# [Intern Proposal] Efficient on Easy Tasks and Effective on Difficult Tasks via Dynamic Sampling.

Intern: [Eric Lan](https://quip-amazon.com/EFO9EA4Un0D)

Mentor: [Lian Xiong](https://quip-amazon.com/KGM9EAhyNmM)

## Background

As the era of autonomous agents making decisions on behalf of users unfolds, e.g., making personalized recommendations as shopping agents, reinforcement learning (RL) advances these language and reasoning models, as demonstrated in systems such as DeepSeek-R1-Zero and DeepSeek-R1 [1].

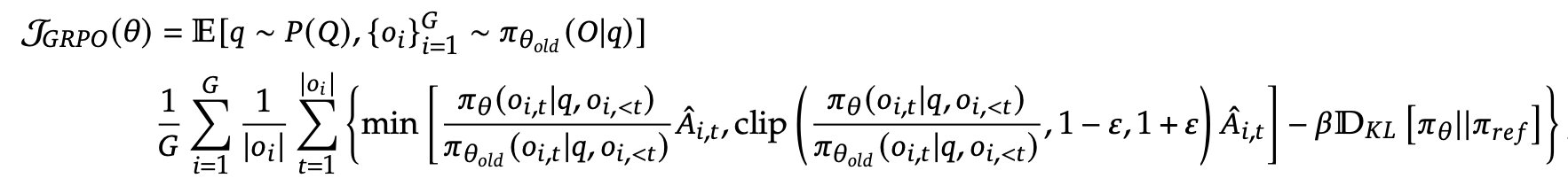
To reduce the computational overhead associated with RL, Group Relative Policy Optimization (GRPO) [1] uses Monte Carlo (MC) method to estimate the advantage functions in reinforcement learning, which eliminates the need for a critic network. This paradigm has shown great success, and leads the recent approaches, e.g., Dr. GRPO [4] and DAPO [5]. However, existing Monte Carlo estimation of advantage functions suffers from stability and exploration limitations. In this project, we aim to analyze these two critical issues in the MC estimation.

In [11], it shows that the larger rollout number brings better performance on difficult tasks in the mathematic domain with LRMs. It also indicates that more rollout samples in training could also bring shorter reasoning chains in inference.

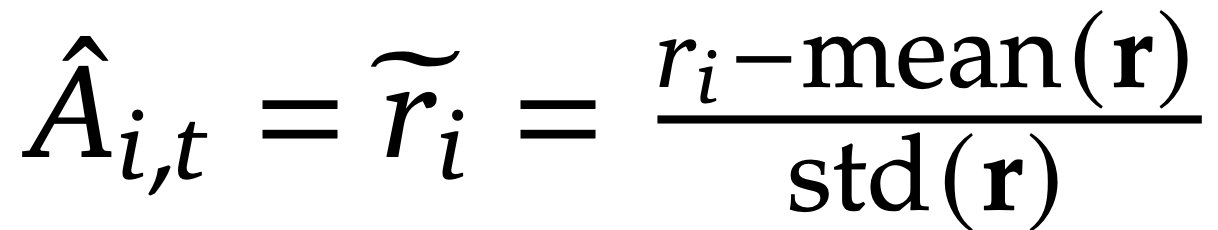
## Problem Statement

1. The **stability** of policy gradients with the MC estimation.

In GRPO, the goal is to maximize the following objective function



The advantage function is estimated with standardization as follows



However, in practical training process, the policy gradient becomes unstable and even NaN (exploding) occasionally [3, 4].

For example, given prompt **x**, the policy model generates 256 responses, where 255 of them have reward 0 and 1 of them has reward -1. The mean is -0.004, and the std is 0.06. For the response with reward -1, the advantage is -16.6, which is significantly beyond the normal range [-1, 1]. This brings instability in the training process.

One current trick in [3] simply removes the std to improve the stability. However, this arbitrary trick makes the estimation sensitive to the reward design such as the reward range.

We aim to analyze the current issue experimentally, and provide theoretical upper bound of the estimation. We then design a method to improve the stability in the training process and be robust to the reward design.

1. The **exploration** in rollout inference.

In all RL implementation, the rollout number is a fixed hyperparameter. However, a fixed rollout number leads to unnecessary computational costs on well-performing tasks, while providing insufficient exploration for tasks with unsatisfactory performance.

DAPO [5] abandons samples with responses that all have low rewards or all have high rewards. However, this approach has low data utilization, especially in the cases that have limited number of high quality data. (Less effective on difficult tasks)

GFPO [11] abandons responses with low rewards, which incorporates the rejecting sampling [14] idea into group policy optimization. (Less efficient on easy tasks)

With full data utilization, we aim to design a method with adaptive exploration ability, which improves both inference efficiency on well-performed tasks and data efficiency on difficult tasks. Specifically, we plan to design a mechanism that the policy model stops with good performance and keeps exploring with unsatisfactory performance. The metrics to evaluate the performance could be rewards from the verifier or the confidence (entropy) [7].

