Re-evaluating Confidence Remasking in Masked Diffusion Language Models

Stipe Frković [*1] **Metod Jazbec** [*1] **Dan Zhang** [2] **Christian A. Naesseth** [1] **Ilija Bogunovic** [3] **Eric Nalisnick** [4]

Abstract

Masked diffusion language models (dLLMs) have recently emerged as a competitive alternative to autoregressive language models, with the promise of faster inference via parallel token generation. A notable limitation of the masked formulation, however, is that once a token has been unmasked it can no longer be revised, leaving dLLMs vulnerable to early sampling mistakes. To address this, a growing body of work has sought to extend masked dLLMs with self-correcting (remasking) capabilities. One appealing subset of these methods does so in a training-free, post-hoc manner based on token confidences, with encouraging early reported results. In this work, we revisit the empirical evaluation of a representative post-hoc remasking method, WINO (Hong et al., 2026), and find that under standard decoding settings (shorter block lengths) it brings little-to-no benefit over confidence-based unmasking alone (Wu et al., 2025). Extending the evaluation to non-greedy decoding, we find that while confidence-based remasking can mitigate errors introduced by increased stochasticity to some extent, it also exacerbates the diversity collapse previously reported for confidence-based unmasking. Overall, our results show that the benefits of post-hoc confidence-based remasking are highly setting-dependent, underscoring the need for a more comprehensive evaluation framework.

1. Introduction

Autoregressive (AR) language models have become the dominant paradigm for text generation, but their sequential left-to-right decoding inherently limits inference throughput. Masked diffusion language models (dLLMs; Sahoo et al. 2024; Shi et al. 2024) have recently emerged as a competitive alternative: starting from a fully-masked sequence, dLLMs progressively unmask tokens in arbitrary order, enabling parallel decoding within each step. Recent work has scaled dLLMs to billions of parameters (Nie et al., 2025; Ye et al., 2025; Bie et al., 2025; Bethune et al., 2026) and shown that they match AR models on a range of downstream tasks while offering the prospect of substantially more efficient inference (Wu et al., 2025; Ben-Hamu et al., 2025; Kim et al., 2025b; Jazbec et al., 2025).

A well-known limitation of the masked diffusion paradigm is that, once a token has been unmasked, the model has no mechanism to revisit it (Wang et al., 2025; von Rütte, 2026), even if, later in the generation process and with more context, another token would be a better fit. This stems directly from the absorbing forward process used during training, and it makes masked dLLMs susceptible to error propagation and limits their overall performance. To address this, a large body of work has recently attempted to equip masked dLLMs with self-correcting capabilities, either through finetuning/continual pre-training (Kim et al., 2025a; Schiff et al., 2026; Chen et al., 2026; Yu et al., 2026), RL post-training (Huang et al., 2025), or post-hoc (Hong et al., 2026; Dong et al., 2025; Xiang et al., 2026), i.e. in a training-free manner. The latter is particularly appealing as it can be applied off-the-shelf to any pretrained dLLM. A representative example is WINO (Hong et al., 2026), which complements confidence-based adaptive unmasking (Fast-dLLM; Wu et al. 2025) with a confidence-based remasking step that reverts already-decoded tokens deemed unlikely under their surrounding context.

In this work, we do not propose a new remasking approach but instead aim to better understand when training-free, confidence-based remasking actually helps in masked dLLMs. To this end, we revisit the empirical evaluation of WINO (Hong et al., 2026), asking how much benefit remasking adds beyond strong confidence-based unmasking like Fast-dLLM (Wu et al., 2025) — a crucial comparison lacking in prior work (Section 3.1). We find that in the standard sampling setting (short block lengths, greedy decoding), WINO performs on par with Fast-dLLM, yielding negligible benefits, especially once the extra latency cost of remasking is accounted for. We trace this ineffectiveness to

[*]Equal contribution [1]UvA-Bosch Delta Lab, University of Amsterdam [2]Bosch Center for Artificial Intelligence [3]University of Basel [4]Johns Hopkins University. Correspondence to: Stipe Frković <stipe.frkovic@student.uva.nl>, Metod Jazbec <m.jazbec@uva.nl>.

Accepted to the Resource-Adaptive Foundation Model Inference workshop at ICML 2026. Seoul, South Korea. PMLR 306, 2026.
Copyright 2026 by the author(s).

a failure of the underlying dLLM to propose alternative, 'better' tokens at positions flagged for remasking (Section 3.2).

Extending the evaluation to stochastic generation (Section 3.3), we find that the picture becomes more nuanced. Under non-greedy decoding, confidence-based remasking helps mitigate some stochasticity-induced errors (indicated by improved pass@1 rates) but aggravates the diversity collapse previously reported for confidence-based unmasking (Ni et al., 2026; Olausson et al., 2026). In contrast, when paired with recently proposed learned Bernoulli unmasking policies (Chen et al., 2025), WINO yields meaningfully larger gains than on top of deterministic Fast-dLLM. Together, these results highlight that the effectiveness of current post-hoc remasking is highly dependent on the decoding setting (block size, unmasking strategy, sampling temperature) and call for more comprehensive evaluation frameworks going forward.

2. Background

We denote a sequence of tokens by $\boldsymbol{x} = (x^1, \ldots, x^L) \in \mathcal{V}^L$, where $\mathcal{V} := \{1, \ldots, V\}$ is the vocabulary and L is the sequence length. Given our focus on masked discrete diffusion, the vocabulary is extended with a special mask token denoted with $\texttt{M}$. Throughout, we use the notation $[L]$ to represent the set $\{1, \ldots, L\}$ for brevity. With *unmasking* we refer to $\texttt{M} \to x$ transitions and with *remasking* to $x \to \texttt{M}$ (where $x \in \mathcal{V}/\{\texttt{M}\}$).

2.1. Masked Diffusion Language Models

Masked Diffusion Models (MDMs) are a form of absorbing discrete diffusion (Austin et al., 2021a; Hoogeboom et al., 2021) whose noise distribution concentrates on a single absorbing state $\texttt{M}$. Under the continuous-time formulation of Campbell et al. (2022), the forward process independently replaces each token of $\boldsymbol{x}_0 \sim p_{\text{data}}$ with a mask token $\texttt{M}$:

$$p_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0) = \prod_{k=1}^{L} \alpha(t)\mathbf{1}[x_t^k = \texttt{M}] + (1-\alpha(t))\mathbf{1}[x_t^k = x_0^k]$$

where $\alpha(t)$ is a masking probability governed by a monotonically increasing schedule $\alpha : [0,1] \to [0,1]$ with $\alpha(0) = 0$ and $\alpha(1) = 1$. The reverse (generative) process is typically modeled by training a network q_θ to approximate the true reverse conditional $p_{0|t}$ directly, since its ELBO admits the simple masked-prediction form (Sahoo et al., 2024; Shi et al., 2024; Ou et al., 2025):

$$-\mathcal{L}(\theta) \triangleq \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{x}_t}\left[\frac{1}{t}\sum_{k=1}^{L}\mathbf{1}[x_t^k = \texttt{M}]\log q_\theta^k(x_0^k \mid \boldsymbol{x}_t)\right],$$

with $t \sim U[0,1]$, $\boldsymbol{x}_0 \sim p_{\text{data}}$, $\boldsymbol{x}_t \sim p_{t|0}$, and $q_\theta^k(\cdot \mid \boldsymbol{x}_t)$ denoting the model's predicted marginal at position k. Recently, such models have been scaled to multi-billion parameters architecture on text data, (Nie et al., 2025; Ye et al., 2025) - we refer to such models as *masked diffusion large language models* (dLLMs) throughout the paper.

2.2. Unmasking in dLLMs

At test time, the reverse process is discretized and starting from a fully-masked sequence $\boldsymbol{x}_T := \texttt{M}^L$ new samples are obtained via progressive *unmasking*. Representing the current generation as $\boldsymbol{x}_t \in \mathcal{V}^L$ with still-masked positions $\mathcal{M}_t := \{k \in [L] \mid x_t^k = \texttt{M}\}$, the next-step generation $\boldsymbol{x}_{t-1}$ is sampled via

$$x_{t-1}^k := \begin{cases} x \sim q_\theta^k(\cdot \mid \boldsymbol{x}_t\,;\tau), & \text{if } k \in \mathcal{U}_t^K, \\ x_t^k\,, & \text{otherwise.} \end{cases}$$

where the set $\mathcal{U}_t^K$ of $K \geq 1$ unmasking positions is sampled uniformly at random (without replacement) from $\mathcal{M}_t$ (Zheng et al., 2025; Nie et al., 2025) and the process is repeated until there are no masked positions left ($\mathcal{M}_t = \emptyset$). Here τ denotes a sampling temperature, with higher temperatures yielding more diverse generated samples. In dLLMs, it is common to set $\tau = 0$ which corresponds to greedy token selection.

