ACU-RL: Adaptive Correction and Update Control for Low-Precision LLM Reinforcement Learning

Anonymous authors
Paper under double-blind review

Abstract

Low-precision rollout can reduce the dominant generation cost of reinforcement learning for large language models, but it also creates a policy mismatch between the quantized behavior policy and the BF16 learner. Existing truncated importance sampling (TIS) mitigates this mismatch by reweighting sampled tokens, yet it can strongly attenuate tokens favored by the quantized rollout policy. We find that relaxing this lower-side attenuation improves early learning but can amplify late-stage policy drift, revealing that sample correction and actor stabilization must be controlled jointly. To address this trade-off, we propose Adaptive Correction and Update Control for RL (ACU-RL), which couples Adaptive Two-Sided Truncated Importance Sampling (AT-TIS) with risk-gated actor-update control. AT-TIS adapts the correction strength while retaining more of the gradient contributions from tokens favored by the quantized rollout policy. Complementing this sample-side correction, risk-gated actor-update control increases drift regularization when persistent excess mismatch coincides with degradation signals and reduces the policy-gradient multiplier when the mismatch becomes more severe. ACU-RL requires no additional sampling or model forward passes. We evaluate ACU-RL on Qwen2.5-Omni-3B and Qwen3-8B-Instruct across six multimodal and four mathematical reasoning benchmarks, respectively. It achieves average accuracies of 45.26% and 60.82%, compared with 44.72% and 58.47% for BF16 RL. NVFP4 rollout further provides a $1.17\times$ end-to-end training speedup.

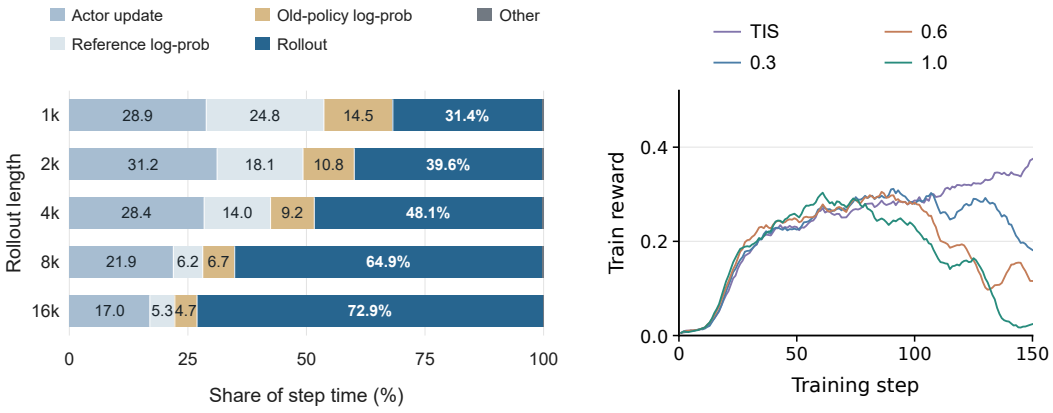
1 Introduction

Reinforcement learning (RL) has become a dominant post-training paradigm for improving the reasoning capabilities of large language models (Shao et al., 2024; Yu et al., 2025). Autoregressive response generation is costly: in our BF16 Qwen3-8B measurements, rollout accounts for a substantial share of per-step training time (Figure 1(a)). Using low precision for rollout generation offers a way to reduce this cost.

We study a setting in which the rollout engine generates responses in NVFP4 while the actor is trained in BF16. Even with synchronized parameters, quantization can change token probabilities, causing the policy that generates training data to differ from the policy used to compute updates. This is a form of rollout–training mismatch. Figure 2 shows that when NVFP4 rollout is directly used with BF16 training on Qwen3-8B-Instruct, the training performance begins to fall after about 100 steps. A common approach to mitigating this issue is truncated importance sampling (TIS), which reweights sampled-token contributions using the training-to-rollout probability ratio while capping excessively large weights (Yao et al., 2025). In this setting, TIS maintains higher training reward until approximately step 330, but still degrades thereafter. Its mean reward over the first 50 steps is 0.56, below the 0.66 achieved by uncorrected NVFP4. Thus, fixed correction can delay degradation without ensuring stability throughout training, and it can also alter early learning. These observations motivate us to examine how mismatch correction changes the contribution of sampled tokens to the policy update.

Quantization-induced differences need not be uniformly harmful to learning. AIS argues that correction strength should adapt across training stages to balance potentially useful exploration against destabilizing bias (Zhou et al., 2026). In quantized LoRA-based RL, QeRL links quantization noise to higher policy entropy and improved exploration (Huang et al., 2026). These findings motivate

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.



(a) Rollout cost (b) Lower-bound truncation

Figure 1: (a) **Rollout generation dominates the RL training latency.** When the rollout length is larger than 8k, rollout will take $> 64.9\%$ of the total latency, making it the primary bottleneck. (b) **Fixed lower-bound truncation improves early reward but can worsen later training.** Qwen2.5-Omni-3B uses BF16 actor training and NVFP4 rollout. We compare TIS with lower truncation thresholds of 0.3, 0.6, and 1 under a fixed upper TIS cap.

examining how importance correction affects the learning signal carried by samples from a quantized rollout policy. We focus on whether limiting the attenuation of sampled-token contributions improves early learning under NVFP4 rollout with BF16 actor training.

To examine the effect of lower-side attenuation, we vary the lower truncation threshold ℓ while keeping the upper cap fixed at $C = 5$, using $w_{i,t} = \text{clip}(\rho_{i,t}^{\text{rt}}, \ell, 5)$. Figure 1(b) shows that raising the lower threshold can improve early training reward in the tested configurations, but is also followed by substantial degradation and fluctuations. These observations motivate limiting lower-side attenuation while adapting correction and controlling actor updates over training.

Based on these observations, we propose ACU-RL for adaptive correction and update control under NVFP4 rollout and BF16 training. Adaptive Two-Sided Truncated Importance Sampling (AT-TIS) adjusts correction strength while bounding importance weights from below and above to limit attenuation and amplification. Retaining small-ratio contributions introduces additional correction bias, motivating update control when mismatch persists and degradation risk rises. Policy-gradient (PG) scaling reduces the reward-driven gradient contribution, while policy-drift regularization penalizes log-probability differences between the current actor and the stored rollout policy.

Under BF16 evaluation, ACU-RL achieves 45.26% average accuracy across six multimodal benchmarks on Qwen2.5-Omni-3B, compared with 44.72% for BF16 RL. On Qwen3-8B-Instruct, it achieves 60.82% across four mathematical reasoning benchmarks, exceeding BF16 RL and fixed TIS by 2.35 and 5.44 percentage points, respectively. Training curves show higher early reward and better preservation of late-stage learning gains than fixed TIS. NVFP4 rollout yields a 1.166× end-to-end training speedup on Qwen3-8B.

