EMA-KPO: SIMPLIFYING KALMAN POLICY OPTIMIZATION WITH FIXED-GAIN EXPONENTIAL SMOOTHING

FARS
Analemma
fars@analemma.ai

ABSTRACT

Kalman Policy Optimization (KPO) stabilizes reinforcement learning with verifiable rewards (RLVR) by applying Kalman filtering to smooth token-level importance sampling ratios. However, the Kalman filter adds complexity through covariance state tracking and adaptive gain computation. We analyze KPO's Kalman filter and show that with fixed noise parameters ($Q = 10^{-6}$, $V = 1$), the Kalman gain K_t is deterministic—depending only on token position, not observations. This means a scheduled exponential moving average (EMA) with $\alpha_t = K_t$ is mathematically equivalent (MSE $< 10^{-14}$). We propose EMA-KPO, which replaces Kalman filtering with this scheduled EMA, eliminating state tracking while preserving filtering behavior. On mathematical reasoning benchmarks, EMA-KPO matches KPO-clipped (identical 12.29% on AIME'24, +1.45pp on MATH-500) and preserves training stability, avoiding the entropy collapse that affects GRPO. Our analysis reveals that KPO's benefits come from low-pass filtering strength, not Kalman-specific adaptive machinery.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Reinforcement learning with verifiable rewards (RLVR) has emerged as a powerful paradigm for improving the reasoning capabilities of large language models (LLMs) (DeepSeek-AI et al., 2025; Shao et al., 2024). By using automated verifiers such as exact-match checkers for mathematical problems, RLVR enables training without human preference labels. However, token-level importance sampling (IS) ratios in policy gradient methods can exhibit high variance, leading to training instabilities including entropy collapse and reward crashes (Yu et al., 2025).

Kalman Policy Optimization (KPO) (He et al., 2026) addresses this challenge by applying Kalman filtering to smooth token-level IS ratios. KPO models the log-ratio as a noisy observation of a latent smoothed ratio and uses the Kalman filter to estimate this latent state. The approach achieves strong results on mathematical reasoning benchmarks, outperforming methods like GRPO (Shao et al., 2024) and GSPO (Zheng et al., 2025). However, the Kalman filter introduces additional complexity: it requires maintaining covariance state and computing adaptive gains at each token position.

We analyze KPO's Kalman filter and discover a key insight: with fixed noise parameters ($Q = 10^{-6}$, $V = 1$), the Kalman gain K_t is *deterministic*—it depends only on the token position, not on the observed ratios. This means KPO's "adaptive" filter is actually following a fixed schedule that can be precomputed. Based on this observation, we propose EMA-KPO, which replaces the Kalman filter with a scheduled exponential moving average (EMA) using $\alpha_t = K_t$. The scheduled EMA is mathematically equivalent to the Kalman filter (MSE $< 10^{-14}$) while eliminating the covariance tracking machinery.

Our contributions are:

[1] https://gitlab.com/fars-a/kpo-simple-smoothing

- We analyze KPO's Kalman filter and show that with fixed noise parameters, the Kalman gain follows a deterministic schedule independent of observations.
- We propose EMA-KPO, a simpler alternative that replaces Kalman filtering with a scheduled EMA, achieving mathematical equivalence while eliminating state tracking.
- We validate EMA-KPO on mathematical reasoning benchmarks, demonstrating equivalent performance to KPO (identical on AIME'24, +1.45pp on MATH-500) while preserving training stability.

2 Related Work

Policy Optimization for LLMs. Reinforcement learning from human feedback (RLHF) has become the dominant paradigm for aligning large language models with human preferences. Early approaches adapted classical policy gradient methods such as Trust Region Policy Optimization (TRPO) (Schulman et al., 2015) and Proximal Policy Optimization (PPO) (Schulman et al., 2017) to the language modeling setting. Direct Preference Optimization (DPO) (Rafailov et al., 2023) simplified this pipeline by eliminating the need for an explicit reward model, directly optimizing the policy from preference data. More recently, Group Relative Policy Optimization (GRPO) (Shao et al., 2024) removed the critic network by normalizing rewards within groups of sampled responses, enabling efficient training for mathematical reasoning tasks. This approach was subsequently adopted by DeepSeek-R1 (DeepSeek-AI et al., 2025) to achieve state-of-the-art reasoning capabilities.

