

BEST PRACTICE CRITIC OPTIMIZATION

Penghui Qi¹, Xiangxin Zhou², Wee Sun Lee¹

¹National University of Singapore ²Tencent Hunyuan
 {penghuiq, leews}@comp.nus.edu.sg

ABSTRACT

Group-based reinforcement learning methods such as GRPO for large language models avoid training a critic by sampling multiple responses for each prompt. A reliable critic could instead estimate token-level advantages from one response, but standard critic-based training recipes are often unstable. We study this instability and develop **Best Practice Critic Optimization (BPCO)**¹, a recipe that combines DPPO, value predictions bounded to the reward range, Monte Carlo value targets, unnormalized policy advantages, and length-adaptive generalized advantage estimation. Because the critic is used only during training, BPCO can also condition it on reward-defining information, such as a reference answer or grading rubric, that is hidden from the policy. Controlled experiments isolate the effect of each design choice. Across mathematical reasoning tasks with models ranging from 1.5B parameters to 30B-A3B mixtures of experts, BPCO improves a strong critic-based baseline consistently, and matches or exceeds a group-based baseline while sampling one response per prompt. The same recipe also improves learning with rubric-based rewards. These results show that a carefully designed critic provides a reliable alternative to group-relative advantage estimation. Code is available at https://github.com/QPHutu/golden_critic.

1 INTRODUCTION

Reinforcement learning (RL) has become a standard approach for improving the reasoning and instruction-following abilities of large language models (LLMs) (Ouyang et al., 2022; Guo et al., 2025; Qi et al., 2026a). Effective RL depends on assigning credit to the sampled tokens (Sutton & Barto, 2018). Group-based methods such as GRPO estimate this signal by sampling several responses for each prompt and comparing their rewards (Shao et al., 2024; Liu et al., 2025). This approach avoids training a value function, but it uses multiple rollouts per prompt and assigns the same outcome-based advantage to every token in a response.

A learned critic offers a direct alternative (Schulman et al., 2017). By estimating the expected return of each response prefix, a critic can construct token-level advantages from one rollout (Schulman et al., 2015b; Hou et al., 2026). In practice, however, critic-based LLM training remains fragile. PPO’s ratio clipping treats low- and high-probability tokens unevenly (Qi et al., 2026b). Bootstrapped value targets can inherit critic error (Yuan et al., 2025), and a fixed GAE parameter gives the terminal reward very different weights in short and long responses (Yue et al., 2025). We identify two additional mismatches in common implementations. First, a linear value head can predict outside the known range of the return. Second, batch-wise advantage normalization forces every batch to have unit-scale advantages, even when the residual policy signal has become small. Our controlled study shows that both choices can destabilize training.

A critic also creates an opportunity that group-relative estimators do not directly exploit. Because the critic is discarded after training, it may receive reward-defining information that is unavailable to the policy. Examples include a reference answer or official solution in mathematical reasoning and a prompt-specific rubric in rubric-based evaluation. Such information is determined by the prompt and therefore does not change the ideal value function. Presenting it explicitly can nevertheless make that function easier to approximate, without changing the policy’s inputs or deployment requirements.

¹The BPCO name also reflects its **B**ounded **P**rivileged Critic **O**ptimization design.

We combine these choices into Best Practice Critic Optimization (BPCO), a single-rollout actor-critic recipe. BPCO uses DPPO (Qi et al., 2026b) to define clipping in terms of the sampled token’s probability change. It bounds value predictions to the reward range and trains the critic directly on observed outcome rewards. For the policy update, it preserves the scale of the raw advantages and adapts the GAE parameter to response length. BPCO can additionally use reward-defining information as privileged critic input when such information is available. Together, these choices align the critic’s output, target, and inputs with the policy signal it produces.

We develop the recipe incrementally in a controlled sanity test (Section 3), where failure to fit a small solvable dataset reveals optimization problems. We then evaluate BPCO on a 40.3K-problem mathematical dataset (Section 4.1), two 30B-A3B mixture-of-experts models (Section 4.2), and a rubric-reward task (Section 4.3). The experiments support three findings. First, BPCO improves the critic-based baseline across model and dataset scales. Second, privileged information can accelerate critic learning, but its policy benefit depends on the task and the degree of overfitting. Third, BPCO matches or exceeds a group-based baseline while using one response per prompt. These results establish a practical recipe for single-rollout critic-based LLM RL.

2 BACKGROUND

2.1 PROXIMAL POLICY OPTIMIZATION

Given a prompt x , a language model with parameters θ generates a response $y = (y_1, \dots, y_T)$ autoregressively. At step t , the state is the prefix $s_t = (x, y_{<t})$, the action is the next token y_t , and the policy is $\pi_\theta(y_t \mid s_t)$. We consider outcome rewards: a completed response receives a scalar reward $R(x, y)$, and all intermediate rewards are zero.

