Enhancing Reasoning for Diffusion LLMs via Distribution Matching Policy Optimization

Yuchen Zhu* Wei Guo* Jaemoo Choi Petr Molodyk Bo Yuan Molei Tao† Yongxin Chen†

Georgia Tech

Diffusion LLMs (dLLMs) & Reinforcement Learning (RL)

■ **Goal**: Improve reasoning of dLLMs with an RL method tailored to masked diffusion structure.

Masked diffusion models learn the conditional distribution: input partially masked sequence $\boldsymbol{x} = (x_1, ..., x_D)$, output $\boldsymbol{\pi}_\theta(\boldsymbol{x}) \in \mathbb{R}^{D \times V}$, where ("UM" = unmasked)

$$\boldsymbol{\pi}_\theta(\boldsymbol{x})_{d,u} \approx \Pr_{\boldsymbol{X} \sim p_{\text{data}}}(X_d = u | \boldsymbol{X}_{\text{UM}} = \boldsymbol{x}_{\text{UM}}), \text{ if } x_d = \mathsf{M}.$$

Sequence probability defined by **random-order autoregressive** generation:

$$p_\theta(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\sigma}}\, p_\theta(\boldsymbol{x}; \boldsymbol{\sigma}), \text{ where } p_\theta(\boldsymbol{x}; \boldsymbol{\sigma}) = \prod_{d=1}^{|\boldsymbol{x}|} \pi_\theta(x_{\sigma_d} | \boldsymbol{x}_{\boldsymbol{\sigma}_{<d}}).$$

ELBO: a surrogate of log-probability, defined as below:

$$-\log p_\theta(\boldsymbol{x}) = -\log \mathbb{E}_{\boldsymbol{\sigma}}\, p_\theta(\boldsymbol{x}; \boldsymbol{\sigma}) \le -\mathbb{E}_{\boldsymbol{\sigma}} \log p_\theta(\boldsymbol{x}; \boldsymbol{\sigma})$$
$$= \mathop{\mathbb{E}}_{m \sim \text{Unif}\{1,...,|\boldsymbol{x}|\}} \left[\frac{|\boldsymbol{x}|}{m} \mathbb{E}_{\mu_m(\widetilde{\boldsymbol{x}}|\boldsymbol{x})} \sum_{d: \widetilde{x}_d = \mathsf{M}} -\log \boldsymbol{\pi}_\theta(\widetilde{\boldsymbol{x}})_{d, x_d} \right] =: \mathcal{L}_\theta(\boldsymbol{x}),$$

where $\mu_m(\cdot|\boldsymbol{x})$ means to sample a uniformly random subset of $\{1, ..., |\boldsymbol{x}|\}$ of size m and mask the corresponding entries in $\boldsymbol{x}$. With i.i.d. samples from p_{data}, the denoising cross-entropy loss for training MDM is simply $\min_\theta \mathbb{E}_{p_{\text{data}}(\boldsymbol{x})} \mathcal{L}_\theta(\boldsymbol{x})$.

For reasoning tasks, we write a sequence as $\boldsymbol{x} = (\boldsymbol{q}, \boldsymbol{o})$ (prompt, response).

■ **Why existing RL methods for LLMs are suboptimal for dLLMs?**

- Sequence likelihood is expensive/inexact for bidirectional generation.
- GRPO-style methods are mostly backward-trajectory based.
- Reward maximization alone tends to be mode-seeking.

Policy Distribution Matching Learning. Given a pretrained dLLM policy $\boldsymbol{\pi}_{\text{ref}}(\boldsymbol{o}|\boldsymbol{q})$ that samples from a distribution $p_{\text{ref}}(\boldsymbol{o}|\boldsymbol{q})$, a reward function $r : (\boldsymbol{q}, \boldsymbol{o}) \mapsto \mathbb{R}$, a set of prompts $\mathcal{D}$, and temperature $\alpha > 0$, learn a dLLM policy $\boldsymbol{\pi}_\theta(\boldsymbol{o}|\boldsymbol{q})$ to produce the desired optimal distribution $p_*(\boldsymbol{o}|\boldsymbol{q})$

$$p_*(\boldsymbol{o}|\boldsymbol{q}) \propto p_{\text{ref}}(\boldsymbol{o}|\boldsymbol{q}) \exp\left(\frac{r(\boldsymbol{q}, \boldsymbol{o})}{\alpha}\right) \text{ via } \min_{\boldsymbol{\pi}_\theta} \mathbb{E}_{\boldsymbol{q} \sim \mathcal{D}}\, \mathcal{F}(p_\theta(\cdot|\boldsymbol{q}), p_*(\cdot|\boldsymbol{q})).$$