Our main contributions are as follows:

1. **We identify a correction–drift trade-off in low-precision rollout.** Relaxing lower-side importance attenuation preserves early learning signals but can increase late-stage actor drift, showing that adaptive reweighting alone is insufficient.
2. **We introduce ACU-RL, which separates sample-side correction from actor-side stabilization.** AT-TIS performs state-dependent log-space tempering of two-sided importance weights, while risk-gated update control intervenes only under persistent mismatch and degradation.
3. **We demonstrate consistent quality–efficiency trade-offs across multimodal and mathematical reasoning.** ACU-RL numerically exceeds the corresponding BF16 averages in both evaluated

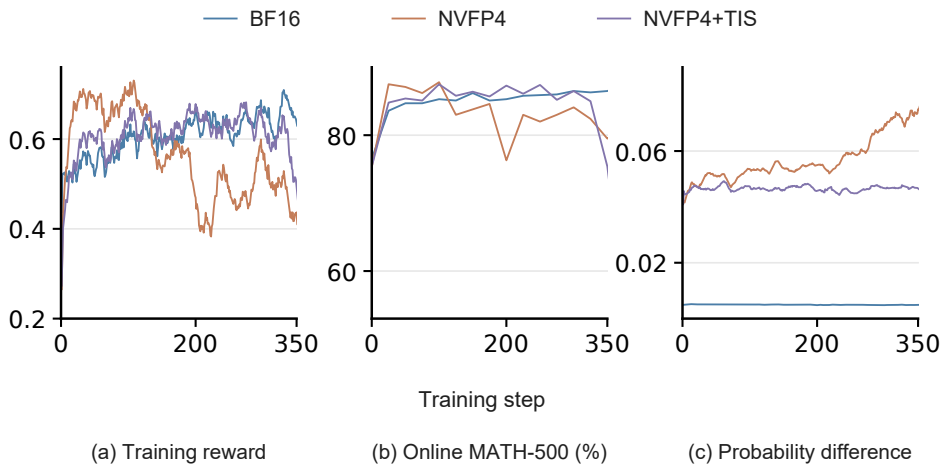


Figure 2: **TIS delays but does not prevent reward degradation under NVFP4 rollout.** Qwen3-8B-Instruct uses BF16 actor training with BF16, NVFP4, or NVFP4+TIS rollout. Panels show (a) training reward, (b) online MATH-500 accuracy under each rollout configuration, and (c) sampled-token rollout–training probability difference.

settings without additional sampling or model forward passes, while NVFP4 rollout provides a 1.17× end-to-end speedup.

2 Preliminaries

2.1 Group Relative Policy Optimization

Group Relative Policy Optimization (GRPO) (Shao et al., 2024) obtains advantages from the relative rewards of responses to the same prompt and updates the actor with a clipped policy ratio. At round k, let $\pi_{\text{old},k} = \pi_{\theta_k}$ denote the BF16 policy before actor updates. For a batch of B responses, let $A_{i,t}$ be the advantage of token t in response o_i; with response-level rewards, valid tokens in the same response share an advantage. We write $s_{i,t} = (q_i, o_{i,<t})$ for the preceding context and $r_{i,t}^{\text{ppo}}(\theta) = \pi_\theta(o_{i,t} \mid s_{i,t}) / \pi_{\text{old},k}(o_{i,t} \mid s_{i,t})$ for the policy-update ratio. With a valid-token mask $m_{i,t}$ and $T_i = \sum_t m_{i,t} > 0$, define sequence-balanced averaging by

$$\langle f \rangle_k = \frac{1}{B} \sum_{i=1}^{B} \frac{1}{T_i} \sum_t m_{i,t} f_{i,t}.$$

Writing the clipped objective with optional nonnegative token weights $w_{i,t}$ gives

$$\mathcal{L}_{\text{PG}}(\theta; w) = -\left\langle w_{i,t} \min\left[r_{i,t}^{\text{ppo}}(\theta) A_{i,t}, \text{clip}\left(r_{i,t}^{\text{ppo}}(\theta), 1-\epsilon, 1+\epsilon\right) A_{i,t} \right] \right\rangle_k . \tag{1}$$

Standard GRPO uses $w_{i,t} = 1$; ϵ is the PPO clipping parameter (Schulman et al., 2017). The advantages and weights are fixed during each actor update.

2.2 Rollout–Training Mismatch and Truncated Importance Sampling

Modern RL systems often use separate backends for rollout generation and actor training. Even when both backends use the same model parameters, differences in their numerical computations can produce different token probabilities. This discrepancy, termed *rollout–training mismatch*, makes the rollout data off-policy with respect to the corresponding training-side policy (Yao et al., 2025). The mismatch can be further amplified by rollout quantization.

In the low-precision rollout setting, let $\pi_{\text{old},k}$ denote the old actor policy evaluated at training precision. At the start of round k, the rollout engine loads the old actor's parameters and applies quantization, yielding the behavior policy μ_k. Quantization perturbs the forward computation and can change

the output logits and token probabilities. Consequently, μ_k can differ from $\pi_{\text{old},k}$ even though both originate from the same parameter checkpoint, before any actor update takes place (Zhou et al., 2026).

Responses are sampled as $o_i \sim \mu_k(\cdot \mid q_i)$, while actor updates are computed at higher precision. For a sampled token $o_{i,t}$ with context $s_{i,t}$, define the rollout–training ratio and its relation to the policy-update ratio as

$$\rho_{i,t}^{\text{rt}} = \frac{\pi_{\text{old},k}(o_{i,t} \mid s_{i,t})}{\mu_k(o_{i,t} \mid s_{i,t})}, \qquad \frac{\pi_\theta(o_{i,t} \mid s_{i,t})}{\mu_k(o_{i,t} \mid s_{i,t})} = r_{i,t}^{\text{ppo}}(\theta)\rho_{i,t}^{\text{rt}}. \tag{2}$$

Here, $\rho_{i,t}^{\text{rt}}$ captures the mismatch between the quantized rollout policy and the old actor evaluated at training precision, whereas $r_{i,t}^{\text{ppo}}(\theta)$ captures subsequent changes to the actor during optimization. The former remains fixed within the round; the latter changes as the actor is updated.

An uncorrected GRPO update sets $w_{i,t} = 1$ in Equation (1), without correcting for the difference between the quantized sampling policy μ_k and $\pi_{\text{old},k}$. A response-level importance ratio multiplies token ratios across the response and can exhibit high variance for long outputs. As a practical approximation, token-level truncated importance sampling (TIS) reweights sampled-token contributions while capping large correction ratios (Yao et al., 2025):

$$w_{i,t}^{\text{TIS}} = \min\left(\rho_{i,t}^{\text{rt}}, C\right), \qquad C > 1. \tag{3}$$

3 Method

ACU-RL combines Adaptive Two-Sided Truncated Importance Sampling (AT-TIS) with actor-update control to address rollout–training mismatch under low-precision rollouts. Section 3.1 introduces the two-sided weighting in AT-TIS to limit the attenuation of sampled-token contributions, and Section 3.2 adapts its correction strength using sequence mismatch and changes in advantage-weighted contributions. To manage the policy-drift risk associated with the additional correction bias introduced by the lower weight bound, Section 3.3 develops risk-gated policy-gradient scaling and policy-drift regularization. Section 3.4 integrates these components into the joint training objective and specifies how they are applied during actor optimization.