Proximal Policy Optimization (PPO) uses a clipped surrogate objective (Schulman et al., 2017). Let μ be the behavior policy that generated the rollouts, and define sampled-token probability ratio as

$$\rho_t(\theta) = \frac{\pi_\theta(y_t \mid s_t)}{\mu(y_t \mid s_t)}.$$

Given an advantage estimate $\hat{A}_t$, PPO maximizes

$$\mathcal{L}_{\text{PPO}}(\theta) = \mathbb{E}_t \left[\min \left(\rho_t(\theta) \hat{A}_t, \text{clip}(\rho_t(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_t \right) \right]. \quad (1)$$

The clipped term removes the incentive to move the ratio farther beyond the clipping boundary in the direction favored by $\hat{A}_t$, forming a trust region to stabilize training (Schulman et al., 2015a).

2.2 DIVERGENCE PROXIMAL POLICY OPTIMIZATION

PPO applies the same ratio threshold to every token. In a large vocabulary, this rule clips small absolute changes to low-probability tokens while allowing much larger absolute changes to high-probability tokens (Qi et al., 2026b). Divergence Proximal Policy Optimization (DPPO) instead defines the clipping boundary in terms of the sampled token’s probability change. The binary total-variation variant used in this work replaces ϵ in Equation (1) with $\epsilon/\mu(y_t \mid s_t)$:

$$\mathcal{L}_{\text{DPPO}}(\theta) = \mathbb{E}_t \left[\min \left(\rho_t(\theta) \hat{A}_t, \text{clip} \left(\rho_t(\theta), 1 - \frac{\epsilon}{\mu(y_t \mid s_t)}, 1 + \frac{\epsilon}{\mu(y_t \mid s_t)} \right) \hat{A}_t \right) \right]. \quad (2)$$

Equivalently, DPPO constrains the probability shift of the sampled token under the policy update, i.e., $|\pi_\theta(y_t \mid s_t) - \mu(y_t \mid s_t)| \leq \epsilon$. This gives sampled tokens a common absolute-probability threshold rather than a common ratio threshold.

2.3 CRITIC-BASED METHODS

Critic-based methods estimate the expected return of each prefix. For rollouts from μ , the value function is

$$V^\mu(s_t) = \mathbb{E}_\mu[R(x, y) \mid s_t],$$

and the critic $V_\phi(s_t)$ approximates this quantity. Let ϕ_{old} denote the frozen critic parameters used to construct targets. Generalized advantage estimation (GAE) (Schulman et al., 2015b) first computes temporal-difference residuals and then forms an exponentially weighted sum:

$$\delta_t = r_t + \gamma V_{\phi_{\text{old}}}(s_{t+1}) - V_{\phi_{\text{old}}}(s_t), \quad (3)$$

$$\hat{A}_t^{\text{GAE}(\lambda)} = \sum_{l=0}^{T-t} (\gamma\lambda)^l \delta_{t+l}. \quad (4)$$

Here $r_t = 0$ for $t < T$, $r_T = R(x, y)$, and $V_{\phi_{\text{old}}}(s_{T+1}) = 0$. In LLM training, $\gamma = 1$ is commonly used. The value of λ controls the degree of bootstrapping. Smaller λ can reduce variance but makes the estimate biased and more sensitive to critic error. With $\lambda = 1$, the sum telescopes to $R(x, y) - V_{\phi_{\text{old}}}(s_t)$, which is an unbiased Monte Carlo estimate without bootstrapping.

Many implementations construct the critic target as

$$\hat{V}_t(\lambda) = \hat{A}_t^{\text{GAE}(\lambda)} + V_{\phi_{\text{old}}}(s_t) \quad (5)$$

and minimize

$$\mathcal{L}_V(\phi) = \mathbb{E}_t \left[\left(V_\phi(s_t) - \hat{V}_t(\lambda) \right)^2 \right]. \quad (6)$$

The policy update uses $\hat{A}_t^{\text{GAE}(\lambda)}$ in Equation (1) or Equation (2).

2.4 GROUP-BASED METHODS

Group-based methods avoid a critic by sampling G responses $\{y^{(i)}\}_{i=1}^G$ for each prompt (Shao et al., 2024). Let $R_i = R(x, y^{(i)})$, and let μ_R and σ_R be the mean and standard deviation of the G rewards. GRPO assigns every token in response i the advantage

$$\hat{A}_{t,i}^{\text{GRPO}} = \frac{R_i - \mu_R}{\sigma_R}. \quad (7)$$

Dr. GRPO removes the standard-deviation normalization, which can otherwise reweight prompts according to their within-group reward variance (Liu et al., 2025). Its advantage is

$$\hat{A}_{t,i}^{\text{Dr.GRPO}} = R_i - \mu_R. \quad (8)$$

2.5 CRITICS FOR LONG-RESPONSE LLM REINFORCEMENT LEARNING

Recent work has revisited critics for long-response LLM RL. VC-PPO decouples the GAE parameters used for policy and critic training and pretrains the critic to reduce initialization error (Yuan et al., 2025). VAPO combines these ideas with length-adaptive GAE and other long-response optimization techniques (Yue et al., 2025). SAO uses a critic for single-rollout asynchronous RL, with additional critic updates and frozen attention to track a changing policy (Hou et al., 2026). BPCO is complementary: it isolates the effects of the critic’s output range, batch-wise advantage normalization, and optional reward-defining inputs in a single-rollout setting.