Distribution Matching Policy Optimization (DMPO)

► Core loss: Weighted Denoising Cross-Entropy (WDCE) [5]:

$$\min_\theta \mathop{\mathbb{E}}_{\boldsymbol{q} \sim \mathcal{D}} \mathop{\mathbb{E}}_{\boldsymbol{\sigma}} \mathop{\mathbb{E}}_{p_{\text{old}}(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma})} \left\{ w(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma}) \mathop{\mathbb{E}}_{m \sim \text{Unif}\{1,...,|\boldsymbol{o}|\}} \left[\frac{|\boldsymbol{o}|}{m} \mathop{\mathbb{E}}_{\mu_m(\widetilde{\boldsymbol{o}}|\boldsymbol{o})} \sum_{d: \widetilde{o}_d = \mathsf{M}} -\log \boldsymbol{\pi}_\theta(\widetilde{\boldsymbol{o}}|\boldsymbol{q})_{d, o_d} \right] \right\}.$$

Treating i.i.d. samples from the old policy p_{old} as weighted samples from the optimal policy p_*: compute the weights by

$$w(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma}) = \frac{p_*(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma})}{p_{\text{old}}(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma})} \propto \exp\left(\frac{r(\boldsymbol{q}, \boldsymbol{o})}{\alpha} + \log \frac{p_{\text{ref}}(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma})}{p_{\text{old}}(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma})}\right) =: e^{\ell(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma})},$$

► **Small-batch issue & fix.** All-positive weights may promote both good and bad responses when rollouts per prompt are limited. We insert negative gradients by baseline subtraction:

$$w_{\text{real}}(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma}) = w(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma}) - w_{\text{base}}(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma}).$$

The simplest one is $w_{\text{base}}(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma}) = 1$.

► Besides WDCE (forward KL style), we also explore **weighted direct discriminative optimization (WDDO)** [4]:

$$\mathcal{F} = -\mathop{\mathbb{E}}_{\boldsymbol{\sigma}} \mathop{\mathbb{E}}_{p_{\text{old}}(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma})} \left[w(\boldsymbol{o}|\boldsymbol{q};\boldsymbol{\sigma}) \log \sigma\left(\log \frac{p_\theta(\boldsymbol{o}|\boldsymbol{q})}{p_{\text{old}}(\boldsymbol{o}|\boldsymbol{q})}\right) + \log \sigma\left(-\log \frac{p_\theta(\boldsymbol{o}|\boldsymbol{q})}{p_{\text{old}}(\boldsymbol{o}|\boldsymbol{q})}\right)\right],$$

