WHEN MISMATCH MISLEADS: STABILIZING LOW-PRECISION REINFORCEMENT LEARNING VIA RELIABLE CORRECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning (RL) has become a key component of large language model post-training, but its substantial computational cost increasingly motivates end-to-end low-precision training. Despite improving training efficiency, it also introduces numerical perturbations that amplify train-inference mismatch and create a tradeoff between efficiency and stability. Existing approaches primarily stabilize low-precision RL by aligning rollout and actor policies. However, actor-rollout alignment alone does not guarantee improvement of the underlying full-precision objective, since the observed mismatch conflates genuine policy mismatch with numerical noise. We therefore formulate low-precision RL as a three-policy alignment problem, where noisy rollout and actor policies provide observable training signals while the inaccessible full-precision policy defines the learning problem. Based on this formulation, we propose **ERIS (Excess-Risk-guided Importance Sampling)**, an excess-risk-guided method that corrects high-reliability mismatch tokens while retaining identity weighting for noise-dominated tokens. Across multiple Qwen model scales and architectures, ERIS consistently improves FP8 training stability and performance. ERIS improves the average mathematical reasoning score over standard FP8 by 14.29% on Qwen3-8B, while achieving performance comparable to BF16. Mechanism analysis further confirms the advantage of three-policy alignment in preserving consistency with the full-precision policy. Meanwhile, ERIS preserves the efficiency benefits of FP8, achieving up to a 1.17× end-to-end throughput improvement over BF16.

1 INTRODUCTION

Reinforcement learning (RL) has become an essential part of modern LLM post-training, driving substantial advances in reasoning capabilities (Shao et al., 2024; Guo et al., 2025). However, these gains come at a substantial computational cost. RL repeatedly alternates between rollout generation and actor updates, making it increasingly expensive as model scale and response length grow. This cost has motivated a shift toward low-precision RL. Early efforts primarily quantize rollout generation to accelerate sampling (Team et al., 2025; Zhou et al., 2026; Gu et al., 2026), while more recent works extend low precision to actor updates, enabling end-to-end low-precision RL (Qiu et al., 2026; Xi et al., 2026; Qwen Team, 2026; Xiaomi, 2026). The efficiency gain, however, comes with a stability risk. Low-precision execution noise affects both the sampling distribution and the optimization signal, amplifying train-inference mismatch and, in se-

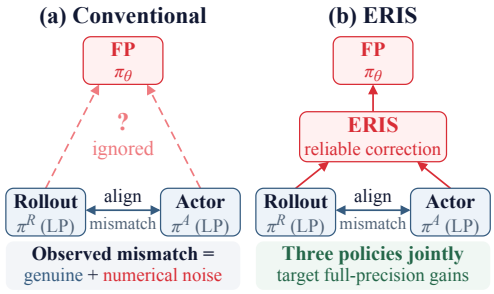


Figure 1: Two-policy vs. three-policy alignment. Conventional methods align low-precision rollout and actor policies, while ERIS additionally corrects toward the full-precision policy. FP denotes Full-Precision; LP denotes Low-Precision.

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.

vere cases, causing training instabilities (Liu et al., 2025b; Zhuge et al., 2026). Low-precision RL therefore exposes a fundamental tradeoff between computation efficiency and optimization stability.

Existing efforts mitigate this instability from several complementary directions. Importance-sampling methods correct the discrepancy between rollout and actor policies by reweighting sampled tokens (Metelli et al., 2018; Liu et al., 2025b; Zhou et al., 2026). Quantization-aware training methods instead modify the optimization procedure to accommodate quantized executions (Huang et al., 2026; Gu et al., 2026; Li et al., 2026b). System-level approaches further reduce numerical discrepancies by unifying precision formats, execution paths, or training and rollout backends (Qiu et al., 2026; Xi et al., 2026; Zhong et al., 2026). Despite their different mechanisms, these approaches share a common principle: stabilizing low-precision RL by reducing actor-rollout mismatch. Although effective for aligning rollout and actor policies, such alignment is defined only by the low-precision policies, leaving the full-precision policy outside the objective. Consequently, a smaller actor-rollout mismatch does not necessarily imply better full-precision optimization, and a performance gap can remain. This suggests that actor-rollout alignment alone is insufficient.

Motivated by this limitation, we formulate low-precision RL as a three-policy problem that jointly considers the rollout, actor and underlying full-precision policies. As illustrated in Figure 1, conventional methods focus only on aligning the rollout and actor policies, whereas our formulation explicitly accounts for the full-precision policy that defines the learning objective. This perspective reveals a key limitation of existing approaches: reducing actor-rollout mismatch can improve stability, but does not necessarily produce beneficial updates for full-precision objective improvement. Building on this formulation, we introduce **Excess-Risk-guided Importance Sampling (ERIS)**, an excess-risk-guided correction method for stable low-precision RL. ERIS applies correction only when its estimated benefit to full-precision objective outweighs the error introduced by low-precision noise, while leaving other tokens uncorrected to avoid unreliable correction. In this way, ERIS retains the benefits of actor-rollout alignment while enabling reliable full-precision objective improvement.

Our contributions are summarized as follows:

- We formulate low-precision RL as a three-policy alignment problem that explicitly accounts for the underlying full-precision objective improvement (Sec. 4.1).
- We propose **ERIS**, an excess-risk-guided correction method that accounts for both actor-rollout mismatch and low-precision numerical noise (Sec. 4.2).
- We evaluate ERIS across different model scales and architectures on a broad range of benchmarks, demonstrating improved training stability and performance (Sec. 5.2).
- Extensive mechanism analyses characterize how ERIS operates, while comprehensive ablations validate the necessity of its core components (Sec. 5.3 & Sec. 5.4).

2 Related Works

Low-Precision RL Training. Low-precision training has emerged as an important direction for improving the efficiency of RL. Early work primarily focuses on quantized rollout: FlashRL (Liu et al., 2025b) and QuRL (Li et al., 2026b) accelerate rollout generation with low-precision inference. More recent work extends low-precision execution throughout the full RL pipeline, with Qwen3.5, MiMo-V2-Flash, Jet-RL and FP8-RL adopting end-to-end FP8 computation for both actor training and rollout generation (Qwen Team, 2026; Xiaomi, 2026; Xi et al., 2026; Qiu et al., 2026). These works reveal an *efficiency-performance tradeoff*: despite gains in throughput and memory efficiency, numerical discrepancies between training and rollout can introduce or amplify train-inference mismatch (TIM), raising concerns about optimization stability.

Stabilizing Low-Precision RL. Existing efforts to stabilize low-precision RL largely focus on reducing the mismatch between rollout and actor policies. At the algorithmic level, FlashRL (Liu et al., 2025b) applies token-level truncated importance sampling (TIS) to correct actor-rollout mismatch, while Liu et al. (2025a) introduces sequence-level masked importance sampling (MIS) to address trajectory-level distribution shift. AIS (Zhou et al., 2026) further adapts batch-level importance correction for improved stability. System-level approaches instead reduce mismatch at its numerical source. Precision-RL (Qi et al., 2025) aligns training and inference precision to FP16 instead of BF16, while Jet-RL (Xi et al., 2026) and FP8-RL (Qiu et al., 2026) unify both under FP8. However, these methods primarily target actor-rollout alignment, while overlooking deviations of low-precision updates from the underlying full-precision objective.

3 PRELIMINARIES

Ideal full-precision RL. Let $q \sim \mathcal{D}$ denote a query sampled from the training dataset and π_θ the full-precision policy. In ideal full-precision RL, the same policy π_θ serves both rollout and actor: it samples responses and provides optimization gradients. For a group of responses $\{o_i\}_{i=1}^{G} \sim \pi_\theta(\cdot \mid q)$ with $o_i = (o_{i,1}, \ldots, o_{i,T})$, full-precision RL optimizes $J(\theta) = \mathbb{E}_{q\sim\mathcal{D},\, \{o_i\}_{i=1}^{G}\sim\pi_\theta(\cdot|q)}[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{T}\sum_{t=1}^{T} L^{\mathrm{CLIP}}(o_{i,t} \mid q, o_{i,<t}; \theta)]$, where L^{CLIP} denotes the token-level PPO clipped surrogate objective (Schulman et al., 2017). Since rollout and actor share the same π_θ, training is inherently on-policy and free from mismatch.

Practical low-precision RL. In practical RL training, rollouts and actor updates often run on different engines, yielding distinct realized policies. Low-precision computation further introduces numerical noise, causing rollout policy π_θ^{R} and actor policy π_θ^{A} to deviate from the full-precision policy π_θ. Prior work mitigates the actor-rollout mismatch via importance sampling (IS) (Liu et al., 2025b). Defining $\rho_{i,t} = \pi_\theta^{\mathrm{A}}(o_{i,t} \mid q, o_{i,<t}) / \pi_\theta^{\mathrm{R}}(o_{i,t} \mid q, o_{i,<t})$ yields $J_{\mathrm{IS}}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\, \{o_i\}_{i=1}^{G}\sim\pi_\theta^{\mathrm{R}}(\cdot|q)}[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{T}\sum_{t=1}^{T} \rho_{i,t} L_{i,t}^{\mathrm{CLIP}}]$, which corrects the observed actor-rollout mismatch. However, it only aligns π_θ^{A} and π_θ^{R}, leaving their mismatch with π_θ unresolved.

Learning problem. We therefore aim to mitigate the mismatch with the underlying full-precision policy under low-precision training. In the following token-wise analysis, the response index i is omitted for simplicity. Let $g_t = \nabla_\theta L^{\mathrm{CLIP}}(o_t \mid q, o_{<t}; \theta)$ denote the token-wise gradient. For each update, a weighting rule assigns non-negative token weights $\mathbf{w} = (w_1, w_2, \ldots, w_T)$, yielding the direction $d(\mathbf{w}) = \frac{1}{T}\sum_t w_t g_t$ and the update $\theta^+(\mathbf{w}) = \theta + \eta d(\mathbf{w})$. The learning problem is

$$\max_{\mathbf{w}(\cdot)} \mathbb{E}\big[J\big(\theta^+(\mathbf{w})\big) - J(\theta)\big] \quad \text{s.t.} \quad \mathbf{w}(\cdot) \geq \mathbf{0},\, \mathbb{E}\left[\|d(\mathbf{w})\|^2\right] < \infty. \tag{1}$$

The weighting rule $\mathbf{w}(\cdot)$ depends on the observed low-precision policies, while improvement is evaluated by the full-precision objective over queries, responses, and numerical noise. Accordingly, the goal is to infer updates from low-precision observations that improve this inaccessible objective.

4 METHODOLOGY

4.1 FROM TWO-POLICY CORRECTION TO THREE-POLICY IMPROVEMENT

Classical IS aligns only rollout and actor policies. For fixed rollout and actor policies, importance sampling provides a valid change of measure, while truncation controls variance, yielding TIS (Liu et al., 2025b). TIS assigns $w_t^{\mathrm{TIS}} = \min(\rho_t, C)$, where C is the truncation threshold. However, this correction depends only on the rollout and actor policies, and contains no information about the full-precision policy that defines the learning problem in Equation 1. The issue is more pronounced under low-precision training, where the observed actor-rollout mismatch conflates genuine policy mismatch with numerical noise, which TIS cannot distinguish. Consequently, reducing actor-rollout mismatch does not guarantee improvements of the full-precision objective.

A certificate for the full-precision objective improvement. Although exact, Equation 1 is unavailable during training because evaluating a weighting rule requires applying its update and measuring the full-precision objective. We therefore derive a certificate for the learning problem. If J is locally L-smooth, every token-weighted update satisfies

$$\mathbb{E}\big[J\big(\theta^+(\mathbf{w})\big) - J(\theta)\big] \geq \eta \underbrace{\nabla J(\theta)^\top \mathbb{E}[d(\mathbf{w})]}_{\text{full-precision alignment}} - \frac{L\eta^2}{2} \underbrace{\mathbb{E}\left[\|d(\mathbf{w})\|^2\right]}_{\text{update cost}}. \tag{2}$$

This bound preserves the original learning target while exposing the two quantities that determine local improvement: the alignment of the expected update with the full-precision gradient and the magnitude of the resulting update. A valid correction rule must therefore control both.

Tractable certification via reliability tempering. The certificate in Equation 2 remains intractable because it depends on the inaccessible full-precision gradient. We therefore restrict the weighting-rule family and derive a tractable lower bound. Let $z_t = \log \rho_t = \bar{z}_t + \epsilon_t$, where $\bar{z}_t$ denotes the genuine mismatch and ϵ_t the residual numerical noise. We then restrict the general non-negative

weights $\mathbf{w} \geq \mathbf{0}$ to a tempered family $w_t(\boldsymbol{\alpha}) \propto \min(\rho_t^{\alpha_t}, C)$ with $\alpha_t \in [0,1]$, which interpolates between identity weighting ($\alpha_t = 0$) and TIS ($\alpha_t = 1$), controlling the extent to which the observed mismatch is reliable for correction. Assuming a positive alignment margin of the genuine-mismatch update with $\nabla J(\theta)$ and a finite token-gradient second moment, under a sufficiently small step size,

$$\mathbb{E}\big[J\big(\theta^+(w(\boldsymbol{\alpha}))\big) - J(\theta)\big] \geq B - K\sqrt{\mathbb{E}\left[\sum_{t=1}^{T}(\alpha_t z_t - \bar{z}_t)^2\right]}, \tag{3}$$

where $B > 0$ and $K > 0$ are independent of $\boldsymbol{\alpha}$. Equation 3 thus converts the intractable certificate into a tractable optimization over $\boldsymbol{\alpha}$ that minimizes the error in recovering genuine mismatch from noisy observations. The detailed bound and constants are given in Appendix B.1.