3 BUILDING BPCO: A CONTROLLED STUDY

We begin from a verl commit from June 16, 2026 and study critic stability in a controlled sanity test (Qi et al., 2025; 2026b). We fine-tune DeepSeek-R1-Distill-Qwen-1.5B (Guo et al., 2025) on 1,460 mathematical problems that the initial model can solve. A suitable training recipe should fit this deliberately small dataset to nearly 100% reward. Failure to do so exposes an optimization problem rather than a lack of model capacity or reward signal.

Each iteration contains 1,024 trajectories. We use a minibatch size of 256 and one optimization epoch, giving four optimizer minibatches per iteration. Following the verl defaults (Sheng et al., 2025), the policy and critic learning rates are 10^{-6} and 10^{-5} , respectively. We observed no benefit from critic warm-up in this small-data setting and therefore update the policy and critic from the first iteration. Each run lasts 1,500 iterations. Because fitting this small dataset can harm generalization,

we monitor AIME 2025 avg@32, the mean accuracy over 32 sampled responses per problem, as a held-out metric.

We modify the default recipe one component at a time. The starting point uses PPO (Section 2.1), standard GAE and critic targets with $\lambda = 1$ (Section 2.3). Unless stated otherwise, each step retains all preceding changes.

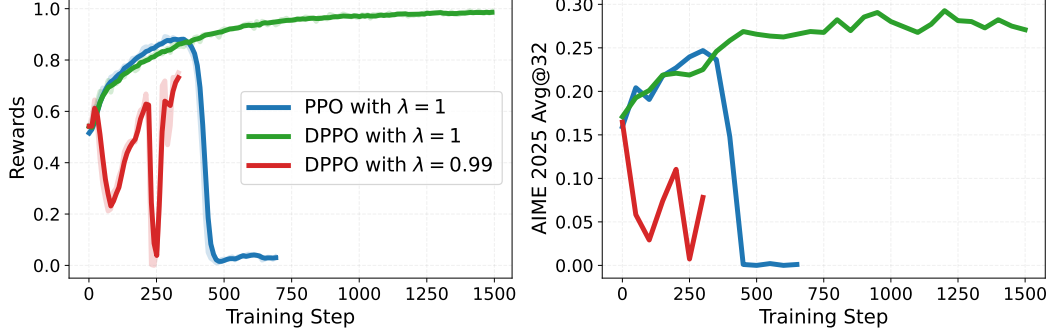


Figure 1: PPO and DPPO in the sanity test. With $\lambda = 1$, PPO’s training reward collapses after an initial increase, whereas DPPO remains stable. Using $\lambda = 0.99$ makes DPPO unstable again.

3.1 STEP 1: REPLACING PPO WITH DPPO

With the PPO objective in Equation (1), the training reward collapses. Replacing it with the DPPO objective in Equation (2) yields stable optimization when $\lambda = 1$, as shown in Figure 1.

Reducing the GAE parameter to $\lambda = 0.99$ makes DPPO unstable again. When $\lambda < 1$, the policy advantage contains bootstrapped critic predictions. Unless $V_\phi(s_t) = V^\mu(s_t)$ for every visited state, approximation error biases the advantage estimate relative to the Monte Carlo estimator obtained with $\lambda = 1$. We therefore use $\lambda = 0.99$ as a stress test in the next steps: stabilizing this setting requires the recipe to control how critic error enters the policy update.

3.2 STEP 2: BOUNDING VALUES TO THE REWARD RANGE

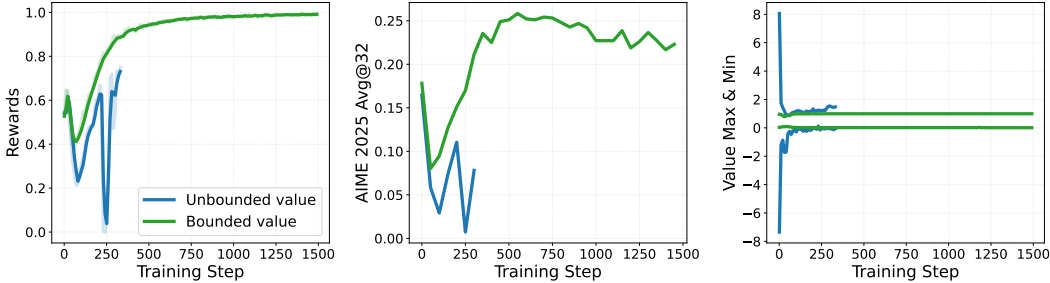


Figure 2: Effect of bounding critic predictions to the reward range. The unbounded linear head predicts values outside the binary-reward range $[0, 1]$ (right), leading to unstable training reward (left) and AIME 2025 avg@32 (middle). The bounded value keeps predictions within the reward range and yields stable training.

