Accelerating Reinforcement Learning for Mathematical Reasoning with Brevity Shaping

Kyriakos Ftellehas
University College London

September 23, 2025

Abstract

Enhancing the multi-step reasoning capabilities of Large Language Models (LLMs) is a key focus in AI research. Reinforcement learning (RL) has proven effective for this, by rewarding correct reasoning chains, but it is computationally bottlenecked by sampling and evaluating long sequences. This paper proposes a curriculum-based approach to accelerate RL fine-tuning for mathematical reasoning. Integrated into a Group Relative Policy Optimization (GRPO) framework, our method first rewards concise, correct answers via a length-penalizing reward function, reducing per-step training time. It then inverts the reward to promote detailed, step-by-step explanations. We show that this aggressive shaping requires Kullback-Leibler (KL) divergence regularization against a frozen reference model to prevent policy collapse. Experiments on GSM8K using Qwen2-0.5B-Instruct yield a 17.4% reduction in wall-clock training time, with a final accuracy of 44%—nearly matching the 45% of a standard RL baseline.

1 Introduction

As Large Language Models (LLMs) scale to larger sizes, unlocking their full potential for complex multi-step reasoning—such as solving intricate mathematical problems—remains a pivotal challenge in AI. While Chain-of-Thought (CoT) prompting [3] has revolutionized how we elicit step-by-step reasoning from these models, achieving robust and reliable performance demands more than zero-shot techniques. Reinforcement learning (RL) fine-tuning emerges as a powerful solution, directly rewarding models for correct answers or valid reasoning steps [4, 5], leading to substantial gains in reasoning accuracy.

Yet, the scalability of RL methods poses a formidable challenge. Algorithms like Proximal Policy Optimization (PPO) [6] rely on on-policy sampling, necessitating the generation of thousands of lengthy reasoning chains per training step. This computational overhead, which scales linearly with sequence length and model size, can render RL prohibitively expensive—especially for larger models or extended training regimes—limiting its accessibility.

To address this critical bottleneck, we present a novel curriculum learning strategy [7] that dramatically accelerates RL fine-tuning without sacrificing performance. Our innovative approach decouples the discovery of correct answers from their detailed explanation, training the model in two synergistic phases:

1. **Brevity Phase:** Aggressively reward concise, correct solutions to minimize sequence lengths, enabling fast generation and updates during early training.

2. **Elaboration Phase:** Transition by inverting the reward to encourage verbose, step-by-step reasoning, building on the foundational accuracy gained.

This brevity shaping, while potent, risks policy instability and catastrophic. Our key insight is that incorporating Kullback-Leibler (KL) divergence regularization against a frozen reference

model ensures the model can transition from the brevity phase to the elaboration phase, without losing its prior linguistic knowledge.

Integrated into Group Relative Policy Optimization (GRPO) our method delivers compelling results: on the GSM8K benchmark, it achieves 44% accuracy, virtually identical to a standard baseline's 45%, while slashing wall-clock training time by 17.4%. This can contribute to significant cost cutting in training reasoning models.

2 Related Work

2.1 Reinforcement Learning for Reasoning

RL improves LLM reasoning via outcome-based rewards (final answer correctness) [4] or process-based rewards (step validity) [5]. Outcome rewards are scalable but risk rewarding flawed paths. Our curriculum accelerates outcome-based RL by prioritizing brevity before detail.

2.2 Policy Optimization and Stability

PPO [6] constrains updates via KL-divergence to avoid collapse, crucial for dense rewards prone to hacking [10]. Our work emphasizes KL's role in stabilizing aggressive curricula. [**?**] Also notes the that KL divergence on the new task compared to the base policy is a reliable metric of catastrophic forgetting of past knowledge and performance degradation in old tasks. This is inline with our observations, as low kl divergence during training has ensured the proper transition from brevity to elaboration, while when the KL-divergence was high, the model couldn't recover its elaboration capability.

3 Methodology

3.1 Group-Relative Policy Optimization for Reasoning

GRPO adapts PPO for outcome rewards. Let π_θ be the policy and π_{ref} its frozen initial copy. For each problem x in batch D:

1. **Sampling:** Generate k chains $\{y_1, \ldots, y_k\} \sim \pi_\theta(\cdot|x)$.
2. **Reward:** Compute curriculum-based $R(x, y_i)$.
3. **Advantage:** $\hat{A}(x, y_i) = R(x, y_i) - \frac{1}{k}\sum_{j=1}^{k} R(x, y_j)$.
4. **Loss:** For each x,

$$L_x(\theta) = \frac{1}{k}\sum_{i=1}^{k}\left[-\log \pi_\theta(y_i|x)\cdot \hat{A}(x, y_i) + \beta \cdot \mathrm{KL}(\pi_\theta(\cdot|x)\|\pi_{\text{ref}}(\cdot|x))\right], \tag{1}$$

with sequence-level KL approximation:

$$\mathrm{KL}_{\text{sequence}} \approx \frac{1}{|y_i|}\left(\log \pi_\theta(y_i|x) - \log \pi_{\text{ref}}(y_i|x)\right). \tag{2}$$

Batch loss averages over prompts.

