On the Rollout-Training Mismatch in Modern RL Systems

Feng Yao[1,2*] FENGYAO@UCSD.EDU
Liyuan Liu[1*] LUCLIU@MICROSOFT.COM
Dinghuai Zhang[2,3] DINZHANG@MICROSOFT.COM
Chengyu Dong[2] CDONG@UCSD.EDU
Jingbo Shang[2] JSHANG@UCSD.EDU
Jianfeng Gao[1] JFGAO@MICROSOFT.COM
[1]*Microsoft Research* [2]*UC San Diego* [3]*Mila*

Abstract

Modern reinforcement learning (RL) systems aim to be efficient by employing hybrid designs for rollout generation (e.g., vLLM) and model training (e.g., FSDP). However, the implementation gap can implicitly turns on-policy RL into off-policy, as the rollout and training policies can produce significantly different token probabilities despite sharing the same model weights. We dive into this rollout-training mismatch problem and propose to use truncated importance sampling (TIS) as a simple yet effective fix. TIS applies importance sampling correction to bridge the distribution gap between rollout and training, enabling stable RL training even with quantized rollouts. We demonstrate TIS's effectiveness across multiple settings, showing it can preserve downstream performance while enabling significant speedups through rollout quantization. Our work provides algorithmic solution to address the systematic mismatch problem in efficient RL training.

1. Introduction

Modern reinforcement learning (RL) frameworks tend to apply hybrid engines to maximize training efficiency, such as using highly optimized inference engines (e.g., vLLM) for rollout generation while using separate backends (e.g., FSDP) for model training [2, 4].

However, this hybrid design brings an unexpected rollout-training mismatch issue [1]. As shown in fig. 1 (left), despite sharing the same model parameters θ, the rollout policy π_{vllm} and the training policy π_{fsdp} can produce significantly different token probabilities. The mismatch problem becomes particularly severe when using quantized rollouts (e.g., INT8, FP8), as quantization further amplifies the distribution gap between rollout and training policies, hurting the training effectiveness.

To address this systematic problem, we propose to use truncated importance sampling (TIS) [3], a simple yet effective algorithmic fix that applies importance sampling correction to bridge the distribution gap. TIS modifies the policy gradient computation by incorporating the importance ratio between the training and rollout policies. As shown in fig. 1 (right), applying TIS to the DAPO-32B RL setting [5] significantly improves the training effectiveness.

2. The Rollout-Training Mismatch Problem

For simplicity, we use the REINFORCE algorithm as an example. The policy gradient update is:

$$\theta \leftarrow \theta + \mu \cdot \mathbb{E}_{a \sim \pi(\theta)}[R(a) \cdot \nabla_\theta \log \pi(a, \theta)].$$

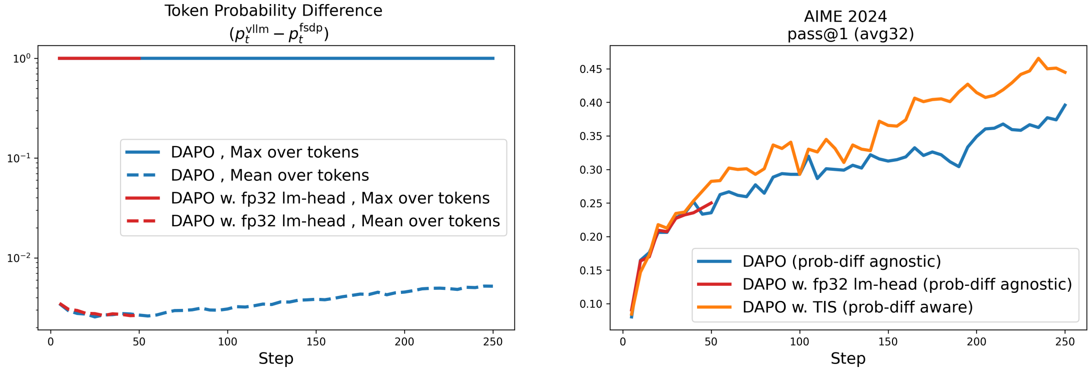

Figure 1: Left: Token probability differences brought by the mismatch problem. Right: Performance comparison between normal RL training and training after fixing the mismatch problem. Experiments are conducted on Qwen2.5-32B dense model using 4 nodes of 8xH100 GPUs.

