FLEXIBLE REVEAL-ORDERS FOR REINFORCEMENT LEARNING OF MASKED DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Masked diffusion language models (MDLMs) have emerged as a promising alternative to autoregressive language models, motivating growing interest in reinforcement learning (RL) to improve their reasoning capabilities. RL for MDLMs, however, requires a tractable surrogate for the intractable order-agnostic likelihood, which marginalizes over a combinatorially large number of possible reveal orders. Existing approaches largely follow two strategies: ELBO-based methods optimize a tractable lower bound, while trajectory-based methods optimize the likelihood along a selected denoising trajectory. In this work, we provide a unified view of these approaches through the distribution over reveal orders underlying the log-likelihood gradient. Specifically, we show that the exact log-likelihood gradient is an average of per-order gradients under the posterior distribution over reveal orders, while ELBO-based and trajectory-based methods correspond to two extremes: a uniform distribution and a Dirac distribution concentrated on a single order, respectively. Motivated by this perspective, we introduce a flexible family of confidence-based distributions over reveal orders that interpolates between these extremes, retaining broad coverage over possible orders while concentrating more weight on plausible ones. Our approach can be integrated into existing RL methods without additional model evaluations. Across mathematical reasoning and planning tasks, confidence-based order measures improve both ELBO-based and trajectory-based RL baselines in most settings under held-out checkpoint selection.

1 INTRODUCTION

Over the past few years, Large Language Models (LLMs) have become the dominant paradigm in Artificial Intelligence (AI), matching or even surpassing human performance on a wide range of tasks (Achiam et al., 2023; Team et al., 2023). While most LLMs rely on Autoregressive (AR) modeling of language, Diffusion Large Language Models (dLLMs) have recently emerged as a promising alternative (Sahoo et al., 2024; Nie et al., 2025; Ye et al., 2025), with potential advantages such as faster decoding through parallel generation (Wu et al., 2026) and greater flexibility for controllable generation (Li et al., 2022).

Following the release of competitive open-weight Masked Diffusion Language Models (MDLMs) (Ye et al., 2025; Nie et al., 2025), a growing body of work has sought to improve their capabilities by adapting post-training algorithms originally developed for AR models (Zhu et al., 2025; Zhao et al., 2025). Among these algorithms, Reinforcement Learning (RL) is the standard approach for improving the reasoning capabilities of LLMs, leading to substantial gains on mathematics and coding benchmarks (Shao et al., 2024; Guo et al., 2025). However, adapting RL to dLLMs is non-trivial, since algorithms such as Group Relative Policy Optimization (GRPO) (Shao et al., 2024) rely on the ratio of generation likelihoods. While this likelihood can be computed straightforwardly for AR models, it is intractable for dLLMs because it requires evaluating all possible denoising trajectories, whose number grows combinatorially with sequence length. Therefore, most works replace the intractable likelihood with tractable surrogates. Depending on the surrogate chosen, these methods fall into two broad families. Evidence Lower Bound (ELBO)-based methods (Zhu et al., 2025; Rojas et al., 2026; Ou et al., 2026) optimize a lower bound on the likelihood, whereas

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

trajectory-based methods (Zhan, 2026; Wang et al., 2026c;b) use the likelihood of the denoising path realized during generation.

In this work, we show that these two families admit a unified view through the distribution over reveal orders used to compute their gradients. Specifically, ELBO-based methods average per-order gradients under a uniform distribution, whereas trajectory-based methods evaluate the gradient under a Dirac measure concentrated on a single order. This reveals a complementary trade-off: the former retain broad coverage over reveal orders but weight them equally, while the latter focus on a generation-relevant order but consider only a single trajectory. In contrast, the exact likelihood gradient averages per-order gradients under the posterior over reveal orders, which assigns greater weight to orders under which the response is more likely. Motivated by this perspective, we introduce a flexible family of confidence-based order measures that interpolates between these extremes, aiming to retain broad coverage over reveal orders while concentrating more weight on plausible ones. We show how these measures can be incorporated into both ELBO-based and trajectory-based RL without additional computational cost. We instantiate our approach in a state-of-the-art method from each family and demonstrate improvements across mathematical reasoning and planning tasks.

Concretely, our contributions are as follows:

- We provide a unified perspective on RL for masked diffusion language models through the distribution over reveal orders. We show that the exact likelihood gradient averages per-order gradients under the posterior over reveal orders, while ELBO-based and trajectory-based methods correspond, respectively, to a uniform measure and a Dirac measure on a selected trajectory.
- Motivated by this perspective, we introduce a flexible family of confidence-based order measures that interpolates between the uniform and single-trajectory extremes, aiming to retain broad coverage over reveal orders while concentrating more weight on plausible ones. We show how to integrate it into existing methods without additional model evaluations.
- We validate our approach on mathematical reasoning and planning tasks, demonstrating improvements in both ELBO-based and trajectory-based RL under equal compute.

2 Background

2.1 Diffusion Language Models

Masked Diffusion Language Models (MDLMs) generate text by progressively revealing tokens on a fixed generation canvas. Given an input prompt x, the model starts from a sequence of L mask tokens, $y^{[\mathtt{M}]} = (\,\mathtt{[M]}, \ldots, \mathtt{[M]}\,)$, and generates the response over T denoising steps. At each step t, it predicts a distribution over the vocabulary at every still-masked position. A subset of these positions is then selected, unmasked, and fixed for the remaining steps. Repeating this process leads to the final response $y = (y_1, \ldots, y_L)$.

During pre-training, the model learns to reverse the standard absorbing-mask process, where positions are masked independently according to the same time-dependent schedule (Sahoo et al., 2024; Nie et al., 2025). Equivalently, one may associate each position with an i.i.d. continuous reveal time. Ties then occur with probability zero, inducing a uniform distribution over the $L!$ possible one-at-a-time reveal orders. The resulting training objective is an ELBO on the model log-likelihood, which admits the following any-order form (Ou et al., 2026):

$$\mathcal{L}_{\text{ELBO}}(\theta) = L\mathbb{E}_{\ell\sim\text{Unif}(\{1,\ldots,L\})}\,\mathbb{E}_{m_\ell\sim\text{Unif}(\mathcal{M}_\ell)}\left[\frac{1}{\ell}\sum_{i=1}^{L} m_{\ell,i}\log p_\theta\big(y_i \mid y_{m_\ell}, x\big)\right], \tag{1}$$

where ℓ denotes the number of masked tokens, $\mathcal{M}_\ell$ the set of masks with ℓ masked positions, $m_{\ell,i} \in \{0,1\}$ indicates whether position i is masked, y_{m_ℓ} is the response with mask m_ℓ applied and $p_\theta\big(y_i \mid y_{m_\ell}, x\big)$ denotes the model's conditional of token y_i given the tokens revealed under mask m_ℓ and the prompt x.

2.2 Reinforcement Learning for LLMs

Reinforcement Learning (RL) with verifiable rewards has become a widely used approach to improve the reasoning capabilities of LLMs, driving substantial gains on mathematics and coding

benchmarks (Shao et al., 2024; Guo et al., 2025; Lambert et al., 2024). In this setting, an LLM is treated as a policy $\pi_\theta(y \mid x)$ that maps a prompt x to a response y, and a reward $r(x, y)$ scores responses automatically, for instance by checking the correctness of a final answer. The objective is to maximize the expected reward $\mathbb{E}_{y\sim\pi_\theta(\cdot\,|x)}[r(x, y)]$. Since the reward itself is not differentiable with respect to θ, policy-gradient methods move the derivative onto the policy (Williams, 1992), giving

$$\nabla_\theta \, \mathbb{E}_{y\sim\pi_\theta(\cdot\,|x)}\big[r(x,y)\big] = \mathbb{E}_{y\sim\pi_\theta(\cdot\,|x)}\big[r(x,y)\,\nabla_\theta \log \pi_\theta(y \mid x)\big]. \tag{2}$$

Equation (2) therefore requires evaluating the sequence-level score $\nabla_\theta \log \pi_\theta(y \mid x)$.