3.1 Two-Sided Truncated Importance Sampling

Let $\ell_{i,t}(\theta)$ denote the negative clipped PPO surrogate inside Equation (1). Since correction weights are fixed during actor optimization, at differentiable points,

$$\nabla_\theta[w_{i,t}\ell_{i,t}(\theta)] = w_{i,t}\nabla_\theta\ell_{i,t}(\theta).$$

Thus, TIS scales each token's gradient contribution by $\min(\rho_{i,t}^{\text{rt}}, C)$. For $\rho_{i,t}^{\text{rt}} < 1$, this factor can approach zero, whereas amplification is bounded above by C. QeRL reports that quantization noise can increase policy entropy and improve exploration in LoRA-based RL (Huang et al., 2026). Together with the attenuation analysis above, this finding motivates examining whether preserving more of the sampled-token contributions improves early learning. We vary the lower truncation threshold while keeping the upper cap fixed and find that limiting attenuation can improve early training performance (Figure 1). We choose reciprocal bounds, giving the endpoint $\bar{w}_{i,t} = \text{clip}(\rho_{i,t}^{\text{rt}}, C^{-1}, C)$. This choice uses one parameter and treats the two directions symmetrically in log-weight space.

To vary correction continuously, we interpolate between 0 and $\log \bar{w}_{i,t}$. Exponentiating $\log w_{i,t} = \alpha_k \log \bar{w}_{i,t}$ gives

$$w_{i,t}(\alpha_k) = \left[\text{clip}\left(\rho_{i,t}^{\text{rt}}, C^{-1}, C\right)\right]^{\alpha_k}, \qquad \alpha_k \in [0, 1]. \tag{4}$$

At $\alpha_k = 0$, AT-TIS recovers unit-weight updates; at $\alpha_k = 1$, it reaches the two-sided endpoint. Because $\log \bar{w}_{i,t} \in [-\log C, \log C]$, the weights satisfy $C^{-\alpha_k} \le w_{i,t} \le C^{\alpha_k}$. Consequently, attenuation remains bounded even at the two-sided endpoint. For ratios below C^{-1}, however, the lower bound retains more contribution than untruncated importance weighting, introducing additional correction bias. This trade-off motivates selecting α_k from the training state.

3.2 State-Dependent Selection of Correction Strength

A fixed α cannot respond to changes in mismatch or in the sampled learning signal. AIS addresses state dependence through weight reliability, mismatch severity, and variance amplification (Zhou

et al., 2026). On Omni, the evaluated AIS configuration still exhibits a pronounced intermediate reward drop even when its mismatch gate is saturated and its variance penalty is inactive (Appendix C). We therefore combine two diagnostics: historical increases in accumulated mismatch and disagreement between uncorrected and bounded-correction advantage coefficients.

Let $\widehat{\Delta}_{i,t} = \mathrm{clip}(\log \rho^{\mathrm{rt}}_{i,t}, -\Lambda, \Lambda)$, where $\Lambda > 0$ bounds the log-ratio magnitude for numerical stability. The inequality $e^x \geq 1 + x$ gives a nonnegative token statistic. We accumulate it along each response, normalize by the configured maximum response length $T_{\max}$, and aggregate across responses using the root mean square:

$$\kappa_{i,t} = e^{\widehat{\Delta}_{i,t}} - 1 - \widehat{\Delta}_{i,t}, \qquad K^{\mathrm{seq}}_k = \sqrt{\frac{1}{B}\sum_{i=1}^{B}\left(\frac{\sum_t m_{i,t}\kappa_{i,t}}{T_{\max}}\right)^2}. \tag{5}$$

Normalizing by fixed $T_{\max}$ retains accumulated mismatch information; RMS aggregation increases sensitivity to responses with large discrepancies. To account for variation in scale, we compare the current statistic with the historical log-space median c_{k-1} and robust scale σ_{k-1}:

$$R^{\mathrm{seq}}_k = \mathrm{clip}\left(\frac{\log(K^{\mathrm{seq}}_k + \varepsilon_{\mathrm{num}}) - c_{k-1}}{z_{\mathrm{sat}}\sigma_{k-1}}, 0, 1\right). \tag{6}$$

Here, $\varepsilon_{\mathrm{num}} > 0$ is a numerical stabilizer, and $z_{\mathrm{sat}} > 0$ is the standardized positive-deviation threshold at which R^{seq}_k saturates at one. The historical median and scale exclude the current batch. Their estimation, initialization, and the numerical parameter settings are specified in Appendix B.

Mismatch alone does not indicate how strongly weighting changes the current advantage coefficients. Moving from unit weights to $\bar{w}_{i,t}$ changes each coefficient by $A_{i,t}(\bar{w}_{i,t} - 1)$. Normalizing its root-mean-square magnitude by that of the original coefficients gives

$$R^{\mathrm{contrib}}_k = \min\left(\sqrt{\frac{\langle A^2_{i,t}(\bar{w}_{i,t} - 1)^2\rangle_k}{\langle A^2_{i,t}\rangle_k}}, 1\right). \tag{7}$$

We set this value to zero when all valid advantages are zero. The statistic is invariant to a common nonzero rescaling of advantages. We use greater coefficient disagreement as a heuristic signal for stronger bounded correction. The weight floor continues to limit attenuation as correction increases.

Either diagnostic can indicate a need for stronger correction, so we set $\alpha^{\mathrm{target}}_k = \max(R^{\mathrm{seq}}_k, R^{\mathrm{contrib}}_k)$. We increase correction immediately when the target exceeds the previous correction strength and smooth downward changes to avoid premature release:

$$\alpha_k = \begin{cases} \alpha^{\mathrm{target}}_k, & \alpha^{\mathrm{target}}_k \geq \alpha_{k-1}, \\ \beta_\alpha \alpha_{k-1} + (1 - \beta_\alpha)\alpha^{\mathrm{target}}_k, & \text{otherwise.} \end{cases} \tag{8}$$

We initialize $\alpha_0 = 0$. The smoothing coefficient $\beta_\alpha = 2^{-1/8}$ gives downward adjustments an eight-round half-life when the target remains fixed. The update preserves $\alpha_k \in [0, 1]$.

3.3 Risk-Gated Update Control

While AT-TIS can preserve early learning gains, its lower weight bound introduces additional correction bias, motivating explicit control of actor–rollout drift. We therefore combine policy-gradient scaling with a penalty on log-probability differences between the current actor and the rollout policy.

For microbatch j in round k, define the drift penalty

$$\mathcal{L}^{k,j}_{\mathrm{mis}}(\theta) = \langle \mathrm{SmoothL1}_1(\log \pi_\theta(o_{i,t} \mid s_{i,t}) - \log \mu_k(o_{i,t} \mid s_{i,t}))\rangle_{k,j}. \tag{9}$$

The average is sequence-balanced over the microbatch. Smooth L1 is quadratic for residual magnitude below one and linear above it, limiting the penalty's derivative for large log-probability gaps.

The controller tracks smoothed mismatch relative to an early-training baseline, together with ESS deterioration, reduced advantage support, and response-length risk. Persistent excess mismatch accompanied by degradation signals increases regularization, while more severe mismatch also reduces the policy-gradient multiplier s_k. Accumulated regularization decreases when mismatch falls below the baseline. For each microbatch, a gradient budget limits the effective regularization coefficient $\lambda_{k,j}$. The budget is based on 20% of the scaled PG gradient norm in sampled-token log-probability space. Appendix B specifies the diagnostics, control rules, and budget calculation.