Algorithm 1 WINO sampling algorithm (Hong et al., 2026). Differences to Fast-dLLM (Wu et al., 2025) unmasking are denoted in blue.

Input: $\boldsymbol{x}_t, b_t, q_\theta, \tau, \lambda_1, \lambda_2$,
Output: $\boldsymbol{x}_{t-1}, b_{t-1}$

$\tilde{\boldsymbol{x}}_t \leftarrow [\boldsymbol{x}_t, \texttt{M}, \ldots, \texttt{M}] \in \mathcal{V}^{L+BL}$
$\mathcal{B} \leftarrow [b_t \cdot BL : (b_t+1)\cdot BL]$
$\mathcal{M}_t \leftarrow \{k \in \mathcal{B} \mid x_t^k = \texttt{M}\}$
$q_t^k \leftarrow q_\theta^k(\cdot \mid \tilde{\boldsymbol{x}}_t)$

$x_0^k \sim q_\theta^k(\cdot \mid \tilde{\boldsymbol{x}}_t\,;\tau)$
$c_t^k \leftarrow q_t^k(x_0^k)$
$\mathcal{U}_t \leftarrow \{k \in \mathcal{M}_t \mid c_t^k > \lambda_1\} \cup \{\arg\max_{k\in\mathcal{M}_t} c_t^k\}$

$\tilde{c}_t^k \leftarrow q_t^{k_s}(x_t^k), k_s = L + (k - b_t \cdot BL)$
$\mathcal{R}_t \leftarrow \{k \in \mathcal{B} \setminus \mathcal{M}_t \mid \tilde{c}_t^k < \lambda_2\}$
$\mathcal{R}_t \leftarrow \texttt{Loop-guard}(\mathcal{R}_t, \mathcal{U}_{t+1})$

$$x_{t-1}^k := \begin{cases} x_0^k\,, & k \in \mathcal{U}_t \\ \texttt{M}\,, & k \in \mathcal{R}_t \\ x_t^k\,, & \text{else} \end{cases}$$

$$b_{t-1} := \begin{cases} b_t + 1\,, & \{k \in \mathcal{B} | x_{t-1}^k = \texttt{M}\} = \emptyset \\ b_t\,, & \text{else} \end{cases}$$

In practice, confidence-based unmasking is often chosen over random unmasking due to better empirical performance. Two particularly popular instantiations are *high-*

confidence sampling (Chang et al., 2022; Nie et al., 2025) where K positions with the highest token confidence[1] $c_t^k \triangleq \max_v q_\theta^k(v \mid \boldsymbol{x}_t)$ are unmasked

$$\mathcal{U}_t^K := \underset{I \subseteq \mathcal{M}_t,\ |I|=K}{\arg\max} \sum_{k \in I} c_t^k \tag{1}$$

and *Fast-dLLM* (Wu et al., 2025) where all positions with token confidence larger than a pre-specified threshold λ_1 are unmasked

$$\mathcal{U}_t^\lambda := \{k \in \mathcal{M}_t \mid c_t^k > \lambda_1\} \cup \{\arg\max_{k \in \mathcal{M}_t} c_t^k\} \tag{2}$$

with a fallback to the position with the highest confidence to ensure progress is made in every sampling step. Due to its adaptive nature, Fast-dLLM was empirically shown to push the frontier of parallel-token generation in dLLMs.

Additionally, confidence-based samplers are often combined with semi-autoregressive (semi-AR; Arriola et al. 2025) sampling, which restricts unmasking to a contiguous block of $BL < L$ tokens at a time and proceeds block-by-block in a left-to-right order (i.e., once all positions in the current block are unmasked, generation moves to the next block). Empirically, semi-AR sampling has proved essential for confidence-based samplers, as using too large block length BL leads to compromised performance (Nie et al., 2025).

2.3. Confidence-based Remasking in dLLMs

In masked dLLMs, once a position has been predicted/unmasked, the model has no option to revisit it, even if at a later time step, and with additional context, another token would be a better fit. This makes masked models susceptible to early sampling mistakes from which they can not recover (Wang et al., 2025). Note that the inability to remask stems from the absorbing (towards M) forward process used during training. To overcome this limitation, a growing body of literature has tried to extend masked dLLMs with self-correcting capabilities (Hong et al. 2026; Kim et al. 2025a; Huang et al. 2025; Schiff et al. 2026; *inter-alia*). While most approaches require some form of retraining or finetuning, we focus here on methods that aim to achieve remasking in a post-hoc, i.e., training-free, manner.

A representative post-hoc remasking approach is WINO (Hong et al., 2026), where the idea of confidence thresholding unmasking (Wu et al., 2025) is extended to remasking. To get informative remasking confidences efficiently, WINO attaches to the current generation $\boldsymbol{x}_t$ an auxiliary block of the so called *shadow* tokens $\tilde{\boldsymbol{x}}_t = [\boldsymbol{x}_t, \mathtt{M}, \ldots, \mathtt{M}] \in \mathcal{V}^{L+BL}$, one per position in the active block b_t. Crucially, each shadow position has the same positional embedding as its corresponding position in the original block but is prevented from attending to the (unmasked) token at the original position via a custom attention mask (see Figure 2). This ensures that the shadow output approximates a re-prediction (starting from M instead of the current token x_t^k) of that position given the surrounding context without the cost of an additional forward pass:[2]

$$q_\theta^{k_s}(\cdot \mid \tilde{\boldsymbol{x}}_t) \approx q_\theta^k(\cdot \mid \boldsymbol{x}_{t,-k}) \tag{3}$$

where $k_s = L + (k - b_t \cdot BL)$ is the shadow position corresponding to the k-th original position and $\boldsymbol{x}_{t,-k}$ corresponds to current generation $\boldsymbol{x}_t$ with the k-th token masked out (i.e., leave-one-out style prediction). Note that original positions do not attend to shadow positions, hence their predictive distribution is not affected by the presence of the shadow block: $q_\theta^k(\cdot \mid \tilde{\boldsymbol{x}}_t) = q_\theta^k(\cdot \mid \boldsymbol{x}_t)$ for $k \leq L$. All presently unmasked tokens in the active block whose shadow confidence falls below a fixed threshold λ_2 are then remasked

$$\mathcal{R}_t := \{k \in \mathcal{B} \setminus \mathcal{M}_t \mid q_\theta^{k_s}(x_t^k \mid \tilde{\boldsymbol{x}}_t) < \lambda_2\} \tag{4}$$

where $\mathcal{B} := [b_t \cdot BL : (b_t + 1) \cdot BL]$ denotes the set of indices in the active block b_t; see also Algorithm 1. The intuition is that if the current token x_t^k would not likely be predicted under the same surrounding context (captured by shadow predictive distribution $q_\theta^{k_s}$), the position is set to M again so that the model can predict a new token for it in the future sampling steps under a new context.

Note that once remasking is allowed, convergence ($\mathcal{M}_t = \emptyset$) is no longer guaranteed: sampling can enter a loop in which a given set of positions is repeatedly unmasked to the same tokens after being remasked. WINO's implementation[3] addresses this with a *loop-guard* mechanism that whenever too many tokens get selected for remasking via confidence thresholding, i.e. $|\mathcal{R}_t| \geq |\mathcal{U}_{t+1}|$, only $K = |\mathcal{U}_{t+1}| - 1$ positions with the smallest shadow confidences are remasked, guaranteeing net progress at every step.

3. Re-evaluation of WINO Remasking

We next turn our attention to the question of whether post-hoc remasking methods such as WINO are effective in improving the performance of masked dLLMs. Note that we find the empirical evaluation in the WINO paper (Hong et al., 2026) insufficient for three main reasons: i) high-confidence sampling (Equation 1) is used as the main baseline, leaving it unclear whether the performance gains of WINO over high-confidence sampling come from adaptive unmasking (Fast-dLLM) or from remasking itself; ii) the evaluation focuses on larger block lengths ($BL \geq 128$), which differs

[1] For non-greedy sampling, the confidence is defined instead as $c_t^k \triangleq q_\theta^k(x \mid \boldsymbol{x}_t),\ x \sim q_\theta^k(\cdot \mid \boldsymbol{x}_t\,; \tau)$.

[2] The approximation is exact for a single-layer transformer; for deeper models, the surrounding tokens' hidden state representations still encode information about x_t^k via self-attention.

[3] https://github.com/Feng-Hong/WINO-DLLM/blob/main/LLaDA/decoding.py

from the standard practice of using smaller block lengths (e.g., $BL = 32$). Moreover, confidence-based unmasking is known to struggle at larger block lengths, making the comparison potentially biased (Nie et al., 2025); and iii) the impact of remasking is studied primarily under greedy decoding ($\tau = 0$), leaving unanswered the question of how remasking affects performance under non-greedy decoding.

3.1. WINO vs Fast-dLLM

We start by investigating whether WINO remasking can bring meaningful performance gains over confidence-threshold unmasking alone (Wu et al., 2025).