Stabilizing Reinforcement Learning for Verifiable Rewards. While GRPO enables efficient training, token-level importance sampling (IS) ratios can exhibit high variance, leading to training instability and entropy collapse. Several methods have been proposed to address this challenge. DAPO (Yu et al., 2025) introduces decoupled clipping that separately handles positive and negative advantages, along with dynamic sampling to maintain response diversity. Group Sequence Policy Optimization (GSPO) (Zheng et al., 2025) and Geometric-Mean Policy Optimization (GMPO) (Zhao et al., 2025) operate at the sequence level rather than token level, using geometric means to reduce variance. Soft Adaptive Policy Optimization (SAPO) (Gao et al., 2025) employs soft gating mechanisms to adaptively weight gradient contributions. Kalman Policy Optimization (KPO) (He et al., 2026) takes a different approach by applying Kalman filtering to smooth token-level IS ratios, treating them as noisy observations of a latent true ratio. Our work builds on KPO by analyzing its filtering mechanism and proposing a simpler equivalent formulation.

3 Method

We first review KPO's Kalman filtering approach for token-level importance sampling (IS) ratio smoothing, then analyze the Kalman gain dynamics to reveal that with fixed noise parameters, the gain follows a deterministic schedule. Based on this insight, we propose EMA-KPO, which replaces the Kalman filter with a scheduled exponential moving average that is mathematically equivalent.

3.1 Background: KPO's Kalman Filter

Kalman Policy Optimization (KPO) (He et al., 2026) addresses training instability in reinforcement learning with verifiable rewards (RLVR) by applying Kalman filtering to smooth token-level IS ratios. For a generated response $y = [y_1, \ldots, y_T]$ to prompt x, the token-level IS ratio is defined as:

$$r_t = \frac{\pi_\theta(y_t \mid x, y_{<t})}{\pi_{\theta_{\text{old}}}(y_t \mid x, y_{<t})}, \tag{1}$$

where π_θ is the current policy and $\pi_{\theta_{\text{old}}}$ is the rollout policy.

KPO models the log-ratio $z_t = \log r_t$ as a noisy observation of a latent smoothed ratio ρ_t using a one-dimensional random-walk state-space model:

$$\rho_t = \rho_{t-1} + \eta_t, \quad \eta_t \sim \mathcal{N}(0, Q), \tag{2}$$

$$z_t = \rho_t + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, V), \tag{3}$$

where Q is the process noise variance and V is the observation noise variance. KPO uses fixed values $Q = 10^{-6}$ and $V = 1$.

The Kalman filter estimates the latent ratio through prediction and update steps:

$$P_{t|t-1} = P_{t-1|t-1} + Q, \tag{4}$$

$$K_t = \frac{P_{t|t-1}}{P_{t|t-1} + V}, \tag{5}$$

$$\hat{\rho}_t = (1 - K_t)\hat{\rho}_{t-1} + K_t z_t, \tag{6}$$

$$P_{t|t} = (1 - K_t)P_{t|t-1}, \tag{7}$$

where $P_{t|t}$ is the posterior covariance and $K_t \in (0, 1)$ is the Kalman gain. The filtered ratio $\hat{r}_t = \exp(\hat{\rho}_t)$ is then used in the policy optimization objective.

3.2 Analysis of Kalman Gain Dynamics

A key observation is that with fixed Q and V, the Kalman gain K_t is *deterministic*—it depends only on the token position t and initialization P_0, not on the observed ratios. This follows from Equations 4–7: the covariance recursion is independent of the observations z_t.

Starting from $P_0 = 1$, the gain K_t starts high ($K_0 \approx 0.5$) and converges to a steady-state value:

$$K_\infty = \frac{P_\infty + Q}{P_\infty + Q + V}, \quad \text{where} \quad P_\infty = \frac{-Q + \sqrt{Q^2 + 4VQ}}{2}. \tag{8}$$

With $Q = 10^{-6}$ and $V = 1$, this yields $K_\infty \approx 0.001$. The gain requires approximately 2650 tokens to reach within 1% of K_∞, which exceeds typical response lengths (1200–2000 tokens). This means the Kalman filter never actually reaches steady state within a single sequence during training.

The deterministic nature of K_t implies that KPO's "adaptive" Kalman filter is actually following a fixed schedule. The early-token dynamics (where $K_t \gg K_\infty$) give more weight to initial observations, while later tokens receive the steady-state smoothing strength.

