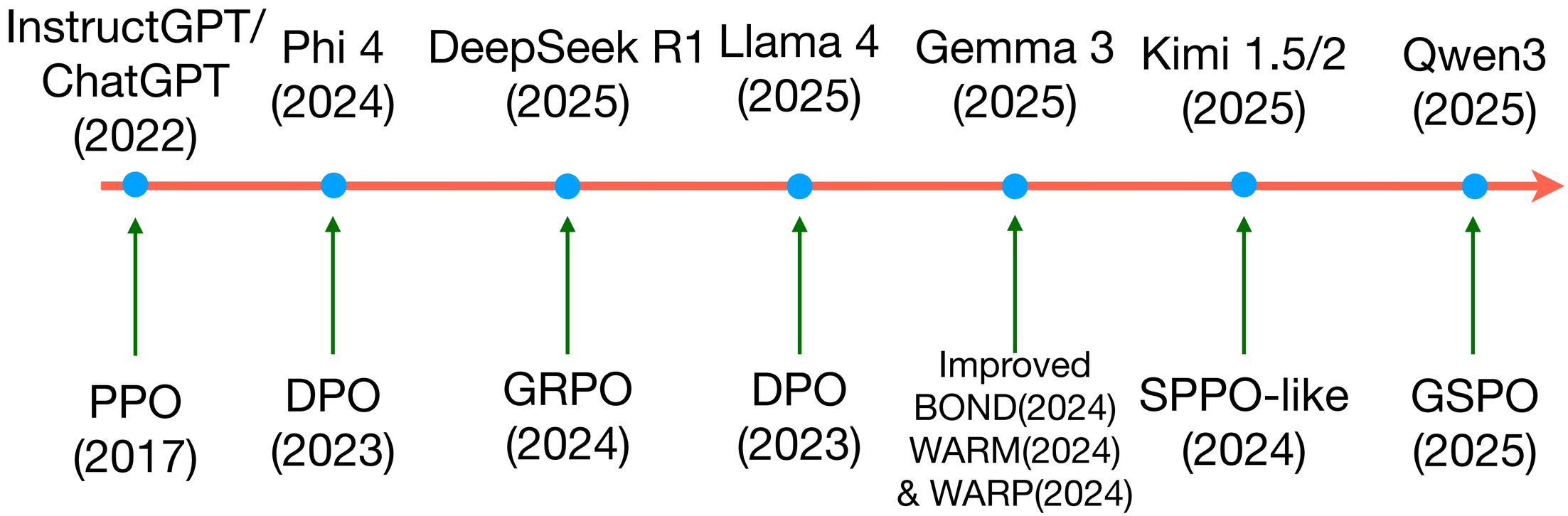
On the Design of KL-Regularized Policy Gradient Algorithms for LLM Reasoning

Yifan Zhang*, Yifeng Liu*, Huizhuo Yuan, Yang Yuan,
Quanquan Gu†, Andrew C. Yao†

April 1, 2026

RLHF in Large Language Model Training



Background

- GRPO(2024):

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim P(Q),\{o_i\}\sim\pi_{\text{old}}}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left(J_{i,t}^{\text{Clip}}(\theta) - \beta\cdot \text{KL}_{\text{est}}\left(\pi_\theta(\cdot\,|\,h_{i,t})\|\pi_{\text{ref}}(\cdot\,|\,h_{i,t})\right)\right)\right]$$

- The derived gradient is a biased estimation of the intended off-policy objective

$$\widehat{KL}_{\text{GRPO-corrected}}(h_{i,t};\theta) = \mathbb{E}_{o_{i,t}\sim\pi_{\text{old}}(\cdot|h_{i,t})}\left[w_{i,t}\; k_3\left(\frac{\pi_{\text{ref}}(o_{i,t}\,|\,h_{i,t})}{\pi_\theta(o_{i,t}\,|\,h_{i,t})}\right)\right]$$

Where $w_{i,t} = \pi_\theta(o_{i,t}\,|\,h_{i,t})/\pi_{\text{old}}(o_{i,t}\,|\,h_{i,t})$, $k_3(y) = y - \log y - 1$

(Consistent with URKL/UKL in RPG framework)

On the Design of KL-Regularized Policy Gradient Algorithms for LLM Reasoning

- We derive policy gradients and corresponding surrogate losses for **Forward/Reverse KL**, in **normalized (KL) and unnormalized (UKL)** forms, under off-policy sampling with importance weights.
- We give both fully differentiable surrogates and **REINFORCE-style** losses (with stop-gradient) and prove their gradient-equivalence to the intended regularized objective.
- We introduce **RPG-Style Clip**, a truncated-importance REINFORCE estimator (PPO-Clip–like) that substantially improves stability and variance control while preserving the RPG gradients.

- We reveal the equality between the k_3 estimator and unnormalized KL, and show that GRPO's KL penalty omits an essential importance weight under off-policy sampling. We provide a **corrected estimator** and loss consistent with the intended objective.
- We present an iterative training framework that periodically updates the reference model to satisfy KL constraints while allowing the policy to depart meaningfully from the initial checkpoint.
- On math reasoning, RPG-REINFORCE (with RPG-Style Clip) yields stable and scalable training and outperforms DAPO by up to **+6** absolute points on AIME24/25.

Preliminaries

- Policy gradient (PG) methods are a cornerstone of modern reinforcement learning (RL), optimizing parameterized policies π_θ by estimating the gradient of an expected objective function $J(\theta)$ with respect to the policy parameters θ.

