TURNING OFF-POLICY TOKENS ON-POLICY: A PLUG-IN APPROACH FOR IMPROVING LLM ALIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning (RL) post-training for large language models (LLMs) follows a efficient paradigm of "rollout then update", which inevitably results in off-policy training data. To resolve this, Importance sampling (IS) is proposed, while the token-level ratios compound over long sequences, causing severe variance exploded. A natural idea is "transferring" these off-policy token into on-policy token, so that the importance scores for correction are unnecessary. Following this idea, we propose *Selective Importance Sampling* (SIS), which is inspired by rejection sampling. Concretely, SIS implements by viewing off-policy model as proposal distribution, and implement a token-level rejection test: accepted tokens are viewed as on-policy, so that receive unit importance score, while rejected tokens retain the standard IS correction. Our proposed SIS is theoretically proved reducing the gap between token-level and sequence-level off-policy gradient estimators. The SIS acts as a plug-in that only modifies the importance ratio in the policy loss, adding negligible wall-clock overhead, and can be combine with a vast vary of RL post-training algorithms. Experiments on dense and MoE LLMs across math and agent benchmarks show that SIS consistently improves all objectives, while providing substantially stronger robustness under off-policy data.

1 INTRODUCTION

Reinforcement learning (RL) (Sutton et al., 1998) has become the dominant paradigm for post-training large language models, driving substantial progress in reasoning (Guo et al., 2025; Chen et al., 2025), code generation (Jiang et al., 2025), tool use (Li et al., 2026b), and broader capabilities (Li et al., 2026a;c; Zhang et al., 2026a;b; Tu et al., 2026). The RL process follows an order of "rollout then update". For efficiency, techniques such as sample reuse (Noukhovitch et al., 2025), asynchronous rollouts (Fu et al., 2026) are widely adopted, but inevitably introduce off-policy data from stale policies. The resulting distribution mismatch introduces bias into gradient estimates and degrades training stability (Ma et al., 2025). This makes effective off-policy gradient correction a fundamental challenge for RL-based post-training methods.

Importance sampling (IS) (Tokdar & Kass, 2010) offers a principled correction to off-policy gradient by multiplying a correction ratio, but its variance can grow rapidly in long-horizon reasoning (Metelli et al., 2020), and is sensitive to some outliers (Schulman et al., 2017). To mitigate this, mainstream methods (Schulman et al., 2017; Shao et al., 2024; Yu et al., 2025) rely on hard clipping. Although effective for stabilization, hard clipping suppresses gradients from highly off-policy samples and loses useful learning signal. Some recent methods (Zheng et al., 2025b; Chen et al., 2025; Gao et al., 2025a) mitigate this issue by soft clipping, but remain heuristic.

Instead of working on clipping method, we address this problem by "transferring" the off-policy tokens into on-policy tokens, which fundamentally resolve the off-policy problem. Our method, *Selective Importance Sampling* (SIS), a **simple yet effective method** conduct this transfer via acceptance-rejection sampling (Robert & Casella, 2004). Concretely, SIS views the behavior policy that generated the training data as a proposal distribution for sampling from the current policy. By implementing a rejection sampling, i.e., a token-level acceptance test, the accepted tokens are **treated as on-policy** and thus need no IS correction; the rejected tokens keep the standard IS ratio. To further improve our SIS, we propose an efficient approximation to the acceptance test that substantially reduces its computational cost while incurring negligible loss in performance. Beyond the practical

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.

method, we also provide a theoretical guarantee for SIS: we prove that it yields a strictly tighter approximation-error bound than standard importance sampling.

Empirically, SIS delivers consistent gains as a plug-in module across diverse backbones models (dense and Mixture-of-Experts (Yang et al., 2025)), RL algorithms, and benchmarks (math (MAA, 2024) and agent tasks (Jin et al., 2025)). Moreover, it enhances training stability under challenging off-policy regimes, and ablation studies confirm that it is insensitive to hyperparameter choices.

Our contributions are as follows:

- We propose SIS, a plug-in method that can be combined with most mainstream RL post-training methods e.g., GRPO (Shao et al., 2024), DAPO (Yu et al., 2025), GSPO (Zheng et al., 2025b). The method mitigates off-policy mismatch in LLM post-training in a rejection sampling regime.
- We provide theoretical justification for SIS, proving that its token-level rejection mechanism yields a strictly sharper upper bound on off-policy approximation error.
- We conduct extensive experiments spanning diverse model architectures, RL algorithms, and task domains, demonstrating that SIS delivers consistent gains and stronger stability under challenging off-policy regimes.

2 BACKGROUND

In this section, we present the necessary background knowledge of this paper.

Off-Policy Policy Gradient. Given a query x from the dataset $\mathcal{D}$, the response denoted as $y = (y_1, y_2, \ldots, y_T)$ is sampled from LLM $\pi_\theta(y \mid x)$. The response is evaluated by a reward model $r(x, y)$. To optimize the policy LLM, we usually maximize the following regularized expected reward.[1]

$$\mathcal{J}(\theta) = \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta(\cdot|x)}[r(x, y)], \tag{1}$$

The policy gradient of this objective with respect to the policy parameters can be computed via the log-trick (Sutton et al., 1998) by

$$\nabla_\theta \mathcal{J}(\theta) = \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta(\cdot|x)} \left[(A(x, y)) \nabla_\theta \log \pi_\theta(y \mid x) \right], \tag{2}$$

where $A(x, y)$ denotes the advantage associated with response y under prompt x, which results in unbiased gradient estimator e.g., $A(x, y) = r(x, y) - c$ for some constant c (Schulman et al., 2017).

In typical large-scale settings, due to the limitations of sample reuse, rollout efficiency e.t.c. (Fu et al., 2026), online sampling from π_θ to estimate policy gradient is infeasible, and training is performed on a fixed set of responses collected from a stale behavior policy $\pi_{\theta_{\text{old}}}$. To reproduce the unbiased gradient estimator, we usually use importance sampling correction ratio $w(\theta) = \frac{\pi_\theta(y|x)}{\pi_{\theta_{\text{old}}}(y|x)}$

$$\nabla_\theta \mathcal{J}(\theta) = \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_{\theta_{\text{old}}}(\cdot|x)} \Big[\underbrace{w(\theta) A(x, y) \nabla_\theta \log \pi_\theta(y \mid x)}_{\boldsymbol{g}_{\text{seq}}} \Big]. \tag{3}$$

The sequence-level correction ratio $w(\theta)$ can be viewed as a factor to account for the discrepancy between the target and behavior policies. In practice, this ratio may become excessively large, leading to exploding gradients and unstable training (Schulman et al., 2017; Yu et al., 2025). To mitigate this, $w(\theta)$ is typically modified through clipping or normalization to ensure more stable training as proposed in (Schulman et al., 2017; Zheng et al., 2025b).

Token-level Approximation of the Importance Ratio and Its Bias. For autoregressive models, the sequence likelihood admits the following factorization

$$\pi_\theta(y \mid x) = \prod_{t=1}^{T} \pi_\theta(y_t \mid x, y_{<t}). \tag{4}$$

Consequently, the sequence-level ratio $w(\theta)$ decomposes into a product of token-level ratios

$$w(\theta) = \prod_{t=1}^{T} w_t(\theta), \quad w_t(\theta) = \frac{\pi_\theta(y_t \mid x, y_{<t})}{\pi_{\theta_{\text{old}}}(y_t \mid x, y_{<t})}. \tag{5}$$

[1] Here we ignore the KL regularization (Ouyang et al., 2022) to simplify the notation, our analysis can be similarly conducted with it involved.

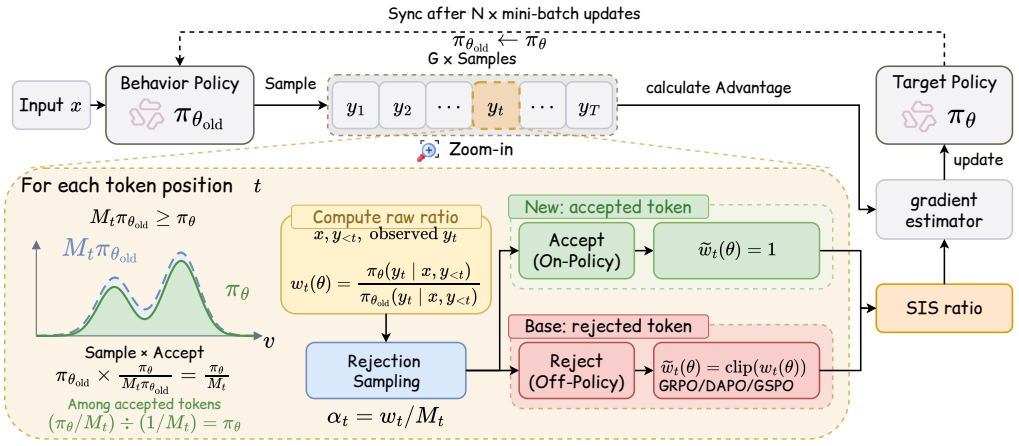

Figure 1: Overview of SIS. At each token position, SIS applies a rejection sampling to determine whether the token can be treated as on-policy ($\widetilde{w}_t(\theta) = 1$) or should retain its off-policy importance ratio. The modified ratios are then plugged into the gradient estimator of any existing algorithm.

This product scales exponentially with sequence length, leading to severe variance in gradient estimation (Metelli et al., 2020). To mitigate this, practical implementations replace the sequence-level importance ratio in gradient estimation by single token-level ratios (Ma et al., 2025) and results in approximated gradient

$$\tilde{g}(\theta) = \mathbb{E}_{y \sim \pi_{\theta_{\text{old}}}(\cdot|x)} \underbrace{\Big[\sum_{t=1}^{T} w_t(\theta) A(x,y) \nabla_\theta \log \pi_\theta(y_t \mid x, y_{<t}) \Big]}_{\boldsymbol{g}_{\text{tok}}}. \tag{6}$$

Although $\tilde{g}(\theta)$ does not equal the true policy gradient $\nabla_\theta \mathcal{J}(\theta)$, it provides a reliable approximation **only when the tokens stay nearly on-policy** (Kakade & Langford, 2002; Zheng et al., 2025a) (i.e., each token-level ratio $w_t(\theta) = \pi_\theta(y_t \mid x, y_{<t}) / \pi_{\theta_{\text{old}}}(y_t \mid x, y_{<t}) \approx 1$). Otherwise, the policy gap grows, yielding biased gradients that destabilize training. Controlling this gap is therefore central to reliable off-policy LLM training; we formalize this in Section 3.3.

3 Method: Turning Off-Policy Tokens On-Policy

In this section, we propose our *Selective Importance Sampling* (SIS), a **simple yet effective method** that uses rejection sampling to turn off-policy tokens into on-policy ones, thereby avoiding the problems caused by off-policy data. Beyond the SIS, we further propose an efficient top-K implementation with provably controllable error, cutting its computational cost while preserving performance, which makes SIS a plug-in for existing algorithms. Finally, we prove the theoretical improvements of SIS in terms of a provably sharper sequence-versus-token approximation-error bound.

3.1 SIS: Converting Off-Policy Tokens via Rejection Sampling

Importance sampling corrects stale rollout data by weighting tokens according to the importance sampling ratio. *Under acceptance-rejection sampling with this ratio, SIS views the behavior policy $\pi_{\theta_{\text{old}}}$ as a proposal distribution to sample from current policy π_θ* (see Figure 1). Then, by a acceptance-rejection test, the accepted tokens becomes on-policy tokens with unit importance ratio. Otherwise, the token retains its original importance ratio as in existing methods. In this way, SIS reduces the variance of importance ratios while preserving gradient signal.