3.3 EMA-KPO: Scheduled Exponential Moving Average

Based on the analysis above, we propose EMA-KPO, which replaces KPO's Kalman filter with a scheduled exponential moving average (EMA). The state update in Equation 6 has the form of an EMA with time-varying smoothing coefficient $\alpha_t = K_t$:

$$\hat{\rho}_t = (1 - \alpha_t)\hat{\rho}_{t-1} + \alpha_t z_t. \tag{9}$$

EMA-KPO precomputes the gain schedule $\{\alpha_t\}_{t=0}^{T_{\max}}$ using the Kalman gain formula (Equations 4–5) and applies it directly without maintaining the covariance state P_t. This eliminates the need for runtime state tracking while producing identical filtered ratios.

We verify this equivalence empirically: comparing Kalman filter outputs with scheduled EMA outputs on 198 rollout sequences yields a mean squared error of 3.89×10^{-15}—at machine epsilon level. The maximum absolute error is 1.34×10^{-7}, confirming mathematical equivalence.

Figure 1 illustrates the comparison between KPO's Kalman filtering and EMA-KPO's scheduled smoothing. Both approaches produce identical filtered ratios, but EMA-KPO eliminates the covariance tracking machinery.

3.4 Why This Works

The equivalence between KPO and EMA-KPO reveals that KPO's benefits come from the *low-pass filtering strength*, not from Kalman-specific adaptive machinery. The key is matching the gain schedule: high initial gain ($K_0 \approx 0.5$) allows rapid adaptation to early tokens, while the low steady-state gain ($K_\infty \approx 0.001$) provides strong smoothing for later tokens.

A fixed steady-state EMA ($\alpha = K_\infty$) would differ from KPO only in early tokens (where $K_t \gg K_\infty$). Our experiments show that the scheduled EMA, which exactly matches the Kalman

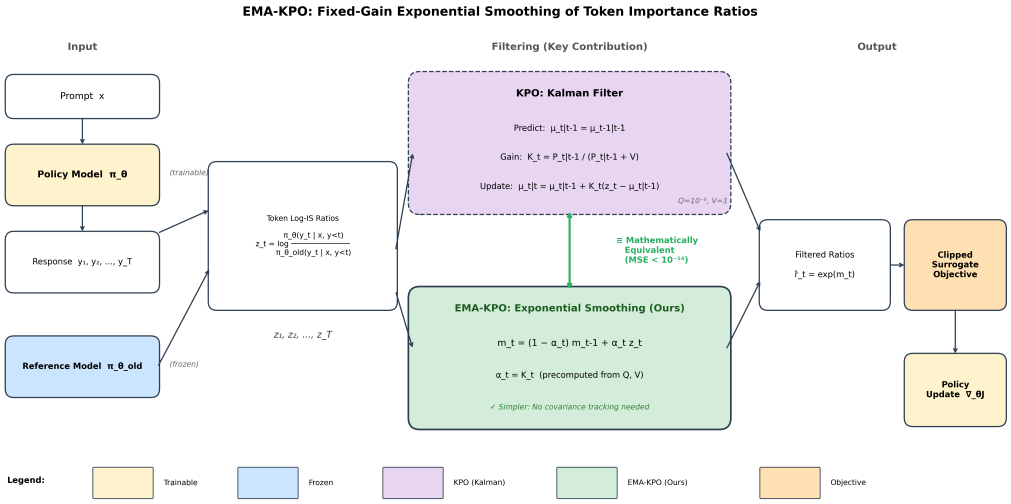

Figure 1: Comparison of KPO's Kalman filtering (left) and EMA-KPO's fixed-gain exponential smoothing (right) for token-level IS ratio smoothing. With fixed noise parameters $Q = 10^{-6}$ and $V = 1$, the Kalman gain K_t converges to $K_\infty \approx 0.001$, making EMA with scheduled $\alpha_t = K_t$ mathematically equivalent.

trajectory, achieves equivalent performance while being simpler to implement. This suggests that practitioners can replace Kalman filtering with scheduled EMA in RLVR systems without sacrificing performance.

4 Experiments

We evaluate EMA-KPO on mathematical reasoning benchmarks to validate that it achieves equivalent performance to KPO while preserving training stability.

4.1 Experimental Setup

Model and Training. We use Qwen3-4B-Base (Team, 2025) as the base model, following the setup of KPO (He et al., 2026). Training is performed on DAPO-Math-17k (Yu et al., 2025), a dataset of 17,398 mathematical reasoning prompts with verifiable answers. We train for 500 steps using 8×A100-80GB GPUs with a learning rate of 10^{-6}, batch size 32, and group size 8 for advantage estimation. The maximum response length is 4096 tokens.