Experiment details We compare Fast-dLLM decoding with WINO decoding on two open-source dLLMs: LLaDA-8B-Instruct (Nie et al., 2025) and Dream-v0-Instruct-7B (Ye et al., 2025). We evaluate on four standard benchmarks: GSM8k (Cobbe et al., 2021), MATH-500 (Hendrycks et al., 2021), HumanEval (Chen et al., 2021a), and MBPP (Austin et al., 2021b). For both decoding methods, we vary the unmasking threshold $\lambda_1 \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ to obtain pareto-frontiers for each approach. For WINO, we set the remasking threshold $\lambda_2 = 0.8$ and use greedy decoding ($\tau = 0$). To understand the impact of different block lengths on the impact of remasking, we vary $BL \in \{32, 128\}$: $BL = 32$ is the current standard setting for LLaDA-style models (Ni et al., 2026; Wu et al., 2025), while $BL = 128$ is the primary setting reported in the WINO paper.

Focusing first on the $BL = 32$ results (Figure 1), we observe that remasking sometimes improves performance (e.g., see LLaDA's performance on HumanEval), but the gains are generally marginal and inconsistent across datasets and along the Pareto frontier. Concretely, WINO improves over Fast-dLLM's top performance by $\sim 0.4\%$ and $\sim 0.5\%$ on LLaDA and Dream models, respectively (averaged across the four datasets considered). Beyond the marginal size of these gains, comparing the efficiency of WINO and Fast-dLLM solely in terms of NFEs is potentially misleading: each WINO forward pass extends the sequence with an additional shadow block, thereby increasing both FLOPs and latency per step. To account for this, we also report results in terms of throughput (tokens/second). As shown in Figure 3, on most datasets WINO reaches its top performance at lower throughput than Fast-dLLM, which, combined with its marginal accuracy improvements, suggests that WINO's gains may not justify its added computational overhead and implementation complexity. Additionally, in Appendix C.1 we ablate several of WINO's design choices (confidence-thresholding mechanism, loop-guard variant, unmasking aggressiveness, etc.) and find that the lack of meaningful gains at $BL = 32$ persists across all configurations.

We next turn to $BL = 128$, the main setting considered in the WINO paper (Hong et al., 2026). Here remasking yields larger gains (Figure 1) when compared to Fast-dLLM, on average improving top performance by $\sim 1.3\%$ and $\sim 1.5\%$ on LLaDA and Dream, respectively. Possibly this is because a larger block length gives WINO more opportunity to correct mistakes over a longer range. However, masked dLLMs like LLaDA are known to degrade under confidence-based samplers like Fast-dLLM as block length grows, due to corrupted `EOS` confidences that produce degenerate generation orders (Jazbec et al., 2025). The fact that WINO at $BL = 128$ underperforms Fast-dLLM at $BL = 32$ thus suggests that remasking is repairing Fast-dLLM's specific failure mode at large block lengths rather than genuinely raising the underlying dLLM's performance ceiling. Moreover, as seen in Figure 3, the latency overhead of WINO due to shadow tokens is, as expected, even larger for $BL = 128$.

Takeaway 1. While WINO can mitigate performance degradation of confidence-based unmasking for larger block lengths ($BL = 128$), it fails to meaningfully increase the performance ceiling of masked dLLMs for more commonly used smaller block lengths ($BL = 32$).

3.2. Investigating WINO's ineffectiveness for $BL = 32$

Following our observation on WINO's limited performance gains for shorter block lengths ($BL = 32$), we now aim to understand the potential reasons behind it.

We first examine the quality of the shadow-token approximation for the remasking confidence (Equation 3). Recall that, ideally, remasking would be based on the oracle, leave-one-out confidence $q_\theta^k(\cdot \mid \boldsymbol{x}_{t,-k})$. However, computing such oracle confidences requires an additional forward pass for every candidate remasking position at each sampling step. We nevertheless compute it here to obtain an upper bound on remasking performance via confidence-thresholding. Interestingly, as reported in Figure 5, WINO's shadow confidences match the performance of oracle confidences while being much cheaper (the oracle approach requires $\sim 14\times$ as many NFEs). We further compare top-1 consistency between shadow and oracle predictive distributions in Figure 6, finding it to be $\sim 100\%$, further validating the quality of shadow token approximation. Moreover, when relying on the original dLLM's confidence $q_\theta^k(\cdot \mid \boldsymbol{x}_t)$ at the unmasked position to decide on remasking, the approach fails to improve over unmasking alone, confirming the necessity of the shadow-tokens.

Since the shadow-token approximation does not explain WINO's limited gains, we next measure the *flip-flop* frequency of WINO, following the analysis proposed in (Xiang et al., 2026). Concretely, the flip-flop frequency is defined as the proportion of remasks that result in no token change,

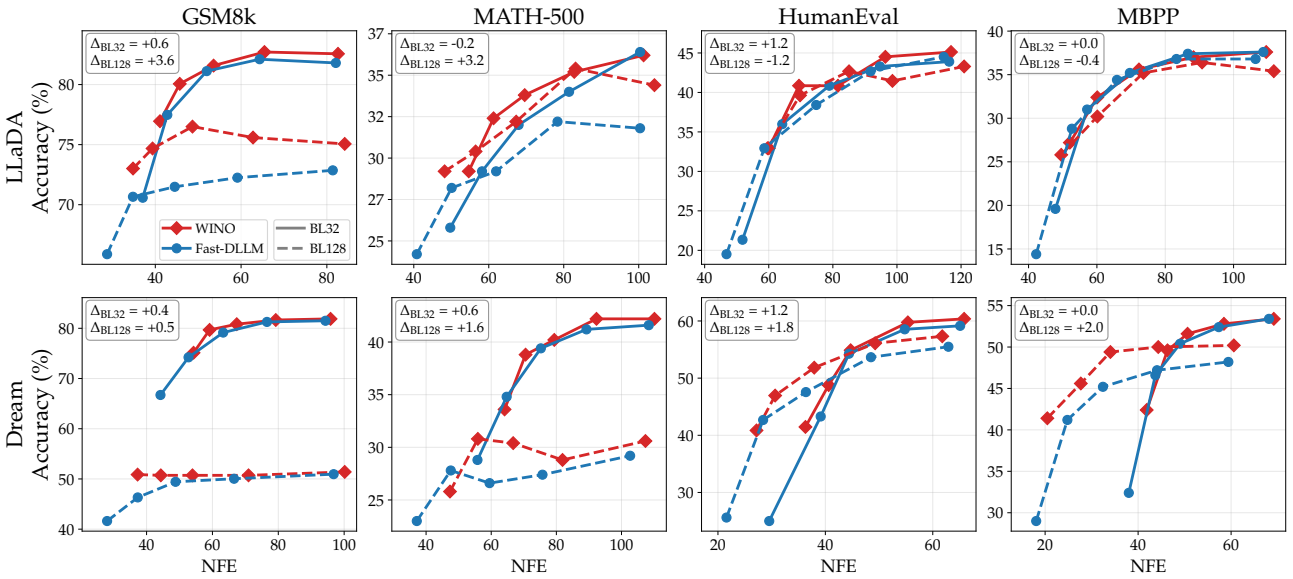


Figure 1. Pareto fronts of accuracy vs. network function evaluations (NFEs) for WINO and Fast-dLLM, sweeping the unmasking threshold $\lambda_1 \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$, with block lengths $BL \in \{32, 128\}$ and WINO remasking threshold $\lambda_2 = 0.8$ (greedy decoding, $\tau = 0$). At the standard short block length ($BL = 32$), WINO yields only marginal and inconsistent improvements over Fast-dLLM; at $BL = 128$ WINO gains are larger; however since WINO $BL = 128$ underperforms Fast-dLLM $BL = 32$ this suggests it primarily compensates for Fast-dLLM's known degradation at large block lengths rather than raising the underlying dLLM's performance ceiling. When reporting Δ, we take the difference between the best accuracy of WINO and the best accuracy of Fast-dLLM, each evaluated at its optimal point along its respective Pareto frontier.

i.e., a position is remasked and then re-predicted to the same token at a later sampling step. As shown in Figure 4, flip-flops occur at high rates across the entire Pareto frontier: on average $\sim$ 75–90% for LLaDA and $\sim$ 85–95% for Dream of remasked positions end up with the same token. Combined with the shadow-token result above, this shows that WINO is effective at *identifying* positions that need remasking (at least relative to the oracle confidence); the underlying dLLM, however, fails to *propose* better alternatives at those positions, as indicated by high flip-flop rates.

One possible explanation for high flip-flop rates is that the later tokens were unmasked conditioned on x_t^k, so even after position k is remasked, the surrounding context still encodes x_t^k and pins the predictive distribution back to it. To test this cascading-dependency hypothesis, we run an ablation in Appendix C.2 in which, whenever a position is flagged for remasking, we additionally remask either i) tokens that were unmasked at the same or the next few sampling steps, or ii) its immediate left and right neighbors, giving the dLLM more flexibility to revise its prediction. Surprisingly, both variants degrade performance and fail to reduce the flip-flop rate relative to the default of remasking only the flagged position. This indicates that cascading dependencies are not the bottleneck: the dLLM's inability to propose a better-fitting token persists even when the conditioning context around k is partially removed.