- Typically, $J(\theta)$ represents the expected cumulative discounted reward over trajectories $\tau = (s_0, a_0, r_0, s_1, \ldots, s_T, a_T, r_T)$ generated by the policy:

$$J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta}[G(\tau)]$$

where $G(\tau) = \sum_{t=0}^{T} \gamma^t r_t$ is the trajectory return (with discount factor γ), and the expectation is taken over the trajectories sampled according to the policy $\pi_\theta(a \mid s)$ and the environment dynamics $p(s' \mid s, a)$.

Preliminaries

- Generalized Policy Gradient Theorem
  - Let $f(x, \theta)$ be a scalar-valued function associated with x, potentially depending on θ. Under suitable regularity conditions, the gradient of the expectation $\mathbb{E}_{x\sim\pi_\theta}[f(x, \theta)]$ with respect to θ is:

$$\nabla_\theta \mathbb{E}_{x\sim\pi_\theta}[f(x,\theta)] = \mathbb{E}_{x\sim\pi_\theta}\left[f(x,\theta)\,\nabla_\theta \log \pi_\theta(x) + \nabla_\theta f(x,\theta)\right]$$

- REINFORCE algorithm (1992)
  - It applies $J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}[G(\tau)]$. In this case, $f(\tau, \theta) = G(\tau)$:

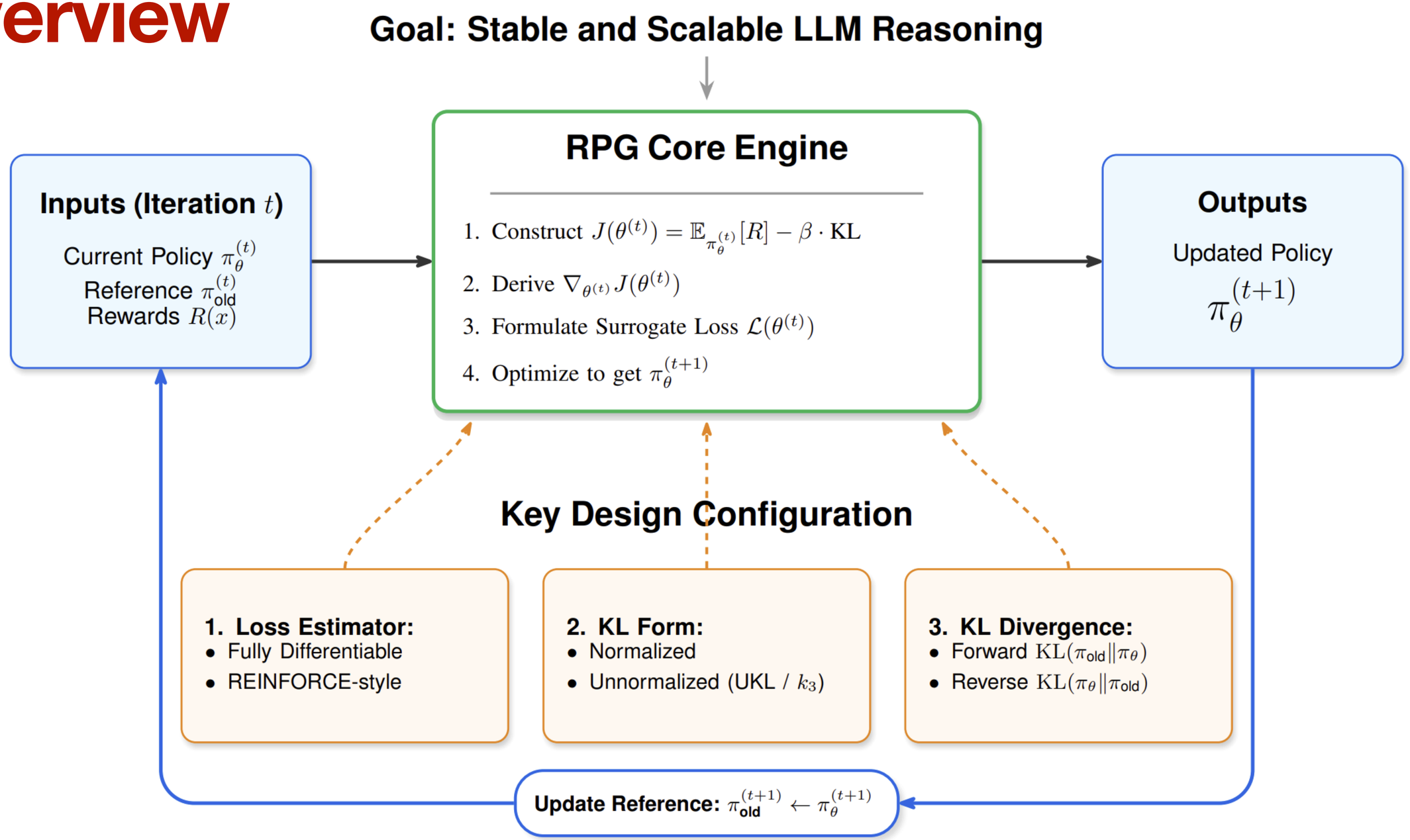
$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} G_t \nabla_\theta \log \pi_\theta(a_t \mid s_t)\right]$$

where $G_t = \sum_{k=t}^{T} \gamma^{k-t} r_k$ is the return-to-go from timestep t.