Certificate-optimal tempering. Equation 3 reduces the learning problem to the following optimization over the tempered family:

$$\min_{\boldsymbol{\alpha}} \ \mathbb{E}\left[\sum_{t=1}^{T}(\alpha_t z_t - \bar{z}_t)^2\right], \qquad \text{s.t.} \quad \boldsymbol{\alpha} \in [0,1]^T \tag{4}$$

To characterize the solution of Equation 4, we impose the following mild condition on the numerical perturbation in $z_t = \bar{z}_t + \epsilon_t$.

Assumption 1 (Centered numerical noise)**.** *For each token t, assume that $\bar{z}_t$ and ϵ_t admit finite second moments, and that ϵ_t is conditionally mean-zero given $\bar{z}_t$, i.e., $\mathbb{E}[\epsilon_t \mid \bar{z}_t] = 0$.*

Theorem 1 (**Optimal tempering coefficient**)**.** *Under Assumption 1, Equation 4 is separable across tokens. For each token with $\mathbb{E}[\bar{z}_t^2] + \mathbb{E}[\epsilon_t^2] > 0$, its unique minimizer is*

$$\alpha_t^\star = 1 - \frac{\mathbb{E}[\epsilon_t^2]}{\mathbb{E}[z_t^2]},$$

where the ratio measures how much of the observed mismatch is attributable to numerical noise.

Remark 1. *The detailed derivation is deferred to Appendix B.2. The solution approaches TIS when genuine mismatch dominates and shifts toward identity weighting as numerical noise increases. By minimizing the error term in Equation 3, $\boldsymbol{\alpha}^\star$ is certificate-optimal within the tempered family. Since $\bar{z}_t$ is latent, the remaining challenge is to estimate the relative contributions of mismatch and noise, which will be addressed in the following subsection.*

4.2 **ERIS**: EXCESS-RISK-GUIDED IMPORTANCE SAMPLING

From population tempering to endpoint decisions. Theorem 1 characterizes the population-optimal coefficient $\alpha_t^\star$. However, directly estimating a continuous coefficient for each token is impractical, as the population statistics required to separate the latent genuine mismatch $\bar{z}_t$ from numerical noise ϵ_t are unavailable in RL training. We therefore restrict α_t to the two endpoints, $\alpha_t \in \{0,1\}$, corresponding to identity weighting and TIS, respectively. Under Assumption 1, the corresponding recovery risks are $\mathbb{E}[\bar{z}_t^2]$ and $\mathbb{E}[\epsilon_t^2]$. We define $D_t = \mathbb{E}[\bar{z}_t^2] - \mathbb{E}[\epsilon_t^2]$ as the *excess risk*. TIS is therefore preferred when $D_t > 0$, equivalently when $\alpha_t^\star > 1/2$, favoring correction only when the genuine mismatch dominates the low-precision noise. Since the preference implied by the population solution is preserved, the practical problem thus reduces to estimating a single risk difference from sampled-token observations.

Estimating excess risk from calibrated noise. We estimate excess risk from quantities available during low-precision training. After actor-rollout synchronization, both backends represent the same policy, so their residual log-probability discrepancy largely reflects numerical noise. The magnitude of this noise varies with token support. We therefore group tokens by the surprisal of the smaller probability assigned by the actor and rollout policies, and maintain lagged estimates of the noise statistics for each group. For token o_t, let $\hat{\mu}_t$ and $\hat{v}_t$ denote the estimated conditional mean and variance of ϵ_t for its corresponding support group. Under exact support-conditioned calibration, $(z_t - \hat{\mu}_t)^2$ has expectation equal to the squared genuine mismatch plus the calibrated noise variance. It follows that $(z_t - \hat{\mu}_t)^2 - 2\hat{v}_t$ provides a direct proxy for the excess risk between identity weighting and mismatch correction. To reduce sensitivity to extreme numerical perturbations, we clip the squared centered mismatch with a fixed threshold τ^2 and define

$$H_t = \min\big((z_t - \hat{\mu}_t)^2, \tau^2\big) - 2\hat{v}_t. \tag{5}$$

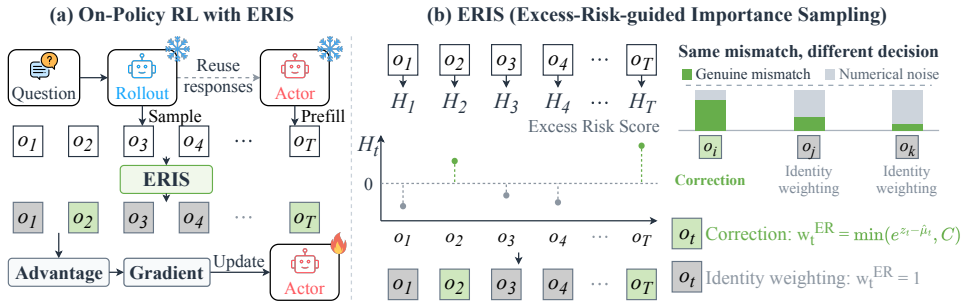


Figure 2: Overview of ERIS. ERIS uses the token-wise excess-risk score H_t to assign reliable corrections, applying truncated importance sampling when $H_t > 0$ and identity weighting otherwise.

Tokens with $H_t > 0$ receive TIS correction, while the rest retain identity weighting. Before calibration is initialized, all tokens use identity weighting. See Appendix B.3 for detailed derivations.

Budgeted correction. A positive excess-risk score identifies a token as eligible for TIS correction. To control the aggregate effect of correction, we restrict the selected tokens to a fraction $p \in [0, 1]$ of the T valid tokens. Let k denote the corresponding token budget. We set $\alpha_t = \mathbb{I}\big[t \in \mathrm{TopK}_+(H_1, \ldots, H_T; k)\big]$, where TopK_+ selects up to k tokens with the largest positive excess-risk scores. Thus, only the most reliable eligible tokens receive correction, while all remaining tokens retain identity weighting. The resulting budget limits the number of corrected tokens and thereby controls the update-cost term in Equation 2. Details can be found in Appendix B.4.

Practical objective instantiation. Given the endpoint decision α_t, ERIS instantiates the token weight as $w_t = 1 + \alpha_t \left[\min(\exp(z_t - \hat{\mu}_t), C) - 1\right]$, where C is the truncation threshold. Thus, $\alpha_t = 0$ recovers identity weighting, whereas $\alpha_t = 1$ applies the calibrated TIS correction. The resulting training objective is

$$J_{\mathrm{ER}}(\theta) = \mathbb{E}_{q \sim \mathcal{D},\, \{o_i\}_{i=1}^{G} \sim \pi_\theta^{\mathrm{R}}(\cdot|q)} \left[\frac{1}{G} \sum_{i=1}^{G} \frac{1}{T} \sum_{t=1}^{T} \mathrm{sg}(w_t) L^{\mathrm{CLIP}}(o_{i,t} \mid q, o_{i,<t}; \theta) \right]. \tag{6}$$

The correction weights are detached during optimization. ERIS therefore modifies only the actor-rollout weighting while preserving the original clipped surrogate. The full-precision objective $J(\theta)$ in Equation 1 remains the reference for evaluating the resulting update.

4.3 Conceptual Interpretation and Positioning of ERIS

Interpretation of ERIS. ERIS is motivated by a fundamental gap between the observable actor-rollout mismatch and the underlying full-precision objective that defines improvement. Under low-precision execution, the observed mismatch also mixes genuine policy mismatch with numerical noise, making direct correction unreliable. Indiscriminately applying TIS propagates noise into the update, whereas identity weighting avoids noise but leaves genuine mismatch uncorrected. As illustrated in Figure 2, ERIS balances these two errors through excess risk by making token-wise decisions between TIS correction and identity weighting, favoring corrections that improve the full-precision objective rather than merely reducing actor-rollout mismatch. Importantly, ERIS is not simply sparse TIS: support-conditioned calibration estimates the reliability of the observed mismatch, the excess-risk proxy determines whether correction is preferable to identity weighting, and the correction budget controls the update magnitude. Together, these components retain informative mismatch correction while suppressing noise-driven reweighting.

Relation to prior works. Classical policy-optimization methods correct distribution shift between rollout and actor policies (Schulman et al., 2017; Metelli et al., 2018). FlashRL (TIS) extends this view to quantized training through token-level correction at a fixed threshold (Liu et al., 2025b). AIS adapts the strength of importance weighting to reduce the estimation error of a gradient relative to an available full-precision learner (Zhou et al., 2026). The key distinction between these methods and ERIS lies in the optimization target, rather than token-wise adaptation. Instead of treating actor-rollout alignment as the primary goal, ERIS views their mismatch as a noisy signal for improving the underlying full-precision objective. The actor-rollout ratio captures only the relation between

the two low-precision policies and is itself affected by numerical perturbations. We therefore derive the correction from a certificate for full-precision improvement, rather than from change-of-measure accuracy or gradient-estimation error alone.

5 EXPERIMENTS

We present the main experimental setup and results in this section. Detailed experimental setup is provided in Appendix C, with extended experiments and analyses included in Appendix D.

5.1 EXPERIMENTAL SETUP

Models and baselines. We evaluate four Qwen models spanning two architectures and multiple scales: Qwen3-1.7B, Qwen3-4B, and Qwen3-8B (Yang et al., 2025a), together with Qwen2.5-7B (Yang et al., 2025b). We compare **ERIS** against four baselines: standard end-to-end **BF16** and **FP8** training, token-level truncated importance sampling (**TIS**), and sequence-level masked importance sampling (**MIS**). Except for the BF16 reference, all methods use end-to-end FP8 execution for both rollout and actor training. This setup enables systematic evaluation across model scales and architectures against both precision references and representative mismatch-mitigation methods.

Training setup. All models are trained on DAPO-Math-17K (Yu et al., 2025) using AdamW (Loshchilov & Hutter, 2017) with a constant learning rate of 1×10^{-6}. We use a PPO minibatch size of 64 prompts and a group size of 8. The maximum prompt length is 1,024 tokens, while the maximum response length is 10,240 tokens for Qwen3-8B and 8,192 tokens for the remaining models. We set $\epsilon_{\text{low}} = 0.2$, $\epsilon_{\text{high}} = 0.28$, and the overlong-response buffer to 1,024 tokens. For ERIS, we set $\tau = \log 2$. All experiments are conducted on 8 NVIDIA RTX PRO 6000 GPUs.

Evaluation setup. We evaluate all models with vLLM on eight benchmarks spanning two categories. (1) Mathematical Reasoning benchmarks assess reasoning capability and include AIME24 (Zhang & Math-AI Team, 2024), AIME25 (Zhang & Math-AI Team, 2025), AMC23 (Math-AI, 2025) and OlympiadBench (He et al., 2024). (2) General Knowledge benchmarks evaluate general capability and include ARC-Easy and ARC-Challenge (Clark et al., 2018), SuperGPQA (Du et al., 2025) and MMLU-4 (Hendrycks et al., 2020), which comprises four MMLU subjects. For generation, we use a temperature of 0.7, nucleus sampling at 0.95, and a maximum length of 16,384 tokens. We report Pass@1 accuracy on all benchmarks.

5.2 MAIN RESULTS

In this subsection, we evaluate ERIS from both benchmark performance and training dynamics. We first compare it with BF16 and FP8 references, together with other baselines across model scales and architectures, and then analyze training reward trajectories to assess training stability.

ERIS improves FP8 training and approaches BF16-level performance across model scales and architectures. We evaluate different methods on mathematical reasoning benchmarks, with results shown in Table 1. First, ERIS substantially improves low-precision training across Qwen3 model scales, achieving average scores of **19.19**, **37.24**, and **40.55** on Qwen3-1.7B, 4B, and 8B, respectively. Second, ERIS largely closes the performance gap between FP8 and BF16 training. It nearly matches BF16 on Qwen3-1.7B, surpasses BF16 by 3.70% on Qwen3-4B, and achieves comparable performance on Qwen3-8B. Third, ERIS consistently outperforms existing mismatch-mitigation baselines by substantial margins. On Qwen3-4B, it improves over TIS and MIS by 15.29% and 21.46%, respectively, while on Qwen3-8B it improves over standard FP8 training by 14.29%. The same trend extends to Qwen2.5-7B, where standard FP8 training diverges while ERIS achieves an average score of **28.22**, approaching the BF16 score of 29.16 and outperforming both TIS and MIS. Overall, these results demonstrate that ERIS provides robust performance improvements under low-precision RL training across model scales and architectures.

