PAIRWISE-SAFE, JOINTLY-UNSAFE: HIGHER-ORDER RESIDUAL RISK IN PARALLEL DECODING OF DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion language models (dLLMs) accelerate generation by committing multiple masked tokens in parallel. Confidence and pairwise dependency estimates, however, need not certify the safety of a complete commit bundle. We characterize the pairwise-safe/jointly-unsafe failure mode and define higher-order residual risk (HOR) as an offline joint-risk proxy unexplained by pairwise risk, confidence, bundle size, and diffusion stage. A lightweight predictor combines pooled hidden states with these scheduler features, enabling a pre-commit gate that accepts, splits, or defers candidate bundles. Across Dream-7B and LLaDA diagnostics, blind-spot bundles constitute 13.9–18.2% of sampled candidate bundles in our Phase-0 diagnostics and exhibit much higher conditional error rates than other bundles. On Dream-7B, across GSM8K, MATH500, HumanEval+, and MBPP+, HOR-aware decoding improves quality over DEMASK by 4.1 percentage points on average, with 8.0% fewer model forward evaluations and 10.3% lower latency. HOR-aware selection also complements within-bundle coordination, providing a practical way to address set-level risk beyond pairwise scheduling.

1 INTRODUCTION

Autoregressive language models decode one token at a time. Masked diffusion language models instead refine partially masked sequences and can commit several positions in a single step (Ho et al., 2020; Austin et al., 2021; Sahoo et al., 2024; Shi et al., 2024; Nie et al., 2025; Ye et al., 2025). More aggressive parallelism reduces decoding depth, but committing tokens predicted from the same incomplete context can discard constraints between positions.

Inference-time schedulers have therefore moved beyond confidence ranking to token order and dependency-aware selection (Kim et al., 2025a; Zhang et al., 2026; Chen et al., 2026; Luo et al., 2026; Kim et al., 2026; Ringel et al., 2026; Zhou et al., 2026). Other work addresses the factorized output distribution itself or coordinates already selected positions (Li et al., 2026; Fang et al., 2026; Yao, 2026). A narrower question remains: can an entire commit set be unsafe even if all available pairwise tests suggest it is safe?

In general, pairwise safety cannot certify joint factorization. A parity construction has pairwise-independent variables obeying a deterministic joint constraint. Rather than claiming a new statistical dependence measure, we isolate an *actionable residual* for dLLM scheduling: how much offline joint-risk evidence remains after accounting for the pairwise and meta-level signals available to a decoder. We call this signal higher-order residual risk (HOR):

$$R_{\mathrm{HOR}}(S) = R_J(S) - \widehat{R}_J^{\mathrm{base}}(S). \tag{1}$$

The HOR gate predicts this residual before accepting a bundle, leaving the upstream pairwise selector intact. It can be composed with CoCommit, which acts *after* set selection. Figure 1 contrasts the prevalence and conditional error rates of blind-spot bundles across two dLLM backbones.

Our contributions are: (i) a quantitative diagnosis of pairwise-safe but jointly-unsafe candidate sets; (ii) an operational residual target and lightweight representation-based predictor; and (iii) a pre-commit scheduling layer that improves the observed quality–efficiency trade-off without changing the backbone weights.

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.