Since the importance sampling ratio can exceed one, we divide it by a shared constant for each position t with prefix $(x, y_{<t})$, computed as

$$M_t = \max_{v \in \mathcal{V}} \frac{\pi_\theta(v \mid x, y_{<t})}{\pi_{\theta_{\text{old}}}(v \mid x, y_{<t})} + \epsilon, \tag{7}$$

where $\mathcal{V}$ denotes the vocabulary, $\pi_{\theta_{\text{old}}}(\cdot \mid x, y_{<t})$, and $\pi_\theta(\cdot \mid x, y_{<t})$ are respectively proposal and target distributions. Here $\epsilon > 0$ is a correction constant, which can be extremely small (e.g., 1e-8) in practice, and used to construct the correction estimator in (29). Given the observed token y_t with

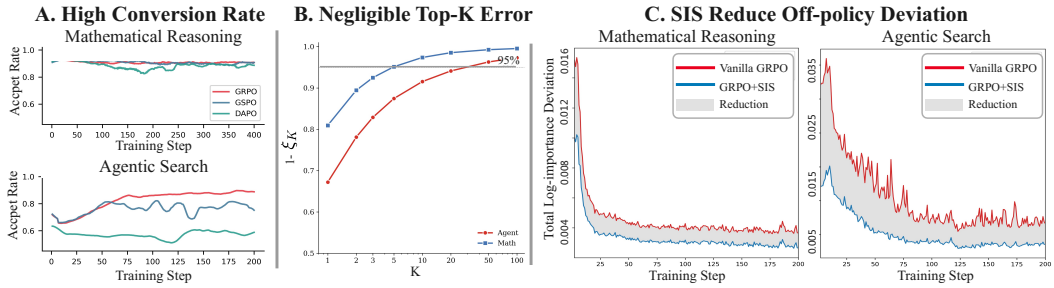

Figure 2: Empirical validation of the three components of SIS on Qwen3-8B-Base. **(A)** Accept rate of rejection sampling on math and agentic search; higher means more off-policy tokens are converted to on-policy (Appendix D.2 for 30B-MoE and 14B). **(B)** Top-K coverage $1-\xi_K$ versus K, where ξ_K is the total-variation bound in (12) of Proposition 2. $1-\xi_K \to 1$ at small K (e.g., $K=5$) means the approximation is nearly exact. **(C)** Total log-importance deviation D (defined in (14)) for vanilla GRPO vs. GRPO+SIS, where $D=0$ means all token on-policy. The lower GRPO+SIS curve shows that SIS drives D closer to 0, tightening its error bound.

ratio $w_t(\theta) = \frac{\pi_\theta(y_t|x,y_{<t})}{\pi_{\theta_{\text{old}}}(y_t|x,y_{<t})}$, we draw an acceptance indicator

$$z_t \sim \text{Bernoulli}\left(\frac{w_t(\theta)}{M_t}\right). \tag{8}$$

Multiplying the probability of sampling token v by its acceptance probability gives

$$\mathbb{P}(y_t = v, z_t = 1 \mid s_t) = \pi_{\theta_{\text{old}}}(v \mid s_t)\frac{\pi_\theta(v \mid s_t)}{M_t \pi_{\theta_{\text{old}}}(v \mid s_t)} = \frac{\pi_\theta(v \mid s_t)}{M_t}.$$

By the standard theory of rejection sampling, acceptance ($z_t = 1$) certifies that y_t follows exactly the target distribution $\pi_\theta(y_t \mid x, y < t)$. We state this process formally as below.

Proposition 1. *Let $y_t \sim \pi_{\theta_{\text{old}}}(\cdot \mid x, y_{<t})$ and $z_t \sim \text{Bernoulli}(w_t(\theta)\,/\,M_t)$ as defined in* (8)*. Then*

$$\mathbb{P}(y_t = v \mid z_t = 1) = \pi_\theta(v \mid x, y_{<t}), \quad \forall\, v \in \mathcal{V}. \tag{9}$$

Proposition 1 states that, although y_t is sampled from $\pi_{\theta_{\text{old}}}$, its conditional distribution over $z_t = 1$ becomes the distribution of π_θ. In other words, accepted tokens are already on-policy samples and require no importance correction. Accordingly, the modified token-level ratio

$$\widetilde{w}_t(\theta) = \begin{cases} 1, & z_t = 1 \quad \text{(on-policy, weight removed)}, \\ w_t(\theta), & z_t = 0 \quad \text{(off-policy, weight retained)}, \end{cases} \tag{10}$$

is then substituted for $w_t(\theta)$ in (6).

Empirical Validation. We visualize the accepted rate of the rejection process in Figure 2(A). As can be seen, the accept rate is **substantial** in both scenarios, demonstrating that a large fraction of tokens can be converted from off-policy to on-policy. This indicates that *the modification introduced by SIS is no trivial*. The lower accept rate in the agentic setting is expected, as interleaving reasoning with diverse tool calls introduces greater distributional shift between $\pi_{\theta_{\text{old}}}$ and π_θ.

Remark 1. *In standard PPO algorithm, $w_t(\theta)$ is processed by hard clipping as in* (16)*. This is originated from Trust-Region Policy Optimization (TRPO) (Schulman et al., 2015; Qi et al., 2026), which regularize the KL divergence between behavior policy and current policy. However, as pointed in (Qi et al., 2026), the clipping heavily relies on the sampled token, which introduces large noise in determining trust-region. However, our rejection process **distributionally** compare the behavior policy and current policy, so that also mitigate the noise introduced by sampling process.*

3.2 Practical Top-K Approximation

While the above rejection sampling procedure provides an principled mechanism for turning off-policy tokens on-policy, computing the maximal constant M_t in (7) requires maximizing over the entire vocabulary, which is computationally prohibitive. To mitigate this, we introduce a top-K approximation to this constant.

Our method is built upon the fact that the probability mass of LLM is concentrated on a small subset of vocabulary, so that we can approximate M_t with the maximization restricted to the top-K. Concretely, we define [2]

$$\widehat{M}_t = \max_{v \in \mathcal{V}_K} \frac{\pi_\theta(v \mid x, y_{<t})}{\pi_{\theta_{\text{old}}}(v \mid x, y_{<t})}, \quad \mathcal{V}_K = \text{TopK}(\pi_{\theta_{\text{old}}}(\cdot \mid x, y_{<t}),\, K), \tag{11}$$

and replace M_t in (8) with it. Under this approximation, when computing acceptance indicator z_t (8), only tokens within $\mathcal{V}_K$ are considered during rejection sampling, while tokens outside this set are still off-policy data. The approximated algorithm is summarized in Algorithm 1 in Appendix B. In this regime, we have the following proposition.

Proposition 2. *When approximating SIS with Algorithm 1, the accepted tokens follow the conditional distribution $\hat{\pi}_\theta(v \mid x, y_{<t}) = \pi_\theta(v \mid x, y_{<t}, v \in \mathcal{V}_K)$. Let $\xi_K^{\text{old}} = \sum_{v \notin \mathcal{V}_K} \pi_{\theta_{\text{old}}}(v \mid x, y_{<t})$ denote the probability mass of the proposal policy outside $\mathcal{V}_K$. Then*

$$D_{\text{TV}}(\hat{\pi}_\theta(\cdot \mid x, y_{<t}) \,\|\, \pi_\theta(\cdot \mid x, y_{<t})) \;\le\; \xi_K, \tag{12}$$

where $\xi_K = \xi_K^{\text{old}} + \sum_{v \notin \mathcal{V}_K} |\pi_\theta(v \mid x, y_{<t}) - \pi_{\theta_{\text{old}}}(v \mid x, y_{<t})|$, and D_{TV} is the total variation distance.

The bound in Proposition 2 sums two terms, both small in practice (Figure 2(B)). The old-policy tail mass outside $\mathcal{V}_K$ is small because LLM next-token logits are sharply peaked, so a few tokens hold nearly all probability. The tail drift is small because RL post-training only mildly reshapes it, mostly re-ordering tokens within the top-K (Yue et al., 2025; Karan & Du, 2026). The top-K approximation therefore stays faithful yet efficient.

Complexity analysis. Beyond approximation quality, the top-K formulation also keeps computational cost low. SIS reuses the old-policy logits from generation and the current-policy logits from training, requiring no extra model forward pass. The only additional work per token is selecting the top-K candidates under the old policy and computing their acceptance probabilities from cached logits, which adds only $\sim 1\%$ wall-clock overhead per training step (see Appendix D.3).

3.3 Why SIS Works: A Theoretical Perspective

The preceding subsections established what SIS does: selectively resetting off-policy tokens to on-policy via rejection sampling (§3.1), and how to implement it efficiently via a top-K approximation process (§3.2). Next, we explain why this mechanism is effective. To do so, we first define the bias

$$\text{Bias}_{\boldsymbol{g}} = \|\mathbb{E}[\boldsymbol{g}] - \nabla_\theta \mathcal{J}(\theta)\| \tag{13}$$

between the expectation of gradient estimator $\boldsymbol{g}$ and the ground-truth $\nabla_\theta \mathcal{J}(\theta)$ (3). Then we have the following theorem.

Theorem 1. *For a sequence of length T, and standard gradient estimator $\boldsymbol{g}_{\text{tok}}$ in* (6)*, we have*

$$\text{Bias}_{\boldsymbol{g}_{\text{tok}}} = \|\mathbb{E}[\boldsymbol{g}_{\text{tok}}] - \nabla_\theta \mathcal{J}(\theta)\| \le \mathbb{E}\Big[|A(x,y)|\,(e^D - 1)\sum_{t=1}^{T}\|s_t\|\Big], \tag{14}$$

where $D = \sum_{t=1}^{T} |\log w_t(\theta)|$, $A(x,y)$ is the advantage and $s_t = \nabla_\theta \log \pi_\theta(y_t \mid x, y_{<t})$.

The theorem is proved in Appendix A.2. As can be seen, the total log-importance deviation $D = \sum_{t=1}^{T} |\log w_t(\theta)|$ serves as the key quantity governing the estimation bias, which is consistent to the fact that for the gradient on each token, the real correction $w(\theta) = \prod_{t=1}^{T} w_t(\theta)$ is replaced with $w_t(\theta)$. This exponential dependence explains why off-policy training destabilizes rapidly as the behavior policy drifts. The drift makes $w_t(\theta) \not\approx 1\,(1 \le t \le T)$ so that $w_t(\theta) \not\approx w(\theta) = \prod_{t=1}^{T} w_t(\theta)$. Next, we use a proposition to illustrate that estimating the policy gradient with our SIS strictly reduces D, thereby tightening the error bound (14).

Proposition 3. *Under the same conditions as Theorem 1, replacing the correction ratios $w_t(\theta)$ in* (6) *with the ratios $\widetilde{w}_t(\theta)$* (10) *in SIS yields gradient estimator $\boldsymbol{g}_{\text{SIS}}$. Then,*

$$\text{Bias}_{\boldsymbol{g}_{\text{SIS}}} = \|\mathbb{E}[\boldsymbol{g}_{\text{SIS}}] - \nabla_\theta \mathcal{J}(\theta)\| \le \mathbb{E}\Big[|A(x,y)| \sum_{t=1}^{T} B_t(x,y)\|s_t\|\Big] \le \mathbb{E}\Big[|A(x,y)|\,(e^D - 1)\sum_{t=1}^{T}\|s_t\|\Big], \tag{15}$$

where $B_t(x,y)$ is latter defined in (32)*.*

[2] Here $\widehat{M}_t$ does not need correction constant ϵ.

The proof to this proposition is in Appendix A.2. Intuitively, the correction ratio $w(\theta) = \prod_{t=1}^{T} w_t(\theta)$ is contributed by each off-policy token $w_t(\theta)$. However, our SIS transfers most off-policy tokens (see Figure 2) into on-policy tokens, which strictly reduces the bias D of Theorem 1 into a $B_t(x, y)$ in Proposition 3. Figure 2(C) corroborates this empirically: SIS consistently maintains a lower deviation than vanilla GRPO on both mathematical reasoning and agentic search tasks, directly translating to a tighter approximation error bound and more stable policy optimization.

3.4 Plugging SIS into Existing Algorithms

Notably, our SIS is algorithm-agnostic: it only modifies the token-level correction ratio $w_t(\theta) \to \widetilde{w}_t(\theta)$ (cf. (10)), while leaving the rest of the components unchanged. Thus it can be combined with any RL-post training algorithms implemented on off-policy data e.g., GRPO (Shao et al., 2024), DAPO (Yu et al., 2025), GSPO (Zheng et al., 2025b). Below we illustrate the combination with with GRPO; instantiations with more algorithms are given in Appendix E.2. For a prompt x from query set $\mathcal{D}$ and a group of G responses $\{y_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(\cdot \mid x)$, we write $\widetilde{w}_{i,t}(\theta)$ for the SIS-modified ratio of the t-th token in y_i.

GRPO + SIS. Applying our proposed SIS to GRPO (Shao et al., 2024) yields

$$\mathcal{J}_{\text{GRPO}}^{\text{SIS}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\{y_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|} \min\Big(\widetilde{w}_{i,t}(\theta)\widehat{A}_i,\ \text{clip}(\widetilde{w}_{i,t}(\theta), 1-\varepsilon, 1+\varepsilon)\widehat{A}_i\Big) - \beta D_{\text{KL}}(\pi_\theta \parallel \pi_{\text{ref}}) \right], \tag{16}$$

where $\widehat{A}_i$ is the standard group-based normalized advantage. The only modification is substituting $w_{i,t}(\theta) \to \widetilde{w}_{i,t}(\theta)$, confirming the plug-in nature of SIS.