ERIS exhibits consistent generalization on out-of-domain benchmarks. We further evaluate different methods on out-of-domain general-knowledge benchmarks to assess their generalization beyond the mathematical reasoning task. As shown in Table 1, ERIS consistently achieves the strongest OOD performance across Qwen3 model scales, with average scores of **59.14**, **72.78**, and **73.67** on Qwen3-1.7B, 4B, and 8B, respectively, surpassing the corresponding BF16 baselines, which indi-

Table 1: Performance comparison across model scales and training methods on mathematical reasoning and general knowledge benchmarks. The best results are marked in **bold**, and the second-best results are underlined. Divergent runs are reported with "–".

Model	Method	Mathematical Reasoning					General Knowledge				
		AIME24	AIME25	AMC23	Olymp.	Avg.	ARC-E	ARC-C	S-GPQA	MMLU-4	Avg.
Qwen3 1.7B	BF16	6.88	**6.04**	37.97	26.52	**19.35**	79.90	69.98	15.97	52.68	54.63
	FP8	6.25	3.13	**38.75**	26.79	18.73	83.24	72.67	15.62	53.42	56.24
	TIS	6.25	3.96	35.47	25.76	17.86	82.53	72.74	15.75	52.44	55.87
	MIS	6.67	5.21	**38.75**	25.07	18.93	67.13	60.08	14.08	44.13	46.36
	ERIS	**7.71**	4.17	37.66	**27.23**	19.19	**86.75**	**76.00**	**16.45**	**57.34**	**59.14**
Qwen3 4B	BF16	19.17	17.08	**61.41**	45.98	35.91	96.33	90.02	29.54	70.76	71.66
	FP8	–	–	–	–	–	–	–	–	–	–
	TIS	14.58	15.42	56.88	42.32	32.30	96.18	89.92	28.29	70.57	71.24
	MIS	15.00	12.08	55.31	40.25	30.66	94.06	87.46	27.38	66.79	68.92
	ERIS	**21.46**	**18.96**	**61.41**	**47.14**	**37.24**	**97.10**	**91.39**	**31.18**	**71.44**	**72.78**
Qwen3 8B	BF16	24.58	17.50	69.22	**50.86**	40.54	96.34	92.28	32.10	**73.09**	73.45
	FP8	18.33	16.67	61.41	45.51	35.48	95.75	91.89	31.01	72.60	72.81
	TIS	**25.83**	17.29	63.43	47.20	38.44	**96.66**	92.19	30.94	72.89	73.17
	MIS	17.71	15.21	57.66	44.18	33.69	93.97	90.58	29.63	70.94	71.28
	ERIS	23.33	**21.25**	**69.38**	48.22	**40.55**	96.53	**92.49**	**33.44**	72.22	**73.67**
Qwen2.5 7B	BF16	11.88	**10.00**	**55.00**	**39.77**	**29.16**	**95.67**	**90.20**	**26.47**	**73.61**	**71.49**
	FP8	–	–	–	–	–	–	–	–	–	–
	TIS	12.71	7.08	52.81	36.09	27.17	95.28	88.84	25.04	71.33	70.12
	MIS	11.46	8.54	51.25	36.75	27.00	95.02	87.65	23.84	70.16	69.17
	ERIS	**13.96**	9.38	52.34	37.18	28.22	95.28	89.39	25.43	71.78	70.47

cates that the benefits of ERIS are not restricted to in-domain mathematical reasoning. Compared with TIS and MIS, ERIS also yields consistent improvements: it outperforms TIS by 3.27, 1.54, and 0.50 points across the three Qwen3 scales while exceeding MIS by substantially larger margins. On Qwen2.5-7B, ERIS reaches **70.47**, outperforming TIS and MIS, and further narrowing the gap to the BF16 score of 71.49. Overall, ERIS achieves consistently stronger OOD generalization than existing baselines, preserving the general capabilities of the underlying model.

ERIS improves FP8 training stability and maintains close alignment with BF16 optimization. As shown in Figure 3, ERIS maintains a reward trajectory that closely tracks the BF16 reference throughout training, indicating that it preserves the optimization behavior of full-precision training under FP8 execution. In contrast, standard FP8 exhibits substantially larger reward fluctuations and severe degradation in the later stage of training. This behavior is particularly evident during steps 180–220, where the FP8 reward repeatedly collapses and eventually becomes unstable, while ERIS remains close to the BF16 trajectory. Compared with TIS and MIS, ERIS also achieves consistently higher rewards in this stage with noticeably smaller deviations from BF16. These results indicate that ERIS effectively stabilizes low-precision RL optimization while retaining training dynamics comparable to BF16.

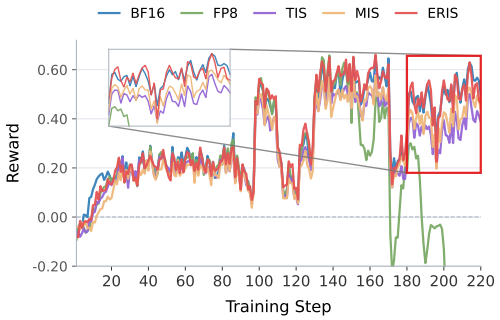


Figure 3: Comparison of training rewards for Qwen3-4B under 5 different methods.

5.3 Empirical Analysis of ERIS's Mechanism

In this subsection, we analyze the mechanism of ERIS from three perspectives. We first examine the advantage of three-policy alignment over conventional actor-rollout alignment, then evaluate

whether the excess-risk score reliably identifies beneficial corrections, and finally compare ERIS with heuristic token selection strategies. All analyses are conducted on Qwen3-4B.

Actor-rollout alignment alone does not ensure full-precision consistency. We examine the relationship between actor-rollout mismatch $|z_t|$, and full-precision mismatch, defined as the root mean square deviation of the actor and rollout log-probabilities from the full-precision policy. As illustrated in Figure 4, standard FP8 exhibits large mismatch along both dimensions, as it imposes no explicit alignment constraint. TIS substantially reduces actor-rollout mismatch, but its full-precision mismatch remains high, indicating that aligning the two low-precision policies alone does not guarantee consistency with the underlying full-precision policy. In contrast, ERIS achieves low mismatch along both axes, simultaneously improving actor-rollout consistency and preserving alignment with the full-precision policy. These results highlight the necessity of three-policy alignment: effective low-precision correction should not only reconcile actor and rollout policies, but also ensure that the resulting updates remain consistent with full-precision behavior.

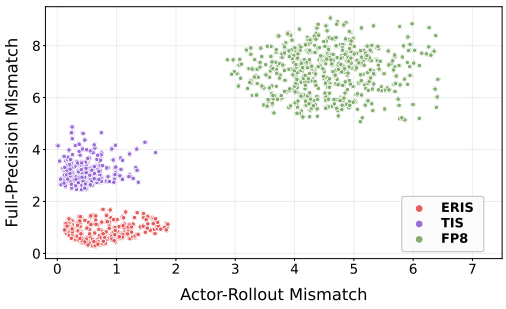


Figure 4: Three-policy alignment under standard FP8, TIS and ERIS.

ERIS improves the identification of beneficial corrections. We further examine whether the excess-risk score identifies tokens for which correction is more beneficial than identity weighting. We use a BF16-based reference oracle Δ_t^{ref} instead of the theoretical population excess risk D_t, which is explained in Appendix C.5. For a selection set $\mathcal{S}$, we define correction enrichment as $\Pr(\Delta_t^{\text{ref}} > 0 \mid t \in \mathcal{S}) / \Pr(\Delta_t^{\text{ref}} > 0)$, where $\Delta_t^{\text{ref}} > 0$ indicates that TIS is preferred to identity weighting under the oracle Δ_t^{ref}. We compare ERIS with a mismatch-only strategy that selects the largest observed mismatch $|z_t|$. As shown in Figure 5, mismatch-only selection achieves a $1.89\times$ enrichment over the all-token base rate, confirming that z_t contains useful genuine-mismatch signal. However, because z_t also contains numerical noise, mismatch magnitude alone is insufficient to identify reliable corrections. By accounting for numerical reliability through H_t, ERIS further increases the enrichment to $4.47\times$. These results show that ERIS better distinguishes informative mismatch from noise-dominated mismatch, enabling more reliable corrections.

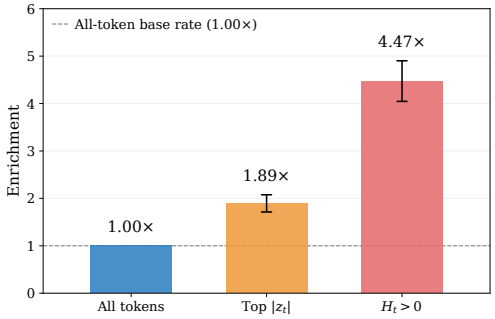


Figure 5: Beneficial correction enrichment. Error bars denote 95% confidence intervals.

ERIS outperforms heuristic token selection strategies. We compare ERIS with two alternative token selection strategies in Table 2, under a correction budget of $p = 0.2$. Random selection, which applies TIS to a uniformly sampled subset of tokens, achieves 35.90 on Math and 71.27 on General benchmarks. Notably, it already exceeds the standard TIS baseline on Math (32.30), suggesting that indiscriminate correction over all tokens can be harmful. In comparison, ERIS achieves 37.24 and 72.78, indicating that sparsity alone does not account for its gains. We further evaluate a mismatch-only strategy that corrects the top 20% of tokens ranked by the observed actor-rollout mismatch $|z_t|$. This strategy obtains only 32.78 on Math and 70.58 on General, substantially underperforming both random selection and ERIS. The degradation indicates that large observed mismatch is not necessarily a reliable indicator for correction under low-precision execution, since $|z_t|$ may reflect either genuine policy mismatch or numerical noise. These results show that ERIS benefits from explicitly accounting for the reliability

Table 2: Comparison of different token selection strategies.

Strategy	Math	General
Random 20%	35.90	71.27
$\lvert z_t \rvert$ Top-20%	32.78	70.58
$H_t > 0$ Top-20% (**ERIS**)	**37.24**	**72.78**

Table 3: Ablation studies of ERIS on Qwen3-4B. Unless otherwise specified, we use the default configuration with correction budget $p = 0.2$, truncation threshold $C = 2$, and the full method. We mark the best results in **bold**. Divergent runs are reported with "–".

(a) Component Analysis

Setting	Math	General
Full	37.24	**72.78**
w/o support condition	36.38	69.88
w/o mean calibration	34.33	71.96
w/o H_t clipping	**37.52**	71.03

(b) Correction Budget

Budget	Math	General
0.05	–	–
0.10	36.11	72.45
0.20	**37.24**	**72.78**
0.50	35.86	72.19

(c) Truncation Threshold

Threshold	Math	General
1.5	–	–
2	**37.24**	**72.78**
5	37.14	71.85
10	36.15	70.83

of observed mismatch, enabling the correction budget to be allocated to tokens with more informative mismatch signals rather than merely large observed mismatch values.

5.4 Ablation Studies

In this subsection, we conduct ablation studies of ERIS from three perspectives. We first analyze the contribution of its key components, then study the effect of the correction budget and finally examine its sensitivity to the truncation threshold. All ablations are conducted on Qwen3-4B.

Ablation of ERIS components. As shown in Table 3a, the full method achieves 37.24 on Math and 72.78 on General benchmarks. Using a global estimated noise mean and variance instead of support-conditioned calibration reduces performance to 36.38/69.88, showing the importance of modeling token-dependent noise. Removing mean calibration and setting $Y_t = z_t$ further degrades performance to 34.33/71.96, highlighting the need to correct systematic bias. Removing the clipping threshold τ in Equation 5 yields 37.52/71.03, indicating that clipping improves robustness to extreme numerical noise. These results demonstrate that support-conditioned calibration, mean calibration, and H_t clipping each contribute to reliable excess-risk estimation.

A moderate correction budget balances correction coverage and update control. As shown in Table 3b, the default $p = 0.2$ achieves the best performance, with 37.24 on Math and 72.78 on General benchmarks, while $p = 0.1$ yields comparable results of 36.11/72.45. Reducing the budget to $p = 0.05$ leads to divergence, suggesting that overly restrictive correction leaves much reliable mismatch uncorrected. In contrast, increasing the budget to $p = 0.5$ degrades performance to 35.86/72.19, consistent with weaker control of update cost and increased exposure to noise-dominated corrections. These results indicate that a moderate correction budget provides the best balance between correcting informative mismatch and controlling the resulting updates.

ERIS performs best with a moderate truncation threshold. According to Table 3c, the default $C = 2$ achieves the highest scores of 37.24 on Math and 72.78 on General benchmarks, while the more restrictive $C = 1.5$ leads to divergence. This suggests that overly aggressive truncation may suppress genuine actor-rollout mismatch and leave useful corrections unapplied. Conversely, relaxing the threshold C to 5 and 10 gradually degrades performance to 37.14/71.85 and 36.15/70.83, respectively. A large threshold permits higher-variance importance weights and makes the correction more sensitive to mismatch dominated by numerical noise. These results favor moderate truncation that preserves informative mismatch correction while limiting noise-induced reweighting.