Takeaway 2. WINO's shadow-token confidence is a good approximation of the oracle, leave-one-out remasking confidence; however, the underlying masked dLLM fails to propose better alternative tokens at the identified remasking positions indicated by high flip-flop rates.

3.3. Impact of remasking for stochastic generation

Lastly, we study whether post-hoc remasking is beneficial under stochastic generation. Although this setting is largely absent from the current literature on remasking in dLLMs, we believe it is particularly relevant for remasking approaches, as the increased stochasticity may produce more erroneous tokens that could benefit from revision. We consider two variants of stochastic generation: i) non-greedy decoding ($\tau > 0$) with deterministic unmasking (Fast-dLLM) and ii) greedy decoding with stochastic unmasking (via learned Bernoulli policies; Jazbec et al. 2025; Chen et al. 2025).

Non-greedy decoding ($\tau > 0$) We compare Fast-dLLM ($\lambda_1 = 0.6$) and WINO ($\lambda_1 = 0.6$, $\lambda_2 = 0.8$) on the LLaDA model with $BL = 32$ under non-greedy decoding. Concretely, we consider $\tau = 0.8$, a standard temperature in RL post-training of dLLMs (Zhao et al., 2025; Tang et al., 2025), as well as $\tau = 1.5$, motivated by the observation that dLLMs

benefit from higher sampling temperatures than AR LLMs (Ni et al., 2026). To measure performance, we report the commonly used `pass@k` metric[4], which tracks the probability that at least one of k generated solutions is correct. As shown in Figure 7, remasking yields decent improvements at `pass@1`, e.g., an average improvement of $\sim$2.6% across the four datasets at $\tau = 0.8$. However, remasking also appears to hurt generation diversity: WINO scales worse than Fast-dLLM as k grows, with gains shrinking to $\sim$0.7% at `pass@64` for $\tau = 0.8$. This suggests that the previously observed diversity collapse of confidence-based unmasking (Ni et al., 2026; Olausson et al., 2026)—where greedily relying on confidence to decide which positions to sample constrains exploration of the solution space—may be further exacerbated by confidence-based remasking, warranting future investigation. Interestingly, we also observe that WINO's performance improvements shrink slightly at larger $\tau = 1.5$, with average improvements dropping to $\sim$1.6% and $\sim$0.3% for `pass@1` and `pass@64`, respectively.

Takeaway 3. For non-greedy decoding, WINO yields decent improvements at `pass@1` but exacerbates the diversity collapse of confidence-based unmasking, with gains shrinking as the number of samples k grows.

Bernoulli unmasking We next consider the case where tokens are still chosen greedily ($\tau = 0$) but the unmasking decisions are stochastic. Such stochastic unmaskers, implemented via learned Bernoulli policies, have recently gained traction as a scalable alternative to confidence-based unmasking heuristics like Fast-dLLM. As a concrete example, we consider dUltra (Chen et al., 2025), which during RL post-training of dLLMs jointly trains an unmasking head that predicts the unmasking probability of each still-masked position at every sampling step:

$$\mathcal{U}_t := \{k \in \mathcal{M}_t \mid b_t^k = 1\}, \tag{5}$$

$$b_t^k \sim \mathrm{Ber}(s_t^k), \tag{6}$$

$$s_t^k = \sigma\Big(f_\phi(\boldsymbol{h}_t^{1:L})_k\Big) \tag{7}$$

where f_ϕ is the unmasking head, $h_t^{1:L}$ denotes the dLLM's last-layer hidden states (before the unembedding matrix), and σ is the sigmoid function. Note that due to the Bernoulli sampling of unmasking decisions, dUltra gives rise to stochastic generation even at $\tau = 0$.

Results comparing dUltra with and without WINO remasking ($\lambda_2 = 0.8$) at $BL = 32$ are shown in Figure 8; we report average accuracy and NFEs on test questions over 3 random seeds (with error bars). Promisingly, adding WINO remasking on top of dUltra improves accuracy by $\sim$3.2% on average across datasets, while increasing NFEs by only $\sim$2 on average. This highlights the potential of post-hoc remasking when paired with stochastic unmasking strategies: it appears effective at correcting low-quality tokens introduced by the randomness of Bernoulli unmasking. Moreover, dUltra produces faster generations than Fast-dLLM (e.g., it reaches $\sim$80% accuracy in only $\sim$20 NFEs, versus $\sim$50 for Fast-dLLM; see Figure 1), which may further explain WINO's stronger gains here—more aggressive generation yields more tokens in need of revision.

Takeaway 4. When paired with stochastic Bernoulli unmasking samplers, WINO remasking proves more effective; suggesting that the added value of post-hoc remasking is sensitive to the choice of unmasking strategy.

4. Conclusion

To better understand the added value of confidence-based remasking in masked dLLMs, we re-evaluated WINO (Hong et al., 2026), a recently proposed training-free remasking approach. Our findings suggest that such post-hoc remasking offers little-to-no benefit over unmasking-only sampling (Fast-dLLM), particularly at shorter block lengths and once the latency overhead of remasking is properly accounted for. We further find that WINO's self-correction yields larger gains under non-greedy sampling, though at the cost of generation diversity, and that its overall impact depends on the unmasking strategy it is paired with. Since our results suggest that remasking's impact is highly setting-dependent (e.g., sampling temperature, unmasking strategy), we hope that future work adopts the evaluation framework outlined here to better understand the limits and potential of post-hoc confidence-based remasking in masked dLLMs.

Limitations and future work While we believe WINO is representative of confidence-based remasking methods, future work should also re-evaluate other recent post-hoc remasking strategies such as ReMDM (Wang et al., 2025), Saber (Dong et al., 2025), and COVER (Xiang et al., 2026) under our proposed evaluation framework, to determine whether our findings are specific to WINO or hold more generally. If the latter, this would suggest that effective remasking in masked dLLMs likely requires either some form of retraining or a shift to uniform discrete diffusion (von Rütte, 2026). Moreover, while we deliberately focused on post-hoc remasking, it would be worthwhile to explore whether the negative impact of remasking on sampling diversity (Figure 7) also extends to approaches that incorporate remasking via retraining or finetuning (Schiff et al., 2026; Chen et al., 2026).

[4]Following Chen et al. (2021b), we report the unbiased estimator $\texttt{pass@}k := \mathbb{E}\big[1 - \binom{n-c}{k} / \binom{n}{k}\big]$, where $n \geq k$ is the number of samples generated per problem and c is the number of those that are correct.

Impact statement This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Arriola, M., Gokaslan, A., Chiu, J. T., Yang, Z., Qi, Z., Han, J., Sahoo, S. S., and Kuleshov, V. Block diffusion: Interpolating between autoregressive and diffusion language models. *arXiv preprint arXiv:2503.09573*, 2025.

Austin, J., Johnson, D. D., Ho, J., Tarlow, D., and van den Berg, R. Structured denoising diffusion models in discrete state-spaces. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 17981–17993. Curran Associates, Inc., 2021a. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/958c530554f78bcd8e97125b70e6973d-Paper.pdf.

Austin, J., Odena, A., Nye, M., Bosma, M., Michalewski, H., Dohan, D., Jiang, E., Cai, C., Terry, M., Le, Q., and Sutton, C. Program synthesis with large language models, 2021b. URL https://arxiv.org/abs/2108.07732.

Ben-Hamu, H., Gat, I., Severo, D., Nolte, N., and Karrer, B. Accelerated sampling from masked diffusion models via entropy bounded unmasking. *arXiv preprint arXiv:2505.24857*, 2025.

Bethune, L., Turrisi, V., Mlodozeniec, B. K., Lopez, P. R., Boominathan, L., Bhendawade, N., Shidani, A., Pelemans, J., Olausson, T. X., Hjelm, D., Dixon, P., Monteiro, J., Ablin, P., Banna, V., Blaas, A., Henderson, N., Noriy, K., Busbridge, D., Susskind, J., Cuturi, M., Belousova, I., Zappella, L., Webb, R., and Ramapuram, J. The design space of tri-modal masked diffusion models, 2026. URL https://arxiv.org/abs/2602.21472.

Bie, T., Cao, M., Chen, K., Du, L., Gong, M., Gong, Z., Gu, Y., Hu, J., Huang, Z., Lan, Z., et al. Llada2. 0: Scaling up diffusion language models to 100b. *arXiv preprint arXiv:2512.15745*, 2025.

Bie, T., Cao, M., Cao, X., Chen, B., Chen, F., Chen, K., Du, L., Feng, D., Feng, H., Gong, M., et al. Llada2. 1: Speeding up text diffusion via token editing. *arXiv preprint arXiv:2602.08676*, 2026.