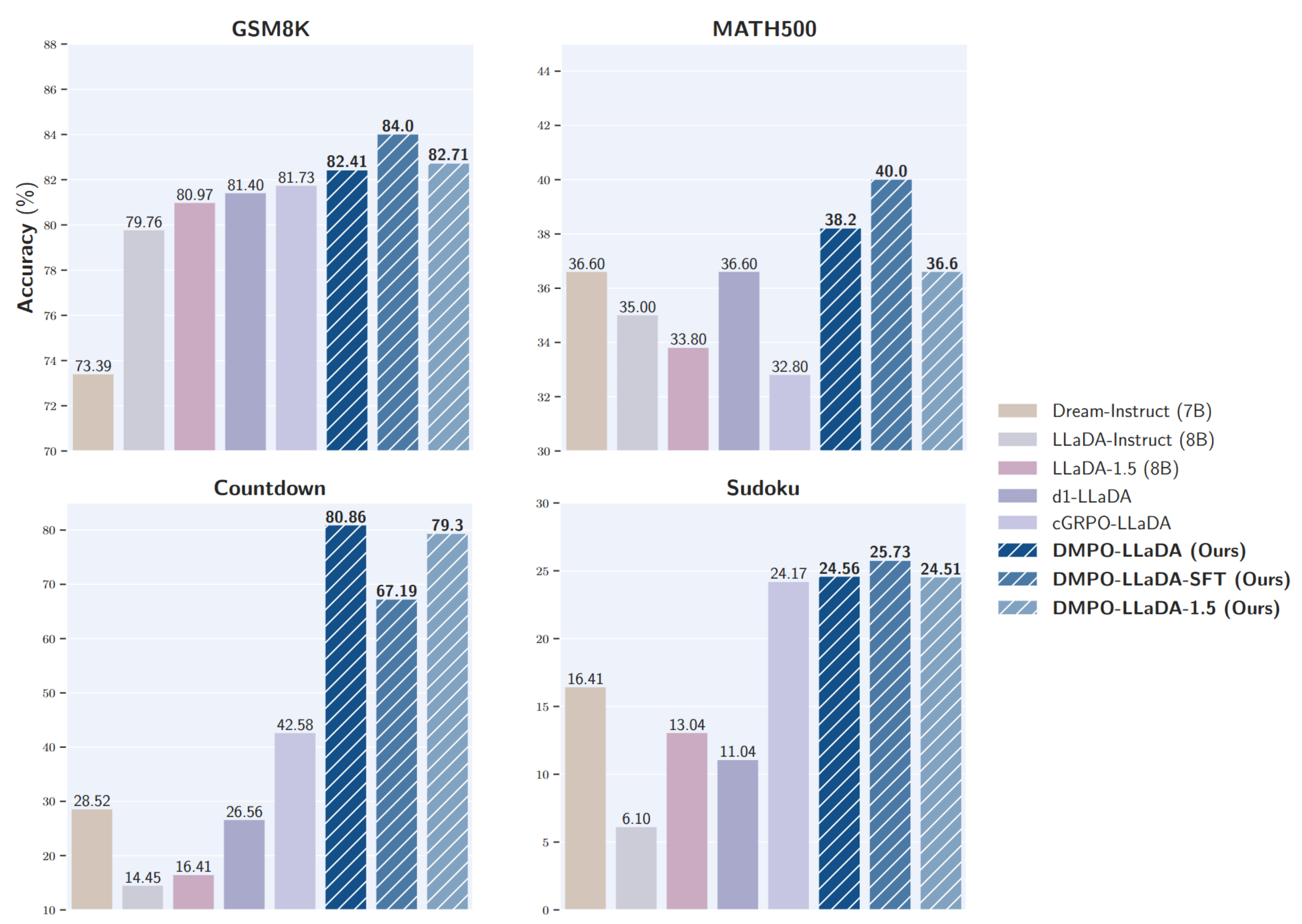
which naturally introduces positive/negative gradient competition and shares the same optimum p_*.

► Practical properties of DMPO.

- Off-policy: supports replay buffer and stale roll-out reuse.
- Forward-only: training leverages cheap noising, not full trajectories.

Results: Strong Reasoning Gains & Efficient Training

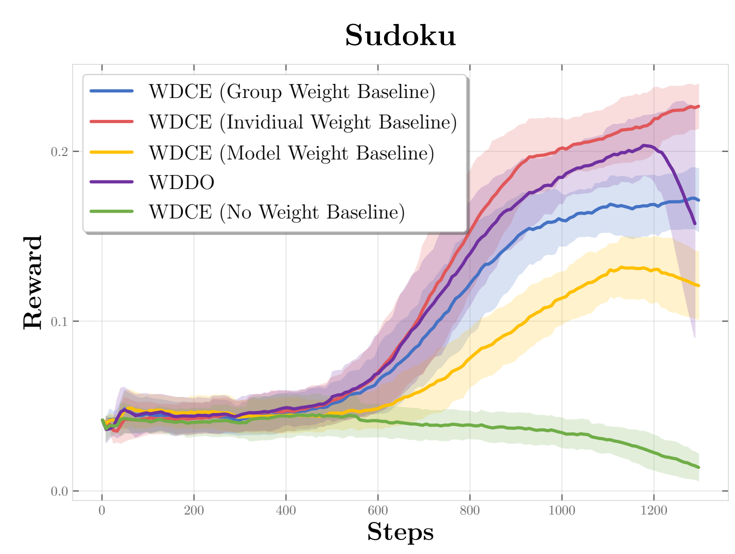
♦ Main benchmark comparison (four tasks, three generation lengths):