6 Conclusion

In this paper, we study the stability of low-precision RL from the perspective of full-precision objective improvement. We formulate low-precision RL as a three-policy problem and propose Excess-Risk-guided Importance Sampling (ERIS), which estimates the excess risk of identity weighting relative importance correction, favoring corrections that improve the full-precision objective rather than merely reducing actor-rollout mismatch. Experiments across model scales and architectures show that ERIS enables stable low-precision RL with a narrowed performance gap to BF16, while preserving the computing efficiency in low-precision training. We discuss the limitations of this paper in Appendix E.

AI USE STATEMENT

We used generative AI tools to assist with manuscript refinement, literature retrieval and review, and the code implementation and debugging of the proposed method. All AI-assisted content was independently reviewed by the authors and, where necessary, revised, corrected, and adapted to ensure consistency with the methodology, experimental procedures, and findings reported in this paper. The authors retain full responsibility for the final manuscript, including its text, citations, technical claims, analyses, experimental results, and code.

ETHICS STATEMENT

This paper focuses on algorithmic methods for improving the stability and efficiency of low-precision reinforcement learning for large language models. Our study does not involve human subjects, collection or processing of personal or sensitive information, or release of new datasets. All experiments are conducted using existing models, datasets, and evaluation benchmarks described in the paper. We do not identify ethical concerns specific to the proposed methodology beyond those generally associated with large language model research and development. We have conducted this work in accordance with the ICLR Code of Ethics and are committed to accurate, transparent, and responsible reporting of our method and experimental results.

REPRODUCIBILITY STATEMENT

We provide the implementation code in the supplementary materials to facilitate reproduction of our results. The main paper describes the models, baselines, training configuration, evaluation benchmarks, and key hyperparameters in Section 5.1, while Appendix C provides the implementation details of ERIS, as well as complete training and evaluation setup, including rollout-correction configurations and evaluation prompts. For the theoretical results, we state the required assumption and theorem in the main text and provide detailed proofs and derivations in Appendix B. Together, these materials are intended to provide sufficient algorithmic, experimental, and theoretical details for reproducing the results reported in this paper.

REFERENCES

Felix Abecassis, Anjulie Agrusa, Dong Ahn, Jonah Alben, Stefania Alborghetti, Michael Andersch, Sivakumar Arayandi, Alexis Bjorlin, Aaron Blakeman, Evan Briones, et al. Pretraining large language models with nvfp4. *arXiv preprint arXiv:2509.25149*, 2025.

Qianben Chen, Jingyi Cao, Jiayu Zhang, Tianrui Qin, LiXiaowan, King Zhu, Dingfeng Shi, He Zhu, Minghao Liu, Xiaobo Liang, Ge Zhang, Jian Yang, Yuchen Eleanor Jiang, and Wangchunshu Zhou. A^2FM: An adaptive agent foundation model for tool-aware hybrid reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=3kvV1nfWVq.

Yongchao Chen, Yueying Liu, Junwei Zhou, Yilun Hao, Jingquan Wang, Yang Zhang, Na Li, and Chuchu Fan. R1-code-interpreter: Llms reason with code via supervised and multi-stage reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 89244–89274, 2026b.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*, 2018.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui

Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3.int8(): 8-bit matrix multiplication for transformers at scale. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 30318–30332. Curran Associates, Inc., 2022. doi: 10.52202/068431-2198. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/c3ba4962c05c49636d4c6206a97e9c8a-Paper-Conference.pdf.

Xeron Du, Yifan Yao, Kaijing Ma, Bingli Wang, Tianyu Zheng, Zhu, Minghao Liu, Yiming Liang, Xiaolong Jin, Zhenlin Wei, Chujie Zheng, Kaixin Deng, Shuyue Guo, Shian Jia, Sichao Jiang, Yiyan Liao, Rui Li, Qinrui Li, Sirun Li, Yizhi Li, Yunwen Li, Dehua Ma, Yuansheng Ni, Haoran Que, Qiyao Wang, Zhoufutu Wen, Siwei Wu, Tianshun Xing, Ming Xu, Zhenzhu Yang, Noah Wang, Junting Zhou, yuelin bai, Xingyuan Bu, chenglin cai, Liang Chen, Yifan Chen, Cheng Chengtuo, Tianhao Cheng, Keyi Ding, Siming Huang, HUANG YUN, Yaoru Li, Yizhe Li, Zhaoqun Li, Tianhao Liang, Chengdong Lin, Hongquan Lin, Yinghao Ma, Zhongyuan Peng, Zifan Peng, Qige Qi, Shi Qiu, Xingwei Qu, Shanghaoran Quan, Yizhou Tan, Zili Wang, Chenqing Wang, Hao Wang, Yiya Wang, Yubo Wang, Jiajun Xu, Kexin Yang, Ruibin Yuan, Yuanhao Yue, Tianyang Zhan, Chun Zhang, Jinyang Zhang, Xiyue Zhang, Owen Zhang, Yue Zhang, Yongchi Zhao, Xiangyu Zheng, ChenghuaZhong, Yang Gao, Zhoujun Li, Dayiheng Liu, Qian Liu, Tianyu Liu, Shiwen Ni, Junran Peng, Yujia Qin, Wenbo Su, Guoyin Wang, Shi Wang, Jian Yang, Min Yang, Meng Cao, Xiang Yue, ZHAO-XIANG ZHANG, Wangchunshu Zhou, Jiaheng Liu, Qunshu Lin, Wenhao Huang, and Ge Zhang. Supergpqa: Scaling llm evaluation

across 285 graduate disciplines. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference. Curran Associates, Inc., 2025. doi: 10.52202/085713-3766. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/a3c5af1f56fc73eef1ba0f442739f5ca-Paper-Datasets_and_Benchmarks_Track.pdf.

Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, Jiashu Wang, Tongkai Yang, Binhang Yuan, and Yi Wu. Areal: A large-scale asynchronous reinforcement learning system for language reasoning, 2025. URL https://arxiv.org/abs/2505.24298.

Hao Gu, Hao Wang, Jiacheng Liu, Lujun Li, Qiyuan Zhu, Bei Liu, Binxing Xu, Lei Wang, Xintong Yang, Sida Lin, et al. Qarl: Rollout-aligned quantization-aware rl for fast and stable training under training–inference mismatch. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 17051–17064, 2026.

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, Sep 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL https://doi.org/10.1038/s41586-025-09422-z.

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

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Jian Hu, Xibin Wu, Zilin Zhu, Xianyu, Weixun Wang, Dehao Zhang, and Yu Cao. Openrlhf: An easy-to-use, scalable and high-performance rlhf framework. *arXiv preprint arXiv:2405.11143*, 2024.

Wei Huang, Yi Ge, Shuai Yang, Yicheng Xiao, Huizi Mao, Yujun Lin, Hanrong Ye, Sifei Liu, Ka Chun Cheung, Hongxu (Danny) Yin, Yao Lu, XIAOJUAN QI, Song Han, and Yukang Chen. Qerl: Beyond efficiency - quantization-enhanced reinforcement learning for llms. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 99338–99361, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/a15a2ece7f0663d1ba7db91103ac61c9-Paper-Conference.pdf.

Xue Jiang, Yihong Dong, Mengyang Liu, Deng Hongyi, Tian Wang, Yongding Tao, Zhi Jin, Wenpin Jiao, and Ge Li. Coderl+: Improving code generation via reinforcement with execution semantics alignment. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3608–3622, 2026.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, SOSP '23, pp. 611–626, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400702297. doi: 10.1145/3600006.3613165. URL https://doi.org/10.1145/3600006.3613165.

Hung Le, Yue Wang, Akhilesh Deepak Gotmare, Silvio Savarese, and Steven Chu Hong Hoi. Coderl: Mastering code generation through pretrained models and deep reinforcement learning. *Advances in Neural Information Processing Systems*, 35:21314–21328, 2022.

Jiazheng Li, Hongzhou Lin, Hong Lu, Kaiyue Wen, Zaiwen Yang, Jiaxuan Gao, Yi Wu, and Jingzhao Zhang. Questa: Expanding reasoning capacity in llms via question augmentation. In *International Conference on Learning Representations*, volume 2026, pp. 144191–144215, 2026a.

Yuhang Li, Reena Elangovan, Xin Dong, Priyadarshini Panda, and Brucek Khailany. Qurl: Efficient reinforcement learning with quantized rollout. *arXiv preprint arXiv:2602.13953*, 2026b.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of machine learning and systems*, 6:87–100, 2024.

Jiacai Liu, Yingru Li, Yuqian Fu, Jiawei Wang, Qian Liu, and Yu Shen. When speed kills stability: Demystifying RL collapse from the training-inference mismatch, September 2025a. URL https://richardli.xyz/rl-collapse.

Liyuan Liu, Feng Yao, Dinghuai Zhang, Chengyu Dong, Jingbo Shang, and Jianfeng Gao. Flashrl: 8bit rollouts, full power rl, August 2025b. URL https://fengyao.notion.site/flash-rl.

Xiaoyuan Liu, Tian Liang, Zhiwei He, Jiahao Xu, Wenxuan Wang, Pinjia He, Zhaopeng Tu, Haitao Mi, and Dong Yu. Trust, but verify: A self-verification approach to reinforcement learning with verifiable rewards. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 130475–130501. Curran Associates, Inc., 2025c. doi: 10.52202/085713-4348. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/bd3d106d8e9789ce4fd40e1edafe71d5-Paper-Conference.pdf.

Zechun Liu, Barlas Oguz, Changsheng Zhao, Ernie Chang, Pierre Stock, Yashar Mehdad, Yangyang Shi, Raghuraman Krishnamoorthi, and Vikas Chandra. Llm-qat: Data-free quantization aware training for large language models. In *Findings of the association for computational linguistics: ACL 2024*, pp. 467–484, 2024.

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, 2025d. URL https://arxiv.org/abs/2503.20783.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Math-AI. AMC 2023. https://huggingface.co/datasets/math-ai/amc23, 2025. Hugging Face Dataset.

Alberto Maria Metelli, Matteo Papini, Francesco Faccio, and Marcello Restelli. Policy optimization via importance sampling. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/6aed000af86a084f9cb0264161e29dd3-Paper.pdf.

Paulius Micikevicius, Dusan Stosic, Neil Burgess, Marius Cornea, Pradeep Dubey, Richard Grisenthwaite, Sangwon Ha, Alexander Heinecke, Patrick Judd, John Kamalu, Naveen Mellempudi, Stuart Oberman, Mohammad Shoeybi, Michael Siu, and Hao Wu. Fp8 formats for deep learning, 2022. URL https://arxiv.org/abs/2209.05433.

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.

Zhaopeng Qiu, Shuang Yu, Jingqi Zhang, Shuai Zhang, Xue Huang, Jingyi Yang, and Junjie Lai. Fp8-rl: A practical and stable low-precision stack for llm reinforcement learning. *arXiv preprint arXiv:2601.18150*, 2026.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Bita Darvish Rouhani, Ritchie Zhao, Ankit More, Mathew Hall, Alireza Khodamoradi, Summer Deng, Dhruv Choudhary, Marius Cornea, Eric Dellinger, Kristof Denolf, Stosic Dusan, Venmugil Elango, Maximilian Golub, Alexander Heinecke, Phil James-Roxby, Dharmesh Jani, Gaurav Kolhe, Martin Langhammer, Ada Li, Levi Melnick, Maral Mesmakhosroshahi, Andres Rodriguez, Michael Schulte, Rasoul Shafipour, Lei Shao, Michael Siu, Pradeep Dubey, Paulius Micikevicius, Maxim Naumov, Colin Verrilli, Ralph Wittig, Doug Burger, and Eric Chung. Microscaling data formats for deep learning, 2023. URL https://arxiv.org/abs/2310.10537.

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, 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.

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.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism, 2020. URL https://arxiv.org/abs/1909.08053.

Joykirat Singh, Raghav Magazine, Yash Pandya, and Akshay Nambi. Agentic reasoning and tool integration for llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2505.01441.