Campbell, A., Benton, J., De Bortoli, V., Rainforth, T., Deligiannidis, G., and Doucet, A. A continuous time framework for discrete denoising models. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 28266–28279. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b5b528767aa35f5b1a60fe0aaeca0563-Paper-Conference.pdf.

Chang, H., Zhang, H., Jiang, L., Liu, C., and Freeman, W. T. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11315–11325, 2022.

Chen, M., Tworek, J., Jun, H., Yuan, Q., de Oliveira Pinto, H. P., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., Ray, A., Puri, R., Krueger, G., Petrov, M., Khlaaf, H., Sastry, G., Mishkin, P., Chan, B., Gray, S., Ryder, N., Pavlov, M., Power, A., Kaiser, L., Bavarian, M., Winter, C., Tillet, P., Such, F. P., Cummings, D., Plappert, M., Chantzis, F., Barnes, E., Herbert-Voss, A., Guss, W. H., Nichol, A., Paino, A., Tezak, N., Tang, J., Babuschkin, I., Balaji, S., Jain, S., Saunders, W., Hesse, C., Carr, A. N., Leike, J., Achiam, J., Misra, V., Morikawa, E., Radford, A., Knight, M., Brundage, M., Murati, M., Mayer, K., Welinder, P., McGrew, B., Amodei, D., McCandlish, S., Sutskever, I., and Zaremba, W. Evaluating large language models trained on code, 2021a. URL https://arxiv.org/abs/2107.03374.

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021b.

Chen, S., Jiao, J., Ratliff, L. J., and Zhu, B. dultra: Ultra-fast diffusion language models via reinforcement learning. *arXiv preprint arXiv:2512.21446*, 2025.

Chen, Z., Fang, G., Ma, X., Yu, R., and Wang, X. Dmax: Aggressive parallel decoding for dllms. *arXiv preprint arXiv:2604.08302*, 2026.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Dong, Y., Ma, Z., Jiang, X., Fan, Z., Qian, J., Li, Y., Xiao, J., Jin, Z., Cao, R., Li, B., et al. Saber: An efficient sampling with adaptive acceleration and backtracking enhanced remasking for diffusion language model. *arXiv preprint arXiv:2510.18165*, 2025.

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., and Steinhardt, J. Measuring mathematical problem solving with the math dataset. *NeurIPS*, 2021.

Hong, F., Yu, G., Ye, Y., Huang, H., Zheng, H., Zhang, Y., Wang, Y., and Yao, J. Wide-in, narrow-out: Revokable decoding for efficient and effective DLLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=XtLQHlNLxy.

Hoogeboom, E., Nielsen, D., Jaini, P., Forré, P., and Welling, M. Argmax flows and multinomial diffusion: Learning categorical distributions. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 12454–12465. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/67d96d458abdef21792e6d8e590244e7-Paper.pdf.

Huang, Z., Wang, Y., Chen, Z., and Qi, G.-J. Don't settle too early: Self-reflective remasking for diffusion language models. *arXiv preprint arXiv:2509.23653*, 2025.

Jazbec, M., Olausson, T. X., Béthune, L., Ablin, P., Kirchhof, M., Monteiro, J., Turrisi, V., Ramapuram, J., and Cuturi, M. Learning unmasking policies for diffusion language models. *arXiv preprint arXiv:2512.09106*, 2025.

Kim, J., Kim, S., Lee, T., Pan, D. Z., Kim, H., Kakade, S., and Chen, S. Fine-tuning masked diffusion for provable self-correction. *arXiv preprint arXiv:2510.01384*, 2025a.

Kim, J., Shah, K., Kontonis, V., Kakade, S., and Chen, S. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. *arXiv preprint arXiv:2502.06768*, 2025b.

Liu, L., Huang, B., Liu, X., Yin, B., and Zhao, T. Teach diffusion language models to learn from their own mistakes. *arXiv preprint arXiv:2601.06428*, 2026.

Ni, Z., Wang, S., Yue, Y., Yu, T., Zhao, W., Hua, Y., Chen, T., Song, J., Yu, C., Zheng, B., et al. The flexibility trap: Why arbitrary order limits reasoning potential in diffusion language models. *arXiv preprint arXiv:2601.15165*, 2026.

Nie, S., Zhu, F., You, Z., Zhang, X., Ou, J., Hu, J., Zhou, J., Lin, Y., Wen, J.-R., and Li, C. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025.

Olausson, T. X., Jazbec, M., Wang, X., Solar-Lezama, A., Naesseth, C. A., Mandt, S., and Nalisnick, E. A tale of two temperatures: Simple, efficient, and diverse sampling from diffusion language models. *arXiv preprint arXiv:2604.09921*, 2026.

Ou, J., Nie, S., Xue, K., Zhu, F., Sun, J., Li, Z., and Li, C. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In Yue, Y., Garg, A., Peng, N., Sha, F., and Yu, R. (eds.), *International Conference on Learning Representations*, volume 2025, pp. 64972–65009, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/a365e37c18fb91af547a2f0012a89e98-Paper-Conference.pdf.

Sahoo, S. S., Arriola, M., Gokaslan, A., Marroquin, E. M., Rush, A. M., Schiff, Y., Chiu, J. T., and Kuleshov, V. Simple and effective masked diffusion language models. In *ICML 2024 Workshop on Efficient and Accessible Foundation Models for Biological Discovery*, 2024. URL https://openreview.net/forum?id=DdU9gP4EXW.

Schiff, Y., Belhasin, O., Uziel, R., Wang, G., Arriola, M., Turok, G., Elad, M., and Kuleshov, V. Learn from your mistakes: Self-correcting masked diffusion models. *arXiv preprint arXiv:2602.11590*, 2026.

Shi, J., Han, K., Wang, Z., Doucet, A., and Titsias, M. Simplified and generalized masked diffusion for discrete data. *Advances in neural information processing systems*, 37:103131–103167, 2024.

Tang, X., Dolga, R., Yoon, S., and Bogunovic, I. wd1: Weighted policy optimization for reasoning in diffusion language models. *arXiv preprint arXiv:2507.08838*, 2025.

Von Rütte, D., Fluri, J., Ding, Y., Orvieto, A., Schölkopf, B., and Hofmann, T. Generalized interpolating discrete diffusion. *arXiv preprint arXiv:2503.04482*, 2025.

von Rütte, D. Why diffusion language models are the future, 2026. URL https://dimitri.ml/posts/why-diffusion-language-models-are-the-future/.

Wang, G., Schiff, Y., Sahoo, S. S., and Kuleshov, V. Remasking discrete diffusion models with inference-time scaling. *arXiv preprint arXiv:2503.00307*, 2025.

Wang, L., Wang, Z., Bai, Y., Deng, W., Lin, G., and Song, Q. Generalized discrete diffusion with self-correction. *arXiv preprint arXiv:2603.02230*, 2026.

Wu, C., Zhang, H., Xue, S., Liu, Z., Diao, S., Zhu, L., Luo, P., Han, S., and Xie, E. Fast-dllm: Training-free acceleration of diffusion llm by enabling kv cache and parallel decoding, 2025. URL https://arxiv.org/abs/2505.22618.

Xiang, Y., Wei, L., Yao, Y., Zhu, Q., Yan, H., Jin, C., Teare, P. A., Zhang, D., Gui, L., Saseendran, A., et al. Stop the flip-flop: Context-preserving verification for fast revocable diffusion decoding. *arXiv preprint arXiv:2602.06161*, 2026.

Ye, J., Xie, Z., Zheng, L., Gao, J., Wu, Z., Jiang, X., Li, Z., and Kong, L. Dream 7b: Diffusion large language models. *arXiv preprint arXiv:2508.15487*, 2025.

Yu, Y., Jian, Y., Wang, J., Zhou, Z., Zhuang, D., Fang, X., Yanamandra, S., Wu, X., Wu, Q., Song, S. L., et al. Introspective diffusion language models. *arXiv preprint arXiv:2604.11035*, 2026.

Zhao, S., Gupta, D., Zheng, Q., and Grover, A. d1: Scaling reasoning in diffusion large language models via reinforcement learning. *arXiv preprint arXiv:2504.12216*, 2025.