2.3 Sequence likelihood for dLLMs

Using RL algorithms on dLLMs requires access to the likelihood $\pi_\theta(y \mid x)$. However a dLLM does not directly provide the likelihood of a generation, but only the conditionals associated with a given denoising trajectory. This trajectory, or unmasking order, is denoted by $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_L)$, where σ_t is the position of the token revealed at step t. Writing $y_{\sigma_{<t}}$ for the sequence in which only the positions in $\{\sigma_1, \sigma_2, \ldots \sigma_{t-1}\}$ are revealed and the remaining ones are masked, the model provides the conditionals $p_\theta(y_i \mid y_{\sigma_{<t}}, x)$ for every masked position i. Chaining these conditionals along σ defines the per-order likelihood of the response y,

$$\pi_\theta^\sigma(y \mid x) = \prod_{t=1}^{L} p_\theta\big(y_{\sigma_t} \mid y_{\sigma_{<t}}, x\big), \tag{3}$$

Marginalizing the latent reveal order yields the sequence likelihood (Hoogeboom et al., 2022),

$$\pi_\theta(y \mid x) = \mathbb{E}_{\sigma\sim\mathrm{Unif}(S_L)}\big[\pi_\theta^\sigma(y \mid x)\big] = \frac{1}{L!}\sum_{\sigma\in S_L} \pi_\theta^\sigma(y \mid x), \tag{4}$$

where S_L is the set of permutations of $\{1, \ldots, L\}$, and $\mathrm{Unif}(S_L)$ is the uniform distribution over it.

Evaluating (4) requires summing over all $L!$ orders, and is therefore intractable for any realistic response length. Therefore, RL methods for dLLMs optimize tractable surrogates of (4). As shown in Figure 1, these methods can be broadly categorized in two main families: ELBO-based and trajectory-based approaches.

ELBO-based methods: A first approach uses the objective already employed during pre-training, (1), as a surrogate for the sequence log-likelihood. Written in terms of reveal orders,

$$\mathcal{L}_{\mathrm{ELBO}}(\theta) = \mathbb{E}_{\sigma\sim\mathrm{Unif}(S_L)}\big[\log \pi_\theta^\sigma(y \mid x)\big] \le \log \mathbb{E}_{\sigma\sim\mathrm{Unif}(S_L)}\big[\pi_\theta^\sigma(y \mid x)\big] = \log \pi_\theta(y \mid x), \tag{5}$$

where the inequality follows from Jensen's inequality. Thus, the ELBO moves the expectation over reveal orders outside the logarithm. Marginalizing over the next reveal and regrouping by the number of masked positions yields the familiar mask-based form in (1), which requires only a single forward pass per Monte Carlo sample. This computational efficiency is a key advantage of ELBO-based methods. Methods such as VRPO (Zhu et al., 2025), GDPO (Rojas et al., 2026), and ESPO (Ou et al., 2026) build on this ELBO surrogate, differing mainly in the construction and variance reduction of their policy-gradient estimators.

Trajectory-based methods. A second approach is to avoid the marginalization over reveal orders altogether. Rather than evaluating the log-sum over the intractable many possible orders, trajectory-based methods select a single meaningful trajectory $\hat{\sigma}$, often taken to be the realized denoising order during generation. This replaces the intractable marginalization with a single trajectory objective

$$\log \pi_\theta^{\hat{\sigma}}(y \mid x) = \sum_{t=1}^{L} \log p_\theta\big(y_{\hat{\sigma}_t} \mid y_{\hat{\sigma}_{<t}}, x\big). \tag{6}$$

Each term conditions on a different partially revealed sequence, requiring $O(L)$ model evaluations. In practice, these methods often reduce the number of denoising steps through multi-token updates or step merging, but still require substantially more model evaluations than the single-forward-pass ELBO objective. Methods such as AGRPO (Zhan, 2026), TraceRL (Wang et al., 2026c), and d2-StepMerge (Wang et al., 2026b) follow this trajectory-based paradigm.

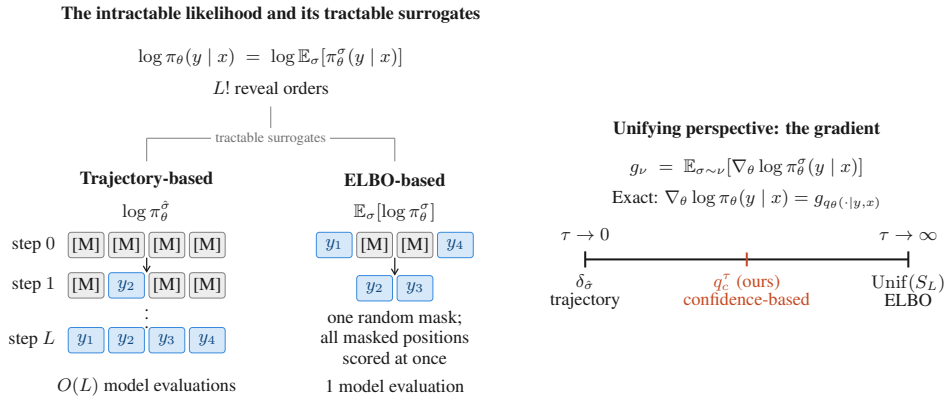


Figure 1: Reveal orders unify ELBO-based and trajectory-based RL for MDLMs. *Left:* log-likelihood surrogates; $\hat{\sigma}$ is the order realized during generation, blue cells are revealed and grey cells masked. *Right:* the gradient as an expectation under a measure ν over reveal orders. $\nu = q_\theta(\cdot \mid y, x)$ is exact but intractable; q_c^τ interpolates between the trajectory Dirac and uniform ELBO measures.

3 A Unified View of Likelihood Gradients for dLLMs

Although ELBO-based and trajectory-based methods are conceptually different surrogates of the log-likelihood, their gradients admit a common form, under which both families and the exact gradient are special cases. We start by the observation that using Fisher Identity, the gradient of the order-agnostic log-likelihood (4) can be rewritten as an average of the per-order gradients, weighted by the likelihood of y under each reveal order. Formally:

Proposition 1 (Posterior gradient). *Denote* $q_\theta(\sigma \mid y, x) := \frac{\pi_\theta^\sigma(y|x)}{\sum_{\sigma' \in S_L} \pi_\theta^{\sigma'}(y|x)}$. *The gradient of the order-agnostic log-likelihood* (4) *satisfies*

$$\nabla_\theta \log \pi_\theta(y \mid x) = \mathbb{E}_{\sigma \sim q_\theta(\,\cdot\,|y,x)} \left[\nabla_\theta \log \pi_\theta^\sigma(y \mid x)\right]. \tag{7}$$

The exact gradient is thus an expectation of per-order gradients, taken under the posterior $q_\theta(\sigma \mid y, x)$, which assigns to each order a weight proportional to the likelihood it gives to y. The gradients of the two families of surrogates take the same form under a different distribution over orders. For the ELBO (5), and for the likelihood of a realized order $\hat{\sigma}$ (6), we have respectively

$$\nabla_\theta \mathcal{L}_{\text{ELBO}} = \mathbb{E}_{\sigma \sim \mathrm{Unif}(S_L)}\left[\nabla_\theta \log \pi_\theta^\sigma(y \mid x)\right], \quad \nabla_\theta \log \pi_\theta^{\hat{\sigma}}(y \mid x) = \mathbb{E}_{\sigma \sim \delta_{\hat{\sigma}}}\left[\nabla_\theta \log \pi_\theta^\sigma(y \mid x)\right] \tag{8}$$

where $\delta_{\hat{\sigma}}$ is the point mass at $\hat{\sigma}$. Hence, from the gradient perspective, the exact likelihood, ELBO, and trajectory-based approaches can all be expressed as

$$g_\nu := \mathbb{E}_{\sigma \sim \nu}\left[\nabla_\theta \log \pi_\theta^\sigma(y \mid x)\right] \tag{9}$$

for a given distribution over orders ν. ELBO-based and trajectory-based families represent two opposite choices: $\mathrm{Unif}(S_L)$ weights all orders equally, whereas a Dirac measure commits to a single order. The exact likelihood gradient instead uses the posterior q_θ, which weights orders according to the likelihood they assign to the response. In Figure 1, we show the unifying perspective in terms of the gradient, in which ELBO-based, trajectory-based and the exact gradient are special cases differentiated by the selected order distribution. Unfortunately, the posterior distribution q_θ is not directly usable, since its normalizer is an intractable sum, also required to compute (4).

3.1 Confidence-based measure for order sampling

Motivated by this unifying perspective, we look for a family of distributions over reveal orders that interpolates between the uniform and the trajectory-based extremes. We propose a one-parameter family of confidence measures, defined as follows. Given a mask $m \in \{0,1\}^L$, a temperature $\tau > 0$ and a masked token i, we define

$$q_{c,i}^\tau(m) := \frac{p_\theta(y_i \mid y_m, x)^{1/\tau}}{\sum_{j=1}^L m_j p_\theta(y_j \mid y_m, x)^{1/\tau}}. \tag{10}$$

An order σ is then constructed by sequentially sampling the next reveal position σ_t according to

$$\sigma_t \sim \mathrm{Categorical}\big(q_c^\tau(m_{\sigma_{<t}})\big), \tag{11}$$

that is, among the currently masked positions, proportionally to the confidence the model assigns to the realized token at that position, tempered by τ. Applied sequentially, (11) induces a distribution over reveal orders, which we also denote by q_c^τ.

