07	↑0.69

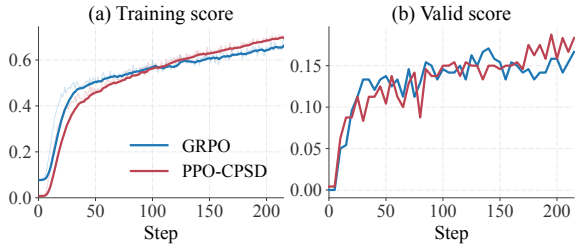


Figure 7: **Comparisons between GRPO and PPO-CPSD.** (a) Accuracy on the training set. (b) Accuracy on the validation set (AIME26).

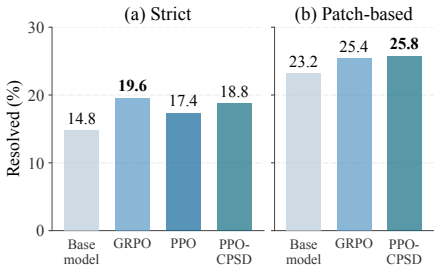


Figure 8: **Comparisons on SWE-bench Verified of Qwen3-8B.**

For the comparison with GRPO, we use matched training hyperparameters and the similar criterion for selecting policy-update samples: GRPO uses dynamic sampling, while PPO-CPSD similarly collects a sufficient number of informative groups. The key distinction is that PPO-CPSD maintains a critic whose parameters accumulate information from past rollouts, whereas GRPO's rule-based advantage estimator does not. Figure 7 compares their performance on the training and validation sets. PPO-CPSD surpasses GRPO on both sets in the later stages of training, *providing empirical evidence consistent with the benefits of the critic's learning signals*.

Agentic coding. Training settings are provided in Section 3.1 and Appendix B. We examine two verification protocols: *Strict* counts trajectories exceeding the task timeout or rollout length limit as failures, whereas *Patch-based* evaluates the patches available at termination regardless of these violations and counts trajectories as successful if their patches pass verification.

Figure 8 presents the results of training Qwen3-8B under both evaluation protocols. Under strict evaluation, PPO undergoes mode collapse due to the length preference discussed above (see Figure 6). GRPO, meanwhile, learns to shorten rollouts appropriately, avoiding length-limit violations and timeouts. PPO-CPSD mitigates this collapse by reducing the proportion of samples from all-failure groups used for policy updates, but still underperforms GRPO. This finding suggests that *PPO could be more sensitive than GRPO to data and format settings*, while GRPO's reliance solely on rewards to differentiate trajectories within each group may confer greater robustness in certain settings (Shao et al., 2024). As a comparison, under the patch-based setting, timeouts and length-limit violations no longer lead to near-certain zero rewards: the success rate in the longest-length bin increases from 0.3% under strict evaluation to 3.9%. The model no longer exhibits a sharp increase in action tokens, and PPO-CPSD trains stably, achieving performance comparable to GRPO.

6 MECHANISM ANALYSIS

In this section, we provide a supplementary discussion of the critic's learning mechanisms.

Do emergent group-comparison patterns in critics depend on group sampling? In our main experiments, each training batch contains multiple rollouts per task, leaving open whether the observed patterns depend on this group sampling scheme. To examine this dependency, we conduct an additional PPO training experiment in which each task contributes only one rollout and all tasks within a batch are distinct. Reward-normalization patterns still emerge under this setting (see Appendix E.1), indicating that *the critic can develop group-comparison patterns without explicit group sampling during training*.

Do the critic's emergent patterns generalize to unseen tasks? Having observed the group-comparison patterns on the training set, we next examine whether they extend to tasks unseen during training. We evaluate the critic on AMC23, which contains held-out mathematical problems, and out-of-domain coding tasks from LiveCodeBench (Jain et al., 2025). Similar patterns emerge on both benchmarks (see Appendix E.2), providing evidence that this behavior *generalizes beyond the training tasks, including across domains*.

Advantage magnitude implicitly reflects sample informativeness. As shown in Figure 2, all-success and all-failure groups exhibit smaller advantage magnitudes. To quantify this association, we use low per-sample $|\bar{A}|$ to predict silent-group membership. From step 40 to 150, ROC-AUC increases from 0.74 to 0.88 (details see Appendix E.1), suggesting that *advantage magnitude could serve as a group-free proxy for sample informativeness*. This finding points to a potential route for extending to asynchronous training without requiring explicit within-group outcomes.

7 RELATED WORK

Adapting PPO to LLM RL settings. Open-Reasoner-Zero (Hu et al., 2025) has showed that vanilla PPO (Schulman et al., 2017) alone can improve LLM reasoning performance. Subsequent work has introduced a range of modifications to better adapt PPO to LLM RL. VC-PPO (Yuan et al., 2025) reduces critic initialization bias through value pretraining and decouples actor and critic GAE parameters, while VAPO (Yue et al., 2025b) introduces length-adaptive GAE for long-CoT training. For asynchronous agentic RL, SAO (Hou et al., 2026) combines skip-observation token-level GAE, single-rollout value learning, and stricter off-policy correction. In addition, a line of work has refined value estimation and other implementation details (Liu et al., 2026; Zhou et al., 2026a; Chen et al., 2026; Fan et al., 2025; Li et al., 2026b; Gong et al., 2026), making PPO better suited to LLM RL and its associated training infrastructure.