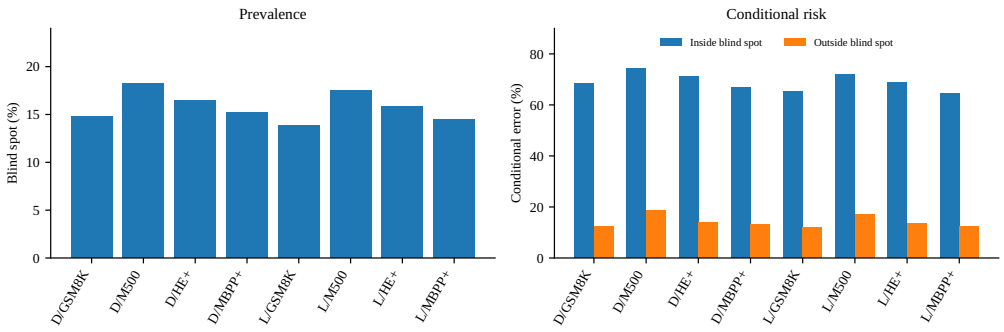


Figure 1: Phase-0 diagnosis across Dream-7B and LLaDA. Pairwise-safe/jointly-unsafe bundles comprise a minority of candidate commits, but have substantially higher conditional error rates than bundles outside the blind spot.

2 Related Work

Masked diffusion and generation order. Discrete diffusion extends denoising to structured symbol spaces (Austin et al., 2021). Modern masked objectives and pretrained dLLMs allow any-order text generation (Sahoo et al., 2024; Shi et al., 2024; Nie et al., 2025; Ye et al., 2025). The order and degree of parallelism affect expressivity and inference efficiency (Kim et al., 2025a; Gwak et al., 2025; Kim et al., 2025b; Svete & Sabharwal, 2026; Zhang et al., 2026; Chen et al., 2026; Zhong et al., 2026; Yeom et al., 2026). ParallelBench provides an information-theoretic and empirical characterization of the speed–quality trade-off induced by parallel decoding and shows that standard math/code benchmarks can underexpose dependency failures (Kang et al., 2026). Our focus is complementary: we diagnose candidate sets that appear safe to a pairwise-aware selector yet retain residual set-level risk, and use that signal online before commitment.

Dependency-aware parallel decoding. DAWN constructs a dependency-aware schedule; DAPD uses attention-based conditional relations; DEMASK estimates pairwise conditional influence; DOS supplies dependency-oriented sampling (Luo et al., 2026; Kim et al., 2026; Ringel et al., 2026; Zhou et al., 2026). KLASS instead uses token-level KL stability for fast masked-diffusion sampling (Kim et al., 2025b). Mean-field parallel decoding coordinates simultaneous commitments through pairwise interactions without an auxiliary predictor (Zoabi et al., 2026). HOR is designed to operate after a dependency-aware selector rather than replace it, and our primary comparison therefore uses DEMASK as the upstream pairwise baseline. The factorization barrier has also motivated coupled or prefix-conditioned token predictors (Li et al., 2026; Fang et al., 2026); CoCommit coordinates an already selected bundle (Yao, 2026). Classic multivariate correlation (Watanabe, 1960) and masked-diffusion parity analyses (Huang & Mirzasoleiman, 2026) motivate the distinction between pairwise and joint structure without establishing that any particular measured regression residual is solely higher-order dependence.

3 Method

3.1 Parallel commitment and offline joint-risk reference

Let $x_t \in (\mathcal{V} \cup \{[\text{MASK}]\})^L$ be a partially denoised sequence, with masked positions M_t. The model emits marginals $p_\theta(x_i \mid x_t)$; a scheduler proposes a bundle $S \subseteq M_t$ and commits under the factorized approximation

$$q_\theta(x_S \mid x_t) = \prod_{i \in S} p_\theta(x_i \mid x_t). \tag{2}$$

For candidate values $\hat{x}_S$, define the parallel and sequential-reveal log-scores

$$\ell_{\mathrm{par}}(S) = \sum_{i \in S} \log p_\theta(\hat{x}_i \mid x_t), \tag{3}$$

$$\ell^{\pi}_{\mathrm{seq}}(S) = \sum_{r=1}^{|S|} \log p_\theta(\hat{x}_{\pi_r} \mid x_t, \hat{x}_{\pi_{<r}}). \tag{4}$$

A computationally expensive *offline proxy*, not the exact full-vocabulary joint divergence, is

$$R_J(S) = \mathbb{E}_{\pi \sim \Pi(S)}\big[|\ell^{\pi}_{\mathrm{seq}}(S) - \ell_{\mathrm{par}}(S)|\big]. \tag{5}$$

For each bundle, we use $\min(8, |S|!)$ distinct permutations. When $|S|! \leq 8$, we use all available orders; otherwise, we sample eight uniformly without replacement. This procedure avoids redundant sequential-reveal evaluations.