GLM Team, Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, Kedong Wang, Lucen Zhong, Mingdao Liu, Rui Lu, Shulin Cao, Xiaohan Zhang, Xuancheng Huang, Yao Wei, Yean Cheng, Yifan An, Yilin Niu, Yuanhao Wen, Yushi Bai, Zhengxiao Du, Zihan Wang, Zilin Zhu, Bohan Zhang, Bosi Wen, Bowen Wu, Bowen Xu, Can Huang, Casey Zhao, Changpeng Cai, Chao Yu, Chen Li, Chendi Ge, Chenghua Huang, Chenhui Zhang, Chenxi Xu, Chenzheng Zhu, Chuang Li, Congfeng Yin, Daoyan Lin, Dayong Yang, Dazhi Jiang, Ding Ai, Erle Zhu, Fei Wang, Gengzheng Pan, Guo Wang, Hailong Sun, Haitao Li, Haiyang Li, Haiyi Hu, Hanyu Zhang, Hao Peng, Hao Tai, Haoke Zhang, Haoran Wang, Haoyu Yang, He Liu, He Zhao, Hongwei Liu, Hongxi Yan, Huan Liu, Huilong Chen, Ji Li, Jiajing Zhao, Jiamin Ren, Jian Jiao, Jiani Zhao, Jianyang Yan, Jiaqi Wang, Jiayi

Gui, Jiayue Zhao, Jie Liu, Jijie Li, Jing Li, Jing Lu, Jingsen Wang, Jingwei Yuan, Jingxuan Li, Jingzhao Du, Jinhua Du, Jinxin Liu, Junkai Zhi, Junli Gao, Ke Wang, Lekang Yang, Liang Xu, Lin Fan, Lindong Wu, Lintao Ding, Lu Wang, Man Zhang, Minghao Li, Minghuan Xu, Mingming Zhao, Mingshu Zhai, Pengfan Du, Qian Dong, Shangde Lei, Shangqing Tu, Shangtong Yang, Shaoyou Lu, Shijie Li, Shuang Li, Shuang-Li, Shuxun Yang, Sibo Yi, Tianshu Yu, Wei Tian, Weihan Wang, Wenbo Yu, Weng Lam Tam, Wenjie Liang, Wentao Liu, Xiao Wang, Xiaohan Jia, Xiaotao Gu, Xiaoying Ling, Xin Wang, Xing Fan, Xingru Pan, Xinyuan Zhang, Xinze Zhang, Xiuqing Fu, Xunkai Zhang, Yabo Xu, Yandong Wu, Yida Lu, Yidong Wang, Yilin Zhou, Yiming Pan, Ying Zhang, Yingli Wang, Yingru Li, Yinpei Su, Yipeng Geng, Yitong Zhu, Yongkun Yang, Yuhang Li, Yuhao Wu, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yuxuan Zhang, Zezhen Liu, Zhen Yang, Zhengda Zhou, Zhongpei Qiao, Zhuoer Feng, Zhuorui Liu, Zichen Zhang, Zihan Wang, Zijun Yao, Zikang Wang, Ziqiang Liu, Ziwei Chai, Zixuan Li, Zuodong Zhao, Wenguang Chen, Jidong Zhai, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models, 2025. URL https://arxiv.org/abs/2508.06471.

Guiyao Tie, Zeli Zhao, Dingjie Song, Fuyang Wei, Rong Zhou, Yurou Dai, Wen Yin, Zhejian Yang, Jiangyue Yan, Yao Su, Zhenhan Dai, Yifeng Xie, Yihan Cao, Lichao Sun, Pan Zhou, Lifang He, Hechang Chen, Yu Zhang, Qingsong Wen, Tianming Liu, Neil Zhenqiang Gong, Jiliang Tang, Caiming Xiong, Heng Ji, Philip S. Yu, and Jianfeng Gao. A survey on post-training of large language models, 2025. URL https://arxiv.org/abs/2503.06072.

Songjun Tu, Chengdong Xu, Qichao Zhang, Yiwen Ma, Yaocheng Zhang, Linjing Li, Dong Li, Xiangyuan Lan, and Dongbin Zhao. Ucob: Learning to utilize and evolve agentic skills via credit-aware on-policy bidirectional self-distillation. *arXiv preprint arXiv:2606.29502*, 2026.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. TRL: Transformers Reinforcement Learning, 2020. URL https://github.com/huggingface/trl.

Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Liyuan Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, Weizhu Chen, Shuohang Wang, Simon Du, and yelong shen. Reinforcement learning for reasoning in large language models with one training example. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 122721–122764. Curran Associates, Inc., 2025. doi: 10.52202/085713-4093. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/b1ea3f93167c55f3fc2999b68a170ff3-Paper-Conference.pdf.

Haocheng Xi, Charlie Ruan, Peiyuan Liao, Yujun Lin, Han Cai, Yilong Zhao, Shuo Yang, Kurt Keutzer, Song Han, and Ligeng Zhu. Jet-rl: Enabling on-policy fp8 reinforcement learning with unified training and rollout precision flow. *arXiv preprint arXiv:2601.14243*, 2026.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International conference on machine learning*, pp. 38087–38099. PMLR, 2023.

LLM-Core Xiaomi. Mimo-v2-flash technical report, 2026. URL https://arxiv.org/abs/2601.02780.

Fengli Xu, Qianyue Hao, Zefang Zong, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, Tianjian Ouyang, Fanjin Meng, Chenyang Shao, Yuwei Yan, Qinglong Yang, Yiwen Song, Sijian Ren, Xinyuan Hu, Yu Li, Jie Feng, Chen Gao, and Yong Li. Towards large reasoning models: A survey of reinforced reasoning with large language models, 2025. URL https://arxiv.org/abs/2501.09686.

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

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin

Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025b. URL https://arxiv.org/abs/2412.15115.

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

Yifan Zhang and Math-AI Team. American invitational mathematics examination (AIME) 2024, 2024.

Yifan Zhang and Math-AI Team. American invitational mathematics examination (AIME) 2025, 2025.

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

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. Sglang: Efficient execution of structured language model programs, 2024. URL https://arxiv.org/abs/2312.07104.

Tianle Zhong, Neiwen Ling, Yifan Pi, Zijun Wei, Tianshu Yu, Geoffrey Fox, Peng Wu, and Xiao Yu. Diagnosing training inference mismatch in llm reinforcement learning. *arXiv preprint arXiv:2605.14220*, 2026.

Jiajun Zhou, Wei Shao, Lingchao Zheng, Yuwei Fan, and Ngai Wong. Ais: Adaptive importance sampling for quantized rl. *arXiv preprint arXiv:2605.13907*, 2026.

Zhengyang Zhuge, Hao Yu, Xin Wang, Zheng Li, Yizhong Cao, Dayiheng Liu, and Jianwei Zhang. Quads: Stabilizing nvfp4 reinforcement learning for moe via quantization-error alignment across dual sides, 2026. URL https://arxiv.org/abs/2607.15810.

Jiaru Zou, Ling Yang, Yunzhe Qi, Sirui Chen, Mengting Ai, Ke Shen, Jingrui He, and Mengdi Wang. Autotool: Dynamic tool selection and integration for agentic reasoning, 2026. URL https://arxiv.org/abs/2512.13278.

A Related Works

Reinforcement Learning for LLMs. Reinforcement learning has become a central paradigm for improving LLM capabilities during post-training (Xu et al., 2025; Tie et al., 2025). At the algorithmic level, DeepSeekMath introduced Group Relative Policy Optimization (GRPO), which replaces the learned critic in PPO with group-relative reward estimation and reduces the memory overhead of RL training (Shao et al., 2024). DeepSeek-R1 subsequently demonstrated that large-scale RL with verifiable rewards can elicit sophisticated reasoning behaviors directly from language models (Guo et al., 2025). Building on this paradigm, subsequent methods further improve optimization stability and scalability: DAPO introduces several optimization and sampling refinements for long-context reasoning training (Yu et al., 2025), Dr. GRPO corrects optimization biases in GRPO that can spuriously favor longer responses (Liu et al., 2025d), and GSPO reformulates importance weighting and clipping at the sequence level for more stable large-scale training (Zheng et al., 2025). Beyond these algorithmic developments, RL has been increasingly adopted to enhance a broad range of reasoning capabilities. In mathematical reasoning, verifiable rewards provide scalable supervision for learning long-horizon reasoning and self-verification (Li et al., 2026a; Shao et al., 2024; Liu et al., 2025c; Wang et al., 2025). For code generation, executable feedback enables direct optimization toward functional correctness while improving reasoning over program semantics (Le et al., 2022; Jiang et al., 2026; Chen et al., 2026b). More recently, RL has been extended to agentic settings, where it supports multi-step decision making, tool use, and skill acquisition through interaction with external environments (Chen et al., 2026a; Zou et al., 2026; Tu et al., 2026; Singh et al., 2025). These developments establish RL as a general post-training framework for improving reasoning, coding, and agentic capabilities in modern LLMs.

Quantization of LLMs. Quantization has become a fundamental technique for reducing the memory and computational costs of LLMs, with existing approaches broadly divided into post-training quantization (PTQ) and quantization-aware training (QAT). PTQ quantizes pretrained models without further optimization, with representative methods including LLM.int8(), SmoothQuant, and AWQ (Dettmers et al., 2022; Xiao et al., 2023; Lin et al., 2024). In contrast, QAT exposes models to quantization effects during optimization, allowing them to adapt to low-precision computation, as exemplified by LLM-QAT (Liu et al., 2024). Meanwhile, quantization formats have evolved from integer-centric representations toward low-precision floating-point formats and increasingly lower bit widths. Micikevicius et al. (2022) introduces two FP8 formats for efficient deep learning, which are E4M3 and E5M2. In order to improve model accuracy and reduce user friction, microscaling formats such as MXFP8 and MXFP4 employ more fine-grained block-wise scaling to support lower-precision computation (Rouhani et al., 2023). More recently, NVFP4 enables practical FP4 training through fine-grained scaling and dedicated numerical techniques (Abecassis et al., 2025). Accordingly, quantization is no longer restricted to deployment-time compression, but has increasingly become an integral and essential component of large-scale model training. Qwen3.5 and MiMo-V2-Flash adopt native FP8 computation for large-scale RL (Qwen Team, 2026; Xiaomi, 2026), while DeepSeek-V4 further incorporates MXFP4-aware QAT and FP4 execution across both training and inference (DeepSeek-AI et al., 2026).

Infrastructure for LLM RL. LLM reinforcement learning places substantial demands on distributed systems, as training repeatedly alternates among rollout generation, reward computation, and actor updates. Early frameworks such as TRL provide trainer-centric abstractions that integrate these components within a unified training workflow (von Werra et al., 2020). As model scale and rollout cost increase, subsequent systems increasingly decouple rollout generation from actor updates to improve resource utilization. OpenRLHF coordinates distributed rollout and training workers through Ray while integrating specialized backends such as DeepSpeed and vLLM (Hu et al., 2024). VeRL further formulates RLHF as a distributed dataflow, enabling flexible device placement and efficient resource remapping between rollout and training (Sheng et al., 2024). More recent systems move toward asynchronous execution. AReaL allows rollout workers to continuously generate trajectories while training proceeds independently, improving pipeline utilization while explicitly managing policy staleness (Fu et al., 2025). Alongside this evolution, RL systems increasingly compose specialized engines for different workloads. Actor updates are commonly handled by distributed training backends such as PyTorch FSDP and Megatron-LM (Zhao et al., 2023; Shoeybi et al., 2020), while rollout generation relies on high-throughput inference engines such as vLLM and SGLang (Kwon et al., 2023; Zheng et al., 2024). Modern RL infrastructure therefore increas-

ingly separates orchestration, training, and inference, enabling each component to be optimized for its distinct computational workload.

B Proofs and Derivations

B.1 Derivation of Equation 3

We fix the current training state, including the parameter vector θ, and take expectations over queries, sampled responses, and numerical perturbations under a common sampling distribution that does not depend on the weighting rule. Following Section 3, we omit the response index and let $g_t = \nabla_\theta L^{\mathrm{CLIP}}(o_t \mid q, o_{<t}; \theta)$ denote the actor-side token gradient. The number of valid tokens $T \geq 1$ may vary across samples. Recall the decomposition $z_t = \log \rho_t = \bar{z}_t + \epsilon_t$, where $\bar{z}_t$ denotes genuine mismatch and ϵ_t denotes residual numerical noise.

For $\alpha_t \in [0, 1]$ and a fixed truncation threshold $1 \leq C < \infty$, consider the non-normalized tempered weights $w_t = w_t(\boldsymbol{\alpha}) = \min(e^{\alpha_t z_t}, C)$. Define the corresponding update direction and its genuine-mismatch reference as

$$d(\mathbf{w}) = \frac{1}{T}\sum_{t=1}^{T} w_t g_t, \qquad \bar{d} = \frac{1}{T}\sum_{t=1}^{T} \min(e^{\bar{z}_t}, C) g_t. \tag{7}$$

The parameter update is $\theta^+(\mathbf{w}) = \theta + \eta d(\mathbf{w})$, with $\eta > 0$, and the weights are detached during optimization. The reference $\bar{d}$ retains the same sampled tokens and actor-side gradients, replacing only the observed mismatch in the weights by its genuine component. In particular, $\bar{d}$ is an auxiliary reference direction, not the full-precision gradient.

Assumption 2 (Local smoothness)**.** *The full-precision objective J is L-smooth, for some $L > 0$, on a region containing all segments from θ to $\theta^+(\mathbf{w}(\boldsymbol{\alpha}))$ under consideration, almost surely.*

Assumption 3 (Finite second moments)**.** *The token-gradient second moment is finite, and each coefficient rule under consideration has finite recovery error:*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} \|g_t\|^2\right] < \infty, \qquad \mathbb{E}\left[\sum_{t=1}^{T} (\alpha_t z_t - \bar{z}_t)^2\right] < \infty.$$

Assumption 4 (Positive reference alignment)**.** *The reference direction has strictly positive expected alignment with the full-precision gradient:*

$$\nabla J(\theta)^\top \mathbb{E}[\bar{d}] > 0.$$

Positive reference alignment is an explicit assumption and does not follow solely from removing numerical noise from the weights.