Understanding the Learning Dynamics of LLM RL. Alongside work on LLM RL algorithm design, analyses of learning dynamics in LLM RL have yielded important insights into RL training mechanisms. Yue et al. (2025a) analyze pass@k dynamics in RLVR, examining the limited extent to which RL expands a model's reasoning frontier. Cui et al. (2025) characterize the evolution of token entropy during training, offering insights into reasoning diversity. Zhu et al. (2025) examine the differing effectiveness of positive and negative samples, providing a complementary perspective on RL training mechanisms. These analytical studies deepen our understanding of RL and inform the design of more effective algorithms.

8 SUMMARY

In this work, we empirically investigate the learning mechanisms of PPO critics at the task, group, and token levels. Critics develop generalizable reward-normalization patterns without explicit group sampling and provide finer-grained value estimates, but this flexibility also makes them susceptible to bias under certain conditions. Our findings highlight that informative sample selection remains important even when critics can provide learning signals for all samples. Motivated by this insight, we propose CPSD, a principled method for prioritizing informative samples in policy updates, further mitigating PPO training instability. The empirical analysis clarifies PPO's strengths and limitations across training settings, informing algorithm design and the study of training dynamics in LLM reinforcement learning.

AI USE STATEMENT

In this work, we used generative AI tools solely to assist with language polishing of the manuscript. We carefully reviewed all AI-assisted revisions to ensure that they faithfully preserved the original meaning, were technically accurate, and introduced no unsupported claims. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of our analysis methods and the proof of the lemma in the main text and appendix. The analysis code is included in the supplementary materials, and the experimental settings and algorithmic modifications are further documented in the appendix. All training datasets and evaluation benchmarks used in this work are publicly available.

REFERENCES

Shiyi Cao, Dacheng Li, Fangzhou Zhao, Shuo Yuan, Sumanth R Hegde, Connor Chen, Charlie Ruan, Tyler Griggs, Shu Liu, Eric Tang, et al. Skyrl-agent: Efficient rl training for multi-turn llm agent. *arXiv preprint arXiv:2511.16108*, 2025.

Zizhe Chen, Jiqian Dong, Yizhou Tian, Garry YANG, Yongqiang Chen, Zhitang Chen, and James Cheng. Hista and numca: Estimate state value effectively for large language model reinforcement learning. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=ff4b0ELdU0.

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

Jasper Dekoninck, Nikola Jovanović, Tim Gehrunger, Kári Rögnvaldsson, Ivo Petrov, Chenhao Sun, and Martin Vechev. Beyond benchmarks: Matharena as an evaluation platform for mathematics with llms. 2026. URL https://arxiv.org/abs/2605.00674.

Guanting Dong, Hangyu Mao, Kai Ma, Licheng Bao, Yifei Chen, Zhongyuan Wang, Zhongxia Chen, Jiazhen Du, Huiyang Wang, Fuzheng Zhang, Guorui Zhou, Yutao Zhu, Ji-Rong Wen, and Zhicheng Dou. Agentic reinforced policy optimization. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=TX4k7BF6aO.

Yiming Dong, Kun Fu, Haoyu Li, Xinyuan Zhu, Yurou Liu, Lijing Shao, Jieping Ye, and Zheng Wang. Probing RLVR training instability through the lens of objective-level hacking. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=KlGj06E8Wa.

Tiantian Fan, Lingjun Liu, Yu Yue, Jiaze Chen, Chengyi Wang, Qiying Yu, Chi Zhang, Zhiqi Lin, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Bole Ma, Mofan Zhang, Gaohong Liu, Ru Zhang, Haotian Zhou, Cong Xie, Ruidong Zhu, Zhi Zhang, Xin Liu, Mingxuan Wang, Lin Yan, and Yonghui Wu. Truncated proximal policy optimization, 2025. URL https://arxiv.org/abs/2506.15050.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for LLM agent training. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=QXEhBMNrCW.

Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, WANG JIASHU, Tongkai Yang, Binhang Yuan, and Yi Wu. AREAL: A large-scale asynchronous reinforcement learning system for language reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=X9diEuva9R.

Xue Gong, Qi Yi, Ziyuan Nan, Guanhua Huang, Kejiao Li, Yuhao Jiang, Ruibin Xiong, Zenan Xu, Jiaming Guo, Shaohui Peng, and Bo Zhou. Segmental advantage estimation: Enhancing ppo for long-context llm training, 2026. URL https://arxiv.org/abs/2601.07320.

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.

Zhenyu Hou, Yujiang Li, Jie Tang, and Yuxiao Dong. Single-rollout asynchronous optimization for agentic reinforcement learning. *arXiv preprint arXiv:2607.07508*, 2026.

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. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=NFM8F5cV0V.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=chfJJYC3iL.

Zishang Jiang, tingyun li, Jinyi Han, Xinyi Wang, Sihang Jiang, Yizhou Ying, Xiaojun Meng, Jiansheng Wei, Jiaqing Liang, and Yanghua Xiao. From outcomes to actions: Leveraging hindsight for long-horizon language agent training. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=iK3yDEvQ4y.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R Narasimhan. SWE-bench: Can language models resolve real-world github issues? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. VinePPO: Refining credit assignment in RL training of LLMs. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=Myx2kJFzAn.

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. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Thanh-Long V. Le, Myeongho Jeon, Kim Vu, Viet Dac Lai, and Eunho Yang. No prompt left behind: Exploiting zero-variance prompts in LLM reinforcement learning via entropy-guided advantage shaping. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=kiXFIESZKv.

Yingru Li, Jiacai Liu, Jiawei Xu, Yuxuan Tong, Ziniu Li, and Baoxiang Wang. Trust region masking for long-horizon LLM reinforcement learning. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=dPaBOu0Xwk.