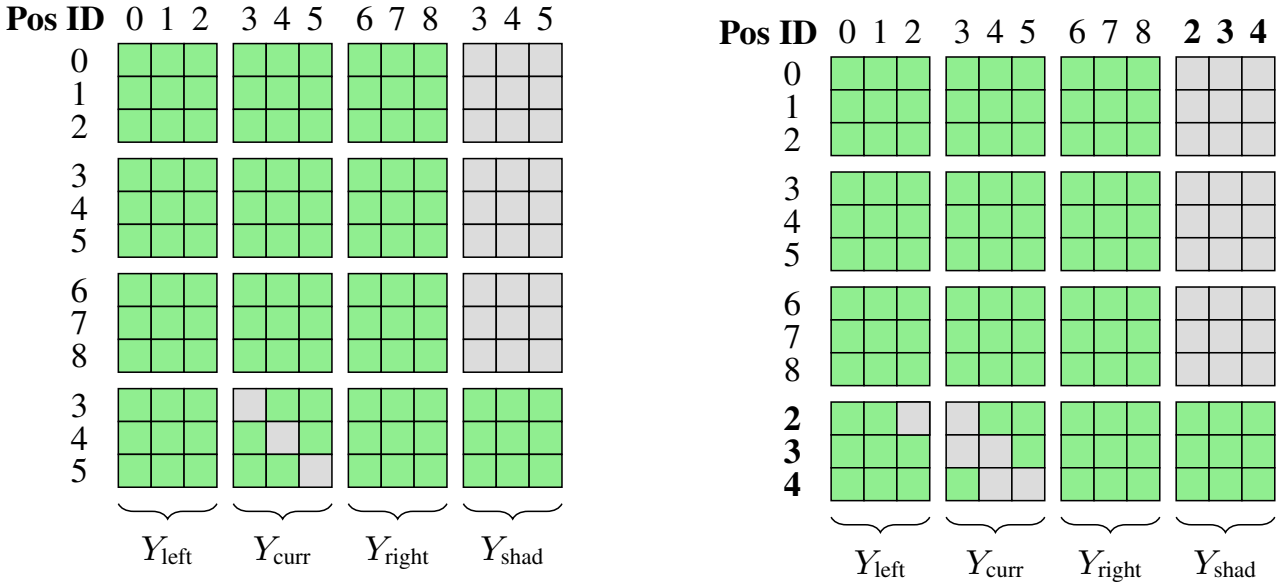
Zheng, K., Chen, Y., Mao, H., Liu, M.-Y., Zhu, J., and Zhang, Q. Masked Diffusion Models are Secretly Time-Agnostic Masked Models and Exploit Inaccurate Categorical Sampling, April 2025.

A. Related Work

Remasking in dLLMs. Several recent works explore remasking in masked dLLMs. One direction focuses on train-time interventions: Kim et al. (2025a) fine-tune dLLMs for provable self-correction, Schiff et al. (2026) train models to identify and revise their own errors, Liu et al. (2026) train a separate correction head on top of a frozen dLLM, Bie et al. (2026) jointly train mask-to-token and token-to-token editing objectives at scale, Yu et al. (2026) introduce introspective objectives, and Huang et al. (2025) use RL post-training to learn when to remask. A related direction is to move away from purely absorbing (masked) diffusion to a uniform one that supports remasking 'naturally': Von Rütte et al. (2025) train hybrid mask+uniform diffusion models, Wang et al. (2026) simplify this with purely uniform transitions, and Chen et al. (2026) convert pretrained masked dLLMs into uniform diffusion models. Our work focuses on a complementary, training-free direction.

Post-hoc remasking. Several recent methods enable remasking without retraining, typically by applying a confidence-based criterion at inference time. WINO (Hong et al., 2026), the focus of our re-evaluation, computes remasking confidences via auxiliary shadow tokens that approximate a leave-one-out re-prediction without an additional forward pass. Although mainly proposed as a train-time intervention, Wang et al. (2025) also introduces a post-hoc variant that interpolates between the absorbing and uniform processes to enable revisions during generation in masked diffusion models. Saber (Dong et al., 2025) proposes to perform confidence-based remasking by tracking drops in the confidences at the unmasked positions. Most directly related to our analysis, Xiang et al. (2026) document the high *flip-flop* rates of confidence-based remasking, which we leverage in Section 3.2 to help characterize WINO's failure mode for shorter block lengths. While these works each report empirical gains over static confidence-based unmasking (Equation 1), the question of how much these gains depend on the evaluation setting–particularly the choice of unmasking strategy and decoding setting–has not been carefully examined, which is a gap we aim to fill with our work.

B. Additional Figures



(a) WINO attention mask and position IDs for LLaDA-8B (Nie et al., 2025).

(b) Adaption for Dream-7B model (Ye et al., 2025).

Figure 2. WINO attention masks and position IDs: LLaDA (left) vs. our adaptation for Dream (right). Note that adaptation is needed since Dream was converted into a dLLM starting from an AR model. Bold indices in (b) highlight the modified position IDs of Y_{shad}. The shadow position IDs were shifted to obtain predictions for the current block, while the masking of main tokens is sub-diagonal and diagonal to prevent information leak due to Dream's shifted predictions.

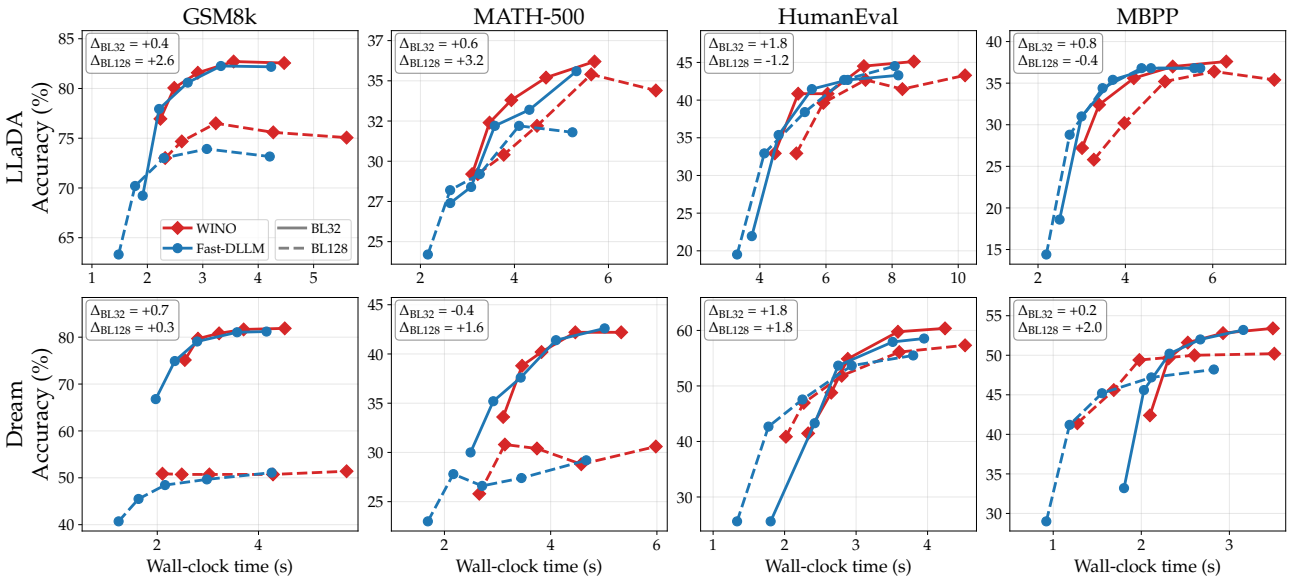


Figure 3. Figure 1 replicated using latency (wall-clock time) as an efficiency measure instead of network function evaluations (NFEs). We see that once the additional overhead of using shadow tokens in WINO is taken into account, WINO's performance gains diminish further.

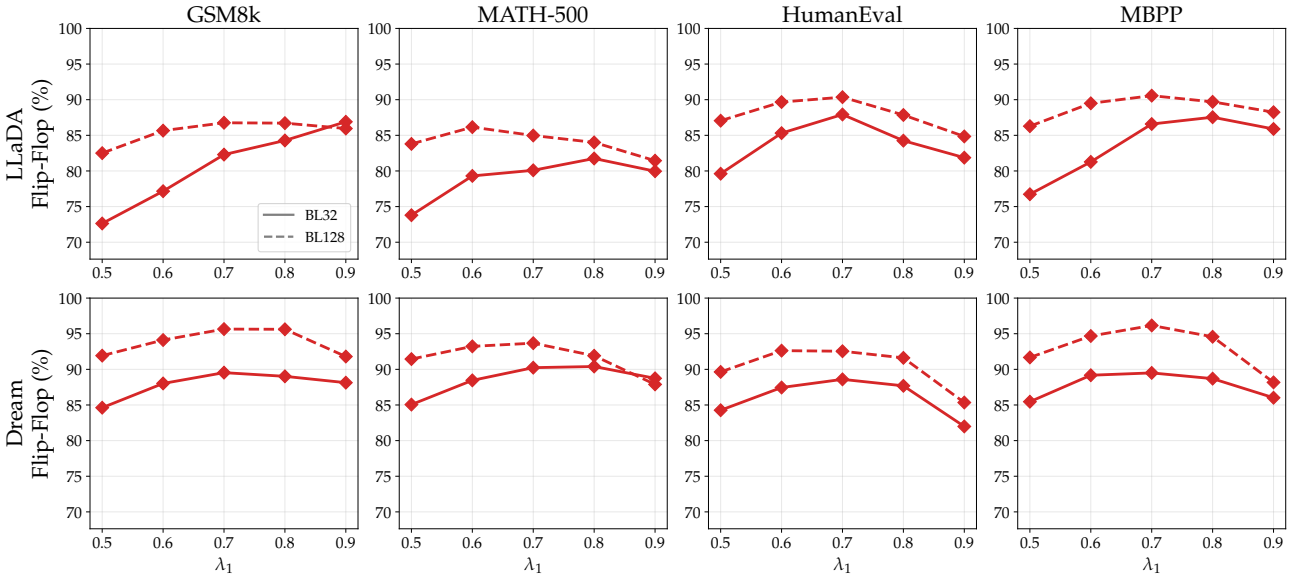


Figure 4. Flip-flop frequency along the WINO ($\lambda_2 = 0.8$) Pareto frontier, defined as the fraction of remasking events in which the position is later re-predicted to the same token (Xiang et al., 2026). Across all four benchmarks, ∼75–90% of remasked positions on LLaDA-8B-Instruct and ∼85–95% on Dream-v0-Instruct-7B are restored to their original token. This indicates that the underlying masked dLLM consistently fails to *propose* alternative tokens for the positions flagged for remasking based on WINO's shadow token confidences.