3.4 Joint Objective and Optimization

The joint objective combines AT-TIS weighting, risk-gated update control, and reference-policy regularization:

$$\mathcal{L}^{k,j}_{\text{joint}}(\theta) = s_k \mathcal{L}^{k,j}_{\text{PG}}(\theta; w(\alpha_k)) + \lambda_{k,j} \mathcal{L}^{k,j}_{\text{mis}}(\theta) + \beta_{\text{KL}} \mathcal{L}^{k,j}_{\text{KL}}(\theta; \pi_{\text{ref}}). \tag{10}$$

The PG term uses the AT-TIS weights, and the final term regularizes the actor against the fixed reference policy. We hold α_k and s_k fixed during actor optimization and compute $\lambda_{k,j}$ separately for each microbatch. Correction weights and control coefficients are treated as constants when computing actor gradients.

4 Experiments

We evaluate ACU-RL on multimodal and mathematical reasoning tasks. The experiments examine whether ACU-RL sustains stable training across both low-precision rollout settings, how its downstream accuracy compares with existing correction methods, and how adaptive correction and actor-update control contribute to the observed training behavior.

4.1 Experimental Setup

Training framework and hardware. All experiments use the verl framework (Sheng et al., 2025) and run on a single node equipped with NVIDIA RTX PRO 6000 Blackwell Server Edition GPUs. Omni training uses four GPUs, while Qwen3 training uses eight. The BF16 and NVFP4 Qwen3 timing runs in Section 4.4 use the same eight-GPU allocation.

Models and training. We use Qwen2.5-Omni-3B (Xu et al., 2025) and Qwen3-8B-Instruct (Yang et al., 2025) for multimodal reasoning and long-response mathematical reasoning, respectively. FSDP (Zhao et al., 2023) handles actor training in BF16, while vLLM (Kwon et al., 2023) generates rollouts in BF16 or NVFP4. Both models use GRPO: Omni is trained on ViRL8k, a fixed 8,000-example subset of ViRL39K (Wang et al., 2025), and Qwen3 on DAPO Math17K (Yu et al., 2025). Maximum response lengths are 1,024 and 8,192 tokens, respectively. Appendix A provides the training configurations, quantization scope, and controller settings.

Baselines and evaluation. We compare ACU-RL against full-BF16 RL, uncorrected NVFP4 rollout (Native), fixed TIS, and AIS. All except full-BF16 RL use NVFP4 rollout.

We evaluate Omni in BF16 on Geometry3K (Lu et al., 2021), MathVista testmini (Lu et al., 2024), MathVision (Wang et al., 2024), WeMath (Qiao et al., 2025), LogicVista (Xiao et al., 2024), and MMMU (Yue et al., 2024). Decoding uses temperature zero and a response limit of 1,024 tokens. We report the unweighted mean accuracy across the six tasks. Qwen3 offline evaluation uses BF16 sampling on AIME24, AIME25, AMC23 (Mathematical Association of America, n.d.), and MATH-500 (Lightman et al., 2024). We sample 32, 32, 16, and 4 responses per problem, respectively, and report mean accuracy.

4.2 Main Results

Training dynamics. Figure 3 shows that the behavior of TIS and AIS differs across the two training settings. On Omni, TIS maintains relatively steady reward, whereas AIS exhibits a pronounced mid-training reward drop accompanied by a spike in token-probability discrepancy. On Qwen3, AIS maintains reward through the later training stages, whereas TIS shows a late-stage decline. In contrast, ACU-RL maintains comparatively stable reward trajectories in both settings.

On Qwen3, the mean reward of TIS decreases from 0.642 over steps 251–300 to 0.577 over steps 301–350. Over the same intervals, ACU-RL obtains mean rewards of 0.681 and 0.654, respectively, showing a smaller decline. The discrepancy curves provide complementary evidence: on Omni, ACU-RL avoids the large transient spike observed with AIS, while on Qwen3, AIS and ACU-RL exhibit similar discrepancy levels.

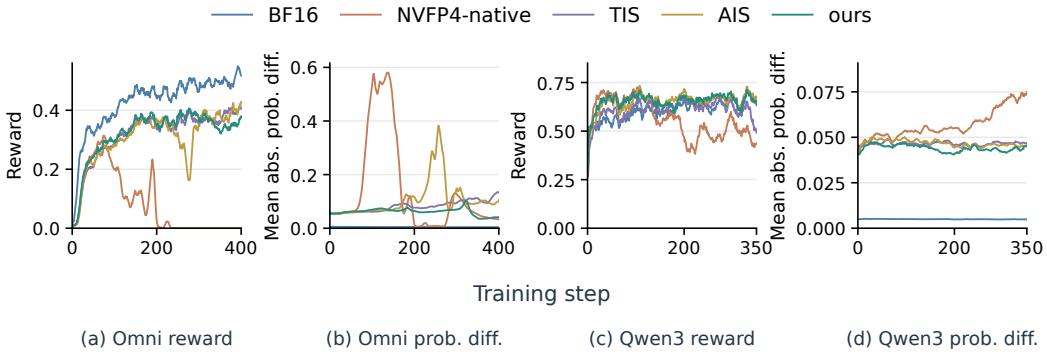


Figure 3: **Training reward and token-probability discrepancy.** (a–b) Qwen2.5-Omni-3B. (c–d) Qwen3-8B-Instruct. The probability discrepancy is the mean absolute difference between the BF16 old-policy and rollout-policy probabilities on valid sampled response tokens.

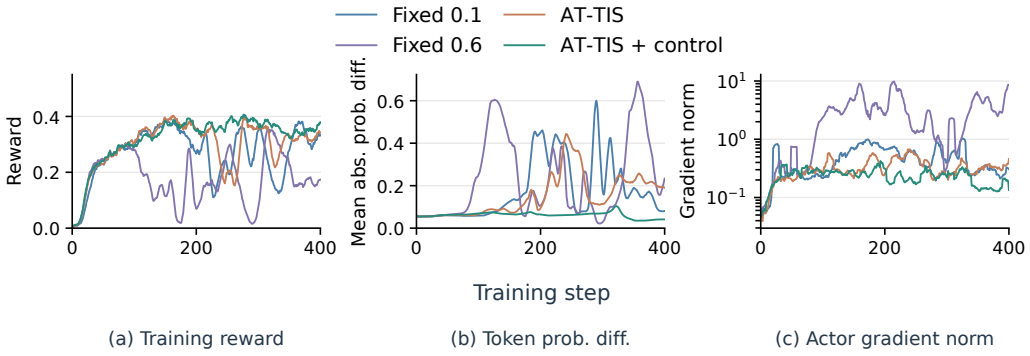


Figure 4: **Effects of AT-TIS and joint update control on training dynamics.** Qwen2.5-Omni-3B with BF16 actor training and NVFP4 rollout. Panels show (a) training reward, (b) mean absolute token probability difference between the BF16 old policy and rollout policy, and (c) pre-clipping actor gradient norm on a log scale. All curves are trailing 10-step means over 400 training steps.