Zihang Li, Rui Zhou, Yingcheng Shi, Wenhan Yu, Zhewen Tan, Zixiang Liu, Zeming Li, Binhua Li, Yongbin Li, Tong Yang, and Jieping Ye. Espo: Early-stopping proximal policy optimization, 2026b. URL https://arxiv.org/abs/2605.29860.

Jiashun Liu, Johan Obando-Ceron, Han Lu, Yancheng He, Weixun Wang, Wenbo Su, Bo Zheng, Pablo Samuel Castro, Aaron Courville, and Ling Pan. Asymmetric proximal policy optimization: mini-critics boost LLM reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=0vgzrcv4Dr.

Wenhan Ma, Hailin Zhang, Liang Zhao, Yifan Song, Yudong Wang, Fuli Luo, and Zhifang Sui. Stabilizing moe reinforcement learning by aligning training and inference routers. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=ZMkfGplf4b.

Mike A Merrill, Alexander Glenn Shaw, Nicholas Carlini, Boxuan Li, Harsh Raj, Ivan Bercovich, Lin Shi, Jeong Yeon Shin, Thomas Walshe, E. Kelly Buchanan, Junhong Shen, Guanghao Ye, Haowei Lin, Jason Poulos, Maoyu Wang, Marianna Nezhurina, Di Lu, Orfeas Menis Mastromichalakis, Zhiwei Xu, Zizhao Chen, Yue Liu, Robert Zhang, Leon Liangyu Chen, Anurag Kashyap, Jan-Lucas Uslu, Jeffrey Li, Jianbo Wu, Minghao Yan, Song Bian, Vedang Sharma, Ke Sun, Steven Dillmann, Akshay Anand, Andrew Lanpouthakoun, Bardia Koopah, Changran Hu, Etash Kumar Guha, Gabriel H. S. Dreiman, Jiacheng Zhu, Karl Krauth, Li Zhong, Niklas Muennighoff, Robert Kwesi Amanfu, Shangyin Tan, Shreyas Pimpalgaonkar, Tushar Aggarwal, Xiangning Lin, Xin Lan, Xuandong Zhao, Yiqing Liang, Yuanli Wang, Zilong Wang, Changzhi Zhou, David Heineman, Hange Liu, Harsh Trivedi, John Yang, Junhong Lin, Manish Shetty, Michael Yang, Nabil Omi, Negin Raoof, Shanda Li, Terry Yue Zhuo, Wuwei Lin, Yiwei Dai, Yuxin Wang, Wenhao Chai, Shang Zhou, Dariush Wahdany, Ziyu She, Jiaming Hu, Zhikang Dong, Yuxuan Zhu, Sasha Cui, Ahson Saiyed, Arinbjörn Kolbeinsson, Christopher Michael Rytting, Ryan Marten, Yixin Wang, Jenia Jitsev, Alex Dimakis, Andy Konwinski, and Ludwig Schmidt. Terminal-bench: Benchmarking agents on hard, realistic tasks in command line interfaces. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=a7Qa4CcHak.

Penghui Qi, Zichen Liu, Xiangxin Zhou, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Defeating the training-inference mismatch via fp16. *arXiv preprint arXiv:2510.26788*, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. Pmlr, 2015.

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

ByteDance Seed, Jiaze Chen, Tiantian Fan, Xin Liu, Lingjun Liu, Zhiqi Lin, Mingxuan Wang, Chengyi Wang, Xiangpeng Wei, Wenyuan Xu, et al. Seed1. 5-thinking: Advancing superb reasoning models with reinforcement learning. *arXiv preprint arXiv:2504.13914*, 2025.

Rulin Shao, Akari Asai, Shannon Zejiang Shen, Hamish Ivison, Varsha Kishore, Jingming Zhuo, Xinran Zhao, Molly Park, Samuel G. Finlayson, David Sontag, Tyler Murray, Sewon Min, Pradeep Dasigi, Luca Soldaini, Faeze Brahman, Wen tau Yih, Tongshuang Wu, Luke Zettlemoyer, Yoon Kim, Hannaneh Hajishirzi, and Pang Wei Koh. DR tulu: Reinforcement learning with evolving rubrics for deep research. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=97NEP1pyS3.

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.

Han Shen. On entropy control in LLM-RL algorithms. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=LqazVN5epT.

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. *arXiv preprint arXiv: 2409.19256*, 2024.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, et al. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*, 2025.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, Eli Gottlieb, Yiping Lu, Kyunghyun Cho, Jiajun Wu, Li Fei-Fei, Lijuan Wang, Yejin Choi, and Manling Li. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning, 2025. URL https://arxiv.org/abs/2504.20073.

Yuxiang Wei, Olivier Duchenne, Jade Copet, Quentin Carbonneaux, LINGMING ZHANG, Daniel Fried, Gabriel Synnaeve, Rishabh Singh, and Sida Wang. SWE-RL: Advancing LLM reasoning via reinforcement learning on open software evolution. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=ULblO61XZ0.

Shirley Wu, Michel Galley, Baolin Peng, Hao Cheng, Gavin Li, Yao Dou, Weixin Cai, James Zou, Jure Leskovec, and Jianfeng Gao. CollabLLM: From passive responders to active collaborators. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=DmH4HHVb3y.

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

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

John Yang, Carlos E Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik R Narasimhan, and Ofir Press. SWE-agent: Agent-computer interfaces enable automated software engineering.

In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024b. URL https://openreview.net/forum?id=mXpq6ut8J3.