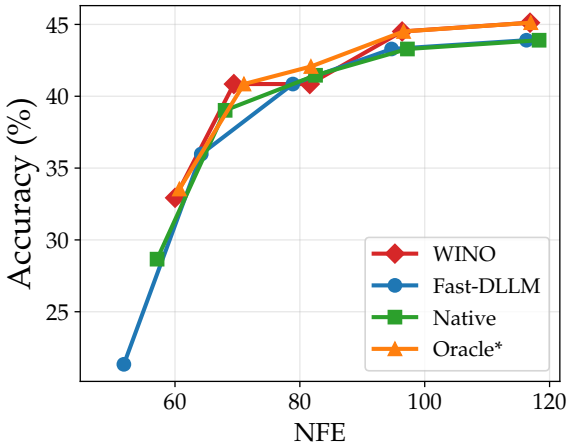


Figure 5. Quality of WINO's shadow-token confidence as an approximation (Equation 3) to the oracle leave-one-out confidence (LLaDA-8B-Instruct, HumanEval, $BL = 32$, $\lambda_2 = 0.8$). *WINO* uses the shadow prediction $q_\theta^{k_s}(\cdot \mid \tilde{\boldsymbol{x}}_t)$; *Oracle* uses the leave-one-out prediction $q_\theta^{k}(\cdot \mid \boldsymbol{x}_{t,-k})$, incurring one extra forward pass per candidate position; *Native* reuses the original dLLM confidence $q_\theta^{k}(\cdot \mid \boldsymbol{x}_t)$ at the already-unmasked position. Shadow confidences match oracle performance while requiring $\sim 14\times$ fewer NFEs, whereas the native confidence yields no gain over unmasking alone. *Note that for oracle approach we plot here number of sampling steps; actual NFEs are $\sim 14\times$ larger.

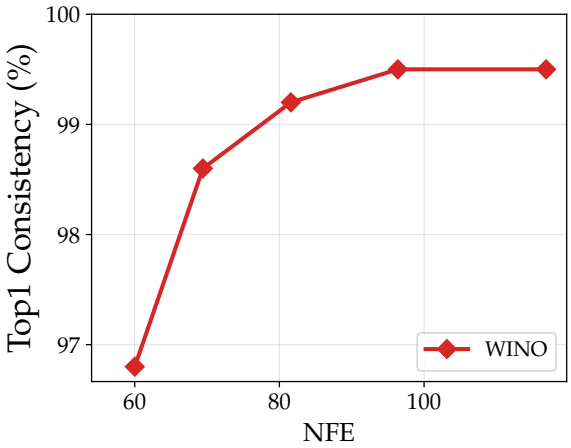


Figure 6. Consistency rate ($\{\arg\max_v q_\theta^{k_s}(v \mid \tilde{\boldsymbol{x}}_t) = \arg\max_v q_\theta^{k}(v \mid \boldsymbol{x}_{t,-k})\}$) of the highest confidence tokens between the shadow and oracle predicted tokens *without* remasking ($\lambda_2 = 0$, LLaDA-8B-Instruct, HumanEval, $BL = 32$). Notably, the consistency rate is very high ($> 97\%$) across all five λ_1 thresholds. Unmasking fewer tokens per step, i.e. increasing the threshold, further improves shadow predictions, potentially due to fewer approximations of KV matrices of other tokens at each step.

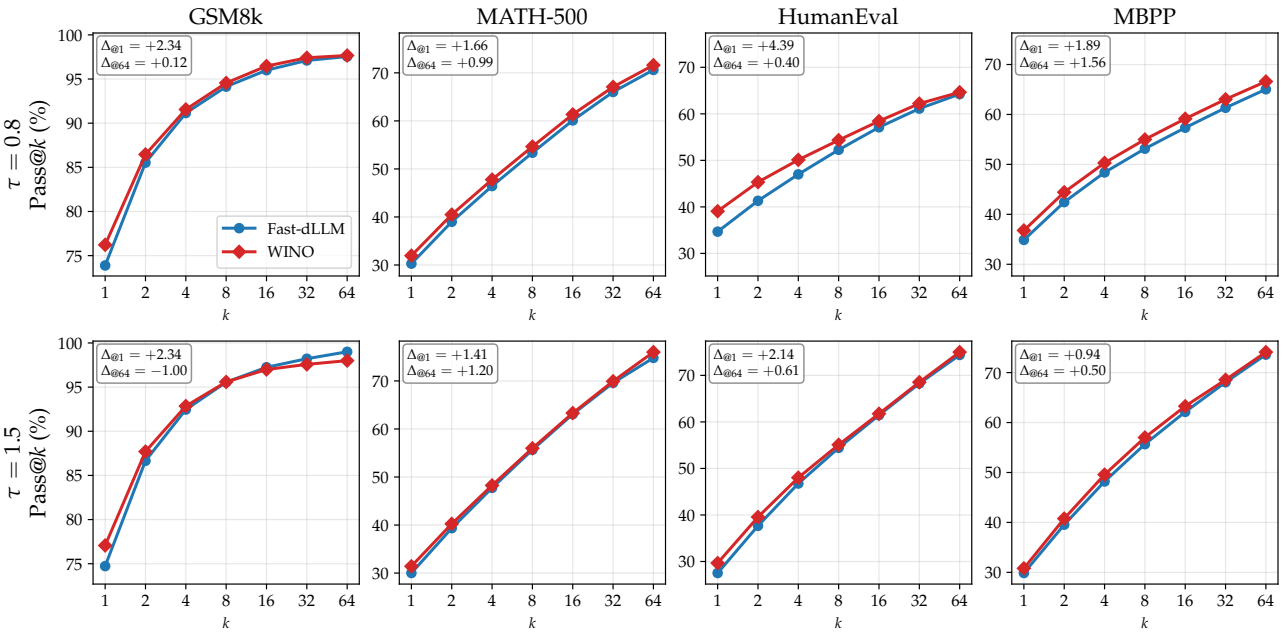


Figure 7. `pass@k` for WINO and Fast-dLLM under non-greedy decoding (LLaDA-8B-Instruct, $BL = 32$, $\lambda_1 = 0.6$, $\lambda_2 = 0.8$) at temperatures $\tau \in \{0.8, 1.5\}$. WINO improves `pass@1` by $\sim$2.6% on average at $\tau = 0.8$, but its advantage diminishes as k grows—shrinking to $\sim$0.7% on average at `pass@64`. This indicates that confidence-based remasking further constrains generation diversity on top of the diversity collapse already documented for confidence-based unmasking (Ni et al., 2026).

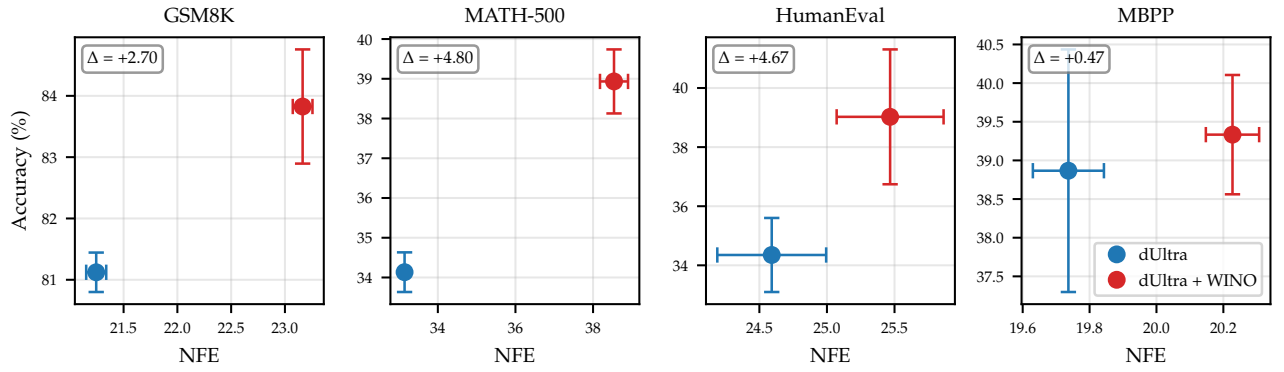


Figure 8. Accuracy and NFEs for dUltra with and without WINO remasking on LLaDA-8B-Instruct ($BL = 32$, $\lambda_2 = 0.8$, $\tau = 0$); means and error bars are over 3 seeds. Adding WINO remasking on top of dUltra's Bernoulli unmasker yields an average accuracy gain of $\sim$3.2% across the four benchmarks at a cost of only $\sim$2 extra NFEs, suggesting that the value of post-hoc remasking depends on the unmasker choice.