Background

- KL Regularization in Policy Gradient
  - Asymmetricity: $KL(P \,||\, Q) \neq KL(Q \,||\, P)$.
  - Minimizing the forward KL $KL(\pi_{\text{ref}} \,||\, \pi_\theta)$ encourages π_θ to cover the support of π_{ref} (zero-forcing)
  - Minimizing the reverse KL $KL(\pi_\theta \,||\, \pi_{\text{ref}})$ encourages π_θ to be concentrated where π_{ref} has high probability mass (mode-seeking).
  - Adding a KL penalty to the RL objective, such as $J(\theta) = \mathbb{E}_{\pi_\theta}[R] - \beta KL(\pi_\theta || \pi_{\text{ref}})$:
    - Helps control the policy update size
    - Prevents large deviations from π_{ref}
    - Encourages exploration near known good policies
    - Mitigate issues like catastrophic forgetting or overly confident outputs

Overview


Goal: Stable and Scalable LLM Reasoning
RPG Core Engine
1. Construct $J(\theta^{(t)}) = \mathbb{E}_{\pi_\theta^{(t)}}[R] - \beta \cdot \mathrm{KL}$
2. Derive $\nabla_{\theta^{(t)}} J(\theta^{(t)})$
3. Formulate Surrogate Loss $\mathcal{L}(\theta^{(t)})$
4. Optimize to get $\pi_\theta^{(t+1)}$
Inputs (Iteration t)
Current Policy $\pi_\theta^{(t)}$
Reference $\pi_{\text{old}}^{(t)}$
Rewards $R(x)$
Outputs
Updated Policy
$\pi_\theta^{(t+1)}$
Key Design Configuration
1. Loss Estimator:
• Fully Differentiable
• REINFORCE-style
2. KL Form:
• Normalized
• Unnormalized (UKL / k_3)
3. KL Divergence:
• Forward $\mathrm{KL}(\pi_{\text{old}} \| \pi_\theta)$
• Reverse $\mathrm{KL}(\pi_\theta \| \pi_{\text{old}})$
Update Reference: $\pi_{\text{old}}^{(t+1)} \leftarrow \pi_\theta^{(t+1)}$

Forward KL Regularization

- Consider the objective function with forward KL regularization:

$$J_{\mathrm{FKL}}(\theta) = \mathbb{E}_{x\sim\pi_\theta}[R(x)] - \beta KL(\pi_{\mathrm{old}} \parallel \pi_\theta)$$

- The gradient of $J_{\mathrm{FKL}}(\theta)$ with respect to θ is:

$$\nabla_\theta J_{\mathrm{FKL}}(\theta) = \mathbb{E}_{x\sim\pi_{\mathrm{old}}}\Big[\big(w(x)R(x) + \beta\big)\,\nabla_\theta \log \pi_\theta(x)\Big]$$

- A corresponding surrogate loss is:

$$\mathscr{L}_{\mathrm{FKL}}(\theta) = \mathbb{E}_{x\sim\pi_{\mathrm{old}}}\big[-w(x)R(x) - \beta \log \pi_\theta(x)\big]$$

- which satisfies $\nabla_\theta \mathscr{L}_{\mathrm{FKL}}(\theta) = -\nabla_\theta J_{\mathrm{FKL}}(\theta)$.

Unnormalized Forward KL Regularization

- In scenarios where distributions might not be normalized (i.e., $\int_x \pi(x)dx \neq 1$), the standard KL divergence might not fully capture the dissimilarity. The unnormalized forward KL divergence addresses this by adding a mass correction term.

- Let $\pi_{\text{old}}(x)$ be a potentially unnormalized reference measure with total mass $Z_{\text{old}} = \int_x \pi_{\text{old}}(x)dx$. Let $\tilde{\pi}_{\text{old}}(x) = \pi_{\text{old}}(x)/Z_{\text{old}}$ be the corresponding normalized probability distribution, such that $\int \tilde{\pi}_{\text{old}}(x)dx = 1.$

Unnormalized Forward KL Regularization

- The unnormalized forward KL divergence is:

$$UKL(\pi_{\text{old}}\|\pi_\theta) = \underbrace{\int_x \pi_{\text{old}}(x)\log\frac{\pi_{\text{old}}(x)}{\pi_\theta(x)}dx}_{\text{Generalized KL}} + \underbrace{\int_x \left(\pi_\theta(x) - \pi_{\text{old}}(x)\right)\ dx}_{\text{Mass Correction}}$$

- Consider the objective using UKL regularization as follows:

$$J_{\text{UFKL}}(\theta) = \mathbb{E}_{x\sim\pi_\theta}[R(x)] - \beta\ UKL(\pi_{\text{old}}\|\pi_\theta)$$

- The corresponding surrogate loss for gradient descent optimization:

$$\mathscr{L}_{\text{UFKL}}(\theta) = Z_{\text{old}}\mathbb{E}_{x\sim\tilde{\pi}_{\text{old}}}\left[-w(x)R(x) + \beta\left(w(x) - \log w(x) - 1\right)\right]$$

Unnormalized Reverse KL Regularization

- The objective of URKL:

$$J_{\mathrm{URKL}}(\theta) = \mathbb{E}_{x\sim\pi_\theta}[R(x)] - \beta\ UKL(\pi_\theta \| \pi_{\mathrm{old}})$$

- A corresponding surrogate loss for URKL:

$$\mathscr{L}_{\mathrm{URKL}}(\theta) = Z_{\mathrm{old}}\mathbb{E}_{x\sim\tilde{\pi}_{\mathrm{old}}}\left[-w(x)R(x) + \beta\Big(w(x)\log w(x) - w(x)\Big)\right]$$

  satisfying $\nabla_\theta \mathscr{L}_{\mathrm{URKL}}(\theta) = -\nabla_\theta J_{\mathrm{URKL}}(\theta)$

- Remark: the k_3 estimator for KL divergence is equivalent to Unnormalized KL Regularization.