John Yang, Kilian Lieret, Carlos E Jimenez, Alexander Wettig, Kabir Khandpur, Yanzhe Zhang, Binyuan Hui, Ofir Press, Ludwig Schmidt, and Diyi Yang. SWE-smith: Scaling data for software engineering agents. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2026. URL https://openreview.net/forum?id=63iVrXc8cC.

Feng Yao, Liyuan Liu, Dinghuai Zhang, Chengyu Dong, Jingbo Shang, and Jianfeng Gao. On the rollout-training mismatch in modern RL systems. In *NeurIPS 2025 Workshop on Efficient Reasoning*, 2025. URL https://openreview.net/forum?id=8MHqvb4lK9.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, YuYue, 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 *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=2a36EMSSTp.

Danlong Yuan, Wei Wu, Zhengren Wang, Xueliang Zhao, Huishuai Zhang, and Dongyan Zhao. SWE-minisandbox: Container-free reinforcement learning for building software engineering agents. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=mMSGU8skL8.

Yufeng Yuan, Yu Yue, Ruofei Zhu, Tiantian Fan, and Lin Yan. What's behind ppo's collapse in long-cot? value optimization holds the secret. *arXiv preprint arXiv:2503.01491*, 2025.

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

Yu Yue, Yufeng Yuan, Qiying Yu, Xiaochen Zuo, Ruofei Zhu, Wenyuan Xu, Jiaze Chen, Chengyi Wang, TianTian Fan, Zhengyin Du, et al. Vapo: Efficient and reliable reinforcement learning for advanced reasoning tasks. *arXiv preprint arXiv:2504.05118*, 2025b.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. Pytorch fsdp: Experiences on scaling fully sharded data parallel, 2023. URL https://arxiv.org/abs/2304.11277.

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, 2025a. URL https://arxiv.org/abs/2507.18071.

Haizhong Zheng, Yang Zhou, Brian R. Bartoldson, Bhavya Kailkhura, Fan Lai, Jiawei Zhao, and Beidi Chen. Act only when it pays: Efficient reinforcement learning for LLM reasoning via selective rollouts. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b. URL https://openreview.net/forum?id=x5lITYXmW2.

Zhijian Zhou, Long Li, Xuan Zhang, Zongkai Liu, Yanting Miao, Yuchen Liu, Deshu Chen, Ke Li, Xing Sun, Ruoxi Jiang, Xiaoyu Tan, Chao Qu, and Yuan Qi. DisPPO: Quantile-based distributional reinforcement learning for large language models. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=Tkdgg5uqNK.

Zhijian Zhou, Long Li, Xuan Zhang, Zongkai Liu, Yulei Qin, Ke Li, Xing Sun, Xiaoyu Tan, Chao Qu, and Yuan Qi. Start classifying: Categorical critics for LLM reinforcement learning. In *Third Conference on Language Modeling*, 2026b. URL https://openreview.net/forum?id=NBhmHJleqt.

Xinyu Zhu, Mengzhou Xia, Zhepei Wei, Wei-Lin Chen, Danqi Chen, and Yu Meng. The surprising effectiveness of negative reinforcement in LLM reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=ftVlLG9cks.

A Proof of lemma

A.1 Lemma 1

Consider a prefix x such that every possible rollout from x yields the same terminal reward, with no intermediate rewards and $\gamma = 1$. Then, for every state-action pair (s, a) reachable from x, $A^\pi(s, a) = 0$.

Proof. Since there are no intermediate rewards and $\gamma = 1$, the return of a rollout is exactly its terminal reward. By assumption, every rollout starting from x has terminal reward c. Moreover, any continuation from a state s reachable from x is also a continuation of x. Therefore, for every reachable state s and every reachable action a,

$$Q^\pi(s, a) = \mathbb{E}[G_t \mid s_t = s, a_t = a] = c.$$

Averaging over actions gives

$$V^\pi(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}\left[Q^\pi(s, a)\right] = c.$$

Hence,

$$A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s) = 0.$$

When the critic converges to V^π, the TD residual at every non-terminal step is

$$\delta_t = r_t + V^\pi(s_{t+1}) - V^\pi(s_t) = 0 + c - c = 0.$$

At the terminal transition, we similarly have

$$\delta_{T-1} = R_T - V^\pi(s_{T-1}) = c - c = 0.$$

Therefore every TD residual along the rollout is zero. Since GAE is a weighted sum of these residuals,

$$\hat{A}_t^{\mathrm{GAE}} = \sum_{l=0}^{T-t-1} \lambda^l \delta_{t+l} = 0,$$

for any $\lambda \in [0, 1]$. □

B Configurations of RL training

Training framework. We use veRL (Sheng et al., 2024) as the RL training framework, vLLM as the rollout engine (Kwon et al., 2023), and FSDP as the training backend (Zhao et al., 2023). For mathematical reasoning, correctness is determined by string matching against the ground-truth answer. For agentic coding, we use SWE-miniSandbox (Yuan et al., 2026) as the execution environment.

Mathematical reasoning. We use Qwen2.5-Math-7B (Yang et al., 2024a) and Qwen3-8B-Base (Yang et al., 2025) as the base models, with DAPO-Math-17K (Yu et al., 2025) as the training dataset after removing duplicate problems, and report performance on AIME 2024–2026 (Dekoninck et al., 2026), AMC23, BeyondAIME Seed et al. (2025), Brumo, HMMT February 2025, GPQA (Rein et al., 2024), and LiveCodeBench (Jain et al., 2025) Following a setup similar to Zhou et al. (2026b), each training step collects 16 rollouts for each of 512 prompts. We perform a policy update on each mini-batch of 32 prompts, corresponding to 512 trajectories, resulting in an off-policy training setting. We set the maximum context length to 3K and the policy learning rate to 1×10^{-6}, with a learning-rate warmup of 10 steps. The policy objective uses token-mean

aggregation and decoupled clipping, with upper and lower clipping parameters of 0.28 and 0.2, respectively (Yu et al., 2025). All mathematical reasoning experiments are conducted on a cluster of NVIDIA A100 GPUs with 80 GB of memory each.