Proof. The function $x \mapsto \min(e^x, C)$ is continuous and piecewise differentiable, with derivative in $[0, C]$ wherever it exists. It is therefore C-Lipschitz, giving

$$\left|w_t - \min(e^{\bar{z}_t}, C)\right| \leq C|\alpha_t z_t - \bar{z}_t|.$$

Applying the triangle inequality and Cauchy–Schwarz over the token sum and sampling randomness yields

$$\begin{aligned}
\mathbb{E}\|d(\mathbf{w}) - \bar{d}\| &\leq C\,\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} |\alpha_t z_t - \bar{z}_t|\,\|g_t\|\right] \\
&\leq C\sqrt{\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}(\alpha_t z_t - \bar{z}_t)^2\right]}\sqrt{\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|g_t\|^2\right]} \\
&\leq C\sqrt{\mathbb{E}\left[\sum_{t=1}^{T}(\alpha_t z_t - \bar{z}_t)^2\right]}\sqrt{\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|g_t\|^2\right]},
\end{aligned} \tag{8}$$

where the last inequality uses $T \geq 1$. Thus, the log-mismatch recovery error controls the expected deviation from the reference direction.

Meanwhile, convexity of the squared norm and $0 < w_t \leq C$ imply

$$\mathbb{E}\|d(\mathbf{w})\|^2 \leq \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} w_t^2 \|g_t\|^2\right] \leq C^2 \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} \|g_t\|^2\right]. \tag{9}$$

This establishes a finite update-cost bound that is uniform over the tempered family.

Applying Equation 2 and separating reference alignment from update deviation gives

$$\begin{aligned}
\mathbb{E}\big[J(\theta^+(\mathbf{w})) - J(\theta)\big] &\geq \eta \nabla J(\theta)^\top \mathbb{E}[d(\mathbf{w})] - \frac{L\eta^2}{2}\mathbb{E}\|d(\mathbf{w})\|^2 \\
&\geq \eta \nabla J(\theta)^\top \mathbb{E}[\bar{d}] - \eta \|\nabla J(\theta)\| \, \mathbb{E}\|d(\mathbf{w}) - \bar{d}\| \\
&\quad - \frac{L\eta^2 C^2}{2}\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} \|g_t\|^2\right] \\
&\geq B - K\sqrt{\mathbb{E}\left[\sum_{t=1}^{T}(\alpha_t z_t - \bar{z}_t)^2\right]},
\end{aligned} \tag{10}$$

where

$$B = \eta \nabla J(\theta)^\top \mathbb{E}[\bar{d}] - \frac{L\eta^2 C^2}{2}\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} \|g_t\|^2\right],$$

$$K = \eta C \|\nabla J(\theta)\| \sqrt{\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} \|g_t\|^2\right]}.$$

Both constants are independent of $\boldsymbol{\alpha}$. Positive reference alignment implies that $\|\nabla J(\theta)\| > 0$ and that the token-gradient second moment is strictly positive. Consequently, $K > 0$ for every $\eta > 0$, while $B > 0$ whenever

$$0 < \eta < \frac{2\nabla J(\theta)^\top \mathbb{E}[\bar{d}]}{LC^2 \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} \|g_t\|^2\right]},$$

provided local smoothness holds along the corresponding update segments. This establishes Equation 3. The proof does not require centered numerical noise or independence between numerical noise and token gradients. □

B.2 Proof of Theorem 1

We consider the population optimization in Equation 4 under Assumption 1. We first fix the number of valid tokens $T \geq 1$ and hold each coefficient $\alpha_t \in [0, 1]$ fixed with respect to the realizations inside the expectation. Recall that $z_t = \bar{z}_t + \epsilon_t$, where $\bar{z}_t$ denotes genuine mismatch and ϵ_t denotes residual numerical noise.

Rationale for Assumption 1. The finite-second-moment requirement is consistent with the finite-recovery-error condition in Appendix B.1. At the identity and correction endpoints, $\alpha_t = 0$ and $\alpha_t = 1$, the squared recovery errors reduce to $\bar{z}_t^2$ and ϵ_t^2, respectively. Their integrability therefore ensures that both endpoint risks are well-defined and finite. For conditional centering, when $\bar{z}_t$ denotes the genuine policy mismatch, ϵ_t is the difference between their numerical log-probability errors. If these errors have equal conditional means given $\bar{z}_t$, their biases cancel, yielding $\mathbb{E}[\epsilon_t \mid \bar{z}_t] = 0$ without requiring either backend to be individually unbiased or their errors to be independent. Consequently, $\mathbb{E}[z_t \mid \bar{z}_t] = \bar{z}_t$, while the noise variance may still depend on genuine mismatch. This describes an idealized centered-noise regime and remains a modeling condition rather than a consequence of low-precision execution alone.

Proof. By linearity of expectation,

$$\mathbb{E}\left[\sum_{t=1}^{T}(\alpha_t z_t - \bar{z}_t)^2\right] = \sum_{t=1}^{T}\mathbb{E}[(\alpha_t z_t - \bar{z}_t)^2]. \tag{11}$$

Each summand depends only on α_t, and the constraints $\alpha_t \in [0, 1]$ are coordinatewise. Hence, the optimization separates across tokens without requiring independence across their observations.

Conditional centering and the law of total expectation give

$$\mathbb{E}[\bar{z}_t \epsilon_t] = \mathbb{E}[\bar{z}_t \, \mathbb{E}[\epsilon_t \mid \bar{z}_t]] = 0. \tag{12}$$

Consequently,

$$\mathbb{E}[z_t^2] = \mathbb{E}[\bar{z}_t^2] + \mathbb{E}[\epsilon_t^2]. \tag{13}$$

Expanding $\alpha_t z_t - \bar{z}_t = (\alpha_t - 1)\bar{z}_t + \alpha_t \epsilon_t$ and using Equation 12, we obtain

$$\mathbb{E}[(\alpha_t z_t - \bar{z}_t)^2] = (1 - \alpha_t)^2 \mathbb{E}[\bar{z}_t^2] + \alpha_t^2 \mathbb{E}[\epsilon_t^2]. \tag{14}$$

Thus, the recovery error balances the genuine mismatch left uncorrected against the numerical noise introduced by correction.

For any token with $\mathbb{E}[z_t^2] > 0$, completing the square yields

$$\mathbb{E}[(\alpha_t z_t - \bar{z}_t)^2] = \mathbb{E}[z_t^2] \left(\alpha_t - \frac{\mathbb{E}[\bar{z}_t^2]}{\mathbb{E}[z_t^2]} \right)^2 + \frac{\mathbb{E}[\bar{z}_t^2]\mathbb{E}[\epsilon_t^2]}{\mathbb{E}[z_t^2]}. \tag{15}$$

The second term is independent of α_t, and the first is a quadratic with a strictly positive coefficient. Therefore, its unique minimizer is

$$\alpha_t^\star = \frac{\mathbb{E}[\bar{z}_t^2]}{\mathbb{E}[z_t^2]} = 1 - \frac{\mathbb{E}[\epsilon_t^2]}{\mathbb{E}[z_t^2]}. \tag{16}$$

Equation 13 ensures that $\alpha_t^\star \in [0, 1]$, so this also solves the constrained problem. If $\mathbb{E}[z_t^2] = 0$, both component second moments vanish, and every $\alpha_t \in [0, 1]$ gives zero recovery error. Thus, the minimizer is unique precisely for the non-degenerate tokens specified in Theorem 1.

Applying this result token-wise establishes the theorem. Under the conditions of Appendix B.1, B and $K > 0$ in Equation 3 are independent of $\boldsymbol{\alpha}$. Minimizing the recovery error therefore maximizes that lower bound over fixed population coefficients. This establishes certificate optimality, rather than optimality for the exact full-precision objective or for observation-dependent coefficient rules.

For variable-length responses, each position with positive occurrence probability contributes

$$\Pr(T \geq t) \, \mathbb{E}\big[(\alpha_t z_t - \bar{z}_t)^2 \mid T \geq t\big]$$

to the population objective. Under the conditional version of Assumption 1, the preceding argument applies to this conditional expectation. Since $\Pr(T \geq t)$ is independent of α_t, the unique minimizer for each non-degenerate position is

$$\alpha_t^\star = 1 - \frac{\mathbb{E}[\epsilon_t^2 \mid T \geq t]}{\mathbb{E}[z_t^2 \mid T \geq t]}.$$

Positions with zero occurrence probability or zero conditional second moment impose no restriction on α_t. The finite-value assumption ensures that the coordinatewise minimizers attain a finite global minimum, including when the response length is not uniformly bounded. □

B.3 Derivation of Equation 5

We derive the excess-risk score from the calibrated endpoint comparison and characterize the effect of clipping on token selection. Throughout, we fix the current training state and the lagged calibration tables. For each sampled token, define its minimum support as

$$s_t = \min\big\{ \pi_\theta^{\mathrm{A}}(o_t \mid q, o_{<t}), \pi_\theta^{\mathrm{R}}(o_t \mid q, o_{<t}) \big\}.$$

As in Section 4.2, tokens are grouped by the surprisal $-\log s_t$, and $\hat{\mu}_t$ and $\hat{v}_t \geq 0$ denote the estimated noise mean and variance assigned to the corresponding group. With the calibration tables fixed, these estimates are fixed functions of s_t. Recall that $z_t = \bar{z}_t + \epsilon_t$ and define the centered observation $Y_t = z_t - \hat{\mu}_t$. We retain the finite-second-moment requirement of Assumption 1 and impose the following condition on the calibrated residual.

Assumption 5 (Exact support-conditioned calibration)**.** *For each token t, the calibration estimates satisfy*

$$\mathbb{E}[\epsilon_t - \hat{\mu}_t \mid \bar{z}_t, s_t] = 0, \qquad \mathbb{E}[(\epsilon_t - \hat{\mu}_t)^2 \mid \bar{z}_t, s_t] = \hat{v}_t$$

almost surely.

This assumption requires calibration to remain valid after conditioning jointly on genuine mismatch and support. It does not follow from Assumption 1 or from matching noise moments conditional on support alone.

Proof. Since $Y_t = \bar{z}_t + \epsilon_t - \hat{\mu}_t$, exact calibration gives

$$\begin{aligned}\mathbb{E}[Y_t^2 \mid \bar{z}_t, s_t] &= \bar{z}_t^2 + 2\bar{z}_t\,\mathbb{E}[\epsilon_t - \hat{\mu}_t \mid \bar{z}_t, s_t] \\ &\quad + \mathbb{E}[(\epsilon_t - \hat{\mu}_t)^2 \mid \bar{z}_t, s_t] \\ &= \bar{z}_t^2 + \hat{v}_t.\end{aligned} \tag{17}$$

In the log-mismatch recovery criterion, identity weighting estimates genuine mismatch by 0, whereas the calibrated correction endpoint uses Y_t. Their conditional expected squared errors are $\bar{z}_t^2$ and $\hat{v}_t$, respectively. Thus, the conditional excess risk of identity weighting relative to calibrated correction is $\bar{z}_t^2 - \hat{v}_t$, and Equation 17 yields

$$\mathbb{E}[Y_t^2 - 2\hat{v}_t \mid \bar{z}_t, s_t] = \bar{z}_t^2 - \hat{v}_t. \tag{18}$$

Therefore, $Y_t^2 - 2\hat{v}_t$ is conditionally unbiased for this excess risk under exact calibration. One variance term removes the noise contribution to Y_t^2, while the other accounts for the recovery error of the calibrated correction endpoint.

Averaging Equation 18 gives the population excess risk of the calibrated endpoints, $\mathbb{E}[\bar{z}_t^2] - \mathbb{E}[\hat{v}_t]$. This agrees with D_t in Section 4.2 when $\hat{\mu}_t = 0$ almost surely. With nonzero mean calibration, the relevant correction error is instead that of the centered residual $\epsilon_t - \hat{\mu}_t$. Under approximate calibration, $Y_t^2 - 2\hat{v}_t$ serves as a plug-in proxy without an unbiasedness guarantee.

To limit the influence of extreme observations on token selection, ERIS clips the squared centered mismatch at a fixed threshold τ^2, where $0 < \tau < \infty$:

$$H_t = \min(Y_t^2, \tau^2) - 2\hat{v}_t. \tag{19}$$

Substituting $Y_t = z_t - \hat{\mu}_t$ recovers Equation 5. Here, τ controls score clipping, whereas C bounds the correction weight $\min(e^{Y_t}, C)$. The following identities do not require any prescribed relation between these thresholds.

Using $\min(Y_t^2, \tau^2) = Y_t^2 - \max\{Y_t^2 - \tau^2, 0\}$, exact calibration yields

$$\begin{aligned}\mathbb{E}[H_t \mid \bar{z}_t, s_t] &= \bar{z}_t^2 - \hat{v}_t \\ &\quad - \mathbb{E}\big[\max\{Y_t^2 - \tau^2, 0\} \mid \bar{z}_t, s_t\big].\end{aligned} \tag{20}$$

Clipping therefore introduces a non-positive conditional bias. It regularizes the selection score rather than preserving unbiased excess-risk estimation.