Regularized Policy Gradients

Table 1: Summary of fully differentiable surrogate loss functions $\mathcal{L}(\theta)$ for unnormalized KL-regularized objectives (main text). Minimizing $\mathcal{L}(\theta)$ corresponds to maximizing $J(\theta) = \mathbb{E}_{\pi_\theta}[R(x)] - \beta \cdot \text{Divergence}$. Samples x are drawn from $\widetilde{\pi}_{\text{old}} = \pi_{\text{old}}/Z_{\text{old}}$. These losses yield $-\nabla_\theta J(\theta)$ via differentiation. Notation: $w(x) = \pi_\theta(x)/\pi_{\text{old}}(x)$, $R(x)$ is reward, β the regularization strength, and $Z_{\text{old}} = \int \pi_{\text{old}}$. Normalized counterparts are in Appendix D (Table 4).

Regularization (Unnormalized)	**Surrogate loss (expectation w.r.t. $\widetilde{\pi}_{\text{old}}$)**
Forward (UFKL)	$Z_{\text{old}}\,\mathbb{E}\big[-w(x)R(x) + \beta\big(w(x) - \log w(x) - 1\big)\big]$
Reverse (URKL)	$Z_{\text{old}}\,\mathbb{E}\big[-w(x)R(x) + \beta\big(w(x)\log w(x) - w(x)\big)\big]$

REINFORCE-Style Regularized Policy Gradients

- Notice that the gradients derived previously share a structural similarity with the REINFORCE estimator:

$$\nabla_\theta J(\theta) = \mathbb{E}_{x\sim\pi_{\text{sampling}}}\left[\text{Weight}(x,\theta)\,\nabla_\theta \log \pi_\theta(x)\right]$$

- REINFORCE-style approach (SG: Stop-Gradient):

$$\mathscr{L}_{\text{REINFORCE-style}}(\theta) = -\,\mathbb{E}_{x\sim\pi_{\text{sampling}}}\left[\text{SG}\left(\text{Weight}(x,\theta)\right)\log \pi_\theta(x)\right]$$

$$\nabla_\theta \mathscr{L}_{\text{REINFORCE-style}}(\theta) \overset{\text{Autodiff}}{=} -\,\mathbb{E}_{x\sim\pi_{\text{sampling}}}\left[\text{SG}\left(\text{Weight}(x,\theta)\right)\nabla_\theta \log \pi_\theta(x)\right]$$

REINFORCE-Style Regularized Policy Gradients

Table 2: REINFORCE-style surrogate losses $\mathcal{L}(\theta)$ for unnormalized KL-regularized objectives using the stop-gradient operator (SG). These losses yield the target gradient via automatic differentiation. Compare with the fully differentiable losses in Table 1. Normalized versions are given in Appendix E.

Regularization (Unnormalized)	**REINFORCE-style loss (sampling $x \sim \widetilde{\pi}_{\text{old}}$)**
Forward (UFKL)	$-\mathbb{E}\Big[\,\mathrm{SG}(Z_{\text{old}}(w(x)R(x) - \beta(w(x)-1)))\log \pi_\theta(x)\Big]$
Reverse (URKL)	$-\mathbb{E}\Big[\,\mathrm{SG}(Z_{\text{old}}w(x)(R(x) - \beta \log w(x)))\log \pi_\theta(x)\Big]$

RPG-Style Clip: dual-clip truncation of importance ratios

- Large importance ratios $w(x) = \dfrac{\pi_\theta(x)}{\pi_{\text{old}}(x)}$ induce high variance and destabilize off-policy updates.
- **RPG-Style Clip** follows the dual-clip method:
  - we clip w into $[1-\epsilon_1, 1+\epsilon_2]$ and additionally impose a lower bound for negative advantages. Let $\hat{A}(x;\theta)$ denote the regularized advantage analogue determined by the chosen objective (e.g., $\hat{A}_{\text{URKL}} = (R-b) - \beta \log w$, $\hat{A}_{\text{RKL}} = (R-b) - \beta(\log w + 1)$). The loss used in our implementation is

$$\mathcal{L}^{\text{RPG-Clip}}(x,\theta) = \begin{cases} \max\Big(-w(x)\,\hat{A}(x;\theta),\ -\text{clip}(w(x), 1-\epsilon_1, 1+\epsilon_2)\,\hat{A}(x;\theta)\Big), & \hat{A}(x;\theta) \geq 0, \\ [0.5ex]\ \min\Big(\max\big(-w(x)\,\hat{A}(x;\theta),\ -\text{clip}(w(x), 1-\epsilon_1, 1+\epsilon_2)\,\hat{A}(x;\theta)\big),\ -c\,\hat{A}(x;\theta)\Big), & \hat{A}(x;\theta) < 0, \end{cases}$$