C. Additional Experiments

C.1. WINO ablations

To ensure that our findings in Section 3.1 are robust across various design choices of WINO, we perform the following ablations and report results in Figure 9:

- **Consistency-based remasking criteria (*EQUAL*)** Recall that WINO selects positions for remasking via confidence thresholding based on shadow position confidences (Equation 4). To remove the effect of the exact choice of the remasking threshold λ_2, we instead consider here the consistency-based approach where the position is flagged for remasking if the max token under the shadow predictive distribution is not the same as the current token:

$$\mathcal{R}_t := \{k \in \mathcal{B} \setminus \mathcal{M}_t \mid \arg\max_{v \in \mathcal{V}} q_\theta^{k_s}(v \mid \tilde{\boldsymbol{x}}_t) \neq x_t^k\} .$$

- **Loop-Guard (*MR1*)** To measure the impact of the exact loop-guard implementation, we consider here an alternative implementation where each position is allowed to be remasked at most once.
- **Shadow Attention Mask (*DIAG*)** Instead of using a full attention mask for the shadow region as originally proposed in WINO (Figure 2a), we use a diagonal mask which prevents shadow positions from attending to each other.
- **Max-Accept (*MA*)** The original WINO implementation imposes a cap on the number of positions that can be unmasked at each step.[5] Since this cap is neither described in the WINO paper nor part of the Fast-dLLM unmasking procedure (Wu et al., 2025), we disable it for the experiments in the main paper and re-enable it here to study its impact.

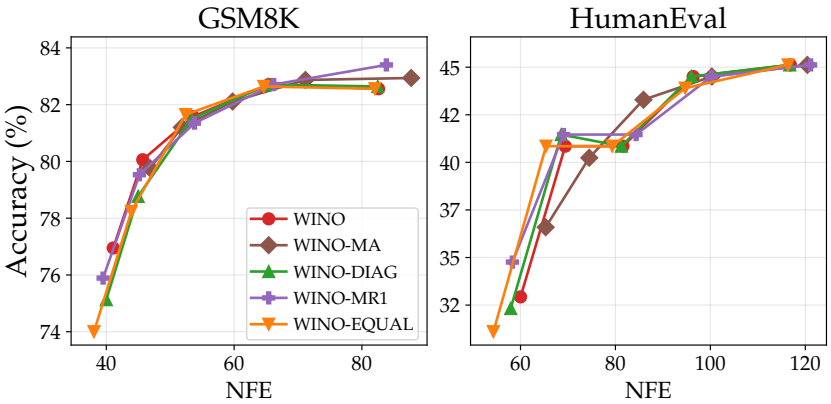


Figure 9. Ablations of four WINO design choices across GSM8k and HumanEval (LLaDA-8B-Instruct, $BL = 32$, $\tau = 0$). The performance across all variants is similar with no variant showing clear performance gains.

C.2. Flip-flop rates: impact of remasking expansions

To better understand the high flip-flop rates observed in Figure 4, we perform additional ablations that vary the set of positions remasked at each sampling step. Beyond the position k flagged by confidence-thresholding (Equation 4), we additionally remask either its *spatial* or *temporal* neighbors. Spatial neighbors are positions at most S tokens to the left or right of k, while temporal neighbors are positions unmasked within T sampling steps starting from when k was unmasked (so $T = 1$ refers to the same step, $T = 2$ to that step and the next, and so on). We denote the resulting variants WINO-SS and WINO-TT. In addition, to ensure convergence, we switch to the aforementioned *MR1* loop-guard, in which each position can be flagged for remasking only once, but can still be remasked as a neighbor of other positions.

As shown in Figure 10, neither expansion method improves over default WINO: across the entire NFE range, both variants consistently *degrade* accuracy, with the gap widening as the neighborhood grows ($T, S = 2$ vs. $T, S = 1$). In addition, Figure 11 shows the flip-flop rate stays roughly the same for expanded remasking ablations. This rules out the cascading-dependency explanation: even when the surrounding context that would supposedly pin the prediction at x_t^k is partially

[5] https://github.com/Feng-Hong/WINO-DLLM/blob/main/LLaDA/decoding.py

removed, the dLLM still re-proposes the same token. The bottleneck therefore appears to lie in the model's predictive distribution at position k itself, rather than in the conditioning structure of the unmasked context.

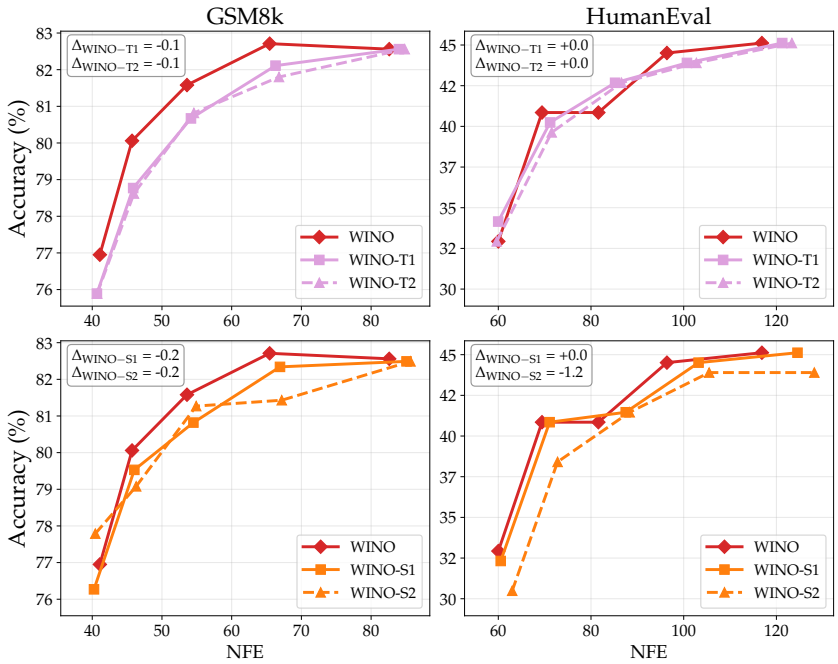


Figure 10. Accuracy of expanded remasking ablations based on additional remasking either spatial or temporal neighbors (LLaDA-8B-Instruct, $BL = 32$).

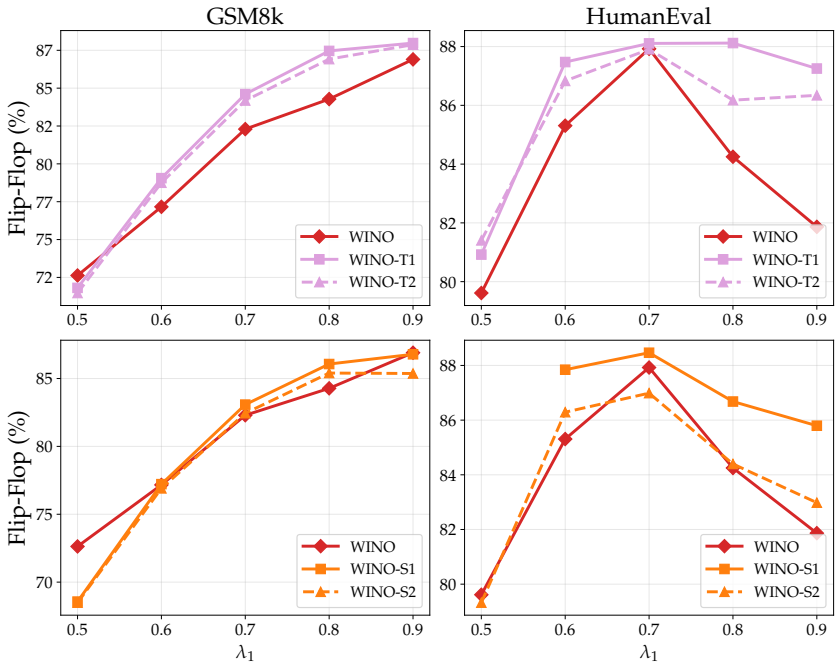


Figure 11. Flip-flop frequencies of expanded remasking ablations based on additional remasking either spatial or temporal neighbors (LLaDA-8B-Instruct, $BL = 32$).