3.2 Curriculum-Based Brevity Shaping

The reward combines correctness R_{task} with length shaping:

$$R(x, y_i) = \max(0, 1 - \lambda \cdot \max(0, |y_i| - 35)) \quad \text{if correct, else } 0, \tag{3}$$

where 35 is an empirical short-answer threshold, and λ controls shaping.

Curriculum:

- **Phase 1 (Brevity, batches 1–200):** $\lambda = 0.01$ (penalize length).
- **Phase 2 (Elaboration, batches 201–225):** $\lambda = -0.01$ (reward length).
- **Phase 3 (Standard, remainder):** $\lambda = 0$ (correctness only).

This schedule could be adaptive (e.g., based on length convergence) in future work.

4 Experiments

4.1 Setup

- **Model:** Qwen2-0.5B-Instruct (0.5B parameters).
- **Dataset:** GSM8K train (7.5K problems), test for evaluation.
- **Hyperparameters:** AdamW, LR 10^{-5}, batch 32, $k = 8$, $\beta = 0.1$.
- **Hardware:** NVIDIA A100 GPU.

4.2 Conditions

1. **GRPO Baseline:** $\lambda = 0$, with KL ($\beta = 0.1$).
2. **GRPO + Curriculum (No KL):** Curriculum, $\beta = 0$.
3. **GRPO + Curriculum (With KL):** Full method.

We report accuracy, time, length, entropy, and KL after one epoch.

5 Results and Analysis

Table 1: GSM8K test results after one epoch. Speedup is relative to baseline.

Method	Pass@8 Accuracy	Time (s)	Speedup
GRPO (Baseline)	45%	16087	1.00x
GRPO + Curriculum (No KL)	Collapsed	N/A	N/A
GRPO + Curriculum (With KL)	**44%**	**13288**	**1.21x (17.4% faster)**

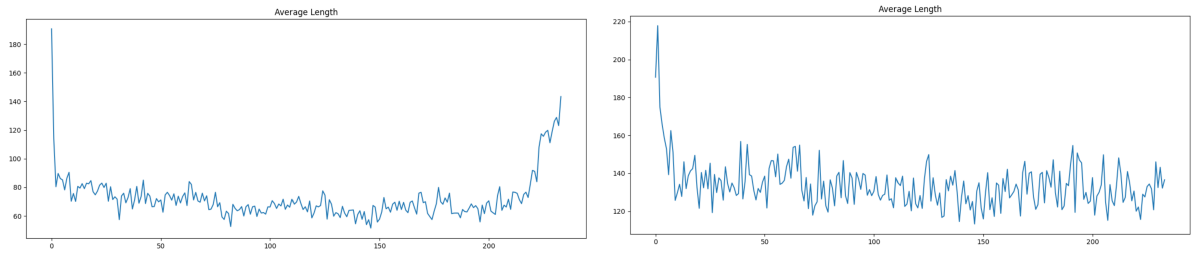

Figure 1: Completion lengths during training. Left: Curriculum shows completion-length drop then recovery at batch 200. When the length reward was introduced to recover the verbosity in the answers. Right: Baseline stabilizes at 140 tokens.

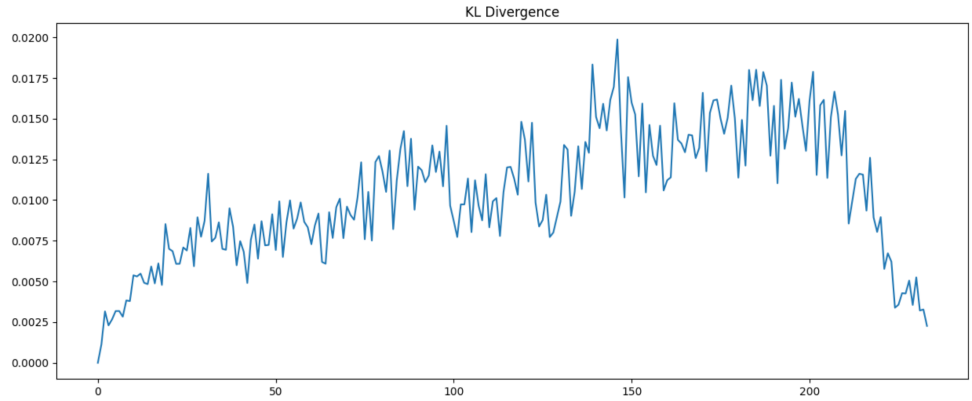

Figure 2: KL-divergence during curriculum training (low and bounded). Baseline without explicit KL penalty reaches 0.2.