3.2 Higher-order residual risk

A set's joint risk also depends on size, confidence and diffusion stage. Let

$$m_S = [R_{\mathrm{pair}}(S), \bar{C}(S), C_{\min}(S), |S|, t] \in \mathbb{R}^5. \tag{6}$$

Throughout the paper, $R_{\mathrm{pair}}(S)$ denotes the cumulative dependency score assigned by the unmodified DEMASK greedy selector to the candidate set S. A separate meta-only baseline regressor f_ψ estimates $\widehat{R}^{\mathrm{base}}_J(S) = f_\psi(m_S)$. We instantiate f_ψ as a two-layer MLP with a 256-unit ReLU hidden layer and scalar output, trained with mean-squared error on the predictor-training partition using AdamW (learning rate 10^{-4}, weight decay 10^{-5}). The fitted f_ψ is frozen before residual targets are formed, leaving the residual

$$R_{\mathrm{HOR}}(S) = R_J(S) - f_\psi(m_S). \tag{7}$$

Positive residual denotes greater offline risk than the baseline expects. Reported mean HOR values are signed residuals under Eq. (7); they are not absolute residual magnitudes. This is an *operational residual*, not a pure information-theoretic higher-order interaction: finite regression capacity, feature omissions, sampling error, and distribution shift can also contribute. We therefore interpret HOR as a scheduler-oriented risk residual rather than an exact decomposition of multivariate information.

Why pairwise safety is insufficient. For $k \geq 3$, take independent $X_1, \ldots, X_{k-1} \sim \mathrm{Bernoulli}(1/2)$ and $X_k = X_1 \oplus \cdots \oplus X_{k-1}$. Each pair is independent, yet the full set is parity constrained. This counterexample establishes the possibility of joint dependence invisible to pairwise statistics, not that natural-language bundles literally form parity systems.

3.3 Lightweight predictor and scheduler

For final-layer hidden states h_i of the candidate positions, use mean pooling $h_S = |S|^{-1} \sum_{i \in S} h_i$ and a two-layer MLP

$$\widehat{R}_{\mathrm{HOR}}(S) = g_\phi(h_S, m_S), \quad \mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} (g_\phi(h_{S_n}, m_{S_n}) - R_{\mathrm{HOR}}(S_n))^2. \tag{8}$$

The predictor is a two-layer MLP with a 256-unit ReLU hidden layer and scalar output. Dream-v0-Base-7B and LLaDA-8B-Base have hidden widths of 3584 and 4096, respectively, giving backbone-dependent input widths after concatenating the five meta features. Backbone-specific predictors are trained separately with the same protocol. Training uses AdamW, learning rate 10^{-4}, weight decay 10^{-5}, up to 30 epochs, and early stopping (patience 5). Development trajectories are generated only from non-test questions: reasoning development uses the official training partitions associated with GSM8K/MATH, while code development uses non-test MBPP training questions; HumanEval+ and MBPP+ test items are reserved exclusively for final evaluation. Within each development pool, trajectories are partitioned 85%/15% for predictor training and validation by original question ID, so every trajectory of a question stays in one partition. The questions used for final decoding accuracy in Table 3 are disjoint from *both* development partitions. Table 2 reports the primary Dream-7B predictor

on its held-out validation partition; the LLaDA predictor is trained separately for cross-backbone diagnostics and scaling analyses. To express predicted joint risk from the residual predictor, we use

$$\widehat{R}_J(S) = f_\psi(m_S) + g_\phi(h_S, m_S). \tag{9}$$

The R^2 column evaluates the reconstructed joint-risk prediction against the offline R_J reference. The feature-ablation models reported in Table 2 are trained as comparison predictors on the same held-out validation partition; the pairwise/meta ablation in that table is not the fixed baseline regressor f_ψ used to construct the residual targets in Eq. (7).