4 Experiments

We empirically verify SIS along three axes: (1) consistent plug-in gains across policy-gradient algorithms, backbones, and tasks; (2) improved training stability under challenging off-policy regimes; (3) robustness to its hyperparameter and interpretable token-level selectivity.

4.1 Experimental Setup

Models and Datasets. We conduct experiments with three Qwen3 (Yang et al., 2025) backbones: the dense Qwen3-8B-Base, Qwen3-14B-Base, and the Mixture-of-Experts (MoE) Qwen3-30B-A3B-Base. We mainly explore our results on math (Shao et al., 2024) and reasoning with search agent (Jin et al., 2025), and train our models on DAPO-Math-17K corpus (Yu et al., 2025) and the merged training sets of NQ (Kwiatkowski et al., 2019) and HotpotQA (Yang et al., 2018), respectively. More experiments on Llama3.2 (Grattafiori et al., 2024) backbones are in Appendix D.6.

Evaluation. We evaluate on 10 benchmarks across three settings: **Math**: MATH500 (Hendrycks et al., 2021), AMC23 (Ouyang et al., 2022), AIME24 and AIME25 (MAA, 2024); **General QA (Agent)**: Natural Questions (NQ) (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), and PopQA (Mallen et al., 2022); **Multi-Hop QA (Agent)**: HotpotQA (Yang et al., 2018), Musique (Trivedi et al., 2022), and Bamboogle (Bamb) (Press et al., 2023). The QA datasets are designed for reasoning with search agent. We report Avg@1 accuracy for most tasks and Avg@32 for the smaller AMC23, AIME24, and AIME25. The experiments on math are conducted on Qwen3-8B-Base and Qwen3-30B-A3B-Base, and the experiments on agent are under Qwen3-8B-Base and Qwen3-14B-Base.

Baselines. We integrate SIS into three representative RL objectives: token-level GRPO (Shao et al., 2024) and DAPO (Yu et al., 2025), and sequence-level GSPO (Zheng et al., 2025b). We also compare against recent methods that address off-policy error through alternative mechanisms: clipping-based CISPO (Chen et al., 2025), divergence-based DPPO-TV (Qi et al., 2026), and entropy-based Clip-Cov (Cui et al., 2025). *Each SIS run is paired with its corresponding baseline under identical hyperparameters and infrastructure; the only modification is replacing $w_{i,t}(\theta)$ with the SIS-modified weight $\widetilde{w}_{i,t}(\theta)$ in* (10). All runs use FSDP (Zhao et al., 2023) for training and vLLM (Kwon et al., 2023) for rollouts. Detailed hyperparameters and descriptions of all comparison methods are provided in Appendix C.

Table 1: Performance of SIS as a plug-in across math and agent benchmarks. Shaded rows denote the application of SIS on top of each baseline. The best accuracy per model is **bolded**. Δ denotes the absolute gain over the corresponding baseline.

Method	Mathematical Reasoning						Agentic Search							
	MATH500	AMC23	AIME24	AIME25	Avg	Δ	NQ	TriviaQA	PopQA	HotpotQA	Musique	Bamb	Avg	Δ
	Qwen3-8B-Base						*Qwen3-8B-Base*							
GRPO	80.4	74.14	23.44	18.23	49.05	–	49.14	66.99	44.68	46.24	19.24	48.8	45.85	–
w/ SIS	**85.6**	79.53	25.83	19.38	52.59	+3.54	49.78	67.47	**49.29**	**47.90**	19.74	**52.8**	**47.83**	+1.98
DAPO	83.2	81.88	21.56	18.33	51.24	–	49.61	65.69	45.95	42.89	16.17	47.2	44.59	–
w/ SIS	84.2	**86.79**	26.98	22.60	**55.14**	+3.90	**50.11**	67.66	47.83	47.83	21.18	51.2	47.64	+3.05
GSPO	**85.6**	78.83	26.46	18.85	52.44	–	49.72	67.53	46.93	47.12	20.52	45.6	46.24	–
w/ SIS	85.2	77.19	**28.33**	**23.44**	53.54	+1.10	49.09	**68.18**	46.97	46.97	**22.26**	**52.8**	47.71	+1.47
	Qwen3-30B-A3B-Base						*Qwen3-14B-Base*							
GRPO	82.8	78.59	25.52	18.23	51.29	–	50.64	68.21	47.42	47.70	22.13	52.1	48.03	–
w/ SIS	**85.8**	83.91	35.94	25.00	57.66	+6.37	50.86	**70.13**	48.54	47.94	**23.91**	**56.2**	49.60	+1.57
DAPO	84.2	81.02	29.17	18.96	53.34	–	50.22	68.51	48.61	48.01	22.09	51.2	48.11	–
w/ SIS	85.2	**84.77**	35.10	**27.40**	**58.12**	+4.76	51.55	69.25	50.05	**49.98**	**23.91**	54.4	**49.86**	+1.75
GSPO	75.0	77.34	34.90	25.52	53.19	–	51.58	69.36	51.36	47.01	20.98	43.2	47.25	–
w/ SIS	85.4	79.38	**35.41**	24.38	56.14	+2.95	**52.22**	68.22	**51.58**	47.47	21.02	46.4	47.82	+0.57

4.2 Main Results

We summarize our main results in Tables 1 and 2. As can be seen in Table 1, our SIS consistently improves all policy-gradient algorithms on dense (Qwen3-8B/14B-Base) and MoE (Qwen3-30B-A3B-Base) architectures across math and agentic reasoning tasks, yielding average gains of up to $+6.37$ on math and $+3.05$ on agent benchmarks.

On other hands, Table 2 unifies recent methods for off-policy correction, which are actually GRPO under varied $w_{i,t}(\theta) = g(w)$. SIS instead *decomposes* in two-folds: accepted tokens take $w_{i,t}(\theta) = 1$, while rejected tokens are handled by any base algorithm $g(\cdot)$. This composability gives SIS a guaranteed floor and a high ceiling: the trivial $g(w) = w$ (no trick) already beats most methods, while pairing $g(\cdot)$ with a strong algorithm e.g., DAPO attains the best accuracy.

Table 2: Trick-free training on Qwen3-8B-Base math benchmarks. All variants build on GRPO; the $w_{i,t}(\theta)$ column gives each method's effective coefficient (definitions of $\mathrm{TV}_{i,t}$, $\mathrm{Cov}_{i,t}$ in Appendix E). SIS sets $\widetilde{w} = 1$ for accepted tokens and applies any trick $g(w)$ to the rejected ones. Best **bolded**, second-best underlined; Δ is the relative Avg gain over GRPO.

Method	$w_{i,t}(\theta)$		MATH500	AMC23	AIME24	AIME25	Avg	Δ (%)
No correction								
w/o IS	1		79.6	67.95	23.04	18.46	47.26	-3.65
w/o clip	w		78.6	72.34	22.82	18.75	48.13	-0.92
Baseline								
GRPO	$\mathrm{clip}(w,\, 1-\varepsilon,\, 1+\varepsilon)$		80.4	74.14	23.44	18.23	49.05	–
Stabilization tricks								
+ Clip-Higher (DAPO)	$\mathrm{clip}(w,\, 1-\varepsilon_{\mathrm{lo}},\, 1+\varepsilon_{\mathrm{hi}})$		83.2	81.88	21.56	18.33	51.24	+4.46
+ Seq-level IS (GSPO)	$\mathrm{clip}(s_i,\, 1\pm\varepsilon),\ s_i = (\prod_t w_{i,t})^{1/\lvert y_i\rvert}$		**85.6**	78.83	26.46	18.85	52.44	+6.91
+ Clipped-IS (CISPO)	$\mathrm{sg}[\min(w,\, 1+\varepsilon_{\mathrm{hi}})]$		84.0	85.55	26.46	21.77	54.44	+11.0
+ Divergence (DPPO-TV)	$w \cdot \mathbf{1}[\mathrm{TV}_{i,t} \le \delta]$		81.4	73.83	23.85	18.85	49.48	+0.88
+ Clip-Cov (Entropy)	$w \cdot \mathbf{1}[\mathrm{Cov}_{i,t} \le \omega]$		80.8	76.56	24.17	18.23	49.94	+1.81
Ours	*accept*	*reject* $g(w)$						
+ SIS (vanilla)	1	w	82.2	76.72	**28.33**	20.00	51.81	+5.63
+ SIS (accept-only)	1	0	84.4	73.98	28.02	21.98	52.10	+6.21
+ SIS (GRPO)	1	$\mathrm{clip}(w,\, 1\pm\varepsilon)$	**85.6**	79.53	25.83	19.38	52.59	+7.22
+ SIS (DAPO)	1	$\mathrm{clip}(w,\, 1-\varepsilon_{\mathrm{lo}},\, 1+\varepsilon_{\mathrm{hi}})$	84.2	**86.79**	26.98	**22.60**	**55.14**	+12.4

4.3 SIS Improves Training Stability

We explore whether SIS improves training stability under off-policy regimes, since it transfers off-policy tokens into on-policy. We design three stress tests: (1) increasing staleness by reusing rollouts for up to 16 gradient updates, (2) amplifying train-inference mismatch via MoE routing divergence, and (3) removing clipping. All comparisons keep the paired protocol from §4.1.

SIS resists degradation from stale rollouts. Reusing rollouts for multiple gradient updates is common practice to amortize generation cost, but increases the degree of off-policy-ness, and results in performance drop (Fu et al., 2026). We vary N, the number of mini-batch updates per rollout, from 4 to 16 on Qwen3-8B-Base to explore whether SIS remains effective as staleness grows. Figure 3 re-

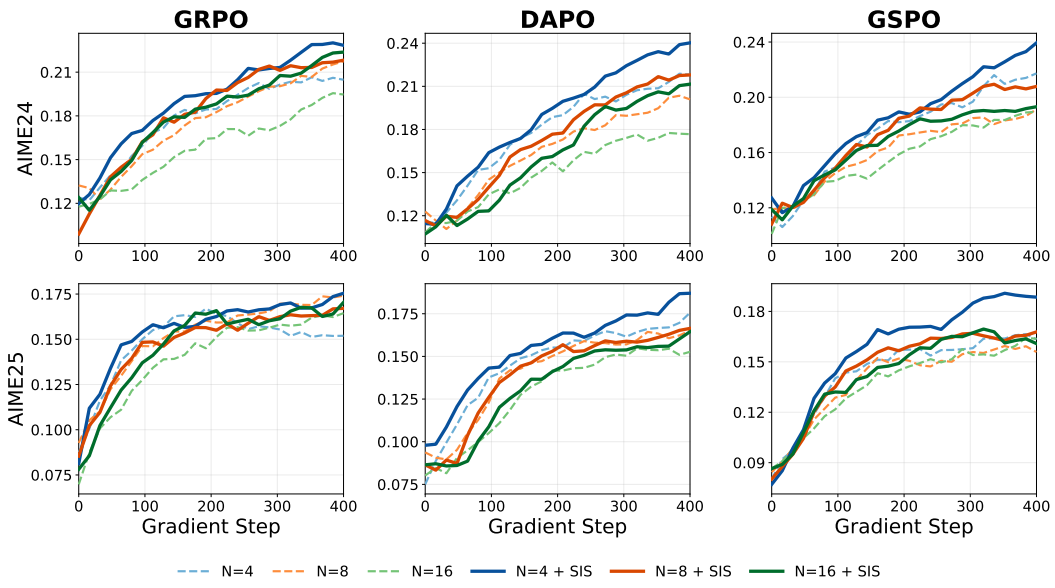

Figure 3: Training dynamics under increasing policy staleness (N=4, 8, 16) on Qwen3-8B-Base. Columns: GRPO, DAPO, GSPO. Rows: AIME24 and AIME25 accuracy. Dashed lines are baselines; solid lines add SIS. Increasing N degrades the baselines, whereas the corresponding SIS variants stay consistently above their baselines at every staleness level.

veals two observations: *(i) staleness is harmful in itself*: increasing N degrades accuracy regardless of whether SIS is applied, confirming that reusing stale off-policy rollouts hurts RL, though increasing computational efficiency; *(ii) SIS yields a stable gain at every staleness level*: each SIS variant stays consistently above its corresponding baseline across all N, because SIS maintains an appreciable accept rate even under large N (Appendix D.5). Together, these results show that SIS reliably recovers performance under stale rollouts, persisting under longer training (detailed in Tab. 8).