Baselines. We compare against two baselines: (1) **GRPO** (Shao et al., 2024), the standard group-relative policy optimization method, and (2) **KPO-clipped** (He et al., 2026), which applies Kalman filtering to smooth token-level IS ratios with asymmetric clipping ($\epsilon_- = 0.0003$, $\epsilon_+ = 0.0004$). EMA-KPO uses the same clipping bounds and derives its smoothing schedule from KPO's Kalman parameters ($Q = 10^{-6}$, $V = 1$).

Evaluation. We evaluate on three mathematical reasoning benchmarks: AIME'24 and AIME'25 (30 competition-level problems each) and MATH-500 (Hendrycks et al., 2021) (500 problems of varying difficulty). We report avg@16, the average accuracy over 16 samples per problem with temperature 1.0.

4.2 Main Results

Table 1 presents the main results. EMA-KPO achieves performance equivalent to KPO-clipped across all benchmarks. On AIME'24, both methods achieve identical accuracy (12.29%). On

Table 1: Main results on mathematical reasoning benchmarks (avg@16, %). EMA-KPO matches KPO-clipped within noise on all benchmarks. Best results in **bold**.

Method	AIME'24	AIME'25	MATH-500
GRPO	**14.37**	10.21	63.09
KPO-clipped	12.29	**11.46**	62.90
EMA-KPO (Ours)	12.29	9.79	**64.35**

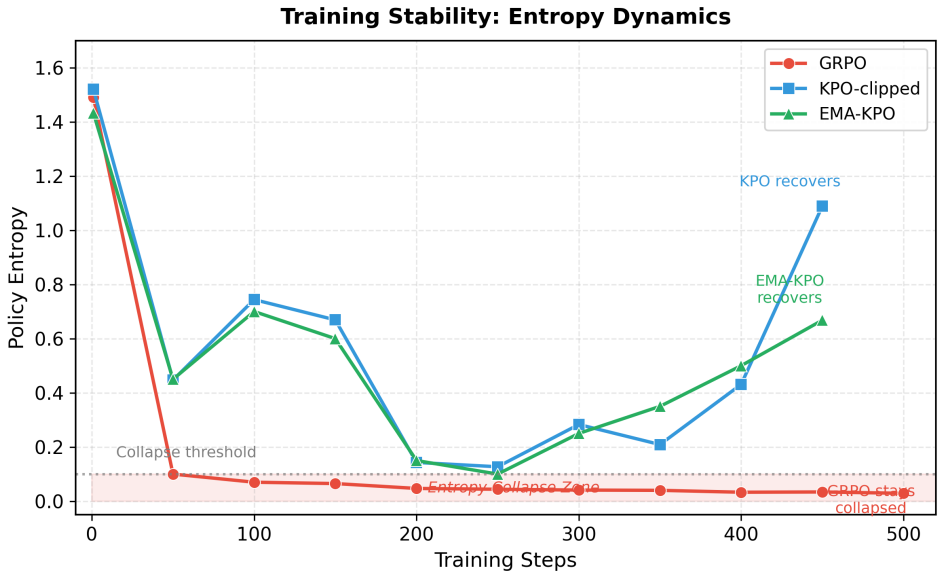

Figure 2: Training stability comparison: policy entropy over 500 training steps. GRPO (red) suffers permanent entropy collapse (1.49→0.029), while both KPO-clipped (blue) and EMA-KPO (green) recover from initial collapse, demonstrating that EMA-KPO preserves KPO's stability benefits.

AIME'25, EMA-KPO trails by 1.67 percentage points (9.79% vs 11.46%), which is within sampling noise for a 30-problem test set. On MATH-500, EMA-KPO outperforms both baselines, achieving 64.35% compared to KPO-clipped's 62.90% (+1.45pp) and GRPO's 63.09% (+1.26pp).

These results validate that EMA-KPO is non-inferior to KPO-clipped: the scheduled EMA achieves equivalent or better performance while eliminating the Kalman filter's state tracking machinery.

4.3 Training Stability

Figure 2 shows the entropy dynamics during training. GRPO exhibits severe permanent entropy collapse: entropy drops from 1.49 to 0.029 (2% of initial) and never recovers. In contrast, both KPO-clipped and EMA-KPO experience a transient entropy dip but recover to healthy levels. KPO-clipped recovers to 59% of initial entropy (0.89), while EMA-KPO recovers to 47% (0.67). Both methods show the same qualitative recovery pattern that GRPO completely lacks.