Token-level screening and scheduling. The HOR predictor scores an entire bundle, so we derive position scores without a separate token-risk head. For each $i \in S$, let $C_i = p_\theta(\hat{x}_i \mid x_t)$ and define

$$r_i = \alpha \, \frac{1 - \cos(h_i, h_S)}{2} + (1 - \alpha)(1 - C_i), \quad \alpha = 0.6. \tag{10}$$

Cosine *dissimilarity*, rather than similarity, is used so that a position farther from the pooled bundle representation receives a larger risk score. Both terms lie in $[0, 1]$. Screening uses the fixed threshold $\tau_{\text{token}} = 0.45$: $H = \{i \in S : r_i > \tau_{\text{token}}\}$ and $L = S \setminus H$. If all positions exceed the threshold, the highest-risk $\lceil |S|/2 \rceil$ positions form the deferred group and the remaining, relatively lower-risk positions are processed in the current iteration; this is an explicit progress safeguard, not a claim that those remaining positions are below threshold. If the bundle gate rejects a set but $H = \varnothing$, we use sequential fallback. In the defer-only configuration, masked high-risk positions return to the upstream selector at the next step; when splitting is also enabled, H is retained as a separate candidate sub-bundle for reconsideration. Risk scores are recomputed from the current state when positions are considered again. The rule for a singleton is different: splitting is bypassed, and a rejected singleton is freshly evaluated and committed once, rather than being repeatedly committed or deferred without progress.

The online commit gate uses $\tau_1 = 0.30$ and $\tau_2 = 0.35$; the separate offline blind-spot thresholds are defined in Section 4.2. The online thresholds, together with $\tau_{\text{token}} = 0.45$, are selected once on development validation trajectories and then frozen globally across downstream benchmarks and decoding seeds. The optional CoCommit coordination pass operates on an accepted bundle after the selection decision (Yao, 2026). The conservative sequential fallback recomputes the token conditional after each individual commitment. All dLLM denoising passes incurred by fallback are included in NFE; HOR inference overhead is relevant to the reported end-to-end latency.

4 Experiments

4.1 Evaluation setup

We conduct Phase-0 diagnosis on Dream-7B and LLaDA and primary decoding comparisons on GSM8K (Cobbe et al., 2021), MATH500 (Hendrycks et al., 2021; Lightman et al., 2023), HumanEval+ and MBPP+ (Chen et al., 2021; Liu et al., 2023). We use the 399-task MBPP+ v0.1.0 release; later EvalPlus releases remove broken tasks and therefore contain fewer items. The comparison includes vanilla dLLM, confidence scheduling, DAWN, DAPD, DEMASK, and DEMASK+CoCommit. The primary checkpoint is `Dream-org/Dream-v0-Base-7B`; the diagnostic LLaDA checkpoint is `GSAI-ML/LLaDA-8B-Base`. Experiments use one A100-80GB, PyTorch 2.4.1, Transformers 4.45.2, CUDA 12.1, batch size 16 and maximum generation length 512. Candidate bundles come from DEMASK's greedy selector, followed by online thresholds $\tau_1 = 0.30$ (pairwise risk) and $\tau_2 = 0.35$ (HOR residual risk); decoding seeds are $\{42, 77, 123\}$. The pairwise score is DEMASK's cumulative dependency score for the selected set, used without rescaling. All reported latency values are mean end-to-end wall-clock measurements across the same three decoding seeds under the same hardware, software, batch-size, and maximum-length settings; they include selector, HOR-predictor, fallback, and CoCommit overhead when applicable. We split development trajectories by question ID (85% training, 15% validation) and report final decoding performance only on test questions disjoint from both development partitions. Final evaluation includes all 1,319 GSM8K test questions, all 500 MATH500 problems, all 164 HumanEval+ problems, and all 399 MBPP+ problems; no additional task-specific filtering is applied.