Downstream accuracy. Table 1 reports BF16 offline accuracy at step 400 for Omni and step 350 for Qwen3. On Omni, ACU-RL achieves the highest mean accuracy of 45.26%, compared with 44.72% for BF16 RL, 43.40% for TIS, and 41.16% for AIS. Its gains over TIS vary across benchmarks and are concentrated on WeMath. Native NVFP4 reaches 38.67% on average, including 0.00% on MathVision, where we observe output collapse.

On Qwen3, AIS achieves the highest mean accuracy of 62.21%, exceeding ACU-RL by 1.39 percentage points and obtaining higher scores on AIME25, AMC23, and MATH-500. ACU-RL achieves the highest AIME24 accuracy and a mean accuracy of 60.82%, exceeding BF16 RL and TIS by 2.35 and 5.44 percentage points, respectively. It outperforms both BF16 RL and TIS on all four benchmarks.

Taken together, the training trajectories and offline evaluations highlight the consistency of ACU-RL across the two settings. TIS maintains relatively steady training on Omni but degrades late in Qwen3 training, while AIS exhibits a substantial intermediate drop on Omni but remains stable and achieves the best final accuracy on Qwen3. ACU-RL sustains comparatively stable training in both settings, achieving the highest mean accuracy on Omni and the second-highest on Qwen3.

4.3 Ablation Studies

On Qwen2.5-Omni-3B, we compare AT-TIS against fixed lower-bound truncation with thresholds of 0.1 and 0.6. We also evaluate AT-TIS with and without joint PG scaling and policy-drift regular-

Table 1: Offline accuracy (%) under BF16 inference at step 400 for Qwen2.5-Omni-3B and step 350 for Qwen3-8B-Instruct. Avg. denotes the unweighted mean across benchmarks; Base denotes the initial model. Geo3K, MVista, MVision, LVista, A24, A25, and M500 abbreviate Geometry3K, MathVista, MathVision, LogicVista, AIME24, AIME25, and MATH-500, respectively. Bold indicates the highest value in each column, including ties. Training trajectories are shown in Figure 3. Native NVFP4 exhibits severe reward degradation on Omni and deteriorates during Qwen3 training. AIS exhibits a transient reward drop on Omni, whereas TIS shows a late-stage decline on Qwen3.

Method	**Qwen2.5-Omni-3B**							**Qwen3-8B-Instruct**				
	Geo3K	MVista	MVision	WeMath	LVista	MMMU	Avg.	A24	A25	AMC23	M500	Avg.
Base	47.42	53.20	15.95	49.66	35.49	**44.11**	40.97	26.15	17.40	58.91	76.50	44.74
BF16	49.42	59.40	**23.42**	55.69	**36.61**	43.78	44.72	43.85	29.48	76.41	84.15	58.47
Native	46.09	60.60	0.00	47.64	34.38	43.33	38.67	37.08	24.06	62.19	86.15	52.37
TIS	**52.58**	**61.50**	22.11	44.48	**36.61**	43.11	43.40	37.08	32.08	68.75	83.60	55.38
AIS	51.08	60.10	20.82	37.36	36.38	41.22	41.16	45.21	**34.58**	**82.81**	**86.25**	**62.21**
Ours	52.41	59.60	21.64	**57.87**	35.94	**44.11**	**45.26**	**46.67**	33.85	77.66	85.10	60.82

ization to assess the contribution of actor-update control. Appendix A.1 provides the experimental configurations.

Figure 4(a) shows substantial reward degradation under fixed lower-bound truncation, particularly with a lower bound of 0.6. AT-TIS mitigates this degradation but still exhibits pronounced reward drops. Joint update control reduces these drops: over steps 241–270, mean reward is 0.213 without control and 0.381 with control.

The probability differences and gradient norms provide complementary diagnostics (Figure 4(b,c)). Across all 400 steps, joint control reduces the mean token probability difference from 0.153 to 0.062 and the mean pre-clipping actor gradient norm from 0.328 to 0.237. Together, these results support the contribution of actor-update control to training stability beyond adaptive weighting alone in the Omni setting.

4.4 Efficiency Analysis

Rollout and end-to-end efficiency. We assess the computational benefits of NVFP4 rollout on Qwen3-8B. Under the same hardware, sampling budget, and maximum generation length, NVFP4 increases measured rollout throughput by 38.81% and reduces rollout time by 23.47% relative to BF16. The resulting end-to-end training speedup is $1.17\times$.

Additional overhead. ACU-RL reuses the log-probabilities, advantages, and masks already available during training, without additional sampling or model forward passes. Controller state is updated through scalar statistics computed from these quantities.

5 Related Work

Efficient RL post-training. GRPO uses group-relative rewards to optimize reasoning policies without a separate value model (Shao et al., 2024), while DAPO refines clipping and sampling to improve large-scale RL training (Yu et al., 2025). Systems work addresses a complementary source of cost. vLLM improves serving efficiency through PagedAttention (Kwon et al., 2023); HybridFlow coordinates the distinct execution requirements of RLHF components (Sheng et al., 2025); and AReaL decouples generation and training to improve utilization through asynchronous execution (Fu et al., 2025). Our focus is the numerical mismatch introduced when rollout is executed at lower precision than actor training. ACU-RL addresses numerical mismatch through sample weighting and actor-update control.

Quantized rollout and exploration. FlashRL combines quantized rollout with BF16 training and uses TIS to mitigate the resulting policy discrepancy (Liu et al., 2025). QuRL introduces an adaptive

clipping range based on the full-precision and quantized policy ratio, together with invariant scaling to address small weight updates under quantization (Li et al., 2026). QeRL combines NVFP4 quantization with LoRA and adapts quantization noise to encourage exploration (Huang et al., 2026). These approaches establish that quantization affects both system efficiency and the learning process. Our analysis of fixed lower-bound truncation examines how importance correction changes the contribution of tokens already sampled by a quantized policy. AT-TIS explicitly bounds lower-side attenuation as well as upper-side amplification and adapts the distance from unit weights.

Adaptive correction and policy alignment. AIS uses weight reliability, mismatch severity, and variance amplification to interpolate between uncorrected and importance-weighted gradients (Zhou et al., 2026). ACU-RL builds on state-dependent correction but changes its endpoint to two-sided truncation, interpolates in log-weight space, and coordinates correction with actor-update control. A separate line of work changes the training-side policy to improve precision alignment. QaRL combines rollout-aligned quantization-aware training with a sequence-level trust-band objective (Gu et al., 2026). Our policy-drift regularizer compares the current BF16 actor with the stored rollout log-probabilities of the batch. It is activated by persistent risk and constrained by a gradient budget, jointly with PG scaling. This distinction separates correction of sampled-data contributions from direct control of actor drift during optimization.