SIS handles train-inference mismatch in MoE models. In MoE architectures, the rollout engine and the trainer may route tokens through different experts, producing mismatched log-probabilities even at the same checkpoint (Zheng et al., 2025a; Zhao et al., 2025). We test whether SIS can handle this system-level off-policy divergence on Qwen3-30B-A3B-Base, comparing against Rollout Routing Replay (R3) (Ma et al., 2025), a method specifically designed for MoE routing mismatch. As shown in Figure 4, *vanilla DAPO collapses entirely*: accuracy peaks and then degrades, entropy drops to near zero, and gradient norms spike by two orders of magnitude. In contrast, both R3 and our SIS mitigate part of this instability, while *combining SIS with R3 yields the strongest accuracy and the most stable optimization*, indicating that SIS and R3 address complementary sources of off-policy divergence.

SIS enables clipping-free training. SIS is itself a stabilization mechanism: with clipping fully disabled, GRPO+SIS still beats clip-based GRPO on math and agent tasks (Appendix D.1). This is because SIS removes the off-policy mismatch at its source rather than merely truncating large ratios, making the hard clipping largely redundant.

4.4 Sensitivity and Token-Level Analysis

We finally ablate the only hyperparameter of SIS and inspect its token-level acceptance behavior.

Top-K sensitivity. SIS introduces a single hyperparameter K that defines the rejection envelope. Sweeping $K \in \{10, 50, 100\}$ on Qwen3-8B-Base, Table 3 shows that *SIS is robust to K*: average accuracy varies by at most $\sim$1 point, and we use K=10 throughout the paper.

Token-level acceptance behavior. We further inspect what the rejection sampler accepts or rejects at the token level. Figure 5 visualizes the most distinctive tokens among accepted (green) and rejected (red) samples. Accepted tokens are dominated by mathematical reasoning terms (e.g., “frac”), while rejected ones contain more formatting and web artifacts (e.g., “$image”), suggesting that SIS preferentially grants on-policy status to tokens aligned with the target reasoning distribution.

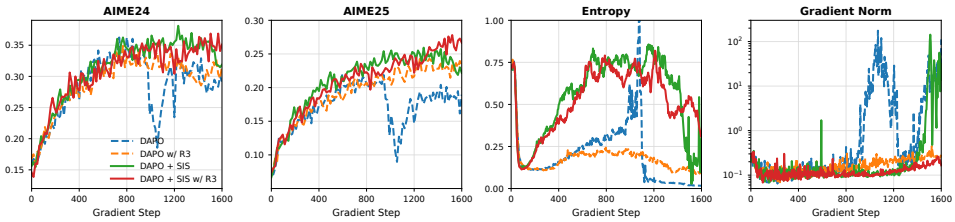

Figure 4: Training dynamics on Qwen3-30B-A3B-Base (MoE) with DAPO. Columns: AIME24/25 accuracy, entropy, gradient norm. Vanilla DAPO (blue dashed) suffers entropy collapse and gradient spikes; R3 (orange dashed) and SIS (green solid) each mitigate part of the instability; combining SIS with R3 (red solid) gives the strongest accuracy and the most stable optimization.

Table 3: Sensitivity to top-K truncation on Qwen3-8B-Base.

Method	K	MATH500	AMC23	AIME24	AIME25	Avg
GRPO + SIS	10	85.6	79.53	25.83	19.38	52.59
	50	84.6	78.83	26.67	20.00	52.53
	100	84.8	77.50	26.88	19.79	52.24
DAPO + SIS	10	84.2	85.00	25.52	22.19	54.23
	50	84.2	84.45	26.98	22.60	54.56
	100	84.0	85.08	26.77	22.19	54.51
GSPO + SIS	10	83.2	76.25	27.92	21.67	52.26
	50	85.2	77.19	28.33	23.44	53.54
	100	83.8	76.95	26.67	22.40	52.46



Figure 5: Token-level SIS decisions. Word clouds of distinctive tokens in accepted(left, green) vs. rejected(right, red) samples. SIS favors mathematical reasoning tokens and rejects formatting/web artifacts.

5 Related Work

Importance Sampling Ratios for LLM RL. Importance-sampling correction for off-policy training dates back to Schulman et al. (2015); Sutton et al. (1998), but the per-token ratio introduces large variance and unstable training (Schulman et al., 2017; Qi et al., 2026). The dominant remedy is hard clipping (Shao et al., 2024; Yu et al., 2025; Cui et al., 2025), which bounds the ratio but discards gradient information from highly off-policy tokens; soft clipping (Chen et al., 2025; Gao et al., 2025a) retains gradients at the cost of sensitive hyperparameters, and other clipping variants (Ömer Veysel Çağatan et al., 2026; Le Roux et al., 2025; Lei et al., 2026) follow the same recipe—shrinking large ratios to suppress variance, which inevitably *attenuates the gradient signal*. SIS instead *transfers* off-policy tokens to on-policy, fixing the off-policy mismatch at its root.

Rejection Sampling and Sample Selection. Rejection sampling has appeared in the LLM pipeline only as a *data curation* step rather than a sampling mechanism: RSO (Liu et al., 2024; Tajwar et al., 2026) filters preference pairs before reward-model training, best-of-N (Gui et al., 2024; Beirami et al., 2025; Sun et al., 2024; Li et al., 2026a) re-ranks candidates at inference, and other work removes OOD or unsafe samples from the training corpus (Guo et al., 2025). Unlike SIS, these response-level pre/post-processing steps do not modify the policy gradient.

Stabilizing RL Post-Training. A parallel line of work stabilizes training at the *systems* side, where staleness arises from asynchronous rollouts (Fu et al., 2026; Noukhovitch et al., 2025; Gao et al., 2025b; Team et al., 2025) or from train–inference engine mismatch in MoE models (Zheng et al., 2025a). Common remedies all *limit* the influence of off-policy data: masking high-deviation tokens (Li et al., 2025), reusing only sparse rollouts (Luo et al., 2026; Wan et al., 2026), replaying inference-time routing via R3 (Ma et al., 2025), or filtering discrepant tokens via IcePop (Zhao et al., 2025). SIS instead *converts* off-policy tokens into on-policy, and is orthogonal to these tricks; in particular, combining SIS with R3 on MoE yields strictly stronger stability than either alone.

6 Conclusion

We present *Selective Importance Sampling* (SIS) to tackle the off-policy bottleneck in post-training of LLMs, where stale rollouts and train–inference mismatch bias the policy gradient. Rather than *suppressing* the correction ratio as in prior remedies, we reframe the problem as *distribution conversion*. Concretely, SIS performs a token-level acceptance–rejection test with a top-K approximation: accepted tokens are treated as fully on-policy, while rejected ones retain the standard IS correction. SIS only modifies the importance ratio, so it plugs into any policy-gradient objective with negligible overhead; it also provably tightens the off-policy approximation-error bound. Empirically, SIS delivers consistent gains across mainstream algorithms and dense/MoE backbones, improves stability under heavy rollout reuse and MoE routing mismatch.

REFERENCES

Ahmad Beirami, Alekh Agarwal, Jonathan Berant, Alexander D'Amour, Jacob Eisenstein, Chirag Nagpal, and Ananda Suresh. Theoretical guarantees on the best-of-n alignment policy. In *Forty-second International Conference on Machine Learning*, 2025.

Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, et al. Minimax-m1: Scaling test-time compute efficiently with lightning attention. Preprint arXiv:2506.13585, 2025.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, et al. The entropy mechanism of reinforcement learning for reasoning language models. Preprint arXiv:2505.22617, 2025.

Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, Jiashu Wang, et al. Areal: A large-scale asynchronous reinforcement learning system for language reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

Chang Gao, Chujie Zheng, Xiong-Hui Chen, Kai Dang, Shixuan Liu, Bowen Yu, An Yang, Shuai Bai, Jingren Zhou, and Junyang Lin. Soft adaptive policy optimization. Preprint arXiv:2511.20347, 2025a.

Jiaxuan Gao, Wei Fu, Minyang Xie, Shusheng Xu, Chuyi He, Zhiyu Mei, Banghua Zhu, and Yi Wu. Beyond ten turns: Unlocking long-horizon agentic search with large-scale asynchronous rl. Preprint arXiv:2508.07976, 2025b.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. Preprint preprint arXiv:2407.21783, 2024.

Lin Gui, Cristina Garbacea, and Victor Veitch. BoNBoN alignment for large language models and the sweetness of best-of-n sampling. In *The Thirty-seventh Annual Conference on Neural Information Processing Systems*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. Preprint arXiv:2501.12948, 2025.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021.

Xue Jiang, Yihong Dong, Mengyang Liu, Hongyi Deng, Tian Wang, Yongding Tao, Rongyu Cao, Binhua Li, Zhi Jin, Wenpin Jiao, et al. Coderl+: Improving code generation via reinforcement with execution semantics alignment. Preprint arXiv:2510.18471, 2025.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. Preprint arXiv:2503.09516, 2025.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Association for Computational Linguistics*, 2017.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the nineteenth international conference on machine learning*, 2002.

Aayush Karan and Yilun Du. Reasoning with sampling: Your base model is smarter than you think. In *International Conference on Learning Representations (ICLR)*, 2026.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, 2020.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 2019.

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 *Symposium on Operating Systems Principles*, 2023.

Nicolas Le Roux, Marc Bellemare, Jonathan Lebensold, Arnaud Bergeron, Joshua Greaves, Alexandre Fréchette, Carolyne Pelletier, Eric Thibodeau-Laufer, Sándor Tóth, and Sam Work. Tapered off-policy REINFORCE: Stable and efficient reinforcement learning for large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2025.

Shiye Lei, Zhihao Cheng, and Dacheng Tao. A step back: Prefix importance ratio stabilizes policy optimization. Preprint arXiv:2601.22718, 2026.

Xiuyu Li, Jinkai Zhang, Mingyang Yi, Yu Li, Longqiang Wang, Yue Wang, and Ju Fan. Ets: Energy-guided test-time scaling for training-free rl alignment. In *Forty-third International Conference on Machine Learning*, 2026a.

Yingru Li, Jiacai Liu, Jiawei Xu, Yuxuan Tong, Ziniu Li, Qian Liu, and Baoxiang Wang. Trust region masking for long-horizon llm reinforcement learning. *arXiv preprint arXiv:2512.23075*, 2025.

Yu Li, Mingyang Yi, Xiuyu Li, Ju Fan, Fuxin Jiang, Binbin Chen, Peng Li, Jie Song, and Tieying Zhang. Reasoning and tool-use compete in agentic rl: From quantifying interference to disentangled tuning. Preprint arXiv:arXiv:2602.00994, 2026b.

Zhaochun Li, Mingyang Yi, Yue Wang, Shisheng Cui, and Yong Liu. Towards a theoretical understanding to the generalization of rlhf. Preprint arXiv:2601.16403, 2026c.

Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. In *The Twelfth International Conference on Learning Representations*, 2024.

Sijia Luo, Xiaokang Zhang, Yuxuan Hu, Bohan Zhang, Ke Wang, Jinbo Su, Mengshu Sun, Lei Liang, and Jing Zhang. Sparse-rl: Breaking the memory wall in llm reinforcement learning via stable sparse rollouts. Preprint arXiv:2601.10079, 2026.

Wenhan Ma, Hailin Zhang, Liang Zhao, Yifan Song, Yudong Wang, Zhifang Sui, and Fuli Luo. Stabilizing moe reinforcement learning by aligning training and inference routers. Preprint arXiv:2510.11370, 2025.

MAA. American invitational mathematics examination (AIME). `https://maa.org/maa-invitational-competitions/`, 2024.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. Preprint arXiv:2212.10511, 2022.

Alberto Maria Metelli, Matteo Papini, Nico Montali, and Marcello Restelli. Importance sampling techniques for policy optimization. *Journal of Machine Learning Research*, 2020.

Michael Noukhovitch, Shengyi Huang, Sophie Xhonneux, Arian Hosseini, Rishabh Agarwal, and Aaron Courville. Faster, more efficient RLHF through off-policy asynchronous learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 2022.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

Penghui Qi, Xiangxin Zhou, Zichen Liu, Tianyu Pang, Chao Du, Min Lin, and Wee Sun Lee. Rethinking the trust region in llm reinforcement learning. In *Forty-third International Conference on Machine Learning*, 2026.

Christian P Robert and George Casella. *Monte Carlo Statistical Methods*. Springer, 2004.

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

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. Preprint arXiv:2402.03300, 2024.

Hanshi Sun, Momin Haider, Ruiqi Zhang, Huitao Yang, Jiahao Qiu, Ming Yin, Mengdi Wang, Peter Bartlett, and Andrea Zanette. Fast best-of-n decoding via speculative rejection. In *The Thirty-seventh Annual Conference on Neural Information Processing Systems*, 2024.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*. MIT press Cambridge, 1998.

Fahim Tajwar, Guanning Zeng, Yueer Zhou, Yuda Song, Daman Arora, Yiding Jiang, Jeff Schneider, Ruslan Salakhutdinov, Haiwen Feng, and Andrea Zanette. Maximum likelihood reinforcement learning. Preprint arXiv:2602.02710, 2026.