Algorithm 1 HOR-aware parallel commitment

Require: State x_t; candidate bundle S from DEMASK; Split/Defer flags

Form h_S, m_S and predict $\widehat{R}_{\mathrm{HOR}}(S)$
safe $\leftarrow [R_{\mathrm{pair}}(S) < \tau_1] \wedge [\widehat{R}_{\mathrm{HOR}}(S) < \tau_2]$
if $|S| = 1$ **then**
    if safe **then**
        Commit the proposed token
    else
        Re-evaluate its conditional and commit the refreshed token
    end if
else if safe **then**
    Commit the candidate tokens in S in parallel
else if neither Split nor Defer is enabled **then**
    Sequentially re-evaluate and commit positions in S, updating the context each time
else
    Compute r_i by Eq. (10) for each $i \in S$
    $H \leftarrow \{i : r_i > \tau_{\mathrm{token}}\}$; $L \leftarrow S \setminus H$
    if $H = \varnothing$ **then**
        Sequentially re-evaluate and commit positions in S
    else
        if $L = \varnothing$ **then**
            $H \leftarrow$ the $\lceil |S|/2 \rceil$ highest-risk positions; $L \leftarrow S \setminus H$
        end if
        Commit the positions in L using the current model predictions
        if Split enabled **then**
            Retain H as a separate masked candidate sub-bundle for a later step
        else
            Keep H masked for re-selection by the upstream scheduler
        end if
    end if
end if
Include all additional dLLM forward passes in NFE

4.2 PAIRWISE-SAFE BUT JOINTLY-UNSAFE BUNDLES

Define the *offline diagnostic* blind spot by $R_{\mathrm{pair}}(S) < \tau_{\mathrm{pair}} = 0.28$ and $R_J(S) > \tau_J = 0.32$, using the unmodified DEMASK pairwise score. These diagnostic thresholds differ from the online gate thresholds in Section 4.1. Phase-0 diagnosis samples candidate bundles from trajectories on the corresponding benchmark tasks solely for post-hoc diagnostic analysis; these sampled bundles are not used for fitting f_ψ or g_ϕ, early stopping, or online-threshold selection. For the expensive sequential-reveal calculation in Eq. (5), we use up to eight distinct permutations per bundle, i.e., $\min(8, |S|!)$; the resulting collection is a sampled diagnostic set rather than an enumeration of every commit event in the main benchmark runs. Consequently, Blind (%) estimates prevalence under the diagnostic sampling distribution rather than the complete online commit distribution. For the binary diagnostic error marker, each sampled reveal order produces a sequential reference token at the moment a position is revealed; the per-token reference is the modal prediction across sampled orders. If multiple tokens tie for the mode, the prediction from the first sampled reveal order provides a deterministic tie-break. A bundle is marked erroneous if at least one parallel proposed token disagrees with that reference. This is a bundle-level disagreement event, not an incorrect final answer. In contrast, Table 3 grades the completed sequence against task reference answers or the EvalPlus tests. Table 1 reports blind-spot prevalence and conditional bundle-level error rates. Across the eight model/task settings, 13.9–18.2% of sampled candidate bundles fall inside the blind spot, with conditional errors of 64.5–74.2%, versus 11.8–18.5% outside it. The repeated association across backbones motivates a pre-commit set-level risk signal.

Table 1: Phase-0 diagnosis. Error rates are conditional on being inside (In) or outside (Out) the pairwise-safe/jointly-unsafe region; Blind denotes its prevalence among candidate bundles.