Effect on correction eligibility. For any non-negative variance estimate $\hat{v}_t$, regardless of calibration accuracy,

$$\begin{aligned}H_t > 0 &\iff \min(Y_t^2, \tau^2) > 2\hat{v}_t \\ &\iff Y_t^2 > 2\hat{v}_t \quad \text{and} \quad \hat{v}_t < \frac{\tau^2}{2}.\end{aligned} \tag{21}$$

Consequently, when $\hat{v}_t < \tau^2/2$, the clipped score is positive exactly when the unclipped proxy $Y_t^2 - 2\hat{v}_t$ is positive. Clipping therefore preserves correction eligibility in this low-estimated-noise region. When $\hat{v}_t \geq \tau^2/2$,

$$H_t \leq \tau^2 - 2\hat{v}_t \leq 0.$$

Such tokens are excluded from positive-score TopK selection regardless of their observed mismatch magnitude. Thus, an arbitrarily large mismatch observation cannot override the estimated-noise cutoff.

Preserving eligibility does not imply preserving the final budgeted selection, since clipping can change the ranking of eligible tokens or introduce ties. In particular, once $Y_t^2 \geq \tau^2$, the score becomes $\tau^2 - 2\hat{v}_t$, so further increases in mismatch magnitude cannot increase selection priority at a fixed variance estimate. Score clipping therefore restricts eligibility to sufficiently low estimated noise while limiting the influence of extreme mismatch observations on subsequent ranking. □

B.4 Justification of the Correction Budget

A positive excess-risk score favors correction under the log-mismatch recovery proxy, but does not directly control the magnitude of the aggregated update. Truncation bounds individual weights, while the correction budget limits how many weights can depart from identity. We make this connection explicit through the update-cost term in Equation 2.

We fix the current training state and take expectations over queries, sampled responses, and numerical perturbations. For $T \geq 1$ valid tokens, let $k = \mathrm{round}(pT)$ with $p \in [0, 1]$. Following Section 4.2, ERIS uses the non-normalized weights

$$w_t = 1 + \alpha_t \left[\min(e^{z_t - \hat{\mu}_t}, C) - 1\right], \qquad \alpha_t \in \{0, 1\}, \qquad \sum_{t=1}^{T} \alpha_t \leq k,$$

where $1 \leq C < \infty$ and $\hat{\mu}_t$ is the estimated noise mean assigned to token t. With $g_t = \nabla_\theta L^{\mathrm{CLIP}}(o_t \mid q, o_{<t}; \theta)$ denoting the actor-side token gradient, the update direction is $d(\mathbf{w}) = T^{-1} \sum_{t=1}^{T} w_t g_t$. The correction weights are detached during optimization.

Under the finite token-gradient second-moment condition in Assumption 3, the update-cost term satisfies

$$\frac{L\eta^2}{2} \mathbb{E}\|d(\mathbf{w})\|^2 \leq \frac{L\eta^2}{2} \mathbb{E}\left[\left(1 + \frac{k}{T}(C^2 - 1)\right) \frac{1}{T} \sum_{t=1}^{T} \|g_t\|^2\right], \tag{22}$$

where L and η are the smoothness constant and step size in Equation 2. This bound holds for any selection satisfying the budget, regardless of calibration accuracy.

Proof. Since α_t is binary, an unselected token has unit weight, whereas a selected token has weight at most C. Consequently,

$$\begin{aligned} \frac{1}{T} \sum_{t=1}^{T} w_t^2 &= 1 + \frac{1}{T} \sum_{t=1}^{T} \alpha_t \left[\min(e^{z_t - \hat{\mu}_t}, C)^2 - 1\right] \\ &\leq 1 + \frac{C^2 - 1}{T} \sum_{t=1}^{T} \alpha_t \\ &\leq 1 + \frac{k}{T}(C^2 - 1), \end{aligned} \tag{23}$$

where the last inequality uses $C \geq 1$ and $\sum_{t=1}^{T} \alpha_t \leq k$. Thus, the budget controls the mean squared weight independently of which tokens are selected.

Applying Cauchy–Schwarz to the token-mean update gives

$$\begin{aligned} \|d(\mathbf{w})\|^2 &= \left\| \frac{1}{T} \sum_{t=1}^{T} w_t g_t \right\|^2 \\ &\leq \left[\frac{1}{T} \sum_{t=1}^{T} w_t^2\right] \left[\frac{1}{T} \sum_{t=1}^{T} \|g_t\|^2\right] \\ &\leq \left[1 + \frac{k}{T}(C^2 - 1)\right] \frac{1}{T} \sum_{t=1}^{T} \|g_t\|^2. \end{aligned} \tag{24}$$

Taking expectations and multiplying by $L\eta^2/2$ establishes Equation 22. Since $0 \leq k/T \leq 1$, the multiplicative factor is at most C^2, ensuring finiteness under the stated moment condition. These

Table 4: Training hyperparameter configurations used in our experiments.

	Hyperparameter	Value
Data	max_prompt_length	1024
	max_response_length	8192 / 10240
	apply_chat_template_kwargs.enable_thinking	False
Actor	optim.lr	1×10^{-6}
	ppo_mini_batch_size	64
	clip_ratio_low	0.20
	clip_ratio_high	0.28
	use_kl_loss	False
Rollout	n	8
	temperature	1.0
	val_kwargs.n	16
	val_kwargs.temperature	1.0
	val_kwargs.top_p	0.95
	max_model_len	17408
Reward	overlong_buffer_cfg.len	1024
	overlong_buffer_cfg.penalty_factor	1.0
Trainer	nnodes	1
	n_gpus_per_node	8
	total_epochs	1

Table 5: Rollout correction configurations for different methods.

Method	Hyperparameter	Value
TIS	rollout_is	token
	rollout_is_threshold	2
	rollout_is_mode	standard
	rollout_rs	null
	rollout_rs_threshold	null
MIS	rollout_is	sequence
	rollout_is_threshold	2
	rollout_is_mode	standard
	rollout_rs	seq_sum_k1
	rollout_rs_threshold	0.5_2.0
ERIS	rollout_is	token
	rollout_is_threshold	2
	rollout_is_mode	excess_risk
	rollout_rs	null
	rollout_rs_threshold	null
	excess_risk_log_clip (τ)	$\log 2$

inequalities hold for every realization and therefore accommodate data-dependent selection and variable T without requiring independence between correction weights and token gradients.

With truncation alone, allowing all tokens to be corrected gives the coefficient C^2 in this uniform bound. The budget replaces it by $1 + (k/T)(C^2 - 1)$, which is strictly smaller when $C > 1$ and $k < T$. At $k = 0$, all weights equal 1, and the coefficient reduces to the corresponding identity-weighting value. Thus, truncation ensures a finite update cost, while the budget tightens its upper bound without lowering the per-token truncation threshold.

Selection within the budget. For fixed observed scores, the positive-score TopK rule in Section 4.2 solves the empirical proxy problem

$$\max_{\boldsymbol{\alpha} \in \{0,1\}^T} \sum_{t=1}^{T} \alpha_t H_t \qquad \text{s.t.} \quad \sum_{t=1}^{T} \alpha_t \leq k. \tag{25}$$

Selecting a negative score decreases the objective, while selecting a zero score leaves it unchanged. Whenever capacity remains, adding an unselected positive score increases the objective. Likewise, replacing a selected smaller score with an unselected larger positive score improves the objective. Hence, selecting the largest positive scores up to the budget, with arbitrary tie-breaking, is optimal. The constraint does not require exhausting the budget, and ERIS selects fewer than k tokens when fewer than k scores are positive.

The score and budget therefore serve complementary purposes: the score prioritizes corrections by estimated benefit, while the budget controls a uniform upper bound on the update cost. A smaller budget need not produce a smaller realized update or greater full-precision improvement, since it also changes gradient alignment and may exclude useful corrections. Likewise, TopK optimality applies to the additive score proxy, not to the exact full-precision objective. □

C Detailed Experimental Setup

C.1 Training Setup

Framework. We conduct all training experiments on the VeRL (Sheng et al., 2024) framework with the DAPO (Yu et al., 2025) algorithm. We use Megatron-LM (Shoeybi et al., 2020) for training and vLLM (Kwon et al., 2023) for rollout generation.

Mathematical Reasoning Evaluation Prompt Template

{PROBLEM} Please reason step by step, and put your final answer within \boxed{}.

General-Knowledge Evaluation Prompt Template

{PROBLEM}

Please reason step by step, and put only the letter of the correct answer within \boxed{} at the end.

Figure 6: Prompt templates used for mathematical reasoning and general-knowledge evaluation. The {PROBLEM} placeholder is replaced by the original benchmark question, including answer choices for multiple-choice benchmarks.

Training hyperparameter configuration. The detailed training hyperparameters are summarized in Table 4. We set the maximum prompt length to 1,024 tokens and use a maximum response length of 10,240 tokens for Qwen3-8B and 8,192 tokens for all other models. The actor is optimized with a learning rate of 1×10^{-6} and a PPO mini-batch size of 64, with clipping ratios $\epsilon_{\text{low}} = 0.2$ and $\epsilon_{\text{high}} = 0.28$, while the KL loss is disabled. For rollout generation, we sample 8 responses per prompt with a temperature of 1.0. During validation, we sample 16 responses with a temperature of 1.0 and nucleus sampling at 0.95. We set the maximum model length to 17,408 tokens and use an overlong-response buffer of 1,024 tokens with a penalty factor of 1. All models are trained for one epoch on a single node with 8 NVIDIA RTX PRO 6000 GPUs.

Rollout correction configuration. Table 5 reports the correction-specific configurations. TIS applies token-level importance sampling with a truncation threshold of 2 under the standard correction mode. MIS instead uses sequence-level importance sampling with the same threshold and enables seq_sum_k1 rejection sampling with threshold 0.5_2.0. ERIS retains token-level importance sampling and the threshold of 2, but replaces the standard correction rule with the proposed excess-risk mode, while rejection sampling is disabled. The clipping threshold for H_t is set to $\log 2$.

C.2 Evaluation Setup

Framework and hyperparameter configurations. We use vLLM (Kwon et al., 2023) for response generation across all evaluation benchmarks. We summarize the evaluation hyperparameter configuration and prompt templates in Table 6 and Figure 6, respectively. We sample 16 responses per problem with a temperature of 0.7, nucleus sampling at 0.95, and a maximum generation length of 16,384 tokens. We disable the model-specific thinking mode during evaluation. We use task-specific prompt templates for different benchmark categories. Mathematical reasoning benchmarks directly request step-by-step reasoning followed by a boxed final answer, while general-knowledge benchmarks use a multiple-choice template that requires the final option letter to be enclosed in a box. For each benchmark, we report the Pass@1 accuracy, computed as the average accuracy over the 16 generated responses per problem.

Table 6: Inference hyperparameter configuration used for evaluation.

Hyperparameter	Value
# responses per problem	16
temperature	0.7
top_p	0.95
max_new_tokens	16384
enable_thinking	False

Dataset curation. We construct the MMLU-4 evaluation set using the `global facts`, `nutrition`, `public relations`, and `sociology` subsets from MMLU, removing two duplicate examples from the `public relations` subset. All benchmarks are downloaded from the Hugging Face Hub [1] and processed into a unified data structure.

[1] https://huggingface.co/

C.3 Implementation Details of ERIS

Token-wise Support-conditioned Calibration. Following each policy synchronization, the updated actor parameters are transferred to the rollout engine, so both backends represent the same underlying policy. We use the subsequent aligned evaluation for calibration, where genuine policy mismatch is largely removed and the observed log-ratio approximates numerical noise, $z_t \approx \epsilon_t$. For each valid token t, we compute the minimum support $s_t = \min(\pi_t^{\mathrm{A}}, \pi_t^{\mathrm{R}})$ and surprisal $u_t = -\log s_t$. We stratify tokens into $N = 8$ fixed-width buckets according to u_t, with the final bucket absorbing the remaining tail. Within each bucket, we estimate the conditional mean and variance of ϵ_t and maintain $\hat{\mu}(s_t)$ and $\hat{v}(s_t)$ using an exponential moving average (EMA) with $\beta = 0.9$. The statistics are initialized to zero, with the first valid observation initializing each bucket and subsequent observations incorporated through EMA. These conditional estimates are then assigned to tokens according to their support buckets for excess-risk estimation.

Non-Compulsory Correction Budget. We implement the correction budget as an upper bound rather than a fixed correction quota. For each training shard containing T valid response tokens, we set $k = \mathrm{round}(pT)$, where p specifies the maximum fraction of tokens that may receive TIS correction and is by default set to 0.2 in our setup. We first restrict the candidate set to tokens with positive excess risk, $H_t > 0$. When the number of eligible tokens exceeds k, we rank them using the excess-risk score and apply correction only to the top-k tokens. Importantly, the budget is non-compulsory: if fewer than k tokens have positive excess risk, all eligible tokens are corrected and the remaining budget is left unused instead of filling with tokens having $H_t \leq 0$. All unselected valid tokens retain identity weighting.

Clipped Excess-Risk Estimation. To reduce the sensitivity of the excess-risk estimate to extreme numerical discrepancies, we clip the magnitude of the calibrated mismatch before computing H_t. Specifically, given the centered mismatch $Y_t = z_t - \hat{\mu}(s_t)$, we use $H_t = \min\big(Y_t^2, \tau^2\big) - 2\hat{v}(s_t)$. Accordingly, mismatch magnitudes beyond τ contribute the same bounded term to the excess-risk estimate, preventing extreme residuals from disproportionately affecting correction decisions. In our default configuration, we use $\tau = \log 2$.