The temperature controls where this distribution sits between the two extremes. As $\tau \to \infty$, the exponent $1/\tau$ vanishes and all masked positions receive the same score, so the next position is drawn uniformly among them and q_c^τ converges to $\mathrm{Unif}(S_L)$. As $\tau \to 0$, the measure concentrates on the highest-confidence position at each step, yielding a point mass on the greedy confidence order. This order closely matches the one realized during generation, since the default LLaDA sampler also reveals the highest-confidence position at each step.[1]

3.2 Order-distribution-aware estimators

Our aim is to use the proposed distribution q_c^τ to build new estimators of the log-likelihood gradient. Existing methods, however, differ not only in the distribution over orders they use, but also in how the resulting estimator is implemented. In this section we show how q_c^τ can be combined with either implementation.

Tractable marginalization for ELBO-based methods. Unlike the exact posterior q_θ, whose conditionals require marginalizing over all possible continuations, q_c^τ is defined sequentially: given the current mask, the distribution of the next revealed position is available directly from (10). The next reveal can therefore be marginalized exactly. Decomposing the per-order gradient over reveal steps gives

$$\begin{aligned} g_{q_c^\tau} &= \mathbb{E}_{\sigma\sim q_c^\tau}\left[\sum_{t=1}^{L} \nabla_\theta \log p_\theta\big(y_{\sigma_t} \mid y_{\sigma_{<t}}, x\big)\right] \\ &= \sum_{t=1}^{L} \mathbb{E}_{\sigma_{<t}}\left[\sum_{i=1}^{L} \mathbb{E}_{\sigma_t|\sigma_{<t}}\big[\mathbb{1}\{\sigma_t = i\}\big]\, \nabla_\theta \log p_\theta\big(y_i \mid y_{\sigma_{<t}}, x\big)\right]. \end{aligned} \tag{12}$$

The conditional at step t depends on the prefix $\sigma_{<t}$ only through the set of positions revealed so far, and not through the order in which they were revealed, so the prefix can be summarized by its induced mask. This allows us to rewrite the equation in terms of masks rather than order, mimicking the ELBO objective (1). Let $\ell = L - t + 1$ be the number of positions still masked. The expectation over prefixes then becomes an expectation over masks, and conditional on m_ℓ the probability that position i is revealed next is $q_{c,i}^\tau(m_\ell)$. Hence

$$\begin{aligned} g_{q_c^\tau} &= \sum_{\ell=1}^{L} \mathbb{E}_{m_\ell}\left[\sum_{i=1}^{L} m_{\ell,i}\, q_{c,i}^\tau(m_\ell)\, \nabla_\theta \log p_\theta(y_i \mid y_{m_\ell}, x)\right] \\ &= L\, \mathbb{E}_{\ell\sim\mathrm{Unif}\{1,\dots,L\}}\, \mathbb{E}_{m_\ell}\left[\sum_{i=1}^{L} m_{\ell,i}\, q_{c,i}^\tau(m_\ell)\, \nabla_\theta \log p_\theta(y_i \mid y_{m_\ell}, x)\right]. \end{aligned} \tag{13}$$

Equation (13) mirrors the standard marginalized ELBO, with the uniform weight $1/\ell$ replaced by the confidence weights $q_{c,i}^\tau(m_\ell)$. For a given mask, both the confidence weights and token probabilities are obtained from the same forward pass, so computing the confidence weights requires no additional model evaluation. As $\tau \to \infty$, the weights become uniform and the standard marginalized ELBO is recovered.

Trajectory-based estimation. Alternatively, q_c^τ can be used directly as an order sampler. Starting from the fully masked response, successive reveal positions are sampled from q_c^τ until a complete order $\sigma \sim q_c^\tau$ is obtained, and the gradient is evaluated along this trajectory.

This separates the choice of order measure from the choice of estimator: q_c^τ can be used either through the marginalized construction above or with trajectory-level estimators such as d2-StepMerge (Wang et al., 2026b).

[1] Our measure uses the confidence assigned to the realized token, whereas the sampler uses the confidence of its current proposal.

4 THE CHOICE OF MEASURE

Having made the order measure flexible, the natural question that follows is how to choose it. To guide this choice, we develop a practical heuristic for screening candidate values of τ before RL. For any full-support measure ν, the gap between its associated gradient g_ν and the true gradient g_{q_θ} can be bounded by two factors: the mismatch between ν and q_θ, and the spread of per-order gradients under ν.

Proposition 2 (Gap of a measure)**.** *For every ν with full support,*

$$\left\|g_\nu - \nabla_\theta \log \pi_\theta(y)\right\| \le \sqrt{\chi^2(q_\theta \| \nu)}\ s_\nu, \qquad s_\nu^2 \coloneqq \mathbb{E}_{\sigma\sim\nu}\left\|\nabla_\theta \log \pi_\theta^\sigma(y) - g_\nu\right\|^2, \tag{14}$$

The bound is the product of two terms. The divergence $\chi^2(q_\theta\|\nu) = \mathbb{E}_\nu[(q_\theta/\nu)^2] - 1$ measures how far the candidate measure sits from the posterior over orders; it is large when the two disagree about which orders are likely. The dispersion s_ν is the spread of the per-order gradients around their mean, i.e. how much the gradient actually depends on the order. For the ELBO, $\nu = \mathrm{Unif}(S_L)$, which is well aligned with the true gradient when q_θ is close to uniform, i.e., when the model is relatively insensitive to the reveal order.

Proposition 2 identifies two sources of error. We focus on the measure mismatch $\chi^2(q_\theta\|\nu)$, which depends only on scalar order likelihoods, whereas estimating s_ν requires the dispersion of high-dimensional gradients and is substantially less practical. However, since the posterior q_θ is itself unavailable, this divergence cannot be evaluated directly. Instead, we use the following equivalent large-sample diagnostic.

Proposition 3 (Effective sample size)**.** *Let $\sigma_1, \dots, \sigma_K \sim \nu$ and define the self-normalized importance weights*

$$w_k = \frac{\pi_\theta^{\sigma_k}(y)/\nu(\sigma_k)}{\sum_{j=1}^K \pi_\theta^{\sigma_j}(y)/\nu(\sigma_j)}.$$

With $N_{\text{eff}} = (\sum_k w_k)^2 / \sum_k w_k^2$,

$$N_{\text{eff}}/K \xrightarrow{K\to\infty} \frac{1}{1+\chi^2(q_\theta\|\nu)}. \tag{15}$$

N_{eff} is the standard importance-sampling diagnostic (Kong, 1992; Owen, 2013). Equation (15) states that its asymptotic value is *exactly* the divergence of Proposition 2, written on a bounded scale. Throughout, we use $N_{\text{eff}}/K \in (0, 1]$ as the bounded, readable image of χ^2.

For block diffusion models, the sequence-level mismatch compounds multiplicatively across blocks, making full-sequence ESS difficult to estimate reliably. We therefore compute ESS blockwise, matching the granularity of the sampler; the corresponding factorization is derived in Appendix B.4.

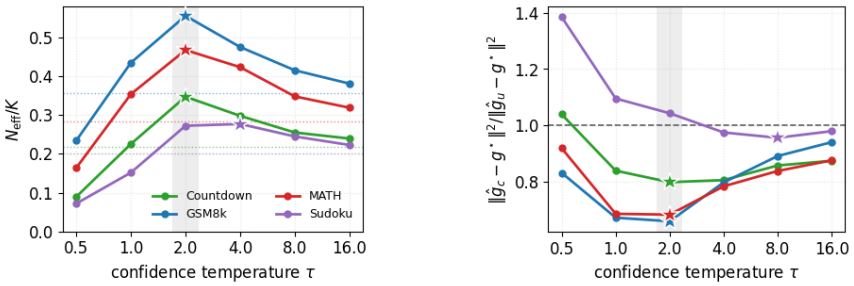
Screening the measure. The ESS controls only the first factor in Proposition 2. Since $\chi^2(q_\theta\|\nu)$ and the dispersion s_ν vary in opposite directions with τ (the divergence grows as ν becomes more concentrated while the dispersion shrinks), minimizing the divergence alone does not minimize the bound. Therefore, we use the average block-level ESS to screen candidate values of τ before RL, avoiding optimization runs for poorly matched measures.