Model	Task	Bundles	Blind (%)	In (%)	Out (%)
Dream	GSM8K	12,450	14.8	68.5	12.2
Dream	MATH500	15,200	18.2	74.2	18.5
Dream	HumanEval+	11,800	16.5	71.0	14.0
Dream	MBPP+	11,200	15.2	66.8	13.1
LLaDA	GSM8K	13,100	13.9	65.2	11.8
LLaDA	MATH500	16,000	17.5	72.0	17.2
LLaDA	HumanEval+	12,200	15.8	68.8	13.5
LLaDA	MBPP+	11,600	14.5	64.5	12.5

Table 2: Primary Dream-7B predictor feature comparison on the held-out validation partition. AUROC and AUPRC evaluate bundle-failure ranking; R^2 evaluates reconstructed joint-risk prediction against the offline reference. All feature configurations use the same validation split.

Features	AUROC	AUPRC	R^2
Confidence only	.612	.385	.215
Bundle size only	.584	.342	.182
Pairwise only	.685	.492	.368
Pairwise + conf.	.742	.581	.495
Pairwise + conf. + size + t	.805	.674	.621
Hidden only	.832	.718	.684
Full HOR predictor	.896	.812	.792

4.3 Can residual risk be predicted?

Table 2 compares predictor feature sets on the same question-grouped held-out validation partition. Adding pooled hidden representations to pairwise and meta features increases error-prediction AUROC from 0.805 to 0.896; the full predictor attains 0.792 R^2 for reconstructed joint risk from Equation (9). We emphasize ranking and regression metrics because these directly match the role of the predictor as a pre-commit risk signal.

4.4 Quality–efficiency trade-off

Table 3 compares quality, model forward evaluations (NFE), and end-to-end latency. HOR improves over DEMASK by 3.7–4.5 percentage points across the four tasks; the displayed task-level deltas average to 4.05 points, reported as 4.1 to one decimal. Averaging relative changes over tasks, NFE decreases by 8.0% and latency by 10.3%. The HOR+CoCommit variant further improves output quality. Full baseline comparisons and average commit sizes appear in Appendix A.

4.5 Ablations and the value of residual learning

Table 4 separates the contribution of the residual-risk gate, pooled hidden features, deferral, and splitting. NFE counts full dLLM denoising forward evaluations, including additional full conditional evaluations triggered by sequential fallback commits. CoCommit's auxiliary partial last-layer coordination pass is not counted as one full NFE; its cost is reflected in end-to-end latency. Deferral keeps selected positions masked for later reconsideration; it does not re-mask already committed tokens. Reported NFE values summarize complete decoding trajectories and do not isolate the local computational cost of a single fallback or split operation. Because NFE and average commit size are reported as separate trajectory-level aggregates, $\mathbb{E}[\mathrm{NFE}]\,\mathbb{E}[\mathrm{Avg.Commit}]$ should not be interpreted as total valid output tokens or as a count of touched masked positions. In the defer-only configuration, high-risk positions are left masked for upstream re-selection; in the full configuration, they are retained as a separate sub-bundle as described in Eq. (10) and Algorithm 1. The matched-head direct-regression comparison in Table 5 trains a two-layer 256-hidden-unit MLP on the same pooled-hidden and meta inputs to predict R_J directly, rather than fitting the residual target. The

Table 3: Primary Dream quality and efficiency, averaged over seeds $\{42, 77, 123\}$. GSM8K/MATH500 use accuracy; HumanEval+/MBPP+ use pass@1. Quality is in percent, NFE counts model forward evaluations, and latency is in seconds.

Task	Method	Quality	NFE	Latency
GSM8K	DEMASK	79.4	88.0	1.80
	HOR	83.1	81.2	1.62
	HOR + CoCommit	84.8	79.0	1.65
MATH500	DEMASK	47.8	96.0	2.10
	HOR	52.3	88.5	1.88
	HOR + CoCommit	54.1	86.0	1.90
HumanEval+	DEMASK	66.1	84.5	1.68
	HOR	70.1	77.5	1.50
	HOR + CoCommit	72.0	75.0	1.52
MBPP+	DEMASK	69.8	82.0	1.60
	HOR	73.8	75.2	1.44
	HOR + CoCommit	75.4	73.5	1.46

Table 4: GSM8K scheduling ablation. All non-baseline variants use the HOR gate. Gate-only configurations use single-token fallback on rejected bundles; the complete scheduler also enables defer and split.