RPG-Style Clip: dual-clip truncation of importance ratios

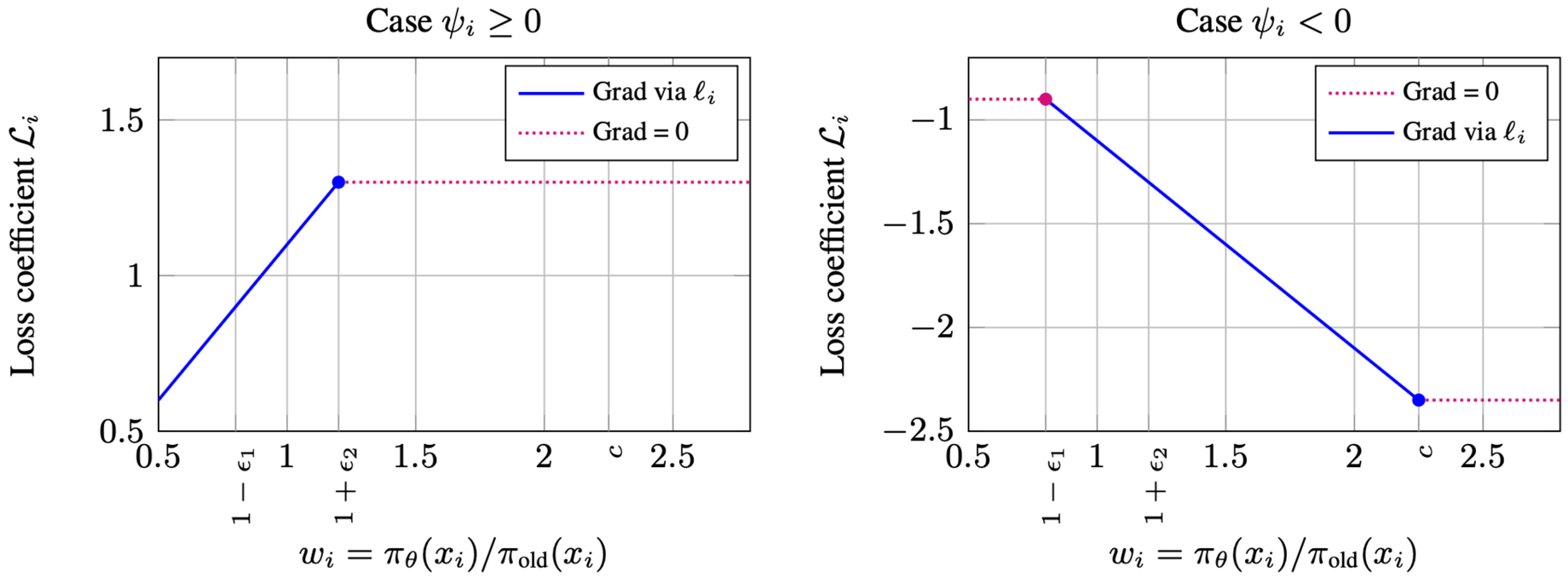


Figure 3: Visualization of the loss coefficient $\mathcal{L}_i$ vs. importance weight w_i based on the specific implementation in Algorithm 2. This version swaps the main branching condition compared to previous versions (branches on $\psi_i > 0$). The plot assumes $\ell_i = -\log \pi_\theta(x_i) = 1$ for visualizing the value of $\mathcal{L}_i$. The line styles indicate the nature of the gradient $\nabla_\theta \mathcal{L}_i$: **Solid blue:** Gradient exists, flowing only via ℓ_i. The coefficient multiplying $\nabla_\theta \ell_i$ depends on $\text{SG}(w_i)$. **Dotted magenta:** Gradient is zero. This occurs when ℓ_i is detached via SG in the loss calculation. Left: Case $\psi_i \geq 0$. Right: Case $\psi_i < 0$.

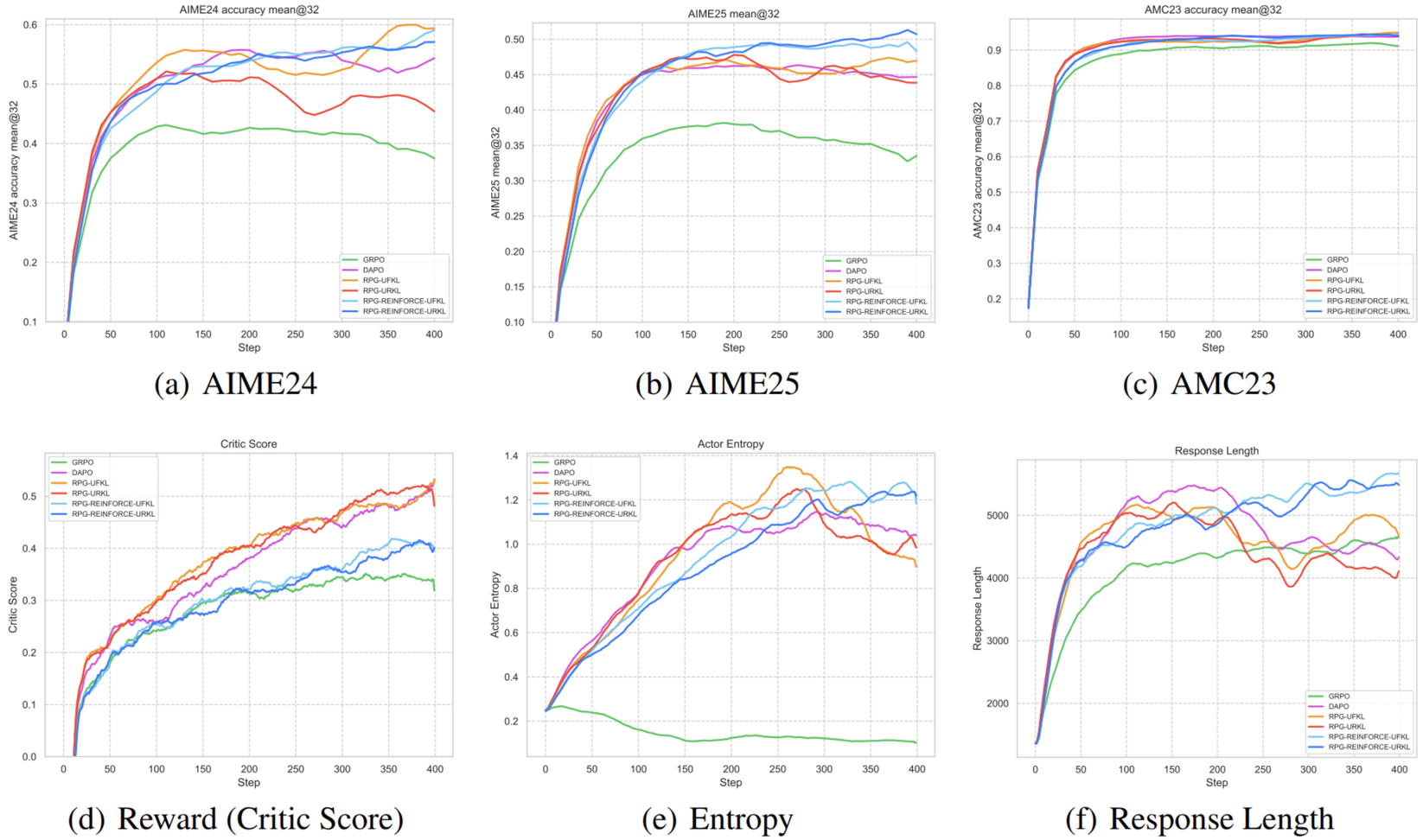
RPG-Style Clip: dual-clip truncation of importance ratios