5 EXPERIMENTS

5.1 CHOICE OF TEMPERATURE τ

We first use the ESS diagnostic to select the temperatures considered during RL. We estimate block-level ESS on the base LLaDA model using $K = 32$ sampled orders for 32 generated responses from each task: GSM8K (Cobbe et al., 2021), MATH500 (Lightman et al., 2024), Countdown, and Sudoku. Responses have length $L = 256$ and are evaluated in blocks of length 32. Following the step-merging approximation of Wang et al. (2026b), we merge $s = 4$ reveal steps, reducing the cost from L to $L/s = 64$ model evaluations per sampled order.

Figure 2a reports ESS across temperatures from 0.5 to 16. The profiles peak near $\tau = 2$ on all four tasks, suggesting that an intermediate confidence measure is better matched to the order posterior



(a) Mean ESS_c/K under the confidence measure (higher is better); dotted lines give the uniform measure ESS.

(b) L_2 distance of the confidence-ordered gradient to the reference $g^\star$, relative to the ELBO at matched K and segmentation (lower is better).

Figure 2: Temperature screening. Frozen LLaDA-8B generations; stars mark each curve's optimum. Selecting τ by maximum ESS recovers the error-minimising τ on three of four tasks, and costs 0.02 in error ratio on the fourth.

than either more concentrated or more uniform measures. Since the profiles are relatively flat around their maxima, ESS identifies a favorable region rather than a single optimal temperature.

As discussed in Section 4, ESS is intended as a heuristic for screening measures whose gradients are expected to be closer to the exact likelihood gradient. We therefore test this connection directly. On 64 blocks per task, we compare each candidate gradient at matched $K = 4$ to a high-fidelity reference $g^\star$ constructed from $K_{\text{ref}} = 64$ held-out samples; details are given in Appendix C.4. As shown in Figure 2b, the temperature maximizing ESS also minimizes the measured gradient error on three of the four tasks, and is within 0.02 of the minimum error ratio on the fourth. This supports the intended use of ESS as a cheap screening heuristic: it requires only the forward evaluations used to score orders, whereas constructing $g^\star$ requires K_{ref} backward passes per block. We therefore select $\tau \in \{1, 2, 4\}$ for RL.

5.2 Experimental Setup

ELBO-based methods. We evaluate our measure in the ELBO-based setting using ESPO (Ou et al., 2026) as our baseline. In principle, Equation (13) requires the intermediate mask m_ℓ to be sampled from the distribution induced by q_c^τ. Sampling from this distribution, however, requires sequentially following q_c^τ across reveal steps, and therefore multiple model evaluations, losing the main computational advantage of the ELBO-based objective. Moreover, the distribution over intermediate mask states is often part of the estimator and variance-reduction design in ELBO-based methods; it can differ substantially across methods (Ou et al., 2026; Wang et al., 2026a). We therefore retain the baseline mask-state sampling procedure and apply q_c^τ conditionally on the sampled mask. Concretely, we build directly on the open-source ESPO training and evaluation code and leave its coupled mask-state sampling procedure unchanged (see Ou et al. (2026) for more details). The only modification is to replace the uniform weight $1/\ell$ over masked positions with the confidence weights of (10). Thus, for a sampled mask m, we replace

$$\sum_{i=1}^{L} \frac{1}{\ell} m_{\ell,i} \log p_\theta(y_i \mid y_m, x) \qquad \text{with} \qquad \sum_{i=1}^{L} \mathrm{sg}\big[q_{c,i}^\tau(m)\big]\, m_{\ell,i} \log p_\theta(y_i \mid y_m, x),$$

where sg denotes stop-gradient. Both the confidence weights and token probabilities are obtained from the same forward pass, so the modification incurs no additional model-evaluation cost over ESPO. We retrain both the ESPO baseline and our τ-ESPO extension under a matched compute budget, the hyperparameters and further training details are given in Appendix C.1.1.

Trajectory-based. For the trajectory-based setting, we use d2-StepMerge (Wang et al., 2026b) as the baseline. Evaluating the trajectory likelihood (6) exactly requires L model evaluations, one per reveal step. d2-StepMerge reduces this cost by partitioning the denoising path into N segments of equal length and scoring all tokens of a segment in parallel, conditioned on the state at the start of that segment. Formally, assuming that the generation length L is a multiple of N, let $S = L/N$ be

the segment length and, for an order σ, let $t(i)$ denote the step at which position i is revealed, i.e. $\sigma_{t(i)} = i$. Position i then belongs to segment $n(i) = \lceil t(i)/S \rceil$, and the resulting estimator of the trajectory log-likelihood is

$$\log \hat{\pi}_\theta^{\sigma,N}(y \mid x) = \sum_{i=1}^{L} \log p_\theta\big(y_i \mid y_{\sigma_{<(n(i)-1)S+1}}, x\big), \tag{16}$$

where all tokens of a segment share the same conditioning sequence and are therefore scored with a single model evaluation. This reduces the cost from L to N evaluations, and (6) is recovered exactly when $N = L$. We apply our method on top of d2-StepMerge by replacing the realized denoising path $\hat{\sigma}$ with an order sampled from q_c^τ, while leaving the estimator (16) unchanged. Sampling requires no additional model evaluations, as the conditionals needed for the confidence weights (10) are already computed during policy-likelihood evaluation. Sampled orders respect the block structure used during generation. We retrain both the d2-StepMerge baseline and our τ-d2-StepMerge extension under a matched compute budget, the hyperparameters and further training details are given in Appendix C.1.2.

Checkpoint selection. Prior dLLM RL evaluations commonly report test accuracy at the best checkpoint over training (Wang et al., 2026a; Rojas et al., 2026; Zhan, 2026). Since selecting and evaluating the checkpoint on the same data can lead to optimistic estimates, we instead use held-out checkpoint selection. For each task, we randomly split the original test set into equal validation and evaluation halves, select the checkpoint that maximizes the average validation accuracy across generation lengths, and report its accuracy on the evaluation half. We use the same split across all methods, temperatures, and generation lengths, and we select the corresponding baseline method checkpoint (ESPO or d2) independently using the same procedure.

To reduce sensitivity to any particular split, we repeat this process over 200 random splits and report the mean accuracy difference relative to the corresponding baseline and the fraction of splits for which this difference is positive (P_+). Results under the conventional best-test-checkpoint protocol are reported in Appendix C.3.

5.3 Results

	GSM8K				MATH				Countdown				Sudoku			
L	LLaDA	baseline	Δ	P_+	LLaDA	baseline	Δ	P_+	LLaDA	baseline	Δ	P_+	LLaDA	baseline	Δ	P_+
ELBO-based setting																
128	70.6	76.1	+0.11	0.52	28.0	30.2	-0.31	0.42	32.8	71.3	+8.82	0.99	24.8	91.7	-0.30	0.40
256	79.0	81.6	+0.30	0.59	31.2	34.5	+1.46	0.73	19.1	71.5	+7.58	0.98	16.2	83.8	+5.17	0.96
512	81.0	82.3	+0.30	0.57	34.4	37.5	+1.49	0.67	20.3	69.4	+5.18	0.84	6.1	75.4	+6.60	0.97
Trajectory-based setting																
128	68.7	72.4	+1.8	0.89	27.8	28.3	+1.8	0.76	18.4	51.2	+9.3	0.99	11.5	90.6	-0.5	0.36
256	75.2	80.8	+1.2	0.87	33.0	35.5	+0.7	0.64	19.9	41.3	+11.5	0.99	6.9	62.4	+1.3	0.68
512	78.5	81.4	+2.7	0.97	36.8	38.5	+1.6	0.70	16.8	39.5	+9.4	0.98	5.8	35.7	+15.0	1.00

Table 1: Average improvement on test accuracy (%) across generation lengths L. For each setting we report the base LLaDA model and the corresponding baseline (ESPO for ELBO-based and d2-StepMerge for trajectory-based), together with the improvement Δ of our method and the probability P_+ that it outperforms the baseline across different validation splits used for selection. Green marks $\Delta > 0$, red $\Delta < 0$; shaded cells are the generation length used during training.

Performance at the training length. Table 1 shows that, at the generation length used during training (shaded cells), our method improves over the corresponding baseline in seven of the eight settings. In the ELBO-based setting, the gains are largest on Countdown ($+7.58$, $P_+ = 0.98$) and Sudoku ($+5.17$, $P_+ = 0.96$), while MATH improves by $+1.46$ ($P_+ = 0.73$) and GSM8K by $+0.30$ ($P_+ = 0.59$). In the trajectory-based setting, we observe gains on GSM8K ($+1.2$, $P_+ = 0.87$), MATH ($+1.6$, $P_+ = 0.70$), and Countdown ($+9.3$, $P_+ = 0.99$), while Sudoku shows a small decrease of -0.5 ($P_+ = 0.36$). LLaDA results are obtained under each settings own evaluation code which explain differences across settings.