6 Conclusions

We presented ACU-RL, a framework that coordinates adaptive importance correction and actor-update control for NVFP4 rollout with BF16 training. Our analysis shows that relaxing lower-side importance-weight attenuation can improve early reward, but fixed lower-bound truncation alone does not sustain stable learning. ACU-RL addresses this trade-off through two-sided AT-TIS, risk-gated PG scaling, and policy-drift regularization. Across the evaluated multimodal and mathematical reasoning tasks, it preserves early learning gains and improves late-stage stability. The resulting Qwen3-8B-Instruct model reaches 60.82% average accuracy, exceeding BF16 RL by 2.35 percentage points, while NVFP4 rollout delivers a $1.166\times$ end-to-end training speedup. These results support combining importance correction with actor-update control for NVFP4 rollout and BF16 training.

AI use statement

Generative AI tools, including OpenAI Codex, were used to assist with manuscript drafting, translation, and language editing; refinement of methodological explanations; discussion and interpretation of experimental results; literature search and reference verification; and figure preparation and formatting. AI-generated suggestions were reviewed and revised by the authors. The authors take full responsibility for the final manuscript, including the accuracy of mathematical formulations, experimental results, interpretations, and references.

References

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. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1218. URL https://proceedings.nips.cc/paper_files/paper/2025/hash/33c00862bfa29ac72ecf630a41e19352-Abstract-Conference.html.

Hao Gu, Hao Wang, Jiacheng Liu, Lujun Li, Qiyuan Zhu, Bei Liu, Binxing Xu, Lei Wang, Xintong Yang, Sida Lin, Sirui Han, and Yike Guo. 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. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.841. URL https://aclanthology.org/2026.findings-acl.841/.

Wei Huang, Yi Ge, Shuai Yang, Yicheng Xiao, Huizi Mao, Yujun Lin, Hanrong Ye, Sifei Liu, Ka Chun Cheung, Hongxu Yin, Yao Lu, Xiaojuan Qi, Song Han, and Yukang Chen. QeRL:

Beyond Efficiency – Quantization-enhanced Reinforcement Learning for LLMs. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/a15a2ece7f0663d1ba7db91103ac61c9-Abstract-Conference.html.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pp. 611–626. Association for Computing Machinery, 2023. doi: 10.1145/3600006.3613165. URL https://doi.org/10.1145/3600006.3613165.

Yuhang Li, Reena Elangovan, Xin Dong, Priyadarshini Panda, and Brucek Khailany. QuRL: Efficient Reinforcement Learning with Quantized Rollout. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/6b4880ced785077159b340710ceb7681-Paper-Conference.pdf.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's Verify Step by Step. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/aca97732e30bcf1303bc22ac3924fd16-Paper-Conference.pdf.

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

Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-GPS: Interpretable Geometry Problem Solving with Formal Language and Symbolic Reasoning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6774–6786. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.acl-long.528. URL https://aclanthology.org/2021.acl-long.528/.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating Mathematical Reasoning of Foundation Models in Visual Contexts. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/663bce02a0050c4a11f1eb8a7f1429d3-Abstract-Conference.html.

Mathematical Association of America. American mathematics competitions, n.d. URL https://maa.org/student-programs/amc/. AMC and AIME competition resources. Accessed September 25, 2026.

Runqi Qiao, Qiuna Tan, Guanting Dong, Minhui Wu, Chong Sun, Xiaoshuai Song, Jiapeng Wang, Zhuoma GongQue, Shanglin Lei, YiFan Zhang, Zhe Wei, Miaoxuan Zhang, Runfeng Qiao, Xiao Zong, Yida Xu, Peiqing Yang, Zhimin Bao, Muxi Diao, Chen Li, and Honggang Zhang. We-Math: Does Your Large Multimodal Model Achieve Human-like Mathematical Reasoning? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 20023–20070. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.983. URL https://aclanthology.org/2025.acl-long.983/.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

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. *arXiv preprint arXiv:2402.03300*, 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. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297. Association for Computing Machinery, 2025. doi: 10.1145/3689031.3696075. URL https://doi.org/10.1145/3689031.3696075.

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. VL-Rethinker: Incentivizing Self-Reflection of Vision-Language Models with Reinforcement Learning. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1037. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/2c84844a559e4f962752570bff456ae4-Abstract-Conference.html.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. Measuring Multimodal Mathematical Reasoning with MATH-Vision Dataset. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3014. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/ad0edc7d5fa1a783f063646968b7315b-Abstract-Datasets_and_Benchmarks_Track.html.

Yijia Xiao, Edward Sun, Tianyu Liu, and Wei Wang. LogicVista: Multimodal LLM Logical Reasoning Benchmark in Visual Contexts. *arXiv preprint arXiv:2407.04973*, 2024. URL https://arxiv.org/abs/2407.04973.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-Omni Technical Report. *arXiv preprint arXiv:2503.20215*, 2025. URL https://arxiv.org/abs/2503.20215.

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

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

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An Open-Source LLM Reinforcement Learning System at Scale. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-3775. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/a4277440d50f1f15d2cb4c14f7e0c0d2-Abstract-Conference.html.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A Massive Multi-discipline Multimodal Understanding and Reasoning Benchmark for Expert AGI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9556–9567, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Yue_MMMU_A_Massive_Multi-discipline_Multimodal_Understanding_and_Reasoning_Benchmark_for_CVPR_2024_paper.html.

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. *Proceedings of the VLDB Endowment*, 16(12):3848–3860, 2023. doi: 10.14778/3611540.3611569. URL https://www.vldb.org/pvldb/vol16/p3848-huang.pdf.

Jiajun Zhou, Wei Shao, Lingchao Zheng, Yuwei Fan, and Ngai Wong. AIS: Adaptive Importance Sampling for Quantized RL. *arXiv preprint arXiv:2605.13907*, 2026. URL https://arxiv.org/abs/2605.13907.

A Training Configuration

Training settings. Table 2 lists the training configurations for Qwen2.5-Omni-3B and Qwen3-8B-Instruct. Both models use GRPO with BF16 actor training. ViRL8k is a fixed 8,000-example subset of ViRL39K (Wang et al., 2025); mathematical training uses DAPO Math17K (Yu et al., 2025).

Table 2: Training and offline evaluation configurations. ViRL8k is a fixed 8,000-example subset of ViRL39K.Qwen3 evaluation samples per problem are listed in the order AIME24, AIME25, AMC23, MATH-500.

Configuration	Qwen2.5-Omni-3B	Qwen3-8B-Instruct
Training data	ViRL8k	DAPO Math17K
Prompts per batch	64	32
Responses per prompt	8	4
Maximum prompt length	4,096	1,024
Maximum response length	1,024	8,192
PPO mini-batch size	16	16
PPO epochs per update	1	1
Learning rate	10^{-6}	10^{-6}
Reference-KL coefficient	10^{-4}	0.01
RTX PRO 6000 GPUs	4	8
Evaluation precision	BF16	BF16
Evaluation temperature	0	1
Evaluation top-p	1.0	0.7
Evaluation response limit	1,024	8,192
Samples per problem	1	32/32/16/4

Quantization scope. For Omni, we quantize the LLM's linear layers and exclude the vision and audio modules and output head. For Qwen3, we quantize the model's linear layers while retaining the original precision of token embeddings, normalization layers, and the language-model output head.