Ling Team, Anqi Shen, Baihui Li, Bin Hu, Bin Jing, Cai Chen, Chao Huang, Chao Zhang, Chaokun Yang, Cheng Lin, et al. Every step evolves: Scaling reinforcement learning for trillion-scale thinking model. Preprint arXiv:arXiv:2510.18855, 2025.

Surya T Tokdar and Robert E Kass. Importance sampling: a review. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2010.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 2022.

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:2606.29502*, 2026.

Xu Wan, Yansheng Wang, Wenqi Huang, and Mingyang Sun. Buffer matters: Unleashing the power of off-policy reinforcement learning in large language model reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. Preprint arXiv:2212.03533, 2022.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. (arXiv:2409.12122), 2024.

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. Preprint arXiv:2505.09388, 2025.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing*, 2018.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. Preprint arXiv:2503.14476, 2025.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Yining Zhang, Mingyang Yi, Chen Wang, Xuwen Xiang, Tianhe Jia, Zedong Dan, Chengqing Zong, and Yue Wang. Htam: Hierarchical transition-attended memory for operator optimization. *arXiv preprint arXiv:2605.29734*, 2026a.

Yuxin Zhang, Meihao Fan, Ju Fan, Mingyang Yi, Yuyu Luo, Guoliang Li, Bin Wu, and Wenchao Zhou. Reward-sql: Boosting text-to-sql via stepwise execution-aware reasoning and process-supervised rewards. *Proceedings of the ACM on Management of Data*, 2026b.

Xin Zhao, Yongkang Liu, Kuan Xu, Jia Guo, Zihao Wang, Yan Sun, Xinyu Kong, Qianggang Cao, Liang Jiang, Zujie Wen, Zhiqiang Zhang, and Jun Zhou. Small leak can sink a great ship–boost rl training on moe with icepop!, 2025. URL `https://ringtech.notion.site/icepop`.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, et al. Pytorch fsdp: experiences on scaling fully sharded data parallel. Preprint arXiv:2304.11277, 2023.

Chujie Zheng, Kai Dang, Bowen Yu, Mingze Li, Huiqiang Jiang, Junrong Lin, Yuqiong Liu, An Yang, Jingren Zhou, and Junyang Lin. Stabilizing reinforcement learning with llms: Formulation and practices. Preprint arXiv:2512.01374, 2025a.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. Preprint arXiv:2507.18071, 2025b.

Ömer Veysel Çağatan, Barış Akgün, Gözde Gül Şahin, and Xuandong Zhao. Clipping-free policy optimization for large language models. Preprint arXiv:2601.22801, 2026.

A PROOFS

A.1 PROOF OF ON-POLICY CERTIFICATE

Proposition 1. *Let* $y_t \sim \pi_{\theta_{\text{old}}}(\cdot \mid x, y_{<t})$ *and* $z_t \sim \text{Bernoulli}(w_t(\theta) \,/\, M_t)$ *as defined in* (8)*. Then*

$$\mathbb{P}(y_t = v \mid z_t = 1) = \pi_\theta(v \mid x, y_{<t}), \quad \forall\, v \in \mathcal{V}. \tag{9}$$

Proof. For any $v \in \mathcal{V}$, applying Bayes' rule gives

$$\mathbb{P}(y_t = v \mid z_t = 1) = \frac{\mathbb{P}(z_t = 1 \mid y_t = v)\, \pi_{\theta_{\text{old}}}(v \mid x, y_{<t})}{\mathbb{P}(z_t = 1)}. \tag{17}$$

The acceptance probability conditioned on $y_t = v$ is

$$\mathbb{P}(z_t = 1 \mid y_t = v) = \frac{w_t(v)}{M_t} = \frac{\pi_\theta(v \mid x, y_{<t})}{M_t\, \pi_{\theta_{\text{old}}}(v \mid x, y_{<t})}. \tag{18}$$

The marginal acceptance probability is

$$\mathbb{P}(z_t = 1) = \sum_{v' \in \mathcal{V}} \frac{\pi_\theta(v' \mid x, y_{<t})}{M_t\, \pi_{\theta_{\text{old}}}(v' \mid x, y_{<t})} \cdot \pi_{\theta_{\text{old}}}(v' \mid x, y_{<t}) = \frac{1}{M_t} \sum_{v' \in \mathcal{V}} \pi_\theta(v' \mid x, y_{<t}) = \frac{1}{M_t}. \tag{19}$$

Substituting back, we have

$$\mathbb{P}(y_t = v \mid z_t = 1) = \frac{\dfrac{\pi_\theta(v \mid x, y_{<t})}{M_t\, \pi_{\theta_{\text{old}}}(v \mid x, y_{<t})} \cdot \pi_{\theta_{\text{old}}}(v \mid x, y_{<t})}{1 \,/\, M_t} = \pi_\theta(v \mid x, y_{<t}). \tag{20}$$

□

A.2 PROOF OF APPROXIMATION ERROR

Theorem 1. *For a sequence of length* T*, and standard gradient estimator* $\boldsymbol{g}_{\text{tok}}$ *in* (6)*, we have*

$$\text{Bias}_{\boldsymbol{g}_{\text{tok}}} = \|\mathbb{E}[\boldsymbol{g}_{\text{tok}}] - \nabla_\theta \mathcal{J}(\theta)\| \le \mathbb{E}\Big[|A(x,y)|\,(e^D - 1) \sum_{t=1}^{T} \|s_t\|\Big], \tag{14}$$

where $D = \sum_{t=1}^{T} |\log w_t(\theta)|$*,* $A(x,y)$ *is the advantage and* $s_t = \nabla_\theta \log \pi_\theta(y_t \mid x, y_{<t})$*.*

Proof. For a single response, the sequence-level gradient is

$$\boldsymbol{g}_{\text{seq}} = A(x,y) \prod_{j=1}^{T} w_j(\theta) \sum_{t=1}^{T} s_t, \tag{21}$$

while the token-level surrogate uses

$$\boldsymbol{g}_{\text{tok}} = A(x,y) \sum_{t=1}^{T} w_t(\theta)\, s_t, \tag{22}$$

with $s_t = \nabla_\theta \log \pi_\theta(y_t \mid x, y_{<t})$, and all y_t are from $\pi_{\theta_{\text{old}}}(y_t \mid x, y_{<t})$. Then the bias is

$$\text{Bias}_{\boldsymbol{g}} = \left\| \mathbb{E}\left[A(x,y) \sum_{t=1}^{T} \left(\prod_{j=1}^{T} w_j(\theta) - w_t(\theta) \right) s_t \right] \right\| \le \mathbb{E}\left[|A(x,y)| \sum_{t=1}^{T} \left| \prod_{j=1}^{T} w_j(\theta) - w_t(\theta) \right| \, \|s_t\| \right]. \tag{23}$$

Define $R = \prod_{j=1}^{T} w_j(\theta)$ and $\Delta_j = \log w_j(\theta)$, so that $|R - w_t(\theta)| = w_t(\theta) \left| \prod_{j \neq t} w_j(\theta) - 1 \right|$. Applying the elementary inequality $|e^x - 1| \le e^{|x|} - 1$ to the product term gives

$$\left| \prod_{j \neq t} w_j(\theta) - 1 \right| = \left| e^{\sum_{j \neq t} \Delta_j} - 1 \right| \le e^{\left|\sum_{j \neq t} \Delta_j\right|} - 1 \le e^{\sum_{j \neq t} |\Delta_j|} - 1. \tag{24}$$

Hence,

$$|R - w_t(\theta)| \le e^{|\Delta_t|} \left(e^{\sum_{j \neq t} |\Delta_j|} - 1 \right) = e^{\sum_{j=1}^{T} |\Delta_j|} - e^{|\Delta_t|} \le e^D - 1, \tag{25}$$

where $D = \sum_{t=1}^{T} |\log w_t(\theta)|$. Plugging this bound into (23) and taking expectation yields our conclusion. □

Proposition 3. *Under the same conditions as Theorem 1, replacing the correction ratios $w_t(\theta)$ in* (6) *with the ratios $\widetilde{w}_t(\theta)$* (10) *in SIS yields gradient estimator $\boldsymbol{g}_{\mathrm{SIS}}$. Then,*

$$\mathrm{Bias}_{\boldsymbol{g}_{\mathrm{SIS}}} = \|\mathbb{E}[\boldsymbol{g}_{\mathrm{SIS}}] - \nabla_\theta \mathcal{J}(\theta)\| \le \mathbb{E}\Big[|A(x,y)| \sum_{t=1}^{T} B_t(x,y)\|s_t\|\Big] \le \mathbb{E}\Big[|A(x,y)|\,(e^D-1)\sum_{t=1}^{T}\|s_t\|\Big], \tag{15}$$

where $B_t(x,y)$ is latter defined in (32).

Proof. Let $\boldsymbol{z} = (z_1, \ldots, z_T) \in \{0,1\}^T$ with z_t defined (8). For each t, we define the

$$\rho_t(\theta) := \frac{(M_t - 1)w_t(\theta)}{M_t - w_t(\theta)}, \tag{26}$$

and

$$w_t^{\mathrm{corr}}(\theta; z_t) := z_t + (1 - z_t)\rho_t(\theta), \qquad R_{\mathrm{corr}}(\theta; \boldsymbol{z}) := \prod_{j=1}^{T} w_j^{\mathrm{corr}}(\theta; z_j) \tag{27}$$

By Bayes's rule, we can show that for both $z_t = 1$ or $z_t = 0$,

$$w_t^{\mathrm{corr}}(\theta; z_t)\pi_{\theta_{\mathrm{old}}}(y_t \mid y_{<t}, x, z_t) = \pi_\theta(y_t \mid y_{<t}, x) \tag{28}$$

and define the unbiased estimator

$$\boldsymbol{g}_{\mathrm{correct}} := A(x,y)R_{\mathrm{corr}}(\theta; \boldsymbol{z})\sum_{t=1}^{T} s_t, \tag{29}$$

we have

$$\begin{aligned}
\mathbb{E}\left[\boldsymbol{g}_{\mathrm{correct}}\right] &= \mathbb{E}_{\boldsymbol{z}}\left[\mathbb{E}_{\pi_{\theta_{\mathrm{old}}}(y|x,\boldsymbol{z})}\left[A(x,y)R_{\mathrm{corr}}(\theta;\boldsymbol{z})\sum_{t=1}^{T} s_t\right]\right] \\
&= \mathbb{E}_{\boldsymbol{z}}\left[\int A(x,y)\prod_{t=1}^{T} w_t^{\mathrm{corr}}(\theta; z_t)\pi_{\theta_{\mathrm{old}}}(y_t \mid y_{<t}, x, z_t)\sum_{t=1}^{T} s_t dy\right] \\
&= \nabla_\theta \mathcal{J}(\theta).
\end{aligned} \tag{30}$$

For the practical estimator $\boldsymbol{g}_{\mathrm{SIS}} := A(x,y)\sum_{t=1}^{T}\widetilde{w}_t(\theta)s_t$, let us define the signed mask-conditioned log gap as

$$\Delta_{t,\boldsymbol{z}} := \log\frac{R_{\mathrm{corr}}(\theta;\boldsymbol{z})}{\widetilde{w}_t(\theta)} = \sum_{\substack{j\neq t \\ z_j=0}} \log\rho_j(\theta) + (1-z_t)\log\frac{\rho_t(\theta)}{w_t(\theta)} \tag{31}$$

and the exact mask-averaged coefficient

$$B_t(x,y) = \mathbb{E}_{\boldsymbol{z}|x,y}\left[\widetilde{w}_t(\theta)\left|e^{\Delta_{t,\boldsymbol{z}}} - 1\right|\right]. \tag{32}$$

Pointwise in (x,y), $B_t(x,y) \le \beta_t(x,y) < e^D - 1$ whenever $w_t(\theta) \neq 1$. Hence, the last comparison in (15) is strict if $w_t(\theta) \neq 1$ and $|A(x,y)|\|s_t\| > 0$ jointly occur with positive probability for at least one t.