Generalization across generation lengths. Table 1 also shows that the gains largely transfer to generation lengths not seen during training. In the ELBO-based setting, our method improves over ESPO in six of the eight unseen-length evaluations, with only small decreases on MATH and Sudoku at $L = 128$. In the trajectory-based setting, it improves over d2-StepMerge at every unseen generation length, including substantial gains on Countdown and Sudoku. These results suggest that the benefit of the confidence-based measure is not tied to the sequence length used during training.

Effect of the temperature. We next study the effect of the confidence temperature τ. Recall that the ESS diagnostic preselected $\tau \in \{1, 2, 4\}$ before RL. Figure 3 reports the performance of τ-ESPO and τ-d2 across these temperatures under our held-out checkpoint-selection protocol. The preferred temperature varies across tasks and estimator families: for example, $\tau = 1$ performs best on Countdown and Sudoku for τ-ESPO, while $\tau = 2$ is preferred on MATH and $\tau = 4$ on several trajectory-based settings. Overall, no single temperature dominates across tasks. This is consistent with the intended role of ESS as a screening heuristic rather than a direct predictor of downstream RL performance. As shown in Section 4, higher ESS is associated with gradients that are closer to the exact likelihood gradient, whereas final RL performance also depends on task-specific effects and the dynamics of optimization throughout training. Moreover, our diagnostics are computed on the frozen base model, while the preferred measure may evolve as the policy changes. We leave adaptive or online measure selection to future work. Full numerical results are reported in Appendix C.2.

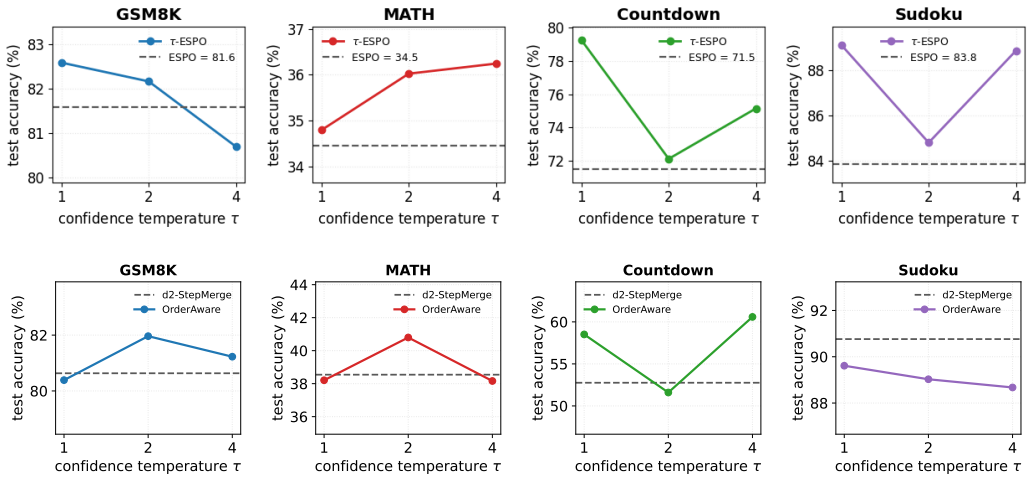


Figure 3: Effect of the confidence temperature τ on test accuracy. The top row shows τ-ESPO in the ELBO-based setting, while the bottom row shows τ-d2Step-Merge in the trajectory-based setting. Dashed lines denote the corresponding baselines. Generation lengths are the same as used during training.

6 Conclusion

In this work, we showed that the exact log-likelihood gradient and the surrogates commonly used for RL in dLLMs admit a common form, differing in the measure over reveal orders under which per-order gradients are averaged. Building on this view, we introduced a one-parameter family of confidence-based measures that interpolates between the uniform measure of ELBO-based methods and the concentrated measure used by trajectory-based methods. Because the choice of measure is separate from the choice of estimator, our approach can be incorporated into both families without additional model evaluations. Across mathematical reasoning and planning tasks, confidence-based measures improve both ELBO-based and trajectory-based baselines in most settings under matched compute.

As a natural extension, our approach allows trajectory-based methods, which typically rely on a single realized reveal order, to sample multiple orders per generation and thereby reduce estimator variance. Beyond post-training, our measure could also be incorporated into the standard ELBO objective during pre-training. We leave both directions to future work.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Moses Charikar, Kevin Chen, and Martin Farach-Colton. Finding frequent items in data streams. In *International Colloquium on Automata, Languages and Programming (ICALP)*, pages 693–703. Springer, 2002.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Emiel Hoogeboom, Alexey A. Gritsenko, Jasmijn Bastings, Ben Poole, Rianne van den Berg, and Tim Salimans. Autoregressive diffusion models. In *International Conference on Learning Representations*, 2022.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Augustine Kong. A note on importance sampling using standardized weights. *University of Chicago, Dept. of Statistics, Tech. Report 348*, 1992.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in neural information processing systems*, 35: 4328–4343, 2022.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pages 39578–39601, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025.

Jingyang Ou, Jiaqi Han, Minkai Xu, Shaoxuan Xu, Jianwen Xie, Stefano Ermon, Yi Wu, and Chongxuan Li. Principled RL for diffusion LLMs emerges from a sequence-level perspective. In *International Conference on Learning Representations*, 2026.

Art B. Owen. *Monte Carlo Theory, Methods and Examples*. 2013.

Kevin Rojas, Jiahe Lin, Kashif Rasul, Anderson Schneider, Yuriy Nevmyvaka, Molei Tao, and Wei Deng. Improving reasoning for diffusion language models via group diffusion policy optimization. In *International Conference on Learning Representations*, volume 2026, pages 1938–1973, 2026.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems*, 2024.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Chenyu Wang, Paria Rashidinejad, Andy (DiJia) Su, Song Jiang, Sid Wang, Siyan Zhao, Cai Zhou, Shannon Shen, Feiyu Chen, Tommi Jaakkola, Yuandong Tian, and Bo Liu. SPG: Sandwiched policy gradient for masked diffusion language models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust, editors, *International Conference on Learning Representations*, volume 2026, pages 24623–24660, 2026a.

Guanghan Wang, Gilad Turok, Yair Schiff, Marianne Arriola, and Volodymyr Kuleshov. d2: Improving reasoning in diffusion language models via trajectory likelihood estimation. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=ldCiNVFt8O.

Yinjie Wang, Ling Yang, Bowen Li, Ye Tian, Ke Shen, and Mengdi Wang. Revolutionizing reinforcement learning framework for diffusion large language models. In *International Conference on Learning Representations*, volume 2026, pages 101476–101501, 2026c.

Kilian Weinberger, Anirban Dasgupta, Josh Attenberg, John Langford, and Alex Smola. Feature hashing for large scale multitask learning. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, pages 1113–1120, 2009.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

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. URL https://openreview.net/forum?id=3Z3Is6hnOT.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models. *arXiv preprint arXiv:2508.15487*, 2025.

Anthony Zhan. Simple policy gradients for reasoning with diffusion language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=89aKcIeUyP.

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. URL https://openreview.net/forum?id=7ZVRlBFuEv.

Fengqi Zhu, Rongzhen Wang, Shen Nie, Xiaolu Zhang, Chunwei Wu, Jun Hu, Jun Zhou, Jianfei Chen, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. LLaDA 1.5: Variance-reduced preference optimization for large language diffusion models. *arXiv preprint arXiv:2505.19223*, 2025.

A AI USE STATEMENT

In this work, we use generative AI tools to provide feedback on experiments, implement methods, support qualitative data analysis, and assist in the writing of proofs. We have not used AI tools to interpret results, generate synthetic datasets, assist with translation, clean and reformat datasets, and perform other required-disclosure tasks that are not applicable to this work. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

B PROOFS

Throughout this section, x and y are fixed, and we write

$$v_\sigma \coloneqq \nabla_\theta \log \pi_\theta^\sigma(y \mid x)$$

for the gradient associated with reveal order σ.