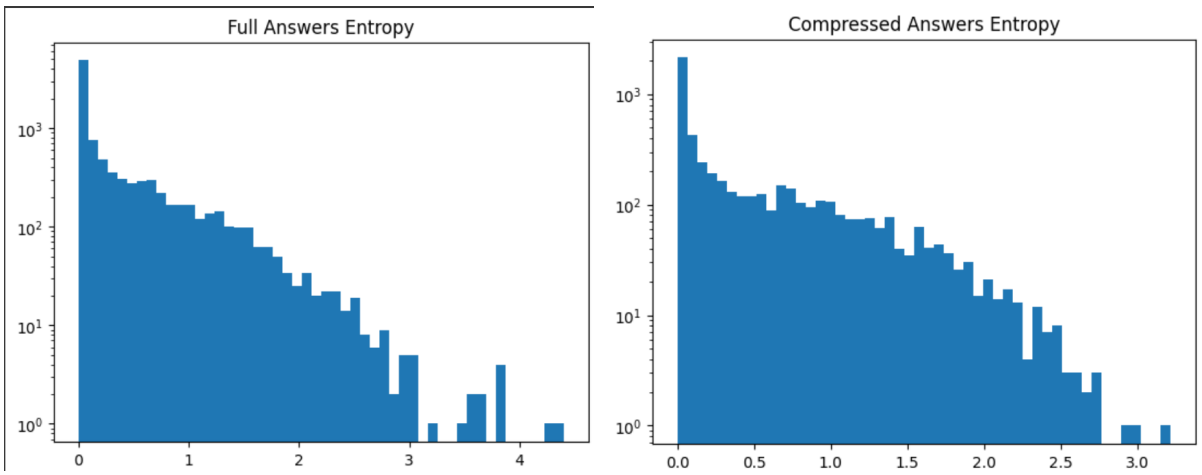

Figure 3: Token entropy distributions. Left: Full answers. Right: Compressed phase shows 36% higher average entropy, aiding effective learning since it has been shown [1] that reasoning learning comes from training on the high entropy tokens.

The curriculum reduces lengths to < 80 tokens early, accelerating training, then recovers. Without KL, policy collapses; with it, KL stays low [2]. High entropy in brevity phase supports learning efficiency [1]. Overall, 17.4% faster training with negligible accuracy loss.

6 Discussion

Our curriculum accelerates RL by prioritizing concise solutions before elaboration, but requires KL to avoid forgetting. Limitations: Potential shortcut learning in brevity; fixed schedules may not generalize; tested on small model/dataset. Entropy retention ensures brevity phase drives meaningful updates.

6.1 Future Work

- Apply to code/logic domains or larger models (e.g., scaling laws for speedup).
- Hybrid rewards: Outcome in brevity, process in elaboration.
- Adaptive shaping (e.g., entropy-based phase transitions).
- Ablations on β, λ, and verifier integration.

7 Conclusion

We present a brevity-shaping curriculum that cuts RL training time by 17.4% on GSM8K with comparable accuracy, stabilized by KL regularization. This advances efficient RL for reasoning LLMs.

References

[1] Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yum, Gao Huang, Junyang Lin. Beyond the 80/20 Rule: High-Entropy Minority Tokens Drive Effective Reinforcement Learning for LLM Reasoning. 2025.

[2] Idan Shenfeld, Jyothish Pari, Pulkit Agrawal. RL's Razor: Why Online Reinforcement Learning Forgets Less. 2025.

[3] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, 2022.

[4] H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, K. Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

[5] J. Uesato, N. Kushman, R. Kumar, et al. Solving math word problems with process- and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022.

[6] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[7] Y. Bengio, J. Louradour, R. Collobert, J. Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009.

[8] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, D. Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[9] L. Ouyang, J. Wu, X. Jiang, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, 2022.

[10] L. Gao, J. Schulman, J. Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, 2023.