In addition to the shared settings above, we enable dynamic sampling for GRPO, filtering out all-success and all-failure groups. For PPO, we incorporate critic pretraining with 30 warmup steps and decoupled GAE (Yuan et al., 2025) with length-adaptive hyperparameters (Yue et al., 2025b). Specifically, we set $\lambda_{\text{critic}} = 1$ and $\lambda_{\text{policy}} = 1 - 1/(\alpha|l|)$, where $|l|$ is the response length of the trajectory and $\alpha = 1.5$. This length-adaptive formulation mitigates the uneven decay of learning signals across trajectories of substantially different lengths under a fixed λ_{policy} (Yue et al., 2025b). The advantage estimates are whitened for policy updates. Following the motivation of faster value updates (Hou et al., 2026), we set the critic learning rate to 2×10^{-6}, twice that of the policy. For PPO-CPSD, the low initial success rate of the base model necessitates repeated sampling to collect sufficient informative groups. Retaining all rollouts for critic training introduces substantial memory overhead during this stage. We therefore cap the number of rollout sampling rounds per training step at 5. This cap also explains the lower training score of PPO-CPSD relative to GRPO at the very beginning of training in Figure 7. For the evaluation in Table 1, we generate 64 rollouts per task for benchmarks with relatively few tasks: AIME24, AIME25, AIME26, AMC23, BeyondAIME, BRUMO, and HMMT25-Feb, and 16 rollouts per task for GPQA and LiveCodeBench, reporting average accuracy. We set the maximum context length to 3k tokens for all mathematical reasoning benchmarks and GPQA, and 2K tokens for LiveCodeBench.

Agentic Coding. We use Qwen3-8B (Yang et al., 2025) with thinking mode enabled, SWE-miniSandbox (Yuan et al., 2026) as the execution environment, and SWE-Agent (Yang et al., 2024b) as the training harness. We train on SWE-smith (Yang et al., 2026), excluding the small number of instances for which sandbox construction fails, and report performance on SWE-bench Verified (Jimenez et al., 2024).

At each training step, we collect 8 rollouts for each of 16 tasks, yielding 128 trajectories. Each policy update uses the full mini-batch of 16 prompts, corresponding to an on-policy training setting. We set the maximum context length to 40K tokens, with a per-turn context limit of 8K tokens. The policy learning rate is 1×10^{-6}. The policy objective uses decoupled clipping, with upper and lower clipping parameters of 0.28 and 0.2, respectively (Yu et al., 2025). We additionally apply KL regularization with a coefficient of 1×10^{-3}, using the k_3 estimator. We impose a per-turn time limit of 300 seconds and an overall task time limit of 900 seconds; exceeding either limit triggers a timeout. As discussed in Section 4, we examine two verification protocols: *Strict* counts trajectories exceeding the time or rollout length limits as failures, whereas *Patch-based* evaluates the patches available at termination regardless of these violations and counts trajectories as successful if their patches pass verification. All agentic coding experiments are conducted on a cluster of NVIDIA H200 GPUs with 141 GB of memory each.

For PPO, we follow the critic training configuration used in the mathematical reasoning experiments, including decoupled GAE (Yuan et al., 2025), length-adaptive λ_{policy} with $\alpha = 1.5$ (Yue et al., 2025b), $\lambda_{\text{critic}} = 1$, and a critic learning rate of 2×10^{-6}. We reduce critic warmup to 10 steps and additionally adopt skip-observation token-level GAE (Hou et al., 2026), which excludes environment observation tokens from the GAE recursion and propagates advantage estimates across consecutive model actions.

C Difference between GRPO and PPO

In this section, we discuss the differences in advantage estimation between GRPO and PPO, complementing our analysis at the *task, group, and token* levels.

We first clarify the three levels of analysis, using GRPO (Shao et al., 2024) as an example.

- **Task level.** *Successful trajectories from different tasks generally receive different advantages.* In GRPO, Eq. (3) uses the group mean reward as a task-specific baseline. Differences in group success rates therefore lead to different advantages for successful trajectories across tasks. We refer to this dependence on task difficulty as the task level.

- **Group level.** *Within a group of trajectories for the same task, GRPO assigns the same advantage to all successful trajectories*, regardless of their other characteristics. We refer to this as the group level, which examines whether trajectories for the same task with the same outcome (success or failure) receive different advantages. GRPO therefore does not differentiate among trajectories at the group level.
- **Token level.** Within each trajectory, GRPO broadcasts the advantage computed using Eq. (3) to every token. We refer to this as the token level, which examines differences in learning signals across tokens within the same trajectory. All tokens consequently receive the same advantage, so GRPO provides no differentiation at the token level.

Unlike GRPO, PPO can in principle differentiate at all three levels. It can assign different advantages to trajectories with the same outcome for the same task and provide fine-grained signals to individual tokens within each trajectory.

D Extend training dynamic analysis

In this section, we provide additional analyses of RL training dynamics and further insights into the underlying mechanisms.

D.1 Training instability of PPO at the saturation stage.