In practice, modern RL frameworks employ hybrid computation designs where rollout generation uses highly optimized inference engines (e.g., vLLM) while model training uses separate backends (e.g., FSDP). This creates a mismatch:

$$\theta \leftarrow \theta + \mu \cdot \mathbb{E}_{a \sim \pi_{\text{vllm}}(\theta)}[R(a) \cdot \nabla_\theta \log \pi_{\text{fsdp}}(a, \theta)].$$

Here, π_{fsdp} denotes the model instantiated with the training backend and π_{vllm} represents the same model loaded with the inference engine. Despite sharing the same parameters θ, these policies can produce significantly different token probabilities, making the training implicitly off-policy.

Empirically, the impact of this rollout-training mismatch is significant. As shown in fig. 1 (left), for certain tokens, they even yield contradictory predictions—$\pi_{\text{vllm}}(a, \theta) = 1$ and $\pi_{\text{fsdp}}(a, \theta) = 0$, which breaks the on-policy assumption and secretly makes the RL training become off-policy.

3. System-level & Algorithm-level Methods

3.1. System-level Solution

One may suspect vLLM implementation as the root cause. We patched vLLM to (i) expose the true sampling probabilities rather than adjusted logprobs, and (ii) cast its lm_head to fp32 to match HuggingFace precision. However, as shown in fig. 1, the rollout–training mismatch persists even after these fixes, suggesting the problem is fundamental to hybrid backend designs.

3.2. Algorithm-level Solution

Instead of trying to eliminate the distribution mismatch at the system level, we propose to adapt the model update to be aware of this mismatch through importance sampling correction. We modify the gradient computation from:

$$\mathbb{E}_{a \sim \pi_{\text{vllm}}(\theta)}[R(a) \cdot \nabla_\theta \log \pi_{\text{fsdp}}(a, \theta)]$$

to:

$$\mathbb{E}_{a\sim\pi_{\text{vllm}}(\theta)}\Big[\frac{\pi_{\text{fsdp}}(a,\theta)}{\pi_{\text{vllm}}(a,\theta)}\cdot R(a)\cdot\nabla_\theta\log\pi_{\text{fsdp}}(a,\theta)\Big].$$

To ensure training stability, we use truncated importance sampling:

$$\mathbb{E}_{a\sim\pi_{\text{vllm}}(\theta)}\Big[\min\Big(\frac{\pi_{\text{fsdp}}(a,\theta)}{\pi_{\text{vllm}}(a,\theta)},C\Big)\cdot R(a)\cdot\nabla_\theta\log\pi_{\text{fsdp}}(a,\theta)\Big]$$

where C is a hyperparameter that controls the maximum importance ratio.

3.3. Extension to PPO

The same principle applies to PPO. The standard PPO policy gradient is:

$$\mathbb{E}_{a\sim\pi_{\theta_{\text{old}}}}\Big[\nabla_\theta\min\Big(\frac{\pi_\theta(a)}{\pi_{\theta_{\text{old}}}(a)}\,\hat{A},\text{clip}\big(\frac{\pi_\theta(a)}{\pi_{\theta_{\text{old}}}(a)},\,1-\epsilon,\,1+\epsilon\big)\,\hat{A}\Big)\Big]$$

With hybrid computation, this becomes:

$$\mathbb{E}_{a\sim\pi_{\text{vllm}}(\theta_{\text{old}})}\Big[\nabla_\theta\min\Big(\frac{\pi_{\text{fsdp}}(a,\theta)}{\pi_{\text{fsdp}}(a,\theta_{\text{old}})}\,\hat{A},\text{clip}\big(\frac{\pi_{\text{fsdp}}(a,\theta)}{\pi_{\text{fsdp}}(a,\theta_{\text{old}})},\,1-\epsilon,\,1+\epsilon\big)\,\hat{A}\Big)\Big]$$