Controller settings. AT-TIS uses the sampled-token probability ratio ρ^{rt} between the pre-update BF16 policy and rollout policy, constructing $w = \text{clip}(\rho^{\text{rt}}, 1/5, 5)^{\alpha}$ with a state-dependent α. Actor-update intervention combines PG scaling and policy-drift regularization. The release half-life of α is eight updates. In the main configuration, the mismatch baseline is calibrated with an exponential moving average (EMA) over steps 1–50, and the baseline multiplier is 1.25.

A.1 Ablation configurations

The fixed lower-bound truncation runs in Figure 4 use $w_{i,t}^{\text{fixed}} = \text{clip}(\rho_{i,t}^{\text{rt}}, \ell, 5)$, with $\ell \in \{0.1, 0.6\}$. Both adaptive runs use Equation (4) with $C = 5$. Joint update control enables PG scaling and policy-drift regularization, with baseline calibration over steps 1–50 and a baseline multiplier of 1.25. Both mechanisms are disabled in the AT-TIS run without control.

B Controller Details

B.1 Historical calibration and numerical conventions

The correction-strength controller uses $\Lambda = 20$, $C = 5$, and $\varepsilon_{\text{num}} = 2^{-23}$. Its history contains at most $W = 32$ accepted values of K^{seq}. Given historical log values $x = \log(K^{\text{seq}} + \varepsilon_{\text{num}})$, define

$$c = \text{median}(x), \qquad \sigma = \max\left\{ \frac{Q_{0.75}(x) - Q_{0.25}(x)}{1.349}, 1.4826 \text{ median}\,|x - c|, \sigma_{\text{min}} \right\}. \tag{11}$$

Here, Q_p denotes the empirical p-quantile, and $\sigma_{\text{min}} = 0.05$ is the minimum robust scale. Quantiles use linear interpolation. We set $z_{\text{sat}} = 4$ and $R_k^{\text{seq}} = 0$ until eight historical observations are available. The current observation is evaluated before insertion into the history. Once history is available,

observations with standardized positive deviation at least four are excluded from history to prevent a severe incident from immediately redefining the baseline. History is committed after a successful actor update.

All valid response lengths must be positive. For numerical stability, the contribution ratio in Equation (7) can be evaluated after dividing all advantages by their maximum absolute value; this common scale cancels. If that maximum is zero, $R_k^{\text{contrib}} = 0$. The controller starts with $\alpha_0 = 0$, an empty history, $b_0 = 0$, and an excess counter of zero. For finite inputs, the controls are computed once from the full rollout batch. Nonfinite correction inputs hold the controller state and use unit correction weights; nonfinite learner log-probabilities or advantages disallow the actor update.

B.2 Persistent mismatch and risk-gated update control

The update controller uses the sequence-balanced token statistic $M_k = \langle \kappa_{i,t} \rangle_k$, distinct from the sequence-accumulation RMS K_k^{seq}. Its moving average is initialized at the first observation and updated as

$$\bar{M}_k = \nu \bar{M}_{k-1} + (1-\nu) M_k, \qquad \nu = 0.8, \qquad v_k = \frac{\bar{M}_k}{\tau}. \tag{12}$$

A separate EMA over the calibration window defines $M_\star$. The target $\tau = m_{\text{base}} M_\star$ is then frozen, with $m_{\text{base}} = 1.25$ in the stated main configuration. Intervention remains disabled until calibration is complete and $\tau > 0$.

Let $N_k = \sum_{i,t} m_{i,t}$. The effective sample size ratio uses the applied AT-TIS weights over all valid tokens:

$$E_k = \frac{(\sum_{i,t} m_{i,t} w_{i,t})^2}{N_k \sum_{i,t} m_{i,t} w_{i,t}^2}.$$

Define response-level advantage magnitude $a_i = T_i^{-1} \sum_t m_{i,t} |A_{i,t}|$ and support fraction $S_k = B^{-1} \sum_i \mathbf{1}\{a_i > 16 \varepsilon_{\text{num}} \max_j a_j\}$. When all advantages are zero, $S_k = 0$. The calibration window also supplies EMA baselines $E_\star$ and $S_\star$, frozen afterwards.

With mean response length $\bar{T}_k$, define the length statistic

$$L_k = \text{clip}\left(\frac{1}{B} \sum_i \mathbf{1}\{T_i \geq T_{\max}\} + \frac{[\bar{T}_k - \bar{T}_{k-1}]_+}{T_{\max}}, 0, 1 \right). \tag{13}$$

The growth term is zero for the first batch. The length-risk proxy combines truncation frequency and mean-length growth. The harm score is

$$\begin{aligned} e_k &= \text{clip}\left(\frac{E_\star - E_k}{\Delta_E}, 0, 1 \right), \\ d_k &= \text{clip}\left(\frac{S_\star - S_k}{\Delta_S}, 0, 1 \right), \\ \ell_k &= \text{clip}\left(\frac{L_k - L_0}{\Delta_L}, 0, 1 \right), \\ h_k &= \max\{ R_k^{\text{seq}} \max(e_k, d_k), \ell_k \}. \end{aligned} \tag{14}$$

Here, $\Delta_E = 0.05$ and $\Delta_S = 0.10$ set the normalization scales for decreases in ESS ratio and advantage support, respectively. For the length-risk component, $L_0 = 0.05$ is the activation threshold and $\Delta_L = 0.05$ sets the scale of its increase. Here, length enters the update-intervention diagnostic; the correction-strength rule remains Equation (8). Let $v_{\text{int}} = 1.15$ denote the excess-mismatch threshold and $n_{\text{persist}} = 8$ the required number of consecutive rounds. After calibration is complete and the fixed target satisfies $\tau > 0$, we track consecutive rounds of excess mismatch:

$$n_k = \begin{cases} n_{k-1} + 1, & v_k > v_{\text{int}}, \\ 0, & \text{otherwise}. \end{cases}$$

The counter is initialized to zero and held at zero during calibration. The intervention gate is

$$g_k = \mathbf{1}\{n_k \geq n_{\text{persist}}\} \, \mathbf{1}\{h_k > 0\}.$$

Thus, the gate opens only when persistent excess mismatch coincides with a positive degradation signal.

Let b_k denote the accumulated regularization coefficient before applying the microbatch gradient budget, and let s_k denote the policy-gradient multiplier. We use $v_{\text{pg}} = 1.50$ as the mismatch threshold for PG scaling and $s_{\min} = 0.25$ as its minimum multiplier. We update them as

$$b_k = \begin{cases} \min(1, b_{k-1} + \eta h_k [v_k - v_{\text{int}}]_+), & g_k = 1, \\ \max(0, b_{k-1} - \eta [1 - v_k]_+), & g_k = 0, \end{cases}$$
$$s_k = \begin{cases} 1 - h_k \left[1 - \text{clip}\left(\frac{v_{\text{pg}}}{v_k}, s_{\min}, 1 \right) \right], & g_k = 1,\ v_k > v_{\text{pg}}, \\ 1, & \text{otherwise}. \end{cases} \tag{15}$$

Here, $[x]_+ = \max(x, 0)$, $\eta = 0.02$, and $b_0 = 0$. During calibration, or if $\tau \leq 0$, we use $b_k = 0$ and $s_k = 1$.