Most existing recipes use a linear head to predict the value directly, which is unbounded even when the return is known to lie in a finite interval. Let $[R_{\min}, R_{\max}]$ be the reward range and let $z_\phi(s_t) \in \mathbb{R}$ be the linear head output. Because the expectation of a bounded random variable lies in the same interval, a valid value prediction must satisfy $V_\phi(s_t) \in [R_{\min}, R_{\max}]$. We enforce this property with a scaled arctangent:

$$V_\phi(s_t) = R_{\min} + (R_{\max} - R_{\min}) \left(\frac{1}{2} + \frac{1}{\pi} \arctan(z_\phi(s_t)) \right). \quad (9)$$

This parameterization maps every finite head output to the open interval $(R_{\min}, R_{\max})$ and approaches either endpoint asymptotically. The sanity test uses binary rewards, so $R_{\min} = 0$ and $R_{\max} = 1$. Empirically, it removes the extreme values produced by the linear head and allows the training reward to approach one (Figure 2).

3.3 STEP 3: USING UNBIASED MONTE CARLO VALUE TARGET

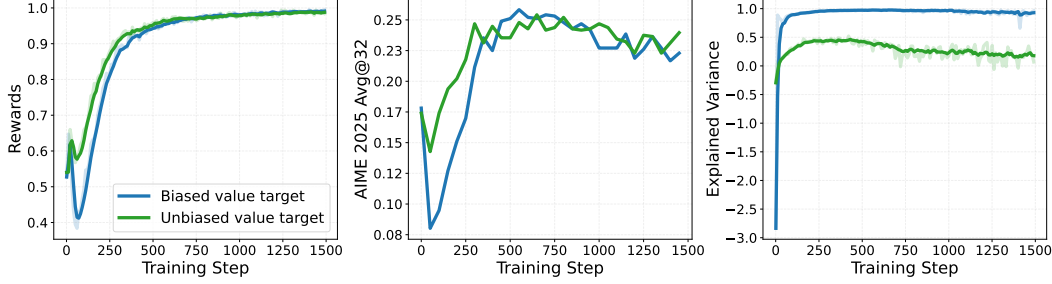


Figure 3: Effect of the unbiased value target. Regressing to the final reward gives more stable training reward (left) and AIME 2025 avg@32 (middle) than a target bootstrapped with $\lambda = 0.99$. Explained variance against the bootstrapped target rapidly approaches one (right) because that target is biased; this unexpectedly high value does not imply accurate prediction of the observed return.

We track how much target variance the critic explains using

$$\text{EV}(V_\phi, \hat{V}) = 1 - \frac{\text{Var}(\hat{V}_t - V_\phi(s_t))}{\text{Var}(\hat{V}_t)}. \quad (10)$$

An explained variance near one normally indicates a close fit to the chosen target. For the standard target in Equation (5), however, $\hat{V}_t(\lambda)$ contains $V_{\phi_{\text{old}}}$ whenever $\lambda < 1$. This self-referential target can be easy for the updated critic to predict even when it is inaccurate with respect to the observed return. In Figure 3, explained variance against the bootstrapped target rapidly approaches one while policy training remains unstable.

Following decoupled GAE in VC-PPO (Yuan et al., 2025), we use separate parameters for the policy advantage and the critic target. We retain $\lambda_\pi = 0.99$ for the policy, but set $\lambda_V = 1$ for critic training. With $\gamma = 1$ and outcome-only rewards, the target telescopes to

$$\hat{V}_t = \hat{A}_t^{\text{GAE}(1)} + V_{\phi_{\text{old}}}(s_t) = R(x, y). \quad (11)$$

For a continuation sampled from μ , the final outcome is an unbiased Monte Carlo sample of $V^\mu(s_t)$. Decoupling the estimators retains the variance reduction of $\lambda_\pi < 1$ for the policy while removing bootstrapping bias for the critic target. This change improves reward stability and convergence speed in Figure 3; the reported explained variance is now measured against the observed outcome reward, and becomes reasonable.

3.4 STEP 4: REMOVING ADVANTAGE NORMALIZATION

Many PPO implementations normalize advantages within each batch before the policy update (Huang et al., 2022). Let $\bar{A}$ and σ_A denote the batch mean and standard deviation of the estimated advantages, respectively. This procedure replaces $\hat{A}_t$ in Equation (1) and Equation (2) with

$$\tilde{A}_t = \frac{\hat{A}_t - \bar{A}}{\sigma_A}. \quad (12)$$

We find this transformation fundamentally problematic. As the policy approaches optimality, the advantages and their variance should both decrease toward zero, causing the policy update to diminish naturally and preserve the policy. Dividing by σ_A , however, removes this desirable behavior. When the standard deviation is small, estimation noise is amplified into a large training signal, preventing

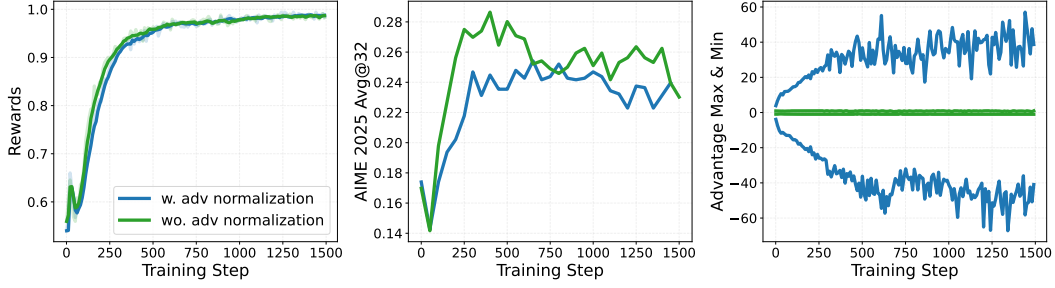


Figure 4: Effect of removing batch-wise advantage normalization. Removing normalization achieves comparable training reward (left) while mitigating the overfitting risk on AIME 2025 (middle). Batch-wise advantage normalization forces small residual advantages to a large, growing range when the policy approaches to optimal (right).

the update from vanishing near the optimum. Moreover, subtracting $\bar{A}$ can reverse the sign of examples with positive advantages that are smaller than the batch mean, thereby hampering exploration.