B.1 POSTERIOR GRADIENT

Proposition 1 (Posterior gradient). Denote $q_\theta(\sigma \mid y, x) \coloneqq \frac{\pi_\theta^\sigma(y|x)}{\sum_{\sigma' \in S_L} \pi_\theta^{\sigma'}(y|x)}$. The gradient of the order-agnostic log-likelihood (4) satisfies

$$\nabla_\theta \log \pi_\theta(y \mid x) = \mathbb{E}_{\sigma \sim q_\theta(\,\cdot\,|y,x)} \left[\nabla_\theta \log \pi_\theta^\sigma(y \mid x) \right].$$

Proof. The result is a direct application of Fisher's identity to the finite mixture over reveal orders. The sequence likelihood is the uniform mixture of the per-order likelihoods,

$$\pi_\theta(y \mid x) = \frac{1}{L!} \sum_{\sigma \in S_L} \pi_\theta^\sigma(y \mid x).$$

Differentiating its logarithm gives

$$\begin{aligned} \nabla_\theta \log \pi_\theta(y \mid x) &= \frac{\frac{1}{L!} \sum_\sigma \nabla_\theta \pi_\theta^\sigma(y \mid x)}{\pi_\theta(y \mid x)} \\ &= \sum_\sigma \frac{\pi_\theta^\sigma(y \mid x)}{L!\, \pi_\theta(y \mid x)} \nabla_\theta \log \pi_\theta^\sigma(y \mid x). \end{aligned}$$

By (4),

$$L!\, \pi_\theta(y \mid x) = \sum_{\sigma'} \pi_\theta^{\sigma'}(y \mid x),$$

so the coefficient multiplying each per-order gradient is precisely $q_\theta(\sigma \mid y, x)$. Hence

$$\nabla_\theta \log \pi_\theta(y \mid x) = \mathbb{E}_{\sigma \sim q_\theta(\,\cdot\,|y,x)}[v_\sigma],$$

which is (7). □

B.2 GAP OF A MEASURE

Proposition 2 (Gap of a measure). For every ν with full support,

$$\left\| g_\nu - \nabla_\theta \log \pi_\theta(y) \right\| \le \sqrt{\chi^2(q_\theta \| \nu)}\ s_\nu, \qquad s_\nu^2 \coloneqq \mathbb{E}_{\sigma \sim \nu} \left\| \nabla_\theta \log \pi_\theta^\sigma(y) - g_\nu \right\|^2.$$

Proof. Proof. Let $r(\sigma) \coloneqq \frac{q_\theta(\sigma|y,x)}{\nu(\sigma)}$. Because ν has full support, r is well defined. Moreover, r is an importance ratio from ν to the posterior, so

$$\mathbb{E}_\nu[r] = 1 \qquad \text{and} \qquad \mathbb{E}_\nu[r\, v_\sigma] = \mathbb{E}_{q_\theta}[v_\sigma] = \nabla_\theta \log \pi_\theta(y \mid x).$$

Hence the difference between the exact gradient and $g_\nu = \mathbb{E}_\nu[v_\sigma]$ can be written as

$$\nabla_\theta \log \pi_\theta(y \mid x) - g_\nu = \mathbb{E}_\nu[r\, v_\sigma] - \mathbb{E}_\nu[v_\sigma] = \mathbb{E}_\nu[(r-1) v_\sigma].$$

Since $\mathbb{E}_\nu[r - 1] = 0$, subtracting the constant vector g_ν inside the expectation does not change its value. Therefore,

$$\nabla_\theta \log \pi_\theta(y \mid x) - g_\nu = \mathbb{E}_\nu[(r-1)(v_\sigma - g_\nu)].$$

This expression makes the two sources of discrepancy explicit: how much the measure ν differs from the posterior, through $r - 1$, and how much the per-order gradient varies under ν, through $v_\sigma - g_\nu$. Applying Cauchy–Schwarz gives

$$\left\| \nabla_\theta \log \pi_\theta(y \mid x) - g_\nu \right\| \le \sqrt{\mathbb{E}_\nu[(r-1)^2]}\, \sqrt{\mathbb{E}_\nu[\|v_\sigma - g_\nu\|^2]} = \sqrt{\chi^2(q_\theta \| \nu)}\, s_\nu.$$

The first factor is exactly the Pearson χ^2 divergence between q_θ and ν, while the second is the gradient dispersion s_ν defined in (14). This proves the result. □

B.3 EFFECTIVE SAMPLE SIZE

Proposition 3 (Effective sample size). Let $\sigma_1, \dots, \sigma_K \sim \nu$ and define the self-normalized importance weights

$$w_k = \frac{\pi_\theta^{\sigma_k}(y)/\nu(\sigma_k)}{\sum_{j=1}^K \pi_\theta^{\sigma_j}(y)/\nu(\sigma_j)}.$$

With $N_{\text{eff}} = (\sum_k w_k)^2 / \sum_k w_k^2$,

$$N_{\text{eff}}/K \xrightarrow{K\to\infty} \frac{1}{1+\chi^2(q_\theta\|\nu)}.$$

Proof. It is convenient to work first with the unnormalized importance ratio

$$r(\sigma) \coloneqq \frac{q_\theta(\sigma \mid y, x)}{\nu(\sigma)}.$$

The unknown normalizing constant in q_θ is common to all samples and therefore disappears after self-normalization, so the weights obtained from $r(\sigma_k)$ are identical to those obtained from $\pi_\theta^{\sigma_k}(y)/\nu(\sigma_k)$.

Writing $r_k = r(\sigma_k)$, the effective sample size can therefore be expressed as

$$\frac{N_{\text{eff}}}{K} = \frac{\left(\frac{1}{K}\sum_{k=1}^K r_k\right)^2}{\frac{1}{K}\sum_{k=1}^K r_k^2}.$$

Because σ_k are i.i.d. under ν, the law of large numbers gives

$$\frac{1}{K}\sum_{k=1}^K r_k \longrightarrow \mathbb{E}_\nu[r] = 1,$$

and

$$\frac{1}{K}\sum_{k=1}^K r_k^2 \longrightarrow \mathbb{E}_\nu[r^2].$$

Finally,

$$\mathbb{E}_\nu[r^2] = \sum_\sigma \nu(\sigma)\left(\frac{q_\theta(\sigma \mid y, x)}{\nu(\sigma)}\right)^2 = 1 + \chi^2(q_\theta\|\nu).$$

Combining the two limits yields

$$\frac{N_{\text{eff}}}{K} \longrightarrow \frac{1}{1+\chi^2(q_\theta\|\nu)},$$

as claimed. □

B.4 BLOCK DIFFUSION MODELS.

Deployed samplers typically decode semi-autoregressively: the response is partitioned into B blocks that are generated left to right, while the reveal order remains free within each block. Writing $\sigma = (\sigma_1, \dots, \sigma_B)$ as a collection of within-block reveal orders, the order posterior factorizes as $q_\theta(\sigma \mid y, x) = \prod_{b=1}^B q_{\theta,b}(\sigma_b \mid y_{<b}, x)$. For a measure ν with the corresponding semi-autoregressive factorization, the χ^2 divergence therefore satisfies

$$1 + \chi^2(q_\theta\|\nu) = \prod_{b=1}^B \left(1 + \chi^2(q_{\theta,b}\|\nu_b)\right),$$

Hence, the sequence-level ESS decays multiplicatively across blocks. Even moderate per-block mismatch can therefore make the full-sequence ESS prohibitively small to estimate. We instead evaluate ESS within each block, matching the granularity at which the sampler makes ordering decisions.

B.5 PRACTICAL τ-ESPO AS A REVEAL-EVENT MEASURE

Equation (13) characterizes the gradient obtained when intermediate mask states are sampled from the distribution induced by the confidence-based order measure q_c^τ. In practical τ-ESPO, however, we retain the mask-state sampling procedure of ESPO and apply q_c^τ only to weight the masked positions conditionally on the sampled mask. This preserves the single-forward-pass structure of ESPO, but the resulting update is not, in general, equal to $g_{q_c^\tau}$. We show that it nevertheless admits an exact interpretation as an expectation over local reveal events.

For a mask m and masked position i, define

$$v_\theta(m, i) \coloneqq \nabla_\theta \log p_\theta(y_i \mid y_m, x),$$

and let a reveal event be $z = (\ell, m, i)$, where ℓ positions remain masked, $m \in \mathcal{M}_\ell$, and $m_i = 1$. The exact posterior over reveal orders induces the event distribution

$$d_\star(\ell, m, i) \coloneqq \frac{1}{L} \Pr_{\sigma \sim q_\theta(\,\cdot\,|y,x)} (m_\ell(\sigma) = m,\ \sigma(L - \ell + 1) = i)\,.$$

By construction, sampling $z \sim d_\star$ is equivalent to sampling an order $\sigma \sim q_\theta$ and then selecting one of its L reveal steps uniformly. Therefore,

$$L\,\mathbb{E}_{z\sim d_\star}[v_\theta(z)] = \mathbb{E}_{\sigma\sim q_\theta}\left[\sum_{t=1}^{L} \nabla_\theta \log p_\theta(y_{\sigma(t)} \mid y_{\sigma(<t)}, x)\right] = \nabla_\theta \log \pi_\theta(y \mid x),$$

where the last equality follows from Proposition 1.