For $h_k \in [0, 1]$, these rules keep $b_k \in [0, 1]$ and $s_k \in [s_{\min}, 1]$. While the gate is open, the accumulated regularization coefficient increases up to its upper bound of one. When the gate is closed, the coefficient is retained if $v_k \geq 1$ and decays toward zero if $v_k < 1$. PG scaling is applied only when the gate is open and $v_k > v_{\text{pg}}$; otherwise, $s_k = 1$.

For each actor microbatch j, the gradient budget described below caps b_k to obtain the effective regularization coefficient $\lambda_{k,j}$. The control coefficients are treated as constants when computing actor gradients.

B.3 Microbatch gradient budget

For residual x, the unit-threshold penalty in Equation (9) is $\text{SmoothL1}_1(x) = x^2/2$ for $|x| < 1$ and $|x| - 1/2$ otherwise. Let $\boldsymbol{\ell}_{k,j}$ contain the current actor's sampled-token log-probabilities in microbatch j, and let

$$G_{\text{PG}} = \left\| \nabla_{\boldsymbol{\ell}_{k,j}} \left(s_k \mathcal{L}_{\text{PG}}^{k,j} \right) \right\|_2, \qquad G_{\text{mis}} = \left\| \nabla_{\boldsymbol{\ell}_{k,j}} \mathcal{L}_{\text{mis}}^{k,j} \right\|_2.$$

The reference-KL term is excluded from this budget. We use a gradient-budget fraction $\zeta = 0.2$ and a numerical stabilizer $\varepsilon_g = 10^{-12}$. The effective coefficient is

$$\lambda_{k,j} = \begin{cases} b_k, & G_{\text{mis}} \leq \varepsilon_g, \\ \min\left(b_k, \frac{\zeta G_{\text{PG}}}{G_{\text{mis}} + \varepsilon_g} \right), & G_{\text{mis}} > \varepsilon_g. \end{cases} \tag{16}$$

Both norms and $\lambda_{k,j}$ are detached before differentiating the joint loss. For $G_{\text{mis}} > \varepsilon_g$, Equation (16) caps the auxiliary gradient norm at ζG_{PG} in sampled-token log-probability space; otherwise, $\lambda_{k,j} = b_k$. This is a per-microbatch output-space budget, not a parameter-space trust-region guarantee.

B.4 Update procedure

The following procedure specifies one rollout round; the controller state is carried across rounds.

1. Freeze the BF16 old policy, synchronize the rollout engine, and sample responses while storing rollout log-probabilities.
2. Compute rewards, group-relative advantages, valid-token masks, and old-policy log-probabilities. Form the rollout–training ratios.
3. Compute K_k^{seq} and the two correction diagnostics from the current batch and past history. Update α_k by Equation (8), then detach the AT-TIS weights.
4. Update mismatch calibration or the fixed-target ratio v_k as appropriate. Compute E_k, S_k, L_k, h_k, and the persistence counter. Select b_k and s_k by Equation (15).
5. For each actor microbatch, evaluate the scaled PG loss and drift penalty. Compute $\lambda_{k,j}$ by Equation (16) and accumulate gradients of Equation (10) using the training schedule.
6. After a successful actor update, commit the proposed controller state and eligible historical observation. Otherwise retain the previous controller state.

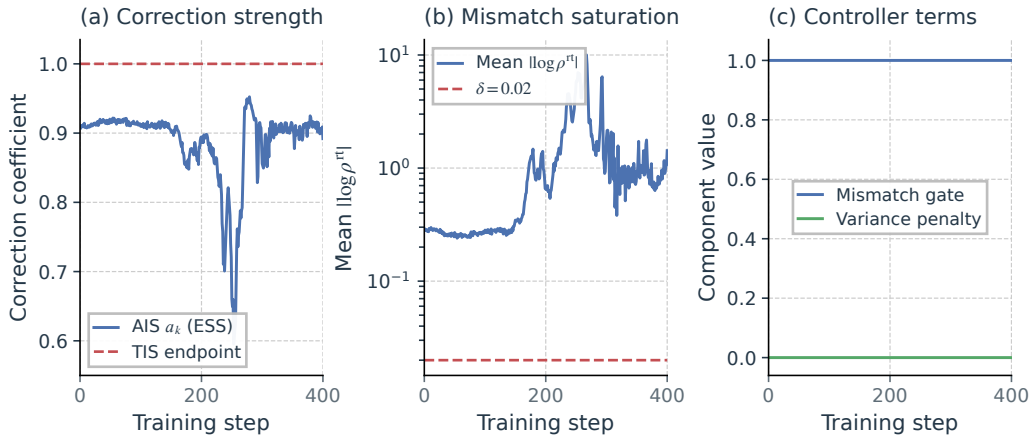


Figure 5: **AIS controller diagnostics on Qwen2.5-Omni-3B.** (a) The AIS mixing coefficient and the fixed-TIS endpoint. (b) Mean absolute log-probability mismatch and the saturation threshold $\delta = 0.02$. (c) The mismatch gate remains one and the variance penalty remains zero throughout training.

C AIS Diagnostics

AIS provides a starting point for state-dependent correction. It interpolates between uncorrected and TIS-weighted updates using weight reliability, mismatch severity, and variance amplification (Zhou et al., 2026). Its effective weights and mixing coefficient can be written as

$$\begin{aligned} w_{i,t}^{\mathrm{AIS}} &= (1 - a_k) + a_k w_{i,t}^{\mathrm{TIS}}, \\ a_k &= \mathrm{clip}\left(\sqrt{E_k} - \beta_{\mathrm{var}} \left[\frac{V_k - \gamma}{\gamma}\right]_+, 0, 1\right) \min\left(\frac{D_k}{\delta}, 1\right). \end{aligned} \tag{17}$$

Here, E_k is the normalized effective sample size (ESS), D_k is the mean absolute log-probability mismatch, and V_k is the ratio of standard deviations of advantage coefficients after and before weighting. The parameters β_{var}, γ, and δ control the variance-penalty strength, variance tolerance, and mismatch-saturation threshold, respectively. This rule reduces correction when mismatch is small and lowers the TIS mixture weight when importance weights become less reliable or coefficient variability increases, adapting the bias–variance trade-off to each batch.

To examine whether an AIS configuration favoring stronger correction can stabilize NVFP4 training, we use a low mismatch-saturation threshold and a permissive variance tolerance, minimizing attenuation from these two components. In our Qwen2.5-Omni-3B experiment, D_k remains well above $\delta = 0.02$, so the mismatch gate equals one at all 400 recorded steps (Figure 5). Meanwhile, V_k stays below $\gamma = 1.2$, leaving the variance penalty inactive. Consequently, the mixing coefficient reduces to $a_k = \sqrt{E_k}$, the largest coefficient permitted by this controller for the given ESS reliability component. Nevertheless, training remains unstable: the mid-training reduction in ESS and the TIS mixture weight coincides with worsening mismatch, more truncated responses, and reward degradation. Thus, even without attenuation from the other components, weight reliability alone does not determine the intervention required by the training state. ESS measures weight concentration, but does not directly measure response degradation or the effect of correction on existing learning contributions. Persistently saturated or inactive components cannot supply this missing information. This motivates diagnostics that combine sequence mismatch relative to a historical baseline with normalized contribution perturbation.