We therefore remove batch advantage normalization and use the raw GAE estimates for policy updates. As shown in Figure 4, removing normalization keeps the advantage range small and stable, whereas normalization causes the magnitude of the normalized advantages to increase during training. Notably, removing advantage normalization also improves validation performance, which is consistent with the less aggressive updates obtained after the training set has nearly been fit.

3.5 STEP 5: PROVIDING PRIVILEGED INFORMATION TO THE CRITIC

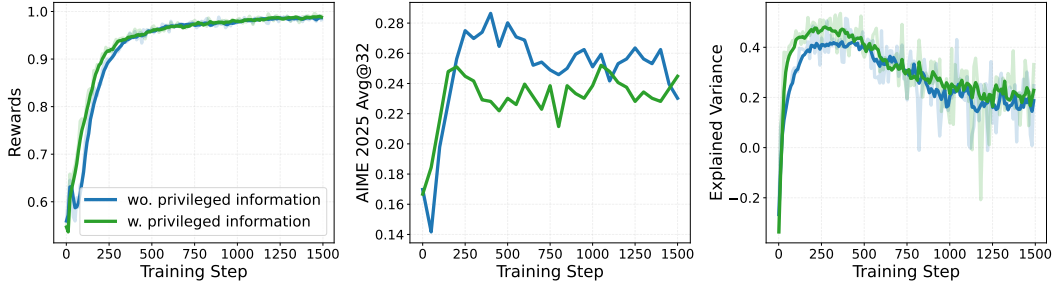


Figure 5: Effect of giving the reference answer only to the critic. Privileged input accelerates training-reward improvement (left) and increases explained variance (right). Its AIME 2025 avg@32 rises faster but peaks earlier (middle), revealing a greater risk of overfitting in this small-data setting.

The critic is needed only during training, so its inputs need not be identical to the policy’s inputs. This observation parallels centralized training with decentralized execution in multi-agent RL (Amato, 2024), including systems that expose hidden game state to a training-time critic (Vinyals et al., 2019; Wang et al., 2021). We apply the same principle to reward-defining information in LLM RL.

Let $q(x)$ denote information used to evaluate responses to prompt x . For mathematical reasoning, $q(x)$ can be the reference answer. A privileged critic estimates

$$V_{\phi}^{\mu}(s_t, q(x)) \approx \mathbb{E}_{\mu}[R(x, y; q(x)) \mid s_t, q(x)]. \quad (13)$$

Because $q(x)$ is fixed by x , exposing it explicitly does not change the optimal value associated with a prompt. It can nevertheless reduce the approximation burden on a finite model. The rollout policy still receives only x and the generated prefix.

As shown in Figure 5, using privileged information leads to faster and more stable training rewards, as well as higher explained variance. However, it also highlights a risk of overfitting: validation performance begins to decline earlier, despite improving more rapidly during the initial stage.

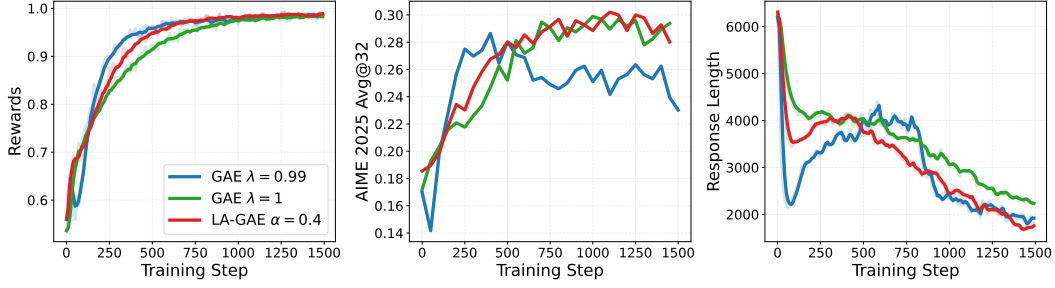


Figure 6: Effect of length-adaptive GAE. A fixed $\lambda_\pi = 0.99$ attains rapid reward improvement (left), but its AIME 2025 avg@32 declines after the initial peak (middle). LA-GAE with $\alpha = 0.4$ retains better training efficiency than $\lambda = 1$ while mitigating the overfitting risk.

3.6 STEP 6: ADOPTING LENGTH-ADAPTIVE GAE

A fixed $\lambda_\pi < 1$ gives the terminal reward exponentially less weight in longer responses. For a token at position t , the coefficient on the terminal TD residual is proportional to λ_π^{T-t} . When $T-t$ is large, early-token advantages depend primarily on bootstrapped critic residuals, and systematic critic error can dominate the policy signal.