Algorithm 1 RPG with Dual-Clip Stabilization

Require: Reference policy π_{old}, Reward function $R(x)$, Initial policy parameters θ_0
Require: Base objective structure J_{chosen} (implies regularization type), Regularization strength $\beta \geq 0$
Require: Learning rate $\alpha > 0$, Batch size $N > 0$, Number of epochs $K \geq 1$ per iteration
Require: Dual Clip parameters: $\epsilon_1 > 0, \epsilon_2 > 0, c > 1$
Require: Baseline method (e.g., batch/group average, value function V_ϕ)

Initialize policy parameters $\theta \leftarrow \theta_0$
Initialize value function parameters ϕ (if baseline uses V_ϕ)
for each training iteration **do**
  Sample batch $\mathcal{D} = \{x_i\}_{i=1}^{N} \sim \pi_{\text{old}}$ ▷ Collect data using old policy
  Compute R_i for $i = 1..N$
  Compute baselines b_i for $i = 1..N$ (e.g., $b_i = \frac{1}{N}\sum_j R_j$ or $b_i = V_\phi(x_i)$)
  for $k = 1$ to K **do** ▷ Multiple optimization epochs on the same batch
    Initialize batch loss $\mathcal{L}_{\text{batch}} = 0$
    for $i = 1$ to N **do**
      $w_i = \frac{\pi_\theta(x_i)}{\pi_{\text{old}}(x_i)}$, $\log w_i = \log \pi_\theta(x_i) - \log \pi_{\text{old}}(x_i)$ ▷ Compute importance weight
      Define Advantage analogue $\widehat{A}_i$ based on J_{chosen}, R_i, b_i, w_i, β.
        ▷ Ex: For RKL, $\widehat{A}_i = (R_i - b_i) - \beta(\log w_i + 1)$. Note: $\widehat{A}_i$ depends on current θ via w_i
      if Dual Clip enabled **then**
        $\text{loss\_term1}_i = -w_i \times \widehat{A}_i$ ▷ Negative of unclipped term, gradient flows through w_i
        $w_{i,\text{clipped}} = \text{clip}(w_i, 1 - \epsilon_1, 1 + \epsilon_2)$
        $\text{loss\_term2}_i = -w_{i,\text{clipped}} \times \widehat{A}_i$ ▷ Negative of clipped term
        $L_{\text{clip}}(i) = \max(\text{loss\_term1}_i, \text{loss\_term2}_i)$
        if $\widehat{A}_i \geq 0$ **then**
          $\mathcal{L}_{\text{term}}(i) = L_{\text{clip}}(i)$
        else ▷ $\widehat{A}_i < 0$
          $\text{loss\_lower\_bound}_i = -c \times \widehat{A}_i$ ▷ Lower bound term
          $\mathcal{L}_{\text{term}}(i) = \min(L_{\text{clip}}(i), \text{loss\_lower\_bound}_i)$
        end if
      else
        ▷ Define base loss term (unclipped) based on chosen objective's negative gradient structure
        ▷ Ex: For RKL loss (no clip): $\mathcal{L}_{\text{term}}(i) = w_i(-(R_i - b_i) + \beta \log w_i)$
        $\mathcal{L}_{\text{term}}(i) = -w_i \times \widehat{A}_i$
      end if
      $\mathcal{L}_{\text{batch}} = \mathcal{L}_{\text{batch}} + \mathcal{L}_{\text{term}}(i)$
    end for
    $\widehat{\mathcal{L}}(\theta) = \frac{1}{N}\mathcal{L}_{\text{batch}}$ ▷ Compute final batch loss for minimization
    $g \leftarrow \nabla_\theta \widehat{\mathcal{L}}(\theta)$ ▷ Compute gradient (flows through w_i and $\widehat{A}_i$)
    $\theta \leftarrow \text{OptimizerUpdate}(\theta, g, \alpha)$ ▷ Update policy parameters
    if using a learned baseline V_ϕ **then**
      Update value function parameters ϕ (e.g., by minimizing $\mathbb{E}[(V_\phi(x_i) - R_i)^2]$ over the batch)
    end if
  end for
end for
return Optimized policy parameters θ