In this section, we provide a more detailed discussion of PPO training dynamics associated with soft silencing (Section 4.1). Figure 9 presents the training dynamics of PPO, including:

- **Reward:** Model performance on the training dataset.
- **KL divergence:** The KL divergence between the old and current policies during policy updates. It measures the shift in token distributions induced by a policy update, with larger values indicating greater distributional shifts (Schulman et al., 2015).
- **Training–inference mismatch:** This mismatch measures the difference in token distributions produced by the same model in the inference and training engines of the RL framework (Yao et al., 2025; Li et al., 2026a; Qi et al., 2025; Ma et al., 2026). We quantify training–inference mismatch as the mean absolute difference between the probabilities assigned to sampled tokens by the rollout policy and the training-side policy before updates:

$$\text{Mismatch} = \frac{\sum_{i,t} m_{i,t} \left|\pi_{\text{train}}(a_{i,t} \mid s_{i,t}) - \pi_{\text{rollout}}(a_{i,t} \mid s_{i,t})\right|}{\sum_{i,t} m_{i,t}}, \tag{10}$$

  where $m_{i,t}$ indicates valid response tokens. Larger values indicate greater disagreement between training and inference. Excessive training–inference mismatch is widely regarded as a major source of instability in reinforcement learning training.
- **Token entropy:** Measures the diversity of the model's outputs (Cui et al., 2025; Shen, 2026).

Figure 9 presents PPO training dynamics over a longer training horizon. As training progresses, the model increasingly fits the training set, and more tasks yield all-success groups, making soft silencing more pronounced. Compared with Figure 5, the KL divergence between the old and current policies grows more rapidly in the later stages, suggesting increasing policy drift as informative training signals become scarce. We also observe an abnormal increase in training–inference mismatch, an important indicator of training instability. Meanwhile, token entropy rises at an accelerating rate in the later stages of training. These trends indicate training instability, accompanied by a gradual decline in reward after step 230 (Shown in panel a).

Table 2: **Success rates and mean action tokens across length bins.**

	Action token	Succ.
Q_1	7.8k	23.1%
Q_2	14.3k	8.4%
Q_3	19.0k	4.4%
Q_4	21.0k	1.1%
Q_5	23.6k	0.3%

Figure 10 compares the training dynamics of GRPO, PPO, and PPO-CPSD. Compared with PPO, PPO-CPSD exhibits less KL divergence drift, relatively stable training–inference mismatch without

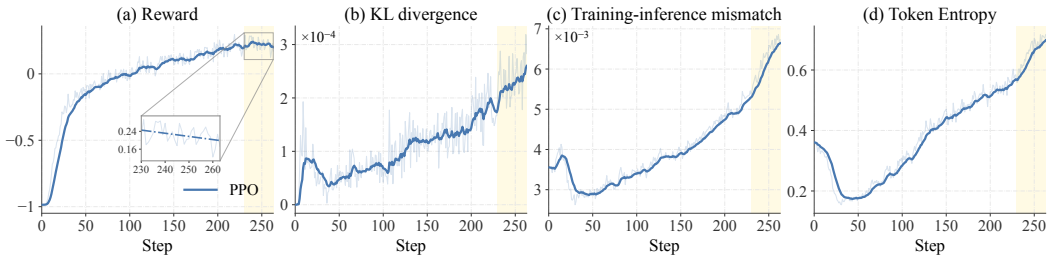


Figure 9: **Extended training dynamics of PPO.** We extend the training run in Figure 5 and show the training dynamic: (a) training reward, (b) KL divergence between the old and current policies, (c) training–inference mismatch, and (d) token entropy. In panel (a), we zoom in on the training dynamics after step 230 and show a linear fit as a dash-dotted line.

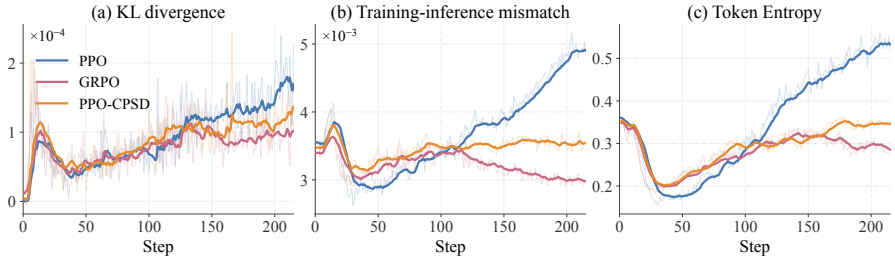


Figure 10: **Training dynamics of GRPO, PPO, and PPO-CPSD.** The three panels show (a) training reward, (b) KL divergence between the old and current policies, and (c) training–inference mismatch.

large abnormal increases, and no accelerating growth in token entropy. Together, these observations indicate improved training stability with PPO-CPSD.

D.2 Pattern collapse in agentic coding training.

Figure 11 shows PPO's training dynamics on agentic coding tasks. The mean number of action tokens per turn increases steadily during training, accompanied by a decline in performance after step 30, primarily due to excessive trajectory lengths and task timeouts. This is also reflected in the model's declining performance on SWE-bench Verified. The model fails to learn to shorten its responses appropriately to prioritize task completion.