Following VAPO and SAO (Yue et al., 2025; Hou et al., 2026), we use length-adaptive GAE. For a response of length $L = |y|$, the policy parameter is

$$\lambda_\pi(L) = 1 - \frac{1}{\alpha L}, \quad (14)$$

where $\alpha > 0$ controls the bias-variance trade-off. The coefficient of the terminal residual in the earliest-token advantage is approximately $(1 - \frac{1}{\alpha L})^L \approx \exp(-1/\alpha)$, which is nearly invariant to response length. We continue to use $\lambda_V = 1$ for critic and do not use privileged inputs in this step.

As shown in Figure 6, fixed $\lambda_\pi = 0.99$ fits the training set fastest but exhibits a pronounced validation decline. Setting $\lambda_\pi = 1$ avoids this decline at the cost of slower optimization. LA-GAE with $\alpha = 0.4$ provides the best trade-off in this study.

4 BROADER EVALUATION

We compare BPCO with group-based and critic-based baselines. The group baseline uses Dr. GRPO (Liu et al., 2025) with 16 responses per prompt. We reduce its number of distinct prompts so that all methods use the same total number of trajectories per iteration. The critic baseline includes two strong existing techniques: decoupled GAE with an unbiased Monte Carlo value target (Yuan et al., 2025) and length-adaptive GAE (Yue et al., 2025). It nevertheless retains an unbounded value head and batch-wise advantage normalization. BPCO differs from this critic baseline only by bounding value predictions and preserving raw advantages. All methods use DPPO for policy optimization (Qi et al., 2026b), which isolates advantage estimation from the choice of policy objective.

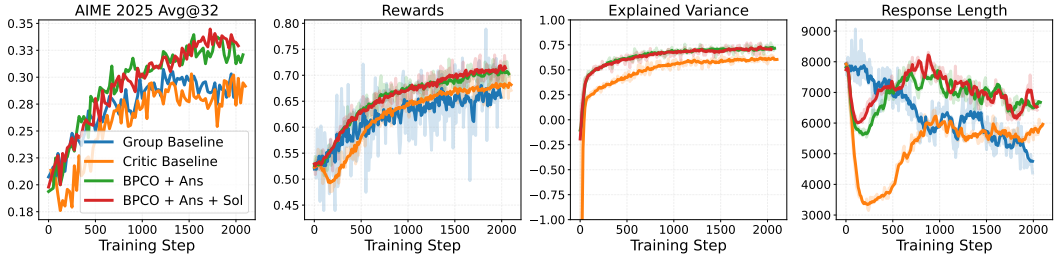


Figure 7: Results on a larger DeepScaleR dataset with DeepSeek-R1-Distill-Qwen-1.5B.

All critic-based methods sample one response for each prompt and warm up the critic during the first 15 iterations. We explicitly denote privileged variants by BPCO+Ans, BPCO+Sol, and BPCO+Ans+Sol. These variants provide the critic with the reference answer, the official solution, or both, respectively. The policy receives the same prompt as all baselines. For mathematical tasks, we report training reward, response length, and AIME 2025 avg@32. Explained variance is computed against the Monte Carlo target in Equation (11).

4.1 SCALING TO A LARGER DATASET

We first test whether BPCO scales beyond the small sanity-test dataset. We fine-tune DeepSeek-R1-Distill-Qwen-1.5B (Guo et al., 2025) on DeepScaleR (Tan et al., 2026), which contains approximately 40.3K mathematical problem-answer pairs. Official solutions are available for about 7.3K problems. We allow generated responses of up to 24,000 tokens.

Figure 7 shows that our BPCO recipes consistently outperform both group-based and critic-based baselines, yielding clear improvements in training and validation performance. Compared with the critic-based baseline, BPCO achieves consistently higher explained variance throughout training, providing strong evidence that it learns a more accurate critic. These results demonstrate that BPCO critic remains effective on a substantially larger training set.

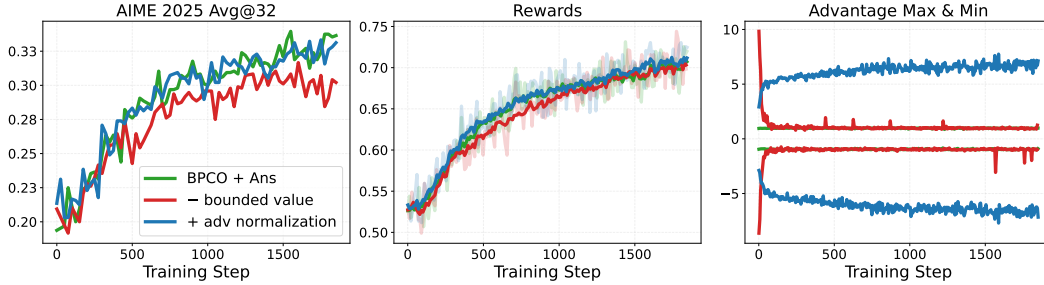


Figure 8: Ablation of bounded value prediction and batch-wise advantage normalization.