C.4 Measuring the Full-Precision Policy for Figure 4

To construct Figure 4, we use a fixed batch of prompts across all compared methods. For each method, we first load its final-step checkpoint on the rollout backend in BF16, using this execution as the full-precision reference policy π_θ. We generate responses and record their token-level reference log-probabilities $\log \pi_\theta(o_t \mid c_t)$. We then load the same checkpoint in FP8 on both the rollout and actor backends and perform teacher-forced evaluation on these fixed responses, obtaining $\log \pi_\theta^{\mathrm{R}}(o_t \mid c_t)$ and $\log \pi_\theta^{\mathrm{A}}(o_t \mid c_t)$, respectively. Given the shared token and context arguments, we measure the actor-rollout mismatch along the horizontal axis as

$$\left|\log \pi_\theta^{\mathrm{A}} - \log \pi_\theta^{\mathrm{R}}\right|,$$

and define the full-precision mismatch along the vertical axis as the root mean square deviation of the two low-precision policies from the common full-precision reference,

$$\sqrt{\frac{1}{2}\left(\log \pi_\theta^{\mathrm{A}} - \log \pi_\theta\right)^2 + \frac{1}{2}\left(\log \pi_\theta^{\mathrm{R}} - \log \pi_\theta\right)^2}.$$

Each method is therefore compared against the BF16 execution of its own checkpoint, with responses held fixed across precision settings within each method.

C.5 Reference-Based Oracle for Figure 5

For Figure 5, we use a token-wise reference-based oracle Δ_t^{ref}, distinct from the theoretical population excess risk D_t. To obtain the reference, we evaluate the final-step checkpoints in BF16 on actor training and rollout backends. We generate responses from a fixed batch of prompts on the rollout backend and evaluate the same responses on the actor training backend, using the resulting BF16 log-probability discrepancy as a reference estimate of the stable mismatch $\bar{z}_t$. We then evaluate the respective checkpoints in FP8 with teacher forcing on the same responses, obtaining the observed mismatch $z_t = \log \pi_\theta^{\mathrm{A}}(o_t \mid c_t) - \log \pi_\theta^{\mathrm{R}}(o_t \mid c_t)$.

We define the reference-based oracle as

$$\Delta_t^{\text{ref}} = \bar{z}_t^2 - (z_t - \bar{z}_t)^2 ,$$

where $\Delta_t^{\text{ref}} > 0$ indicates that the correction endpoint z_t has lower squared recovery error than the identity endpoint 0 relative to the BF16 reference. Unlike D_t, which compares population risks, Δ_t^{ref} compares realized token-wise errors against an empirical reference.

For a token selection set $\mathcal{S}$, we compute

$$\text{Enrichment}(\mathcal{S}) = \frac{\Pr(\Delta_t^{\text{ref}} > 0 \mid t \in \mathcal{S})}{\Pr(\Delta_t^{\text{ref}} > 0)}.$$

We compare the set of tokens with $H_t > 0$ against a mismatch-only set containing tokens with the largest observed mismatch $|z_t|$. For a fair comparison, both sets contain the same number of tokens. The all-token baseline has enrichment 1 by definition.

D Extended Experimental Results

D.1 Token-dependent Support Calibration

We examine the numerical-noise variance estimated by ERIS across minimum-support surprisal buckets. As shown in Figure 7, the estimated numerical-noise variance increases sharply with $-\log s_t$, spanning several orders of magnitude across support regions. This strong heterogeneity motivates the support-conditioned calibration in ERIS rather than a single global noise estimate. Moreover, Equation 5 implies that a positive excess-risk score requires $\hat{v}(s_t) < \tau^2/2$. With $\tau = \log 2$ used throughout training, only the highest-support bucket satisfies this condition, indicating that positive H_t values are concentrated on tokens to which both actor and rollout policies assign non-negligible probability. ERIS therefore preferentially corrects mismatches in well-supported regions while suppressing corrections where numerical uncertainty dominates.

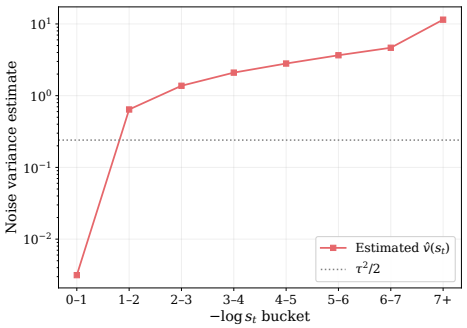


Figure 7: Calibration statistics of noise variance.

D.2 Semantic Analysis of Selected Tokens

We examine the semantic characteristics of tokens selected by ERIS. As shown in Figure 8, although ERIS relies solely on numerical reliability rather than semantic information, its token selection exhibits a clear semantic structure. Selected tokens are predominantly associated with reasoning and deliberation, reflecting uncertainty and exploration during problem solving. In contrast, unselected tokens are largely concentrated on mathematical notation, numerical quantities, variables, and other symbolic expressions. This distinction suggests that reliable actor-rollout mismatch tends to concentrate at semantically meaningful reasoning positions, while deterministic symbolic generation requires less correction. This semantic separation provides qualitative evidence that ERIS can identify distinct regions of the reasoning trajectory without semantic supervision, supporting the advantage of selective correction over uniform token correction.



Figure 8: Token-level semantic analysis of ERIS on selected and unselected tokens.

D.3 Generalization of ERIS on BF16

We further evaluate the applicability of ERIS to BF16 training on Qwen3-4B. As shown in Table 7, TIS improves the Math score from 35.91 to 36.25, while the General score remains nearly un-

Table 7: Performance comparison on Qwen3-4B across different training methods on mathematical reasoning and general knowledge benchmarks with BF16 training. The best results are marked in **bold**, and second-best results are underlined.

Model	Method	Mathematical Reasoning					General Knowledge				
		AIME24	AIME25	AMC23	Olymp.	Avg.	ARC-E	ARC-C	S-GPQA	MMLU-4	Avg.
Qwen3 4B	BF16	19.17	17.08	61.41	45.98	35.91	96.33	90.02	29.54	70.76	71.66
	TIS	18.96	15.42	63.44	**47.19**	36.25	96.16	90.22	28.90	71.28	71.64
	MIS	16.67	18.33	63.28	45.57	35.96	96.03	89.84	29.56	**71.68**	71.78
	ERIS	**21.67**	**20.00**	**63.75**	45.36	**37.70**	**96.43**	**90.36**	**30.09**	70.32	**71.80**

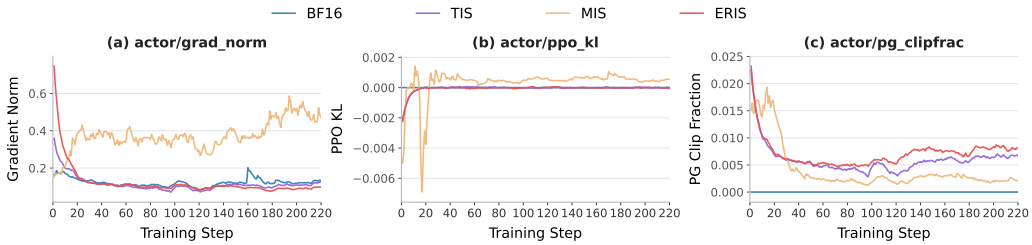


Figure 9: Training dynamics of BF16 and FP8 training with TIS, MIS, and ERIS, showing (a) actor gradient norm, (b) PPO KL, and (c) policy-gradient clipping fraction.

changed at 71.64 compared with 71.66 for standard BF16. MIS achieves 35.96 on Math and 71.78 on General, providing only limited improvements over the standard BF16 baseline. In comparison, ERIS yields stronger gains, increasing the Math score to **37.70** and the General score to **71.80**, outperforming standard BF16 as well as both importance-sampling baselines. Although BF16 provides substantially higher numerical precision than FP8, it remains a reduced-precision format relative to FP32 and may still introduce finite-precision discrepancies. These results show that ERIS can effectively address such residual numerical discrepancies in BF16 training, demonstrating its applicability across different precision formats.

D.4 Training Dynamics

We compare the training dynamics of different methods using the actor gradient norm, PPO KL, and policy-gradient clipping fraction. As shown in Figure 9, ERIS maintains a low and smooth gradient norm throughout training, closely following BF16 and TIS, whereas MIS exhibits substantially larger gradient norms and stronger fluctuations, especially in the later stages. A similar trend is observed in PPO KL, where ERIS remains consistently close to zero, while MIS shows pronounced fluctuations in the early stages and a persistently larger deviation thereafter, indicating more aggressive policy updates. For the clipping fraction, ERIS follows a smooth trajectory without the large early-stage variations observed for MIS. Although MIS reaches a lower clipping fraction later in training, this alone does not imply more stable optimization, given its substantially larger gradient norms and PPO KL deviations. Overall, these results show that ERIS maintains more stable and controlled updates throughout low-precision RL training.

D.5 Effect of Training Infrastructure

We evaluate the sensitivity of different methods to training infrastructure by replacing vLLM with SGLang as the rollout backend in additional Qwen3-4B experiments, assessing whether ERIS retains its performance gains across rollout implementations. As shown in Table 8, standard FP8 training also diverges under SGLang. TIS achieves average scores of 34.76 on Math and 70.86 on General benchmarks, approaching the corresponding BF16 scores of 35.66 and 71.42 but remaining below ERIS in both categories. MIS yields substantially lower average scores of 28.19 and 69.14, respectively, with its Math performance further degrading from 30.66 under vLLM to 28.19 under SGLang. In contrast, ERIS achieves the highest average scores of 38.65 on Math and 71.84 on Gen-

Table 8: Performance comparison on Qwen3-4B across different training methods on mathematical reasoning and general knowledge benchmarks using SGLang as the rollout backend. The best results are marked in **bold**, and second-best results are underlined. Divergent runs are reported with "–".

Model	Method	Mathematical Reasoning					General Knowledge				
		AIME24	AIME25	AMC23	Olymp.	Avg.	ARC-E	ARC-C	S-GPQA	MMLU-4	Avg.
Qwen3 4B	BF16	20.00	15.00	60.94	**46.70**	35.66	96.08	89.69	**29.59**	70.32	71.42
	FP8	–	–	–	–	–	–	–	–	–	–
	TIS	19.38	16.88	57.34	45.44	34.76	95.97	89.71	27.61	70.13	70.86
	MIS	14.79	10.42	47.50	40.06	28.19	95.10	88.20	26.60	66.67	69.14
	ERIS	**22.50**	**23.33**	**64.06**	44.72	**38.65**	**96.53**	**90.40**	28.69	**71.74**	**71.84**

eral, exceeding BF16 by 2.99 and 0.42 points, respectively. These results show that ERIS maintains consistent performance gains across both rollout backends, supporting its robustness to different training infrastructure.

D.6 Training Efficiency

We quantify the efficiency gains of ERIS in Figure 10 using per-GPU end-to-end training throughput, measured as the number of processed tokens per second (TPS). In addition to improving model performance under low-precision training, ERIS preserves most of the computational advantage of standard FP8 execution while consistently outperforming BF16 in throughput. On Qwen3-1.7B, ERIS achieves 1021 TPS, compared with 997 TPS for BF16 and 1007 TPS for FP8, corresponding to a 1.02× speedup over BF16. The gains become more pronounced at larger scales. ERIS reaches 844 TPS on Qwen3-4B and 695 TPS on Qwen3-8B, yielding 1.16× and 1.17× speedups over BF16, respectively, while remaining close to the corresponding FP8 throughputs of 863 and 720 TPS. A similar trend holds for Qwen2.5-7B, where ERIS achieves 685 TPS compared with 625 TPS for BF16 and 697 TPS for FP8, providing a 1.10× speedup over BF16. These results demonstrate that ERIS introduces only modest overhead relative to standard FP8 execution while retaining substantial throughput improvements over BF16, preserving the efficiency benefits of low-precision training.

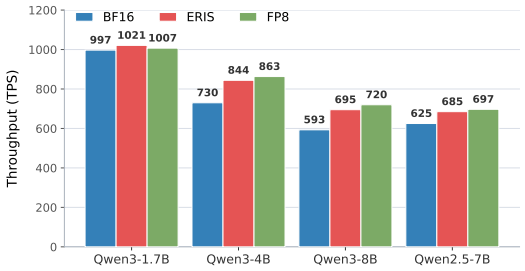


Figure 10: End-to-end training throughput of BF16, FP8 and ERIS across different model scales and architectures.

E Limitations

Our work has several limitations. First, we focus on dense architectures and have not yet evaluated ERIS on other architectural designs, such as Mixture-of-Experts (MoE) models or alternative attention variants. Second, our evaluation is limited to mathematical reasoning tasks. Whether the observed benefits extend to other RL settings, including code generation and multi-turn agentic tasks, remains to be validated. Third, we have not tested ERIS on more aggressive low-precision formats such as FP4. Finally, our work addresses the stability issue in low-precision RL primarily from an algorithmic perspective. System-level remedies that reduce numerical discrepancies through unified quantization kernels, execution paths, or training and rollout backends are complementary to ERIS but remain outside the scope of this study.