As shown in Figure 6(a), GRPO successfully learns to shorten its outputs, reducing task failures caused by excessively long responses. Figure 6(b) further shows that GRPO induces a larger shift in output behavior from the base model, reflected in greater KL divergence. PPO exhibits no compara-

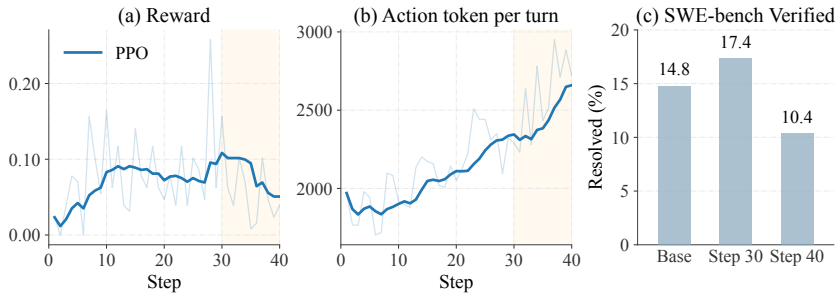


Figure 11: **Training dynamics of PPO for agentic coding tasks.** The three panels show (a) training reward, (b) mean action tokens per turn, and (c) validation score on SWE-bench Verified.

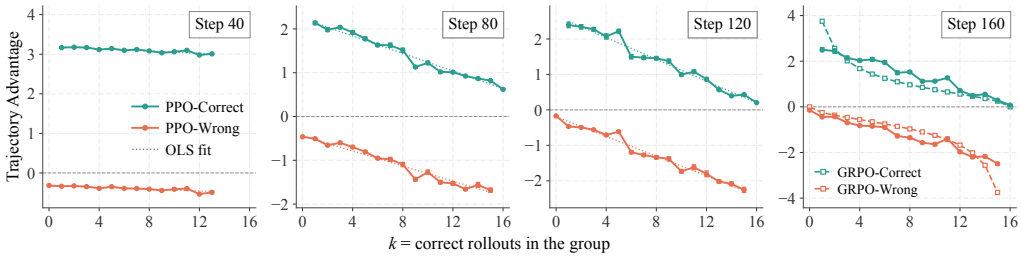


Figure 12: **Emergent group normalization in critic models under single-sample rollout settings.** Trajectory-level advantages versus the number of successful samples k out of 16 per task. The four panels show the critic at training steps 40, 80, 120 and 160, respectively. The final panel additionally includes GRPO's analytical curves for comparison.

ble shift, remaining relatively "lazy" in its adaptation. (Note that the KL divergence here is measured between the current and base models, rather than between the current and old policies during a single policy update.) Differences in advantage estimation between the two methods induce distinct behaviors. We *hypothesize* a potential feedback loop that can lead to collapse during PPO training. Longer trajectories are more likely to fail due to length limits or task timeouts, leading the critic to assign them lower value estimates. In advantage computation, these lower estimates translate into smaller penalties for long, unsuccessful trajectories, implicitly encouraging the policy to generate even longer responses. When informative learning signals are scarce, for example, when most trajectories fail because the tasks exceed the model's capabilities, this implicit bias may increasingly drive training.

E Extended Mechanism Analysis

In this section, we extend the analysis of critic learning mechanisms in Section 6, examining the relationship between group sampling and emergent group-comparison patterns, the generalization of these patterns, and the association between advantage magnitude and sample informativeness.

E.1 Relationship between group sampling and emergent group-comparison patterns

In the main experiments, we use the same sampling strategy for GRPO and PPO to facilitate comparison: each batch contains multiple rollouts per task, a scheme known as group sampling, although PPO does not require it. This setup leaves open whether the critic's emergent group-comparison patterns depend on observing diverse responses to the same task within a batch.

To examine this possibility, we conduct an additional experiment with the other settings unchanged in Section B. We replace the PPO sampling configuration from 512 tasks with 16 rollouts per task at each step, to 512×16 tasks and one rollout per task, preserving the total number of trajectories while removing group sampling. We deduplicate the training data to ensure that each task appears only once within a batch. Under this configuration, we repeat the analysis in Figure 2 to assess whether group-comparison patterns still emerge.

As shown in Figure 12, the critic still develops GRPO-style reward-normalized advantage estimates under the single-rollout setting, demonstrating that *group-comparison patterns can emerge without explicit group sampling during training.*

E.2 Generalization of emergent learning mechanisms of critic models

Section E.1 shows that the critic develops group-comparison patterns reflecting task difficulty on the training set, even without within-batch comparisons enabled by group sampling. This finding establishes that these patterns emerge across different training settings. We next examine how well they generalize: *can the critic retain this difficulty awareness on new tasks, including those outside the training domain?*

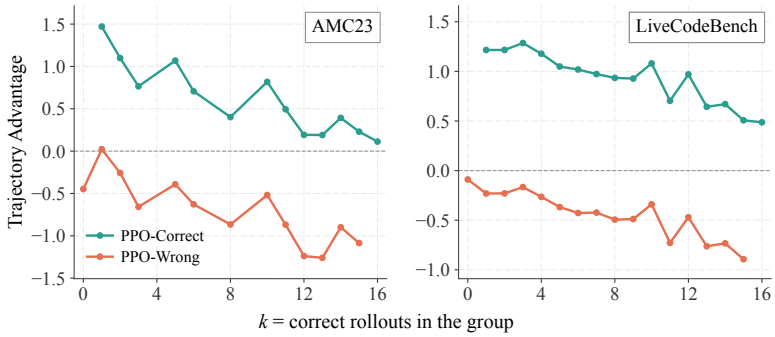


Figure 13: **Generalization of emergent group normalization in critic models.** Trajectory-level advantages versus the number of successful samples k out of 16 per task. The two panels show the critic for AMC23 and LiveCodeBench, respectively.