We fix this with TIS:

$$\mathbb{E}_{a\sim\pi_{\text{vllm}}(\theta_{\text{old}})}\Big[\min\Big(\frac{\pi_{\text{fsdp}}(a,\theta_{\text{old}})}{\pi_{\text{vllm}}(a,\theta_{\text{old}})},C\Big)\cdot\nabla_\theta\,\min\Big(\frac{\pi_{\text{fsdp}}(a,\theta)}{\pi_{\text{fsdp}}(a,\theta_{\text{old}})}\,\hat{A},\text{clip}\Big(\frac{\pi_{\text{fsdp}}(a,\theta)}{\pi_{\text{fsdp}}(a,\theta_{\text{old}})},1-\epsilon,1+\epsilon\Big)\,\hat{A}\Big)\Big]$$

4. Experiments

4.1. Main results

We conduct experiments using Qwen2.5-32B dense model with the popular DAPO [5] recipe. We conduct evaluate on the AIME2024 benchmark.

As shown in fig. 1 (right), fixing the mismatch with TIS brings significant gains on the downstream performance of DAPO-32B training.

4.2. Fix the Mismatch in Quantized Rollout

We note that the mismatch problem can become more severe when using quantized rollouts (e.g., INT8, FP8) to accelerate RL training, while TIS can still effectively fix the problem.

We conduct regular PPO training on GSM8K dataset, both with a standard setup (bf16 rollouts), and a setup with INT8 quantized rollouts. As shown in fig. 2 (left), when employing quantized rollouts, the token probability difference between rollout (e.g., vLLM) and training (e.g., FSDP) becomes even larger. This leads to significant performance degradation as shown in fig. 2 (right). This again confirms that the distribution gap between rollout and training policies significantly affects RL training effectiveness.

However, applying TIS manages to mitigate the gap greatly, effectively allowing quantized rollouts to achieve a similar performance with standard rollouts, while retaining the training efficiency gain since TIS induces little computation overhead.

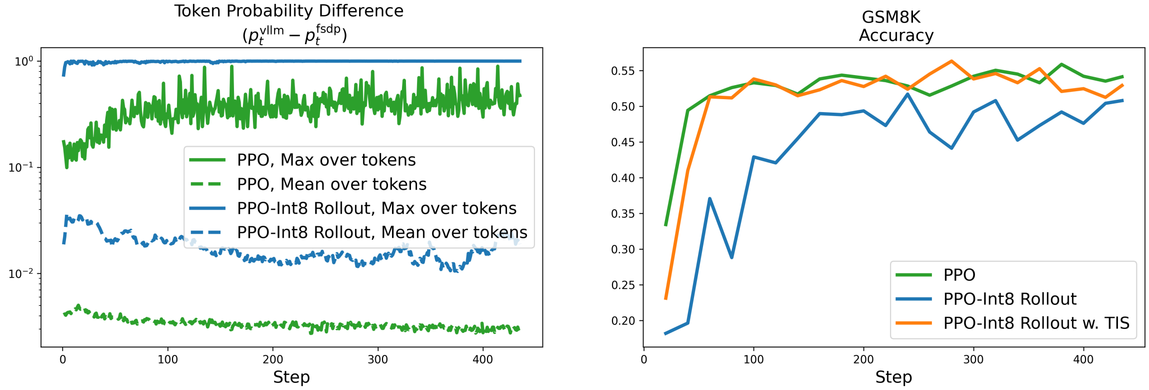

Figure 2: Left: Token-level probability differences. Right: Performance comparison for normal RL training on GSM8K and RL training with INT8 quantized rollouts. Experiments are conducted on Qwen2.5-0.5B dense model using one node of 4xA6000 GPU

4.3. Comparison with Alternative Approaches

To assess the effectiveness of TIS and understand the impact of its design choices, we conducted experiments comparing TIS with two variants below. As shown in Figure 3, TIS outperforms both variants consistently, especially in cases where the gap is large (e.g., FP8/INT8).