Then we have

$$\begin{aligned}
\mathrm{Bias}_{\boldsymbol{g}}^{\mathrm{SIS}} &= \|\mathbb{E}[\boldsymbol{g}_{\mathrm{SIS}} - \boldsymbol{g}_{\mathrm{correct}}]\| \\
&\le \mathbb{E}_{x,y}\left[|A(x,y)|\sum_{t=1}^{T}\mathbb{E}_{\boldsymbol{z}|x,y}[|R_{\mathrm{corr}}(\theta;\boldsymbol{z}) - \widetilde{w}_t(\theta)|]\,\|s_t\|\right] \\
&= \mathbb{E}_{x,y}\left[|A(x,y)|\sum_{t=1}^{T} B_t(x,y)\|s_t\|\right].
\end{aligned} \tag{33}$$

Next, let us analyze the upper bound in above. Defining $H_{-t} := \prod_{j\neq t} w_j^{\mathrm{corr}}(\theta; z_j)$, we have

$$|R_{\mathrm{corr}} - \widetilde{w}_t| \le w_t^{\mathrm{corr}}\,|H_{-t} - 1| + |w_t^{\mathrm{corr}} - \widetilde{w}_t|\,. \tag{34}$$

Due to the definition, we get

$$|H_{-t} - 1| \le \prod_{j\neq t}\left(1 + \left|w_j^{\mathrm{corr}} - 1\right|\right) - 1. \tag{35}$$

Some simple algebra shows

$$\begin{aligned}\mathbb{E}_{z_t|x,y}[w_t^{\text{corr}}] &= w_t(\theta),\\ \mathbb{E}_{z_j|x,y}\big[\big|w_j^{\text{corr}}-1\big|\big] &= |w_j(\theta)-1|,\\ \mathbb{E}_{z_t|x,y}[|w_t^{\text{corr}}-\widetilde{w}_t(\theta)|] &= \frac{w_t(\theta)|w_t(\theta)-1|}{M_t}.\end{aligned} \tag{36}$$

Combining (34)–(36) with conditional independence yields

$$B_t(x,y) \le w_t(\theta)\left[\prod_{j\neq t}\big(1+|w_j(\theta)-1|\big)-1\right] + \frac{w_t(\theta)|w_t(\theta)-1|}{M_t} = \beta_t(x,y). \tag{37}$$

Let $D_{-t} := \sum_{j\neq t} |\log w_j(\theta)|$. The inequality $1+|a-1| \le e^{|\log a|}$ for $a>0$ implies

$$\beta_t(x,y) \le w_t(\theta)(e^{D_{-t}}-1) + \frac{w_t(\theta)|w_t(\theta)-1|}{M_t}. \tag{38}$$

If $w_t(\theta) > 1$, the difference between $e^D - 1$ and the right-hand side of (38) is

$$(w_t(\theta)-1)\left(1-\frac{w_t(\theta)}{M_t}\right) > 0. \tag{39}$$

If $0 < w_t(\theta) < 1$, the same difference equals

$$(1-w_t(\theta))\left[e^{D_{-t}}\frac{1+w_t(\theta)}{w_t(\theta)} - 1 - \frac{w_t(\theta)}{M_t}\right] > 0. \tag{40}$$

For $w_t(\theta) = 1$, the two components are equal. Thus $B_t(x,y) \le \beta_t(x,y) \le e^D - 1$, with strict inequality whenever $w_t(\theta) \neq 1$. Combining (37) and (38) proves our conclusion. □

A.3 Proof of top-K Approximation

Proposition 2. *When approximating SIS with Algorithm 1, the accepted tokens follow the conditional distribution $\hat{\pi}_\theta(v \mid x, y_{<t}) = \pi_\theta(v \mid x, y_{<t}, v \in \mathcal{V}_K)$. Let $\xi_K^{\text{old}} = \sum_{v\notin\mathcal{V}_K} \pi_{\theta_{\text{old}}}(v \mid x, y_{<t})$ denote the probability mass of the proposal policy outside $\mathcal{V}_K$. Then*

$$D_{\text{TV}}(\hat{\pi}_\theta(\cdot \mid x, y_{<t}) \,\|\, \pi_\theta(\cdot \mid x, y_{<t})) \;\le\; \xi_K, \tag{12}$$

where $\xi_K = \xi_K^{\text{old}} + \sum_{v\notin\mathcal{V}_K} |\pi_\theta(v \mid x, y_{<t}) - \pi_{\theta_{\text{old}}}(v \mid x, y_{<t})|$, and D_{TV} is the total variation distance.

Proof. Let $p(v) = \pi_\theta(v \mid x, y_{<t})$ denote the target distribution and $q(v) = \pi_{\theta_{\text{old}}}(v \mid x, y_{<t})$ the proposal. Define $\mathcal{V}_K = \text{TopK}(q, K)$ and let $\bar{\xi}_K = \sum_{v\notin\mathcal{V}_K} p(v)$ denote the target tail mass outside $\mathcal{V}_K$.

Within $\mathcal{V}_K$, the approximate envelope constant is $\widehat{M} = \max_{v\in\mathcal{V}_K} \frac{p(v)}{q(v)}$. The acceptance probability for $v \in \mathcal{V}_K$ is $\mathbb{P}(A=1 \mid v) = \frac{p(v)}{\widehat{M}\, q(v)}$, and for $v \notin \mathcal{V}_K$, $\mathbb{P}(A=1 \mid v) = 0$.

The conditional distribution of accepted tokens is

$$\hat{\pi}_\theta(v) = \mathbb{P}(v \mid A=1) = \frac{\mathbb{P}(A=1 \mid v)\, q(v)}{\sum_{v'\in\mathcal{V}_K} \mathbb{P}(A=1 \mid v')\, q(v')} = \frac{p(v)\,/\,\widehat{M}}{\sum_{v'\in\mathcal{V}_K} p(v')\,/\,\widehat{M}} = \frac{p(v)}{\sum_{v'\in\mathcal{V}_K} p(v')}, \tag{41}$$

for $v \in \mathcal{V}_K$, and $\hat{\pi}_\theta(v) = 0$ for $v \notin \mathcal{V}_K$, so that proves our first conclusion. Next, we compute the total variation distance

$$\begin{aligned}D_{\text{TV}}(\hat{\pi}_\theta \,\|\, p) &= \frac{1}{2}\sum_{v\in\mathcal{V}} |\hat{\pi}_\theta(v) - p(v)|\\ &= \frac{1}{2}\sum_{v\in\mathcal{V}_K}\left(\frac{p(v)}{1-\bar{\xi}_K} - p(v)\right) + \frac{1}{2}\sum_{v\notin\mathcal{V}_K} p(v)\\ &= \frac{1}{2}\sum_{v\in\mathcal{V}_K} p(v)\cdot\frac{\bar{\xi}_K}{1-\bar{\xi}_K} + \frac{1}{2}\bar{\xi}_K\\ &= \frac{1}{2}(1-\bar{\xi}_K)\cdot\frac{\bar{\xi}_K}{1-\bar{\xi}_K} + \frac{1}{2}\bar{\xi}_K = \bar{\xi}_K.\end{aligned} \tag{42}$$

Finally, since $\mathcal{V}_K = \mathrm{TopK}(q, K)$ is the top-K set under the proposal $q = \pi_{\theta_{\text{old}}}$, we bound the target tail mass $\bar{\xi}_K$ by the proposal tail and the policy drift:

$$D_{\mathrm{TV}}(\hat{\pi}_\theta \,\|\, p) = \bar{\xi}_K = \sum_{v \notin \mathcal{V}_K} p(v) = \sum_{v \notin \mathcal{V}_K} q(v) + \sum_{v \notin \mathcal{V}_K} \big(p(v) - q(v)\big) \\ \leq \sum_{v \notin \mathcal{V}_K} q(v) + \sum_{v \notin \mathcal{V}_K} \big|p(v) - q(v)\big| = \xi_K, \tag{43}$$

where $\xi_K = \xi_K^{\text{old}} + \sum_{v \notin \mathcal{V}_K} |p(v) - q(v)|$ and $\xi_K^{\text{old}} = \sum_{v \notin \mathcal{V}_K} q(v)$ is the proposal mass outside $\mathcal{V}_K$. Both terms of ξ_K are small in the near-on-policy regime: ξ_K^{old} vanishes because $\mathcal{V}_K$ collects the highest-probability tokens of q, and the residual L_1 term is bounded by the policy drift and only accumulates over the low-mass tail $v \notin \mathcal{V}_K$. □

B Algorithm

We summarize the complete SIS procedure with the top-K envelope approximation in Algorithm 1. For each token position, the algorithm first identifies the top-K most probable tokens under the behavior policy and restricts the rejection envelope to this set. Tokens falling inside $\mathcal{V}_K$ undergo an acceptance test: accepted tokens are certified as on-policy and receive unit importance weight, while rejected tokens and those outside $\mathcal{V}_K$ retain their original importance ratio. The resulting modified ratios are then directly substituted into any standard policy-gradient objective.

Algorithm 1 SIS with Top-K Envelope Approximation

Require: Prompt x; responses $\{y_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(\cdot \mid x)$; top-$K$ parameter K
Ensure: Modified importance ratios $\{\widetilde{w}_{i,t}(\theta)\}$

for each response y_i, $i = 1, \ldots, G$ **do**
  for each token position $t = 1, \ldots, |y_i|$ **do**
    Compute token-level importance ratio $w_{i,t}(\theta)$
    Select top-K set $\mathcal{V}_K$ under $\pi_{\theta_{\text{old}}}(\cdot \mid x, y_{i,<t})$
    Compute approximation constant $\widehat{M}_t$ restricted to $\mathcal{V}_K$ ▷ Eq. (7)
    if $y_{i,t} \in \mathcal{V}_K$ **then**
      Accept token with probability $w_{i,t}(\theta)/\widehat{M}_t$; set $z_t \leftarrow 1$ if accepted ▷ Eq. (8)
    else
      $z_t \leftarrow 0$ ▷ Tokens outside $\mathcal{V}_K$ remain off-policy
    end if
    Set $\widetilde{w}_{i,t}(\theta) \leftarrow 1$ if $z_t = 1$, else retain $w_{i,t}(\theta)$ ▷ Eq. (10)
  end for
end for
Plug $\{\widetilde{w}_{i,t}(\theta)\}$ into the policy-gradient objective ▷ e.g., GRPO, DAPO, GSPO

C Experimental Setup Details

C.1 Math and Agent Experiments

Table 4 (math) and Table 5 (agent) summarize the hyperparameters. For each baseline, the vanilla run and the SIS run share the same training and evaluation hyperparameters; SIS only adds the top-K approximation parameter used to compute the modified importance ratio. For agent training, we follow the settings of Search-R1 (Jin et al., 2025), using an E5 retriever (Wang et al., 2022) and the 2018 Wikipedia dump (Karpukhin et al., 2020) as the corpus.

For all training-dynamics figures we plot only the first 400 gradient steps: beyond that the policy has largely converged and $\pi_\theta \approx \pi_{\theta_{\text{old}}}$, so the curves flatten and the off-policy gap of interest is no longer visible.

Table 4: Hyperparameters used in math experiments.

Hyperparameter	GRPO	DAPO	GSPO
Learning rate	1×10^{-6}		
Mini-batch size (mbs)	256		
Responses per prompt	8		
Max prompt length	1,024		
Max response length	3,072		
Gradient steps	1,600		
Train temperature	1.0		
Eval temperature / top-p	1.0 / 0.7		
Entropy loss coefficient	0		
Training engine	FSDP		
Inference engine	vLLM		
KL loss coefficient	1×10^{-3}	0	0
SIS top-K (dense / MoE)	10 / 10	30 / 30	50 / 30
ε_{low}	0.2	0.2	3×10^{-4}
$\varepsilon_{\text{high}}$	0.2	0.28	4×10^{-4}
Global batch size (gbs)	$256 \times N$		
GPUs	32 (colocated)		

Table 5: Hyperparameters used in agent experiments.

Hyperparameter	GRPO	DAPO	GSPO
Learning rate	1×10^{-6}		
Mini-batch size (mbs)	64		
Responses per prompt	8		
Max prompt length	4,096		
Max response length	4,096		
Gradient steps	800		
Train temperature	1.0		
Eval temperature / top-p	1.0 / 1.0		
Entropy loss coefficient	0		
Training engine	FSDP		
Inference engine	vLLM		
KL loss coefficient	1×10^{-3}	0	0
SIS top-K (dense)	50	50	50
ε_{low}	0.2	0.2	3×10^{-4}
$\varepsilon_{\text{high}}$	0.2	0.28	4×10^{-4}
Global batch size (gbs)	256		
GPUs	16 (colocated)		

C.2 Reward Design

All experiments use a purely outcome-based, rule-based reward; we apply no learned reward model, no format or length shaping, and no intermediate-step rewards. For a prompt x and a generated response y, let $\hat{a}(y)$ be the answer extracted from y and $a^\star$ (or $\mathcal{A}^\star$) denote the gold answer set.