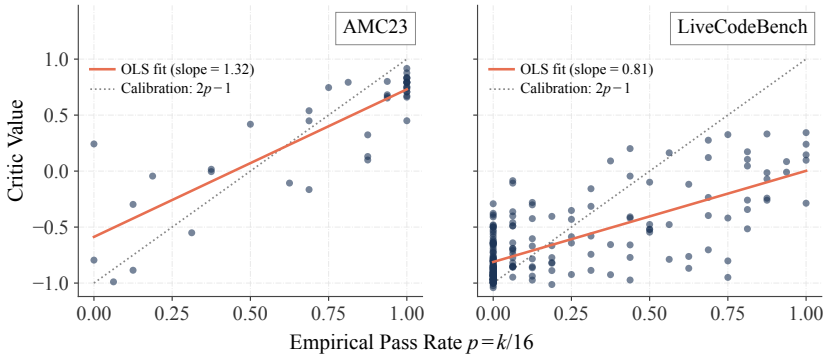


Figure 14: **Critic values track problem difficulty on held-out benchmarks.** Critic values versus empirical pass rates $p = k/16$ on AMC23 (left) and LiveCodeBench (right), with each point representing one task. Orange lines show linear fits, while gray dotted lines denote the ideal calibration reference $2p - 1$. Positive fitted slopes indicate that the critic assigns higher values to problems with higher success rates.

To investigate this question, we examine group-comparison patterns on two benchmarks:

- **AMC23:** Mathematical problems from the same domain as DAPO-Math-17K but not encountered during training, testing generalization within the training domain.
- **LiveCodeBench (Jain et al., 2025):** Coding problems outside the mathematical training domain, testing out-of-domain generalization.

We use the final PPO-CPSD checkpoint of policy model and critic model from Figure 7. For each task in both benchmarks, we generate 16 rollouts using the policy model and compute their advantages using the critic from the same checkpoint.

Figure 13 shows that the critic retains difficulty awareness on tasks not encountered during training. Groups with more successful responses indicate easier tasks, for which the critic assigns smaller advantages to successful trajectories and larger penalties to failed ones.

We further investigate the origin of this advantage-estimation pattern. Figure 14 plots the critic's value estimate at the first response token against the empirical success rate of each task. The critic generally assigns lower values to harder tasks with lower empirical success rates and higher values to easier tasks. These estimates serve as baselines in advantage computation, giving successful responses to harder tasks larger advantages and failed responses smaller penalties.

The critic exhibits stronger difficulty awareness on the in-domain AMC23 benchmark than on the out-of-domain benchmark, as indicated by a steeper slope and closer alignment with the $2p - 1$

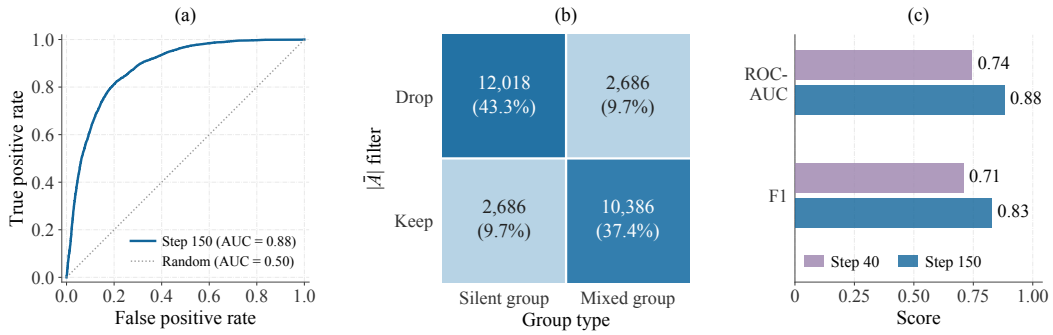


Figure 15: **Advantage magnitude as a proxy for sample informativeness.** (a) ROC curves for identifying silent-group samples by thresholding advantage magnitude. (b) Agreement with silent-group filtering at step 150 under matched discard counts. (c) ROC-AUC and best F1 across thresholds at different training steps.

calibration line, where p denotes the empirical success rate. *Overall, these results suggest that this mechanism generalizes beyond the training data, including to tasks from a different domain.*

E.3 Association between advantage magnitude and sample informativeness

In the main experiments, we use group rewards as the filtering criterion for CPSD to facilitate comparison with GRPO with dynamic sampling. However, advantage magnitude itself could serve as a *group-free proxy* for sample informativeness. As shown in Figure 2, silent groups, which contain less informative samples, tend to exhibit smaller advantage magnitudes.

Figure 15 shows the statistics for identifying samples from silent groups by thresholding advantage magnitude. We treat samples from silent groups as the positive class and predict a sample as belonging to a silent group when its $|\bar{A}|$ falls below a threshold. Varying this threshold yields the ROC curves shown in the figure. The horizontal axis represents the false positive rate: the fraction of mixed-group samples incorrectly classified as silent-group samples. For Figure 15(a), the vertical axis represents the true positive rate (recall): the fraction of silent-group samples correctly identified. Curves closer to the upper-left corner indicate better identification of silent-group samples with fewer false positives. The gray diagonal represents random classification, with an AUC of 0.5. Figure 15(b) shows that, at step 150, advantage-based filtering agrees with silent-group filtering on 81% of sample selection decisions when both discard the same number of samples. Figure 15(c) shows that, from step 40 to 150, ROC-AUC increases from 0.74 to 0.88, while the best F1 score across thresholds rises from 0.71 to 0.83, indicating that advantage magnitude becomes more effective at identifying silent-group samples. Here, F1 is the harmonic mean of precision P and recall R, $F_1 = 2PR/(P+R)$, with silent-group samples treated as the positive class. These results suggest that advantage magnitude alone, computed from a single rollout, can serve as a useful predictor of sample informativeness.