- **PPO-IS**: Directly incorporating importance sampling into PPO clipping

$$\mathbb{E}_{a\sim\pi_{\text{vllm}}(\theta_{\text{old}})}\Big[\nabla_\theta \min\Big(\frac{\pi_{\text{fsdp}}(a,\ \theta)}{\pi_{\text{vllm}}(a,\ \theta_{\text{old}})}\,\hat{A}, \text{clip}\Big(\frac{\pi_{\text{fsdp}}(a,\ \theta)}{\pi_{\text{vllm}}(a,\ \theta_{\text{old}})}, 1-\epsilon,\ 1+\epsilon\Big)\ \hat{A}\Big)\Big] \tag{1}$$

- **Vanilla-IS**: Using untruncated importance ratios

$$\mathbb{E}_{\pi_{\text{vllm}}(\theta_{\text{old}})}\Big[\underbrace{\frac{\pi_{\text{fsdp}}(a,\theta_{\text{old}})}{\pi_{\text{vllm}}(a,\theta_{\text{old}})}}_{\text{importance ratio}}\cdot\nabla_\theta \min\Big(\frac{\pi_{\text{fsdp}}(a,\ \theta)}{\pi_{\text{fsdp}}(a,\ \theta_{\text{old}})}\,\hat{A}, \text{clip}\Big(\frac{\pi_{\text{fsdp}}(a,\ \theta)}{\pi_{\text{fsdp}}(a,\ \theta_{\text{old}})}, 1-\epsilon,\ 1+\epsilon\Big)\ \hat{A}\Big)\Big] \tag{2}$$

5. Conclusion

In this paper, we show that modern RL systems with hybrid backends suffer from a rollout–training mismatch issue, where policies with identical parameters still produce different token distributions. This gap appears even in standard BF16 training and becomes more severe with quantized rollouts, breaking the on-policy assumption and degrading performance. We propose to use truncated importance sampling (TIS) as a simple and effective fix, and demonstrated that it restores stable training and preserves downstream accuracy while unlocking the efficiency gains of quantized rollouts. These results highlight the mismatch as a fundamental challenge in today's RL frameworks and establish TIS as a practical solution moving forward.

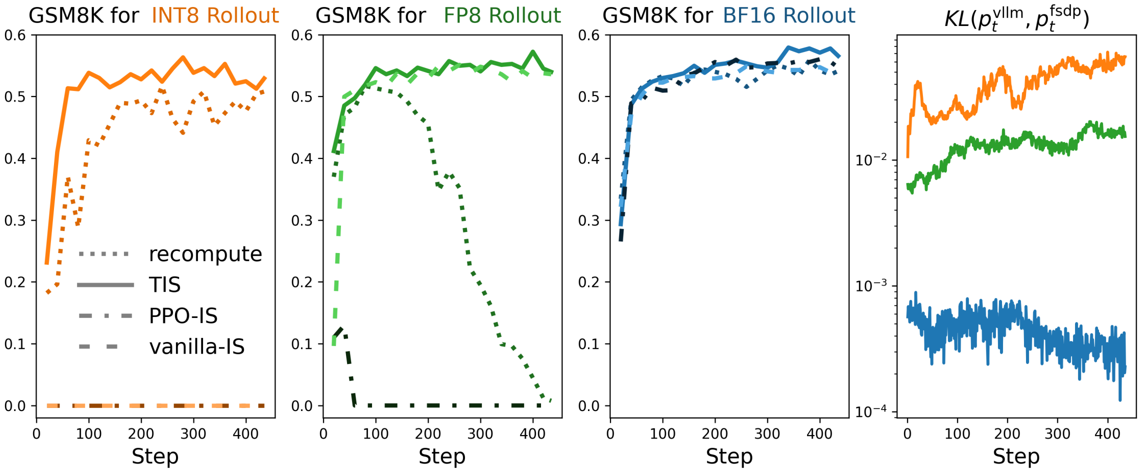

Figure 3: Ablation on different rollout-training mismatch mitigation strategies on Qwen2.5-0.5B with GSM8k. Note that PPO-IS and Vanilla-IS achieves near 0 accuracy for INT8 rollouts thus being highly overlapped. The KL divergence between the distributions of vLLM and FSDP engines is on the right.

References

[1] 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. *arXiv preprint arXiv:2506.13585*, 2025.

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

[3] Edward L Ionides. Truncated importance sampling. *Journal of Computational and Graphical Statistics*, 17(2):295–311, 2008.

[4] Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

[5] 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. *arXiv preprint arXiv:2503.14476*, 2025.