Experiments on Qwen3-4B

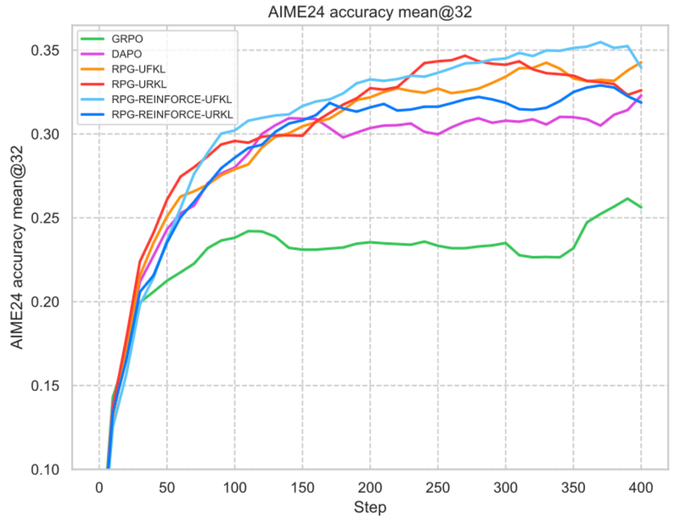


(a) AIME24 (b) AIME25 (c) AMC23

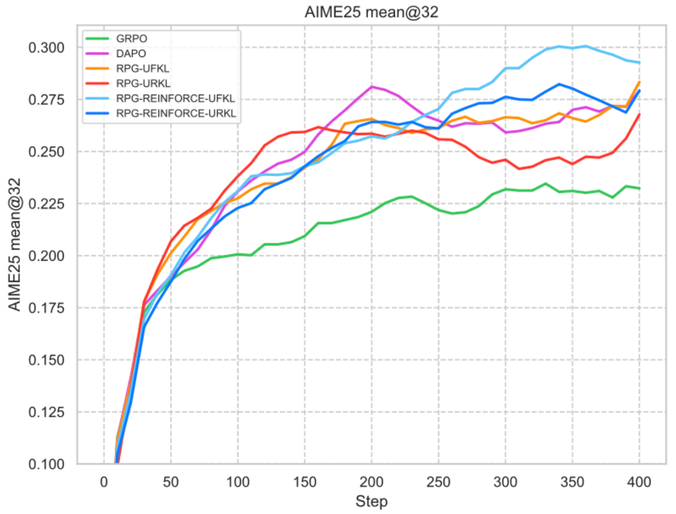
(d) Reward (Critic Score) (e) Entropy (f) Response Length

Figure 4: Training dynamics and benchmark performance for fully differentiable Regularized Policy Gradient (RPG) and REINFORCE-Style RPG (RPG-REINFORCE) compared to baselines (GRPO and DAPO) with 8k context length.

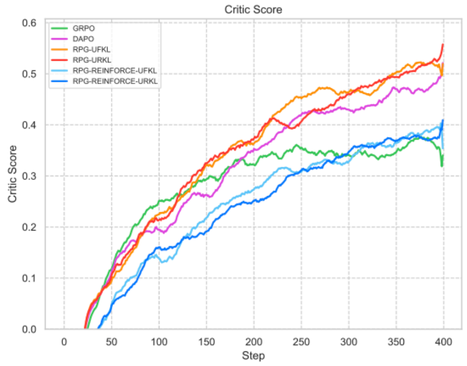
Experiments on Qwen3-4B



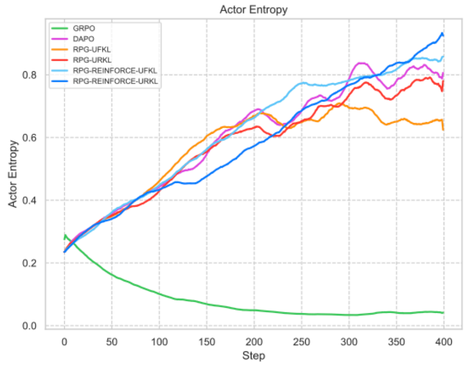
(a) AIME24



(b) AIME25



(c) Reward (Critic Score)