Variant	Gate	Hidden	Split	Defer	Acc.	NFE	Sec
DEMASK	–	–	–	–	79.4	88.0	1.80
Meta-only HOR	Y	–	–	–	81.2	85.2	1.72
+ Hidden	Y	Y	–	–	82.1	83.5	1.68
+ Defer	Y	Y	–	Y	82.7	82.0	1.65
+ Split	Y	Y	Y	Y	83.1	81.2	1.62

reported comparison favors the residual formulation on both bundle-error AUROC and reconstructed joint-risk R^2; this establishes a predictor-level comparison, not a mathematical identification of HOR with pure higher-order information.

4.6 Risk scaling and complementary coordination

Figure 2 and Appendix Table 9 show increasing mean HOR and error rates for larger candidate sets. For stagewise analysis we use three disjoint intervals: early $t \in (0.7, 1.0]$, middle $t \in (0.3, 0.7]$, and late $t \in [0, 0.3]$. Bundle-size and stage scaling are reported for GSM8K, MATH500, and HumanEval+; MBPP+ is not part of these scaling experiments. On GSM8K, the wrong-joint-commit rate is 11.2% for DEMASK, 4.1% for HOR, and 2.3% for HOR+CoCommit (Table 8), consistent with selection and within-set coordination addressing different stages of commitment. Wrong-joint percentages in Table 8 are measured on main inference trajectories, whereas Table 1 uses sampled offline diagnostic bundles with multi-permutation R_J evaluation; their aggregate error rates are therefore not directly comparable.

4.7 Limitations

Our sequential-reveal reference is a tractable diagnostic rather than an exact full-vocabulary joint divergence. HOR is an operational signed residual and can reflect baseline model misfit, omitted features, or distribution shift in addition to multivariate effects. Predictor features and risk thresholds can require recalibration for other diffusion backbones or candidate selectors. Phase-0 diagnosis uses sampled bundles rather than every commit event in the main benchmark runs, so its prevalence statistics are conditional on the diagnostic sampling distribution. Our quality benchmarks cover mathematical reasoning and code generation; extension to open-ended generation, ParallelBench, KL-stability samplers such as KLASS, and additional hardware configurations remains outside the present evaluation. Latency is reported from a single A100-80GB environment and should not

Table 5: Same-architecture prediction-head comparison between residual and direct joint-risk targets. AUROC evaluates bundle-error classification and R^2 evaluates offline joint-risk prediction.

Predictor target	AUROC	Joint-risk R^2
Direct R_J	0.863	0.741
HOR residual (ours)	0.896	0.792