Now let $\mu_\ell(m)$ denote the marginal distribution over masks used by the ESPO estimator when ℓ positions are masked. Practical τ-ESPO has expected update

$$\widetilde{g}_\tau = L\,\mathbb{E}_{\ell\sim \mathrm{Unif}\{1,\dots,L\}} \mathbb{E}_{m\sim\mu_\ell}\left[\sum_{i=1}^{L} m_i q_{c,i}^\tau(m) v_\theta(m, i)\right].$$

Defining

$$d_\tau^{\mathrm{ESPO}}(\ell, m, i) \coloneqq \frac{1}{L}\mu_\ell(m) m_i q_{c,i}^\tau(m),$$

gives the exact representation

$$\widetilde{g}_\tau = L\,\mathbb{E}_{z\sim d_\tau^{\mathrm{ESPO}}}[v_\theta(z)].$$

Thus, practical τ-ESPO does not generally correspond to the full reveal-order measure q_c^τ, but it does correspond exactly to a well-defined distribution over reveal events.

The argument of Proposition 2 applies directly on this event space. For any event distribution d whose support contains that of $d_\star$,

$$\|L\mathbb{E}_d[v_\theta(z)] - \nabla_\theta \log \pi_\theta(y \mid x)\| \le L\sqrt{\chi^2(d_\star \| d)}\, s_d,$$

where

$$s_d^2 \coloneqq \mathbb{E}_d \left\|v_\theta(z) - \mathbb{E}_d[v_\theta(z)]\right\|^2.$$

In particular, the bound applies to $d = d_\tau^{\mathrm{ESPO}}$.

Finally, writing

$$d_\star(\ell, m, i) = \frac{1}{L}\rho_{\star,\ell}(m) q_\star(i \mid m), \qquad d_\tau^{\mathrm{ESPO}}(\ell, m, i) = \frac{1}{L}\mu_\ell(m) q_c^\tau(i \mid m),$$

makes the approximation explicit. Practical τ-ESPO differs from the exact posterior event measure through both the mask-state distribution, $\rho_{\star,\ell}$ versus μ_ℓ, and the conditional reveal distribution, $q_\star(\,\cdot \mid m)$ versus $q_c^\tau(\,\cdot \mid m)$. Retaining μ_ℓ preserves the computational efficiency of ESPO, while confidence weighting modifies only the conditional distribution over which masked positions contribute to the update.

For completeness, the same argument also quantifies the gap between practical τ-ESPO and the ideal gradient $g_{q_c^\tau}$. Let $\rho_{\tau,\ell}(m)$ denote the mask-state distribution induced by sequentially sampling from q_c^τ, and define

$$d_\tau^\star(\ell, m, i) \coloneqq \frac{1}{L}\rho_{\tau,\ell}(m) q_c^\tau(i \mid m).$$

Then $g_{q_c^\tau} = L\,\mathbb{E}_{z\sim d_\tau^\star}[v_\theta(z)]$, while practical τ-ESPO uses d_τ^{ESPO}. Hence,

$$\left\|\widetilde{g}_\tau - g_{q_c^\tau}\right\| \le L\sqrt{\chi^2(d_\tau^\star \| d_\tau^{\mathrm{ESPO}})}\, s_{d_\tau^{\mathrm{ESPO}}}.$$

Because the two measures share the same conditional reveal distribution $q_c^\tau(i \mid m)$, this gap isolates the effect of retaining the ESPO mask-state distribution instead of sampling masks from the occupancy induced by q_c^τ.

C Experiments

C.1 Further implementation details

C.1.1 τ-ESPO

We build our method on top of the ESPO implementation and follow their training recipe unless stated otherwise. We apply LoRA (Hu et al., 2022) with rank $r = 128$ and scaling factor $\alpha = 64$, and optimize with AdamW (Loshchilov and Hutter, 2019) using $\beta_1 = 0.9$, $\beta_2 = 0.99$, a constant learning rate of 3×10^{-6}, weight decay 0.01, and gradient clipping at 0.2, with a KL coefficient of $\beta = 10^{-2}$. Each update uses $\mu = 8$ policy updates and $M = 2$ Monte Carlo samples of the ELBO. The per-dataset generation and optimization settings are reported in Table 2.

	Generation		Optimization		
Dataset	L	Steps	G	μ	Batch size
Countdown	256	128	6	8	96
Sudoku	256	128	6	8	96
GSM8K	256	128	6	8	96
MATH500	256	128	16	8	256

Table 2: Experimental settings per dataset for τ-ESPO. L is the generation length and Steps the number of denoising steps, used both during training and at evaluation. For the optimization, G is the number of rollouts per prompt and μ the number of policy updates per generation.

C.1.2 τ-d2

We build our method on top of the implementation of Wang et al. (2026b), and follow their setup unless stated otherwise. We apply LoRA (Hu et al., 2022) with rank $r = 128$, scaling factor $\alpha = 64$ and dropout 0.05, and optimize with AdamW (Loshchilov and Hutter, 2019) using $\beta_1 = 0.9$, $\beta_2 = 0.99$, a constant learning rate of 3×10^{-6}, weight decay 0.1, and gradient clipping at 0.2. The GRPO clipping range is $\epsilon = 0.5$ and the KL coefficient is $\beta = 0.04$, with the reference model synchronized every 64 steps. Generation uses semi-autoregressive decoding with a block length of 32 and low-confidence remasking, and prompts are randomly masked with probability 0.15 when computing the policy likelihood.

	Generation			Optimization		
Dataset	L	N	Steps	G	μ	Batch size
Countdown	128	16	64	6	6	24
Sudoku	128	16	64	6	6	24
GSM8K	256	16	128	6	12	12
MATH	512	32	256	6	12	24

Table 3: Experimental settings per dataset for τ-d2. L is the generation length, N the number of segments used by the estimator, and Steps the number of denoising steps, set to $L/2$. For the optimization, G is the number of rollouts per prompt, μ the number of policy updates per generation, and the batch size is reported as the number of prompts per gradient update times G.

The per-dataset generation and optimization settings are reported in Table 3. The number of segments N differs from the original paper for GSM8K and MATH: because sampled orders respect the block structure of the generation, choosing segments of the same size as a block would make our estimator coincide with the baseline, so we use a finer segmentation instead. During generation the number of denoising steps is set to half the generation length, which corresponds to revealing two tokens per step. Rather than reporting the numbers of Wang et al. (2026b), we rerun their implementation ourselves, so that both methods are compared under the same training regime and compute budget.

C.2 Sensitivity to the confidence temperature

Tables 4 and 5 report the full results for the fixed temperatures $\tau \in \{1, 2, 4\}$ underlying the temperature-sensitivity analysis in the main text. Checkpoints are selected using the held-out procedure described above. We report performance at the training generation length, together with the mean difference Δ relative to the corresponding baseline and the fraction of splits for which this difference is positive (P_+). The effect of τ is task dependent: several temperatures can improve over the baseline, while the preferred temperature varies across tasks and estimator families.

	GSM8K			MATH			Countdown			Sudoku		
Method	**ESPO**	**Δ**	P_+	**ESPO**	**Δ**	P_+	**ESPO**	**Δ**	P_+	**ESPO**	**Δ**	P_+
$\tau = 1$	81.58	+1.00	0.74	34.45	+0.34	0.51	71.48	**+7.74**	0.99	83.84	**+5.26**	0.98
$\tau = 2$	81.58	+0.58	0.69	34.45	+1.57	0.76	71.48	+0.61	0.51	83.84	+0.95	0.59
$\tau = 4$	81.58	-0.88	0.20	34.45	+1.79	0.81	71.48	+3.64	0.80	83.84	+5.01	0.89

Table 4: Sensitivity across fixed τ values at training generation length (256) for ELBO-based method. For each τ, the best checkpoint is selected using the validation set. ESPO reports the average baseline test accuracy. Δ denotes the average test-accuracy difference relative to ESPO, and $P_+ = P(\Delta > 0)$ is the probability that the selected checkpoint outperforms ESPO.