Math. We extract the final answer from the $\texttt{\textbackslash boxed}\{\}$ field of y and compare it against the reference with a symbolic equivalence checker $\mathrm{Eq}(\cdot,\cdot)$ that handles numeric and algebraic normalization. The reward is binary,

$$r_{\text{math}}(x,y) \;=\; \begin{cases} 1, & \hat{a}(y) \text{ is parseable from } \texttt{\textbackslash boxed}\{\} \text{ and } \mathrm{Eq}(\hat{a}(y),\, a^\star) = \text{True}, \\ 0, & \text{otherwise.} \end{cases} \tag{44}$$

Responses that omit a parseable $\texttt{\textbackslash boxed}\{\}$ answer receive reward 0.

Agent. Following Search-R1 (Jin et al., 2025), the reward is the exact-match (EM) score of the final answer:

$$r_{\text{agent}}(x,y) \;=\; \mathrm{EM}(a_{\text{pred}},\, a_{\text{gold}}), \tag{45}$$

where a_{pred} is the final answer extracted from the model response y (enclosed in the <answer> · </answer> tags), and a_{gold} denotes the ground-truth answer. Search-engine invocations and retrieved passages are not rewarded directly; they affect the reward only through the final answer.

C.3 Prompt Template

For math reasoning, we adopt the same prompt format as Qwen-Math (Yang et al., 2024). For agent reasoning, we follow Search-R1 (Jin et al., 2025) and use a template that enforces a minimal yet sufficient structure without introducing content-specific biases. The template organizes the model output into three iterative stages: (i) a reasoning phase, (ii) a search engine invocation phase, and (iii) a final answer. We illustrate the templates in Table 6.

D Additional Experimental Results

D.1 SIS Enables Stable Training Without Clipping

To test whether SIS alone suffices for training stability, we disable clipping entirely in GRPO+SIS on Qwen3-8B-Base. Table 7 shows that *SIS without clipping still outperforms the clip-based GRPO* on both math and agent tasks, confirming that SIS is a stabilization mechanism in its own right, and one that does not sacrifice gradient information.

D.2 Accept Rate Beyond Dense Math

To show that the on-policy certificate of SIS remains effective beyond the dense-math regime, we further inspect its token-level accept rate on the larger MoE backbone (Qwen3-30B-A3B-Base)

Table 6: Prompt templates for math and agent tasks. The placeholder `question` is replaced by the actual query during both training and inference.

Task	Prompt Template
Math	Please reason step by step, and put your final answer within \boxed{}. question.
Agent	Answer the given question. You must conduct reasoning inside <think> and </think> first every time you get new information. After reasoning, if you find you lack some knowledge, you can call a search engine by <search> query </search>, and it will return the top searched results between <information> and </information>. You can search as many times as you want. If you find no further external knowledge needed, you can directly provide the answer inside <answer> and </answer> without detailed illustrations. For example, <answer> xxx </answer>. Question: question.

Table 7: SIS does not require clipping on Qwen3-8B-Base. Best is **bolded**.

Math					
Method	MATH500	AMC23	AIME24	AIME25	Avg
GRPO	80.4	74.14	23.44	18.23	49.05
+ SIS	**85.6**	**79.53**	**25.83**	**19.38**	**52.59**
+ SIS (w/o Clip)	82.2	76.72	28.33	20.00	51.81

Agent						
Method	NQ	TriviaQA	PopQA	HotpotQA	Musique	Bamb
GRPO	49.14	66.99	44.68	46.24	19.24	48.8
+ SIS	**49.78**	67.47	**49.29**	**47.90**	19.74	**52.8**
+ SIS (w/o Clip)	49.58	**67.90**	46.88	46.56	**20.94**	52.0

and on the agentic search setting (Qwen3-14B-Base). As shown in Figure 6, SIS maintains an appreciable accept rate across all three policy-gradient algorithms in both settings. The accept rate is naturally lower on the agentic task, since interleaving reasoning with diverse tool calls induces a larger distributional shift between $\pi_{\theta_{\text{old}}}$ and π_θ; nevertheless it stays well above 0.6, indicating that a substantial fraction of tokens remain effectively on-policy under π_θ.

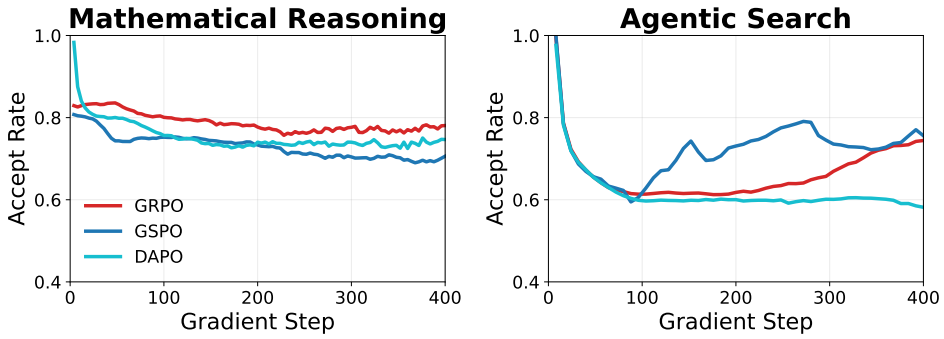

Figure 6: Accept rate of SIS on Qwen3-30B-A3B-Base under math reasoning (left) and Qwen3-14B-Base under agentic search (right). SIS sustains an appreciable accept rate across all algorithms in both settings. The lower rate on the agentic task reflects the larger distributional shift introduced by interleaved tool calls between $\pi_{\theta_{\text{old}}}$ and π_θ.

D.3 Negligible Time Cost of SIS

We profile the wall-clock cost introduced by SIS on top of vanilla GRPO. The setup follows our main math experiments: Qwen3-8B-Base, batch size 512 with $n = 8$ rollouts per prompt, response length 4,096, trained on 8×NVIDIA A800-80GB GPUs. We instrument the five SIS-specific operations: the top-K extraction during the `old_log_prob` stage, and the per-microbatch off-policy-to-on-policy correction during `update_actor`. We sweep $K \in \{10, 50, 100\}$ to assess how the overhead scales.

Figure 7 decomposes the per-step wall-clock into rollout, $\pi_{\theta_{\text{old}}}$ log-prob, actor update, and the additional SIS time. The total SIS overhead is $\sim 5.2\,\text{s}$ per step, which corresponds to roughly 1% of the 465–499 s step time across all three values of K. Furthermore, the overhead is essentially K-invariant: because the dominant cost is the off-policy-to-on-policy ratio reconstruction, which is fully parallelized on GPU and scales with $O(B \cdot L)$ rather than K, varying K from 10 to 100 leaves the wall-clock virtually unchanged. The only price paid by a larger K is GPU memory for the cached top-K logits, not compute time.

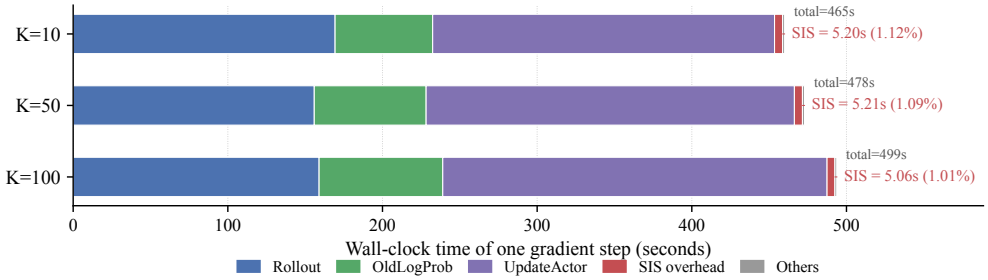

Figure 7: Per-step wall-clock decomposition of GRPO+SIS on Qwen3-8B-Base across $K \in \{10, 50, 100\}$. The red segment is the SIS overhead: near-constant $\sim 5.2\,\text{s}$ ($\sim 1\%$ of step time) and K-invariant.

D.4 Training Dynamics

Beyond final accuracy and accept rate, we further visualize three training-side dynamics for all SIS runs in Table 1: token-level training reward, policy entropy, and gradient norm, summarized in Figure 8.

Training reward. The first column of Figure 8 shows that the three policy-gradient algorithms paired with SIS all exhibit smooth, monotonically increasing reward curves on Qwen3-8B (math), Qwen3-30B-A3B-Base (math), Qwen3-8B (agentic search) and Qwen3-14B (agentic search), and converge to comparable levels within each setting. This indicates that SIS optimizes the training objective stably regardless of the underlying policy-gradient algorithm or the model scale.

Policy entropy. The middle column plots the policy entropy over the same horizon. Across all four settings, SIS keeps the entropy bounded away from zero, avoiding the entropy collapse commonly observed for aggressive off-policy training. On math tasks the entropy first drops as the policy sharpens on the training distribution, then stabilizes; GSPO+SIS in particular exhibits a mild early rebound before settling into a healthy range. On the agentic search tasks the entropy decreases more gradually due to the longer trajectories induced by tool calls, but again remains well above zero throughout training.

Gradient norm. The right column shows the gradient norm on a log scale. The gradient norm stays bounded within roughly 10^{-2}–10^{1} throughout training and does not diverge in any of the four settings. Occasional spikes appear on the agentic search runs, mostly on GSPO, reflecting the higher variance induced by variable-length tool-augmented rollouts; however these spikes remain within one order of magnitude of the median and do not lead to sustained instability, consistent with the smooth reward curves in the first column.

D.5 Robustness Under Stale Rollouts

We vary N, the number of mini-batch updates per rollout, to increase the drift from $\pi_{\theta_{\text{old}}}$ to π_θ within each rollout cycle. Table 8 reports final scores after 1600 gradient steps under all three settings $N = 4$, 8, and 16 grouped by algorithm, so the effect of staleness is directly comparable within each method. Two patterns emerge. First, staleness is harmful in itself: as N grows from 4 to 16, average accuracy drops for every algorithm regardless of whether SIS is applied, confirming that reusing stale off-policy rollouts degrades RL. Second, SIS yields a stable gain at every staleness level: it improves over the corresponding baseline at all N for GRPO, DAPO, and GSPO, and this advantage holds throughout the full 1600-step training.

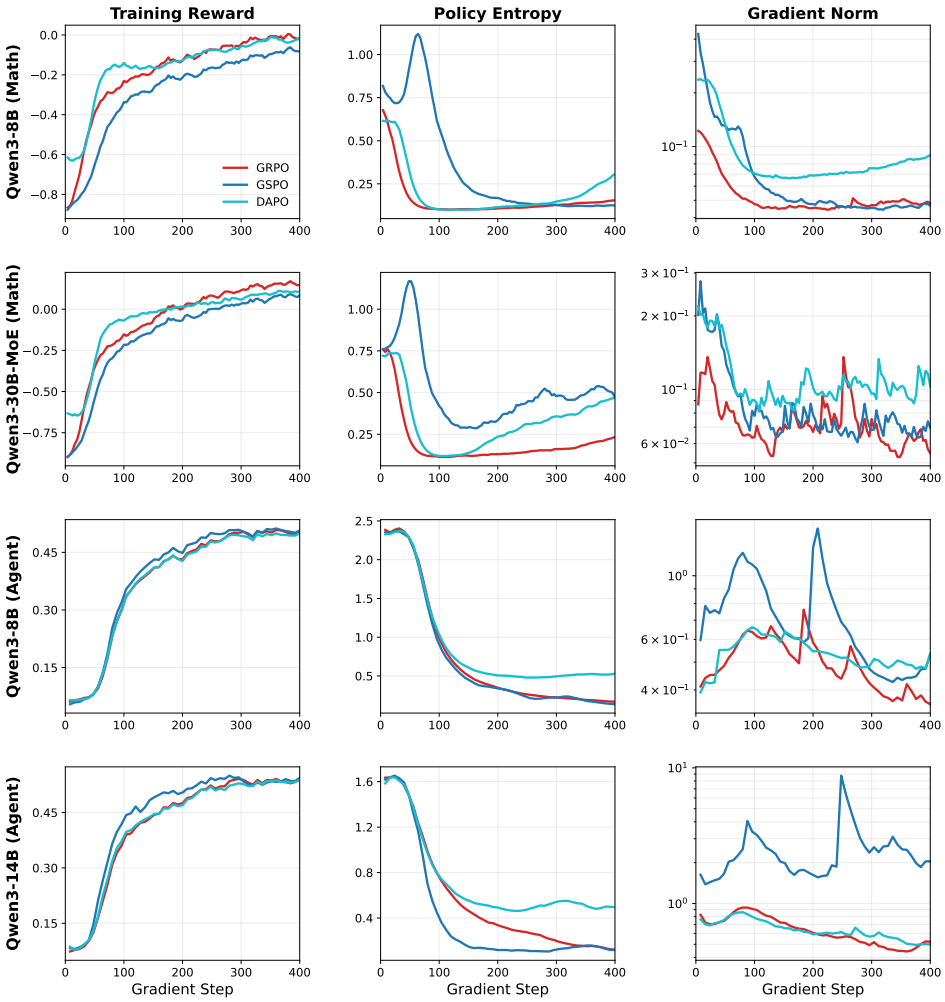

Figure 8: Training dynamics of SIS over the first 400 gradient steps for all entries in Table 1. Rows: Qwen3-8B/Qwen3-30B-A3B-Base on math and Qwen3-8B/Qwen3-14B on agentic search. Columns: training reward, policy entropy, and gradient norm (log scale). SIS produces stable reward growth and non-collapsed entropy across all backbones and algorithms; occasional gradient-norm spikes on agentic search runs are transient and do not lead to sustained instability.