**Evidence**:

1. Multiple **independent inference** brings diversity in the **semantic** of reasoning paths, and temperature influences the diversity.

[12] shows that multiple times of independent inference bring more diversity in the semantic of reasoning paths.

1. Larger rollout number brings more exploration and better performance on difficult tasks [11].

**Basic Approach**:

(1) Dynamic rollout.

Keep sampling with multiple inference times until the model finds a better reasoning path. It saves cost on relatively easy tasks, and achieves better results on relatively difficult tasks.

**Dynamic++**:

* Intuition: If the model struggles to find a better reasoning path, encourage the model to explore more aggressively.
* Method: If the model behave insufficiently, encourage exploration by increasing the temperature sampling in the next round.
* Challenge: The sampling distribution keep changing in each round, which brings the **off-policy samples** for the on-policy algorithms.
* Solution: As the change is known, we can use dynamic importance sampling to balance the distribution shift.

In modern RL training frameworks (e.g., VERL), different implementations are used for rollout generation (e.g., vLLM) and model training (e.g., FSDP and Megatron). Fortunately, this **two-engine system** makes it possible to adapt the importance sampling mechanism in RL training.

This approach do not need data pre-processing, which is heavily labor intensive, and could adjust the training adaptively according to the current performance.

**Remark**: We do **NOT** aim to propose a new RL algorithm and compare to the previous baselines. This project aims to focus on the rollout estimation of advantage functions in policy optimization, which offers a pipeline and is general for all RL methods with the rollout estimation process, e.g., Dr. GRPO [4], DAPO [5], GSPO [6], and GMPO [13].

We aim to address the above issues with both theoretical analysis and empirical verification.

## Technical Approach

### 1. Data

We plan to test the idea on mathematic data first, e.g., GSM8K, MATH500, AIME. We also plan to extend the training on logic tasks, e.g., Zebra Puzzle and OrderLogic.

### 2. Model

We plan to start at Qwen2.5-7B-Instruct. We then adapt the method to different scales, including Qwen2.5-{1.5B, 3B, 14B}-Instruct models, and different series, including Llama and Mistral. We also plan to test the performance on large reasoning models (LRMs), including DeepSeek-R1-Distill models and Qwen3 models.

### 3. RL Training Framework

We plan to use the efficient RL training framework VERL.

## Expected Impact

### 1. Business Impact

* **Efficient Pipeline**: Our improvements streamline current training pipelines in shopping agents, reducing labor-intensive processes such as data preparation in curriculum learning.
* **Competitive Advantage**: Advanced reasoning capabilities will differentiate Amazon's AI systems from competitors.
* **Stable RL training:** stable policy gradient estimation will help current Neo RL behavior training, which suffers from rewards collapse when mixing different behavior RL tasks.

### 2. Technical Contributions

* **Novel RL Frameworks**: Development of new RL methodologies with better training stability and dynamic exploration according to the difficulty.
* **Enhanced Performance:** With improved MC estimation, we could achieve better final performance and enhance overall post-training efficiency and stability.

## Timeline

* Week 13: Wrap up.
* Week 11-12: Accumulate results and ablation study.
* Week 8-10: Scale the method to other models and GRPO variants.
* Week 6-7: Design the adaptive exploration in rollout inference.
* Week 3-5: Analyze the stability problem as the first step.
* Week 2-3: Proposal ready and onboard of MC-RL.
* Week 1: Onboarding setup, and meet team mates.

## Reference

[1] DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv preprint arXiv:2501.12948, 2025.

[2] DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv preprint arXiv:2402.03300, 2024.

[3] Part I: Tricks or Traps? A Deep Dive into RL for LLM Reasoning. arXiv preprint arXiv:2508.08221, 2025.

[4] Understanding R1-Zero-Like Training: A Critical Perspective. arXiv preprint arXiv:2503.20783, 2025.

[5] DAPO: An Open-Source LLM Reinforcement Learning System at Scale. arXiv preprint arXiv:2503.14476, 2025.

[6] Group Sequence Policy Optimization. arXiv preprint arXiv:2507.18071, 2025.

[7] Deep Think with Confidence. arXiv preprint arXiv:2508.15260, 2025.

[8] Truncated Importance Sampling. Journal of Computational and Graphical Statistics, 2008.

[9] Tapered Off-Policy REINFORCE: Stable and Efficient Reinforcement Learning for LLMs. arXiv.2503.14286, 2025.

[10] Your Efficient RL Framework Secretly Brings You Off-Policy RL Training. Feng Yao's Notion, 2025. https://fengyao.notion.site/off-policy-rl

[11] Sample More to Think Less: Group Filtered Policy Optimization for Concise Reasoning. arXiv preprint arXiv:2508.09726, 2025.

[12] Self-Consistency Improves Chain of Thought Reasoning in Language Models. ICLR 2023.

[13] Geometric-Mean Policy Optimization. arXiv preprint arXiv:2507.20673, 2025.

[14] RAFT: Reward rAnked FineTuning for Generative Foundation Model Alignment. TMLR 2023.