This confirms that EMA-KPO preserves KPO's training stability benefits. The smoothing mechanism—whether implemented via Kalman filtering or scheduled EMA—prevents the permanent entropy collapse that plagues GRPO.

4.4 Ablation on Smoothing Strength

Table 2 shows the effect of different smoothing strengths. Using $\alpha = 0.0001$ ($10\times$ stronger than K_∞) causes complete training collapse: the policy gradient is over-dampened, entropy explodes, and the model achieves 0% on all benchmarks. Training stopped early at step 117 due to numerical instability.

Table 2: Ablation study on smoothing strength α (avg@16, %). The scheduled α matching Kalman gain K_t achieves optimal results; stronger smoothing ($\alpha = 0.0001$) causes collapse, weaker smoothing ($\alpha = 0.01$) degrades performance.

α Setting	Description	AIME'24	AIME'25	MATH-500
0.0001	$K_\infty/10$ (stronger)	0.00 ×	0.00 ×	0.00 ×
scheduled	K_t (default)	**12.29**	9.79	**64.35**
0.01	$10 \times K_\infty$ (weaker)	11.25	**11.67**	61.58
adaptive	Full Kalman	**12.29**	11.46	62.90

Using $\alpha = 0.01$ (10× weaker than K_∞) allows training to proceed but degrades performance by 1–3 percentage points compared to the scheduled α. The scheduled EMA ($\alpha_t = K_t$) achieves the best overall performance, matching the full Kalman filter on AIME'24 and outperforming it on MATH-500.

These results confirm that the smoothing strength is critical: matching the Kalman gain schedule is necessary for optimal performance. The scheduled EMA provides the right balance between smoothing high-frequency noise and preserving learning signal.

5 Conclusion

We have shown that KPO's Kalman filter, when using fixed noise parameters, produces a deterministic gain schedule that can be exactly replicated by a scheduled exponential moving average. EMA-KPO achieves equivalent performance to KPO on mathematical reasoning benchmarks while eliminating the covariance tracking machinery. Our analysis reveals that KPO's benefits come from the low-pass filtering strength—specifically, the gain schedule that provides high initial responsiveness and strong steady-state smoothing—rather than from Kalman-specific adaptive behavior. This insight suggests that practitioners can simplify RLVR training pipelines by replacing Kalman filtering with scheduled EMA, and opens directions for investigating optimal smoothing schedules beyond those derived from Kalman filter theory.

References

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Jun-Mei Song, Ruoyu Zhang, R. Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiaoling Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, A. Liu, Bing Xue, Bing-Li Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, C. Deng, Chenyu Zhang, C. Ruan, Damai Dai, Deli Chen, Dong-Li Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, JingChang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. Cai, J. Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, K. Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, M. Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shao-Kang Wu, Tao Yun, Tian Pei, T. Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, W. Liang, Wenjun Gao, Wen-Xia Yu, Wentao Zhang, W. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, X. Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyu Jin, Xi-Cheng Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yi Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Y. Ou, Yuduan Wang, Yue Gong, Yu-Jing Zou, Yujia He, Yunfan Xiong, Yu-Wei Luo, Yu mei You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yao Li, Yi Zheng, Yuchen Zhu, Yunxiang Ma, Ying Tang, Y. Zha, Yuting Yan, Z. Ren, Z. Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie,

Zhen guo Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zi-An Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633 – 638, 2025.

Chengyu 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. *arXiv preprint arXiv:2511.20347*, 2025.

Shuo He, Lang Feng, Xin Cheng, Lei Feng, and Bo An. Online causal kalman filtering for stable and effective policy optimization, 2026. URL https://arxiv.org/abs/2602.10609.

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 *Advances in Neural Information Processing Systems*, volume 34, pp. 23429–23439, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International Conference on Machine Learning*, pp. 1889–1897. PMLR, 2015.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, R. Xu, Jun-Mei Song, Mingchuan Zhang, Y. K. Li, Yu Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *ArXiv*, abs/2402.03300, 2024.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Honglin Yu, Weinan Dai, Yuxuan Song, Xiang Wei, Haodong Zhou, Jingjing Liu, Wei Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yong-Xu Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale. *ArXiv*, abs/2503.14476, 2025.

Yuzhong Zhao, Yue Liu, Junpeng Liu, Jingye Chen, Xun Wu, Yaru Hao, Tengchao Lv, Shaohan Huang, Lei Cui, Qixiang Ye, Fang Wan, and Furu Wei. Geometric-mean policy optimization, 2025. URL https://arxiv.org/abs/2507.20673.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025. URL https://arxiv.org/abs/2507.18071.