Accept rate under staleness. To further explain why SIS remains effective as N grows, we plot the token-level accept rate during training in Figure 9. Across all three algorithms and all staleness levels, the accept rate stays consistently in the 0.80–0.95 range and never collapses. This confirms that even with $N = 16$ a substantial fraction of tokens generated by $\pi_{\theta_{\text{old}}}$ are still on-policy under π_θ, so the on-policy certificate of SIS continues to provide a meaningful correction signal under high staleness.

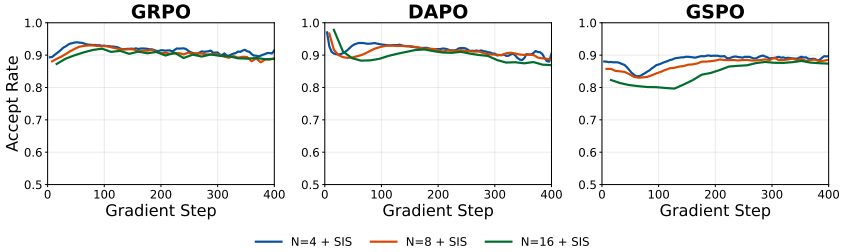

Figure 9: Token-level accept rate of SIS during the first 400 gradient steps on Qwen3-8B-Base under staleness $N \in \{4, 8, 16\}$. Across all three algorithms, the accept rate stays in 0.80–0.95 throughout training, indicating that most tokens remain effectively on-policy even at $N = 16$.

Table 8: Performance of SIS under varying policy staleness on Qwen3-8B-Base after 1600 gradient steps, where staleness N is the number of mini-batch updates per rollout. Results are grouped by algorithm so that the effect of increasing N is directly comparable within each method. Shaded rows denote the application of SIS on top of each baseline. Within each algorithm–staleness pair, the higher accuracy per metric is **bolded**.

N	Method	MATH500	AMC23	AIME24	AIME25	Avg
			GRPO			
4	Base	80.4	74.14	23.44	18.23	49.05
	w/ SIS	**85.6**	**79.53**	**25.83**	**19.38**	**52.59**
8	Base	81.6	71.33	22.91	**20.73**	49.14
	w/ SIS	**83.8**	**78.59**	**24.08**	20.10	**51.64**
16	Base	83.0	69.53	23.54	18.23	48.58
	w/ SIS	**84.6**	**77.03**	**24.69**	**21.35**	**51.92**
			DAPO			
4	Base	83.2	81.88	21.56	18.33	51.24
	w/ SIS	**84.2**	**86.79**	**26.98**	**22.60**	**55.14**
8	Base	83.6	78.59	**24.79**	19.17	51.54
	w/ SIS	**84.2**	**83.28**	24.69	**21.46**	**53.41**
16	Base	83.4	73.83	20.42	17.60	48.81
	w/ SIS	**85.0**	**81.17**	**24.48**	**21.04**	**52.92**
			GSPO			
4	Base	**85.6**	**78.83**	26.46	18.85	52.44
	w/ SIS	85.2	77.19	**28.33**	**23.44**	**53.54**
8	Base	83.4	73.75	25.31	18.02	50.12
	w/ SIS	83.4	**78.20**	**25.52**	**21.77**	**52.22**
16	Base	82.0	71.72	22.92	**19.48**	49.03
	w/ SIS	**83.8**	**73.83**	**23.85**	19.27	**50.19**

D.6 Generalization Beyond Qwen

To verify that our findings generalize beyond a single model family, we further evaluate SIS on Llama-3.2-3B-Instruct (Grattafiori et al., 2024). We use the same DAPO-Math-17K training corpus and keep all hyperparameters identical to the Qwen3-8B-Base math setting (Table 4), ensuring a controlled comparison.

As presented in Table 9, SIS again delivers consistent plug-in gains across all three algorithms, with the largest absolute lift on DAPO ($+4.47$ Avg). In relative terms the improvements are even more pronounced ($+11.0\%$ for GRPO and $+14.4\%$ for DAPO), notably larger than the corresponding relative gains on Qwen3-8B-Base. We attribute this to the smaller Llama model exhibiting a wider policy–rollout gap, making the off-policy correction provided by SIS even more impactful. These results confirm that SIS delivers robust gains independent of the underlying model architecture and family.

E Stabilization Tricks Through the Importance-Sampling Lens

We organize Table 2 by writing every stabilization trick as a token-level rescaling of the importance-sampling (IS) coefficient

$$w_{i,t}(\theta) \;=\; \frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})}. \tag{46}$$

Under this lens, each method amounts to a specific surrogate $g(w_{i,t}(\theta))$ used in place of $w_{i,t}(\theta)$ inside the policy-gradient objective. Clipping-based methods (GRPO, DAPO, GSPO, CISPO) reshape w directly; DPPO-TV and Clip-Cov instead constrain the update through an auxiliary signal, which we recast as multiplying w by an indicator that masks out-of-region tokens. We first restate the six representative methods in this unified form, and then use DAPO and GSPO as concrete cases to illustrate how SIS is instantiated on top of an existing trick.

Table 9: Performance of SIS as a plug-in on Llama-3.2-3B-Instruct across math benchmarks. Shaded rows denote the application of SIS on top of each baseline. The best accuracy per column is **bolded**. Δ denotes the absolute gain over the corresponding baseline.

Method	MATH500	AMC23	AIME24	AIME25	Avg	Δ
GRPO	50.0	44.77	12.40	0.10	26.82	–
w/ SIS	51.2	53.36	13.75	**1.77**	29.76	+2.94
DAPO	57.2	52.74	13.96	0.21	31.03	–
w/ SIS	**58.0**	**65.47**	**17.50**	1.04	**35.50**	+4.47
GSPO	50.2	43.36	11.15	0.52	26.31	–
w/ SIS	50.6	43.36	13.94	0.63	27.13	+0.82

E.1 Representative stabilization methods

GRPO (Shao et al., 2024). The vanilla token-level PPO-style ratio with a symmetric clip bound ε,

$$g_{\mathrm{GRPO}}(w) \;=\; \mathrm{clip}(w,\, 1-\varepsilon,\, 1+\varepsilon). \tag{47}$$

DAPO (Yu et al., 2025). Decoupled clip bounds $\varepsilon_{\mathrm{low}}, \varepsilon_{\mathrm{high}}$ with the KL regularization removed,

$$\mathcal{J}_{\mathrm{DAPO}}(\theta) = \mathbb{E}_{(x,a)\sim\mathcal{D},\{y_i\}\sim\pi_{\theta_{\mathrm{old}}}} \left[\frac{1}{\sum_i |y_i|} \sum_{i,t} \min\Big(w_{i,t}(\theta)\widehat{A}_{i,t},\; \mathrm{clip}(w_{i,t}(\theta), 1-\varepsilon_{\mathrm{low}}, 1+\varepsilon_{\mathrm{high}})\widehat{A}_{i,t} \Big) \right], \tag{48}$$

restricted to prompts with $0 < |\{y_i \mid \texttt{is_equivalent}(a, y_i)\}| < G$.

GSPO (Zheng et al., 2025b). A sequence-level ratio obtained by averaging log-ratios,

$$s_i(\theta) \;=\; \exp\left(\frac{1}{|y_i|} \sum_{t=1}^{|y_i|} \log \frac{\pi_\theta(y_{i,t}|x, y_{i,<t})}{\pi_{\theta_{\mathrm{old}}}(y_{i,t}|x, y_{i,<t})} \right), \tag{49}$$

which then enters a sequence-level clipped objective with bounds $\varepsilon_{\mathrm{low}}, \varepsilon_{\mathrm{high}}$.

CISPO (Chen et al., 2025). A stop-gradient one-sided clip that detaches the high tail,

$$g_{\mathrm{CISPO}}(w) \;=\; \mathrm{sg}[\min(w,\, 1+\varepsilon_{\mathrm{hi}})]\,. \tag{50}$$

DPPO-TV (Qi et al., 2026). Instead of bounding w, it bounds the per-token total-variation distance between the current and behavior token distributions,

$$\mathrm{TV}_{i,t}(\theta) \;=\; \tfrac{1}{2}\big\|\pi_\theta(\cdot \mid x, y_{i,<t}) - \pi_{\theta_{\mathrm{old}}}(\cdot \mid x, y_{i,<t})\big\|_1, \tag{51}$$

and discards the gradient of any token whose policy drift exceeds a threshold δ, which is equivalent to the effective coefficient $w \cdot \mathbf{1}[\mathrm{TV}_{i,t} \le \delta]$.

Clip-Cov (Cui et al., 2025). It attributes entropy collapse to a small set of tokens with large log-probability–advantage covariance,

$$\mathrm{Cov}_{i,t}(\theta) \;=\; \big(\log \pi_\theta(y_{i,t} \mid x, y_{i,<t}) - \overline{\log \pi}\big)\big(\widehat{A}_{i,t} - \overline{A}\big), \tag{52}$$

where $\overline{\log \pi}$ and $\overline{A}$ are batch means. It detaches the gradient of tokens whose covariance exceeds a threshold ω, giving the effective coefficient $w \cdot \mathbf{1}[\mathrm{Cov}_{i,t} \le \omega]$.

E.2 Instantiating SIS: DAPO and GSPO as cases

Since SIS only replaces the token-level ratio $w_{i,t}(\theta)$ with its on-policy-equivalent counterpart $\widetilde{w}_{i,t}(\theta)$ (cf. (10)), instantiating SIS on top of any of the methods above amounts to a one-line substitution $w_{i,t} \to \widetilde{w}_{i,t}$ in the corresponding objective. We illustrate this with DAPO and GSPO, the two cases used in our experiments; analogous substitutions apply to the remaining methods and are left to future empirical study.

Case 1: DAPO + SIS. Substituting $w_{i,t} \to \widetilde{w}_{i,t}$ into the DAPO objective yields

$$\mathcal{J}^{\mathrm{SIS}}_{\mathrm{DAPO}}(\theta) = \mathbb{E}_{(x,a)\sim\mathcal{D},\{y_i\}_{i=1}^{G}\sim\pi_{\theta_{\mathrm{old}}}(\cdot|x)} \left[\frac{1}{\sum_{i=1}^{G}|y_i|} \sum_{i=1}^{G}\sum_{t=1}^{|y_i|} \min\Big(\widetilde{w}_{i,t}(\theta)\widehat{A}_{i,t},\ \mathrm{clip}(\widetilde{w}_{i,t}(\theta), 1-\varepsilon_{\mathrm{low}}, 1+\varepsilon_{\mathrm{high}})\widehat{A}_{i,t}\Big) \right], \quad (53)$$
$$\text{s.t. } 0 < |\{y_i \mid \texttt{is_equivalent}(a, y_i)\}| < G,$$

where a is the ground-truth answer to the prompt x.

Case 2: GSPO + SIS. For sequence-level GSPO, accepted tokens contribute a zero log-ratio, giving the SIS-modified sequence ratio

$$\widetilde{s}_i(\theta) = \exp\left(\frac{1}{|y_i|} \left(\sum_{(i,t)\in\mathcal{A}} \log \frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\mathrm{sg}[\pi_\theta(y_{i,t} \mid x, y_{i,<t})]} + \sum_{(i,t)\notin\mathcal{A}} \log \frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\mathrm{old}}}(y_{i,t} \mid x, y_{i,<t})} \right) \right), \quad (54)$$

which then enters the GSPO clipped objective

$$\mathcal{J}^{\mathrm{SIS}}_{\mathrm{GSPO}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\{y_i\}_{i=1}^{G}\sim\pi_{\theta_{\mathrm{old}}}(\cdot|x)} \left[\frac{1}{G}\sum_{i=1}^{G} \min\Big(\widetilde{s}_i(\theta)\widehat{A}_i,\ \mathrm{clip}(\widetilde{s}_i(\theta), 1-\varepsilon_{\mathrm{low}}, 1+\varepsilon_{\mathrm{high}})\widehat{A}_i\Big) \right]. \quad (55)$$