We next ablate our proposed techniques starting from a well-performing BPCO+Ans run. Specifically, we remove bounded value prediction and reintroduce batch-wise advantage normalization. The results are presented in Figure 8.

Bounded Value Prediction. Removing the value bound slows the improvement in training reward and reduces the AIME 2025 avg@32 score. Aligning the critic’s output range with that of the return therefore remains beneficial, even when the training dataset is substantially larger.

Removing Advantage Normalization. Reintroducing batch-wise advantage normalization causes the advantage magnitude to grow during training, although this effect is less pronounced than in the sanity test. Removing advantage normalization provides only modest performance gains in this setting, likely because training has not yet fully converged. Nevertheless, we recommend removing advantage normalization as a general-purpose default.



Figure 9: Ablation of privileged information for critic training.

Privileged Information. Figure 9 ablates the effects of privileged information within our BPCO recipe. Providing the reference answer as privileged information leads to faster training, higher explained variance, and better AIME 2025 performance. Using the official solution also yields a modest improvement, even though only 7.3k of the 40.3k problems include this information. These results indicate that privileged information can substantially improve critic training when the dataset is sufficiently large and overfitting is not yet a concern.

4.2 SCALING TO LARGER MODELS

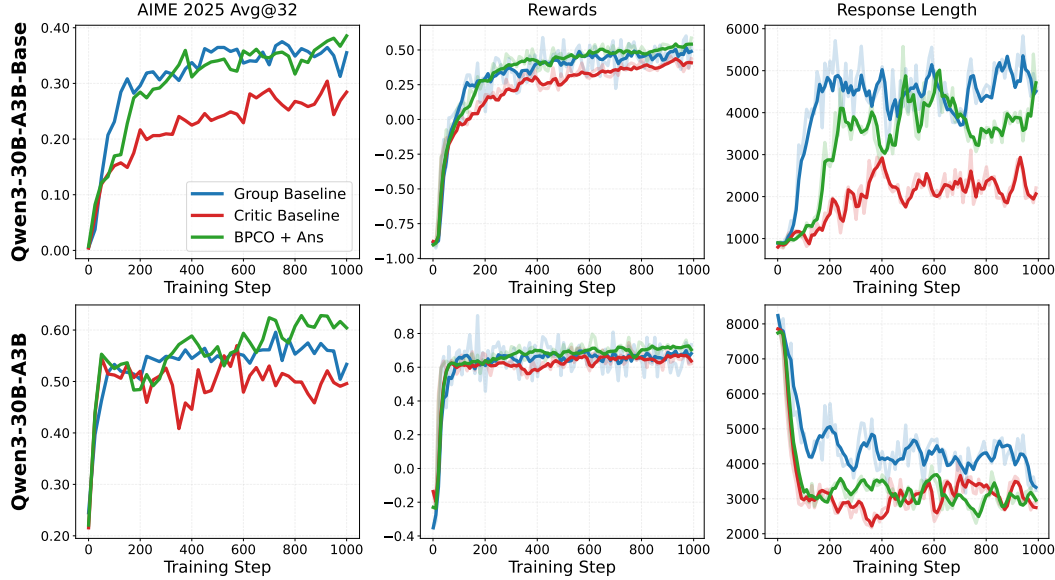


Figure 10: Results with larger MoE models on DAPO-Math-17k dataset.

We further evaluate BPCO with Qwen3-30B-A3B-Base and Qwen3-30B-A3B (Yang et al., 2025) on DAPO-Math-17K (Yu et al., 2026). We set the maximum generation length to 12,000 tokens.

Figure 10 shows that BPCO continues to improve upon the critic baseline at this scale. On Qwen3-30B-A3B, the critic baseline fails to improve AIME 2025 accuracy beyond the first 100 training steps, indicating unstable optimization. In both settings, BPCO achieves substantially higher AIME 2025 accuracy, demonstrating that it learns a more effective critic than the previous recipe.

Compared with the group baseline, BPCO performs better on Qwen3-30B-A3B and comparably on Qwen3-30B-A3B-Base. These results show that BPCO is a strong alternative to the widely adopted group-based method, without requiring group sampling (Xu & Ding, 2026; Hou et al., 2026).

4.3 RUBRIC-BASED REWARDS

We finally consider open-ended prompts evaluated by a rubric-based judge (Gunjal et al., 2026). We initialize both the policy and critic from Qwen3-4B-Base (Yang et al., 2025) and train on Open-Rubrics (Liu et al., 2026). A frozen Qwen3-4B-Instruct-2507 judge scores each generated response against its prompt-specific reference rubric. The policy observes only the prompt. BPCO+Rubrics additionally exposes the same rubric to the training-time critic.

As shown in Figure 11, both BPCO variants learn faster than the group and critic baselines, although the group baseline eventually reaches comparable performance. The critic baseline achieves a slightly lower final reward, likely because of its suboptimal training recipe and low explained variance. Privileged information does not improve performance despite yielding higher explained variance, possibly because the task is relatively simple. Nevertheless, the stronger performance of BPCO without privileged information indicates that bounded value prediction and unnormalized advantages remain beneficial when rewards are provided by a rubric-based judge.