	GSM8K			MATH			Countdown			Sudoku		
Method	**d2**	**Δ**	P_+	**d2**	**Δ**	P_+	**d2**	**Δ**	P_+	**d2**	**Δ**	P_+
$\tau = 1$	80.8	0.00	0.48	38.5	+0.8	0.65	51.2	+7.3	0.97	90.6	-0.5	0.36
$\tau = 2$	80.8	+1.2	0.87	38.5	+1.6	0.70	51.2	+0.6	0.56	90.6	-2.5	0.07
$\tau = 4$	80.8	+0.7	0.75	38.5	+0.6	0.61	51.2	+9.3	0.99	90.6	-3.6	0.02

Table 5: Sensitivity across fixed τ values at training generation length for the trajectory-based method. For each τ, the best checkpoint is selected using the validation set. d2 reports the average baseline test accuracy. Δ denotes the average test-accuracy difference relative to d2, and $P_+ = P(\Delta > 0)$ is the probability that the selected checkpoint outperforms d2.

C.3 Results with best-test-checkpoint selection

For comparison with the conventional best-test-checkpoint protocol, we repeat the evaluation without a held-out validation split. For each task and method, we select the checkpoint that maximizes the average test accuracy across generation lengths and report its performance at each length in Table 6. This mirrors the selection criterion used in our main evaluation, but selects directly on the reported test data. The resulting numbers are generally more optimistic, motivating our use of held-out checkpoint selection in the main experiments.

C.4 Gradient-distance measurement

Candidate gradients. All estimators considered here average the same type of per-order block gradient, $\nabla_\theta \log \pi_\theta^\sigma(y_b)$, but under different distributions over orders. For each block, we draw disjoint sets of orders for the two candidate estimators and for the reference. The uniform candidate,

Method	GSM8K			MATH			Countdown			Sudoku		
	128	256	512	128	256	512	128	256	512	128	256	512
ELBO-based setting												
LLaDA (ESPO eval)	70.6	79.0	81.0	28.0	31.2	34.4	32.8	19.1	20.3	24.8	16.2	6.1
+ ESPO	76.8	82.3	82.6	31.4	34.4	38.2	72.7	72.7	72.7	**91.8**	85.4	75.0
τ-ESPO	**77.2**+0.4	**82.5**+0.2	**82.9**+0.3	**31.4**+0.0	**37.0**+2.6	**39.0**+0.8	**79.3**+6.6	**80.9**+8.2	**77.0**+4.3	91.3-0.5	**89.1**+3.7	**82.3**+7.3
Trajectory-based setting												
LLaDA (d2 eval)	68.7	75.2	78.5	27.8	33.0	36.8	18.4	19.9	16.8	11.5	6.9	5.8
d2-StepMerge	73.4	81.4	81.8	29.4	36.6	39.4	**53.1**	41.0	44.1	**90.8**	62.7	37.0
τ-d2-StepMerge	**74.5**+1.1	**82.4**+1.0	**84.9**+3.1	**30.2**+0.8	**36.8**+0.2	**42.6**+3.2	**60.5**+7.4	**55.1**+14.1	**50.0**+5.9	90.1-0.7	**64.0**+1.2	**50.6**+13.6

Table 6: Test accuracy (%) when the checkpoint is selected to maximize average test accuracy across generation lengths. Bold values denote the best performance within each setting, dataset, and generation length. Coloured values report the difference relative to the corresponding baseline (ESPO or d2-StepMerge).

g_u, averages K orders sampled uniformly and corresponds to the ELBO gradient at matched compute. The confidence candidate, g_c^τ, averages K orders sampled from q_c^τ. We use $K = 4$ for both candidates, matching the sampling budget used by the estimator, with the same block segmentation.

Reference gradient. We compare both candidates to a higher-budget approximation $g^\star$ of the posterior-measure gradient. The reference is constructed by self-normalized importance sampling from $K_{\text{ref}} = 64$ independent draws under the fixed proposal

$$\nu_\star = \tfrac{1}{2}\,\mathrm{Unif}(S_L) + \tfrac{1}{2}\,q_c^{\tau_0}, \qquad \tau_0 = 2.$$

The proposal is held fixed for all candidate temperatures, so the reference does not change as τ varies. Its mixture form covers both the uniform and confidence-based regimes and avoids constructing the reference from either candidate sample set. Because the confidence component is fixed at $\tau_0 = 2$, the proposal is not perfectly symmetric with confidence candidates at other temperatures; we return to this limitation below. Across the measured blocks, the effective sample size of the reference ranges from approximately 15 to 30 out of the 64 draws.

Distances without materializing gradients. The full per-order gradients contain approximately $P \approx 8 \times 10^9$ parameters and cannot be materialized simultaneously. We therefore compress each per-order gradient $v_j \in \mathbb{R}^P$ with a signed random hash into $\phi(v_j) \in \mathbb{R}^m$ using CountSketch (Charikar et al., 2002; Weinberger et al., 2009), which preserves inner products in expectation; we use $m = 10^4$.

Let $\{v_j\}$ denote the pooled set of per-order gradients used by the candidates and reference. Each estimator can then be written as a weighted combination

$$\hat{g} = \sum_j c_j v_j,$$

with zero coefficients for samples not used by that estimator. The candidate weights are uniform within their respective sample sets, while the weights of $g^\star$ are the self-normalized importance weights. Defining the sketched Gram matrix

$$G_{jl} = \langle \phi(v_j), \phi(v_l) \rangle\,,$$

the squared distance between an estimator and the reference is approximated by

$$\|\hat{g} - g^\star\|^2 \approx (c - c^\star)^\top G (c - c^\star),$$

without explicitly forming a full P-dimensional gradient. All gradients used in this analysis are full-parameter gradients.

Reporting. For each block and temperature, we report the relative squared distance

$$R_\tau = \frac{\|g_c^\tau - g^\star\|^2}{\|g_u - g^\star\|^2}.$$

Values below one indicate that the confidence estimator is closer to the reference gradient than the matched-compute ELBO estimator. We aggregate these ratios across blocks using the geometric mean, and additionally report the median and the fraction of blocks for which $R_\tau < 1$. We evaluate 64 blocks per task.

Validation and limitations. We validate the sketch-based computation on a small enumerable model for which full autograd gradients can be accumulated explicitly. Sketched squared distances agree with explicitly computed distances to approximately 10^{-2}, and the same calculation using the exact Gram matrix agrees to approximately 10^{-5}. The per-order log densities entering the importance weights also agree with an independent replay of the proposal to approximately 10^{-4}. We additionally tested restricting the comparison to the output head, but found that it poorly reproduces full-parameter distances (rank correlation ≈ 0.2), and therefore do not use this approximation.

The measurement nevertheless has two important limitations. First, $g^\star$ is itself a finite-sample importance-sampling estimate rather than the exact posterior gradient. Consequently, error in the reference can affect both distances in R_τ, and the reported ratios should be interpreted as relative distances to a high-budget reference rather than exact distances to the true gradient. Second, although the reference proposal is fixed across the temperature sweep, its confidence component uses $\tau_0 = 2$. The temperature minimizing the measured distance therefore retains some dependence on this choice. We observe this dependence to be mild in our measurements, while the independent ESS diagnostic of Figure 2a, which does not use $g^\star$, identifies the same broad temperature region.

C.5 Synthetic validation

To verify the identities and approximations used throughout the paper independently of Monte Carlo error, we construct a small synthetic model for which all reveal orders can be enumerated exactly. The model has $L = 6$ positions and a binary vocabulary, with conditionals controlled by a single parameter that varies the degree of order dependence from nearly order-invariant to strongly order-sensitive. Since $6! = 720$, all quantities can be computed by direct summation over reveal orders.

Across this sweep, the posterior-gradient identity of Proposition 1 and the covariance representation underlying Proposition 2 hold to numerical precision. The corresponding block factorization identities also hold to numerical precision. The bound in (14) remains within a factor of approximately 1.8–6 of the exact gradient gap across the sweep, indicating that it captures the scale of the discrepancy rather than serving only as a vacuous upper bound.

The synthetic model also allows us to verify the ESS diagnostic directly. When samples are drawn from a uniform proposal and reweighted toward the posterior, the empirical ratio N_{eff}/K converges to the asymptotic value $1/(1+\chi^2)$ predicted by Proposition 3. At small sample budgets the resulting estimator remains noisy, however: its error decreases from 0.67 at $K = 4$ to 0.14 at $K = 1024$. This illustrates why directly estimating the posterior gradient by importance sampling is impractical at the budgets used during RL.

Finally, the optimal confidence temperature is generally interior rather than concentrated at either extreme, with $\tau^\star \in [1, 2]$ across the sweep. This mirrors the behavior observed in the ESS experiments of Section 4.