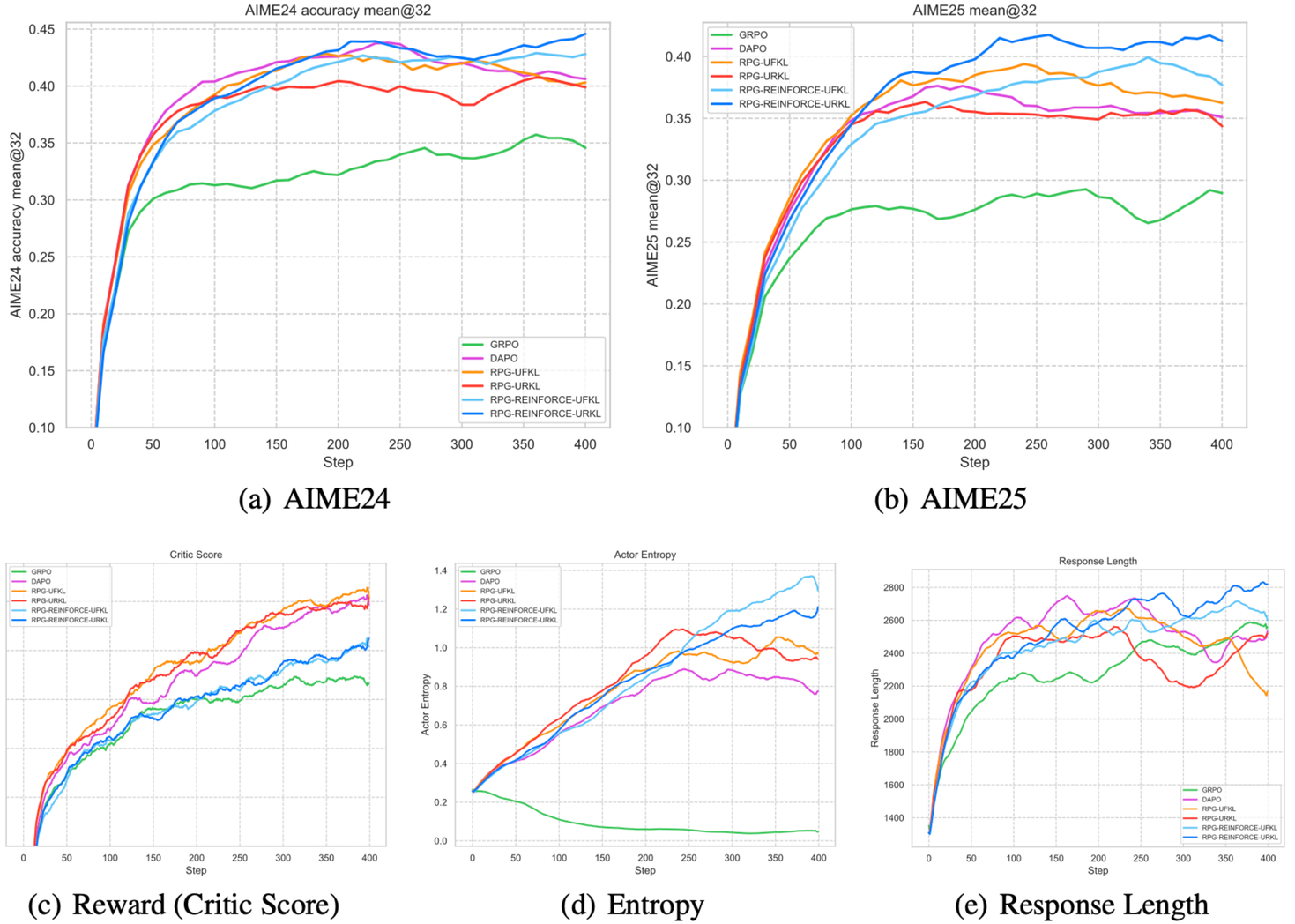


(d) Entropy

(e) Response Length

Figure 5: Training dynamics and benchmark performance for fully differentiable Regularized Policy Gradient (RPG) and REINFORCE-Style RPG (RPG-REINFORCE) compared to baselines (GRPO and DAPO) with 2k context length.

Experiments on Qwen3-4B



(a) AIME24 (b) AIME25

(c) Reward (Critic Score) (d) Entropy (e) Response Length

Figure 2: Performance of RPG and REINFORCE-Style Regularized Policy Gradient (RPG-REINFORCE) methods compared to baselines with 4k context length.

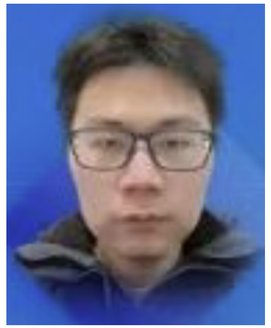
Experiments on Qwen3-4B

2K context length	AIME24		AIME25	
	Last	Best	Last	Best
GRPO	0.2563	0.2708	0.2323	0.2479
DAPO	0.3229	0.3281	0.2792	0.2844
RPG-UFKL	**0.3427**	0.3479	<u>0.2833</u>	0.2833
RPG-URKL	0.3260	<u>0.3594</u>	0.2677	0.2677
RPG-REINFORCE-UFKL	<u>0.3396</u>	**0.3625**	**0.2927**	**0.3083**
RPG-REINFORCE-URKL	0.3188	0.3417	0.2792	<u>0.2938</u>
4K context length	**Last**	**Best**	**Last**	**Best**
GRPO	0.3458	0.3677	0.2896	0.3042
DAPO	0.4063	<u>0.4479</u>	0.3510	0.3938
RPG-UFKL	0.4031	0.4396	0.3625	0.3979
RPG-URKL	0.3990	0.4219	0.3438	0.3792
RPG-REINFORCE-UFKL	<u>0.4281</u>	0.4375	<u>0.3771</u>	<u>0.4042</u>
RPG-REINFORCE-URKL	**0.4458**	**0.4531**	**0.4125**	**0.4313**

Takeaways

- We provided derivations for policy gradients and surrogate loss functions covering forward/reverse KL, normalized/ unnormalized distributions, and both fully differentiable and REINFORCE-style estimators.
- Beyond derivations, we revisited the classical REINFORCE algorithm and made it viable off-policy through RPG-Style Clip and iterative reference updates.
- On LLM reasoning, these design choices deliver stable and scalable training with competitive and superior accuracy relative to strong baselines.

Acknowledgement

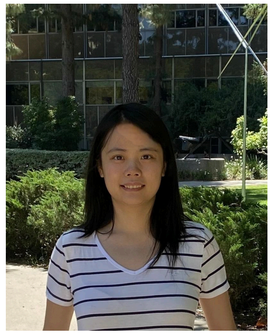


Yifan Zhang

Princeton

Yifeng Liu

UCLA



Huizhuo Yuan

UCLA

Yang Yuan

Tsinghua University

Quanquan Gu

UCLA

Andrew C. Yao

Tsinghua University

Paper and Code

- Paper:

https://arxiv.org/pdf/2505.17508

- Code:

https://github.com/complex-reasoning/RPG

Thank you!