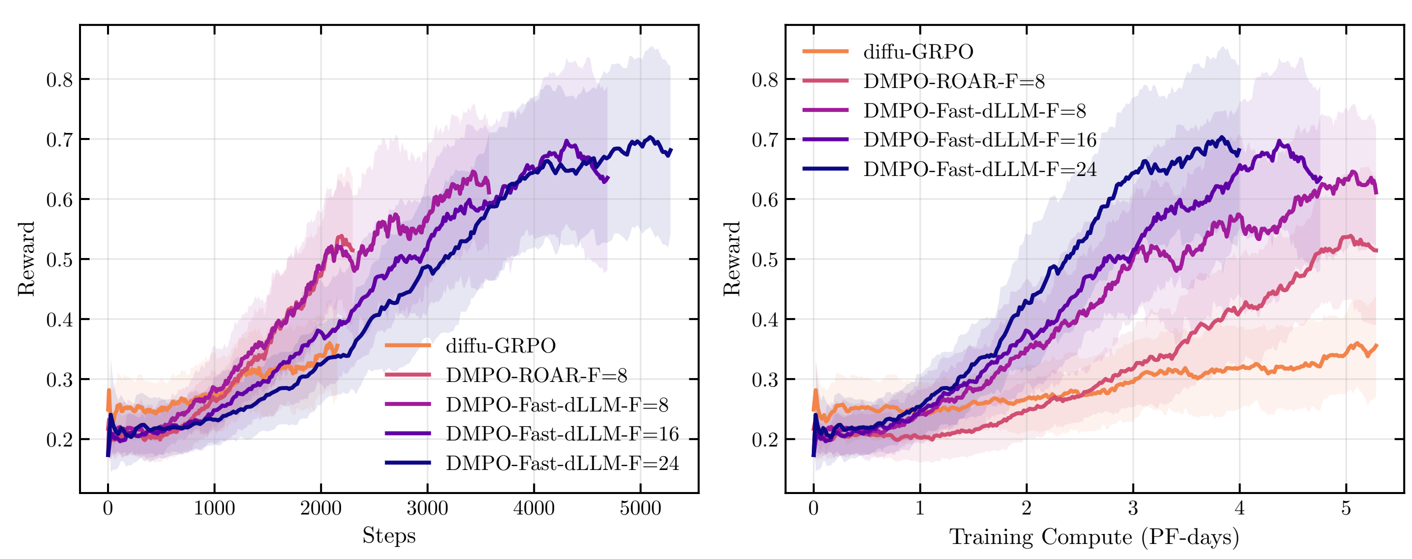
- DMPO consistently outperforms dLLM baselines on GSM8K, MATH500, Countdown, and Sudoku.
- Versus d1 [3] & cGRPO [1], gains are especially large on planning-heavy tasks (Countdown/Sudoku).

♦ Ablation and compute efficiency:

◇ Baseline subtraction is crucial for stable learning at small batch size.



◇ DMPO is sample-efficient (buffer reuse) and compute-efficient (benefits from fast-dLLM [2] sampling).



References

[1] Shansan Gong, Ruixiang Zhang, Huangjie Zheng, Jiatao Gu, Navdeep Jaitly, Lingpeng Kong, and Yizhe Zhang. DiffuCoder: Understanding and improving masked diffusion models for code generation. In *The Fourteenth International Conference on Learning Representations*, 2026.

[2] Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dLLM: Training-free acceleration of diffusion LLM by enabling KV cache and parallel decoding. In *The Fourteenth International Conference on Learning Representations*, 2026.

[3] Siyan Zhao, Devaansh Gupta, Qinqing Zheng, and Aditya Grover. d1: Scaling reasoning in diffusion large language models via reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[4] Kaiwen Zheng, Yongxin Chen, Huayu Chen, Guande He, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Direct discriminative optimization: Your likelihood-based visual generative model is secretly a GAN discriminator. In *Forty-second International Conference on Machine Learning*, 2025.

[5] Yuchen Zhu, Wei Guo, Jaemoo Choi, Guan-Horng Liu, Yongxin Chen, and Molei Tao. MDNS: Masked diffusion neural sampler via stochastic optimal control. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Paper link on ArXiv: `https://arxiv.org/abs/2510.08233`

Contact: `{yzhu738, wei.guo, mtao, yongchen}@gatech.edu`