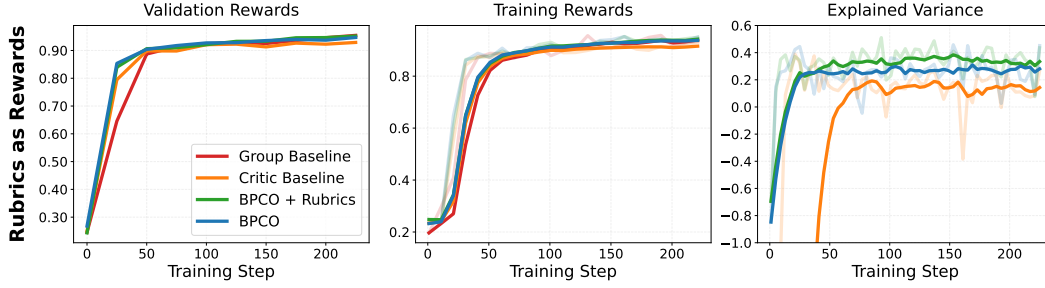


Figure 11: Results under rubric-based rewards.

5 CONCLUSION

We studied why critic-based RL for LLMs can become unstable and assembled the resulting fixes into Best Practice Critic Optimization. BPCO combines DPPO with reward-range-bounded value predictions, unbiased Monte Carlo critic targets, unnormalized policy advantages, and length-adaptive GAE. It can additionally provide the training-only critic with privileged information, such as a reference answer, solution, or grading rubric, while leaving the policy’s inputs unchanged.

Across controlled sanity tests, larger datasets, 1.5B and 30B-A3B models, and rubric-based rewards, BPCO consistently improves a strong critic baseline. It also matches or exceeds group-based optimization while using a single response per prompt. Privileged information further improve critic learning when they provide useful reward context, but their gains are task dependent and can be offset by overfitting in small-data regimes. Overall, these results show that the critic itself is not an inherent weakness of LLM RL. When its output range, target, inputs, and induced policy signal are designed coherently, it provides a stable and efficient alternative to group-based estimation.

Limitations. Evidence is limited to mathematical and rubric rewards. BPCO assumes a known reward range, privileged variants require evaluator information, and critic training adds computation and memory not captured by trajectory-matched comparisons.

REFERENCES

- Christopher Amato. An introduction to centralized training for decentralized execution in cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:2409.03052*, 2024.
- Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. In *International Conference on Learning Representations*, volume 2026, pp. 127924–127945, 2026.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- Zhenyu Hou, Yujiang Li, Jie Tang, and Yuxiao Dong. Single-rollout asynchronous optimization for agentic reinforcement learning. *arXiv preprint arXiv:2607.07508*, 2026.
- Shengyi Huang, Rousslan Fernand Julien Dossa, Antonin Raffin, Anssi Kanervisto, and Weixun Wang. The 37 implementation details of proximal policy optimization. In *ICLR Blog Track*, 2022. URL <https://iclr-blog-track.github.io/2022/03/25/ppo-implementation-details/>. <https://iclr-blog-track.github.io/2022/03/25/ppo-implementation-details/>.
- Tianci Liu, Ran Xu, Tony Yu, Ilgee Hong, Carl Yang, Tuo Zhao, and Haoyu Wang. Openrubrics: Towards scalable synthetic rubric generation for reward modeling and llm alignment. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 17417–17437, 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. *arXiv preprint arXiv:2503.20783*, 2025.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.
- 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.
- Penghui Qi, Zichen Liu, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Optimizing anytime reasoning via budget relative policy optimization. *Advances in Neural Information Processing Systems*, 38:23429–23451, 2026a.
- Penghui Qi, Xiangxin Zhou, Zichen Liu, Tianyu Pang, Chao Du, Min Lin, and Wee Sun Lee. Rethinking the trust region in llm reinforcement learning. *arXiv preprint arXiv:2602.04879*, 2026b.
- 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, 2015a.
- John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015b.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- 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*, pp. 1279–1297, 2025.

- Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.
- Sijun Tan, Michael Luo, Justin Wong, Colin Cai, Xiaoxiang Shi, William Yuan Tang, Manan Roongta, Tianjun Zhang, Li Erran Li, Raluca Ada Popa, and Ion Stoica. Deepscaler: Effective RL scaling of reasoning models via iterative context lengthening, 2026. URL <https://openreview.net/forum?id=I6GzDCne7U>.
- Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *nature*, 575(7782):350–354, 2019.
- Xiangjun Wang, Junxiao Song, Penghui Qi, Peng Peng, Zhenkun Tang, Wei Zhang, Weimin Li, Xiongjun Pi, Jujie He, Chao Gao, et al. Scc: An efficient deep reinforcement learning agent mastering the game of starcraft ii. In *International conference on machine learning*, pp. 10905–10915. PMLR, 2021.
- Zhongwen Xu and Zihan Ding. Single-stream policy optimization. In *International Conference on Learning Representations*, volume 2026, pp. 140925–140944, 2026.
- 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*, 2025.
- Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.
- 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.
- Yu Yue, Yufeng Yuan, Qiyang 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*, 2025.