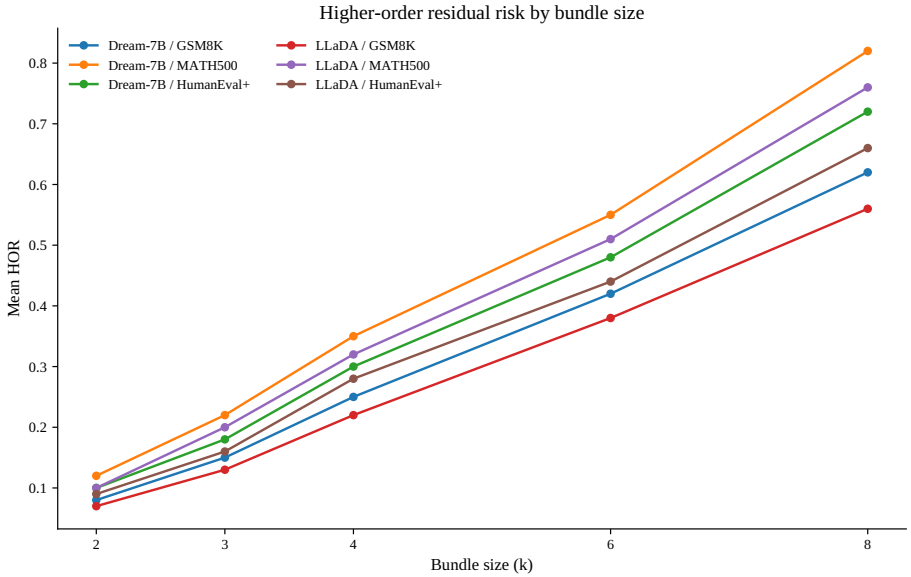


Figure 2: Mean HOR as a function of candidate bundle size on Dream and LLaDA. Lines connect aggregate observations at the evaluated bundle sizes.

be interpreted as hardware-independent speedup; we do not claim kernel-level microbenchmark equivalence across implementations. Predictor feature results use the held-out development validation partition that also determines early stopping, which may mildly favor selected hyperparameters; main task results are reported as averages over three seeds without additional uncertainty intervals.

5 Conclusion

Pairwise indicators alone do not guarantee safe joint commitment in masked diffusion decoding. HOR measures an operational residual from offline sequential-reveal risk, predicts that residual using bundle representations and scheduler features, and uses the estimate to decide whether to commit, split, or defer. Across Dream-7B and LLaDA, the diagnostics identify a recurring pairwise-safe/jointly-unsafe blind spot. On Dream-7B across four reasoning/code benchmarks, HOR-aware decoding improves the quality–efficiency trade-off over the pairwise-aware DEMASK scheduler. The method is complementary to within-set coordination and motivates set-level risk modeling as a component of reliable parallel decoding.

Reproducibility Statement

Section 3 specifies the offline joint-risk reference, residual construction, predictor architecture, token-level risk score, thresholds, and complete scheduling pseudocode. Section 4 reports checkpoints, software versions, hardware, batch size, generation length, data-separation protocol, benchmark versions, random seeds, and evaluation metrics. Appendix A provides the complete baseline table, coordination results, and scaling diagnostics. The test sets used for final task metrics are disjoint from predictor-development questions by construction.

AI USE STATEMENT

Generative AI tools assisted with manuscript restructuring and editing, reconstruction of the LaTeX project from an earlier manuscript PDF, methodological feedback, and preparation of vector figures from tabulated results. The authors take responsibility for the content, claims, numerical results, and citations in the submitted manuscript.

REFERENCES

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

Sitan Chen, Kevin Cong, and Jerry Li. Optimal inference schedules for masked diffusion models. In *Proceedings of the Thirty-Ninth Conference on Learning Theory*, volume 336, pp. 1279–1311. PMLR, 2026. URL https://proceedings.mlr.press/v336/chen26e.html.

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. URL https://arxiv.org/abs/2110.14168.

Xun Fang, Yunchen Li, Hang Yuan, and Zhou Yu. Factorization-error-free discrete diffusion language model via speculative decoding. *arXiv preprint arXiv:2605.14305*, 2026. URL https://arxiv.org/abs/2605.14305.

Daehoon Gwak, Minseo Jung, Junwoo Park, Minho Park, ChaeHun Park, Junha Hyung, and Jaegul Choo. Reward-weighted sampling: Enhancing non-autoregressive characteristics in masked diffusion LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.emnlp-main.1754/.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Jonathan Ho, Ajay N. Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Jianhao Huang and Baharan Mirzasoleiman. Tuning the implicit regularizer of masked diffusion language models: Enhancing generalization via insights from k-parity. *arXiv preprint arXiv:2601.22450*, 2026. URL https://arxiv.org/abs/2601.22450.

Wonjun Kang, Kevin Galim, Seunghyuk Oh, Minjae Lee, Yuchen Zeng, Shuibai Zhang, Coleman Hooper, Yuezhou Hu, Hyung Il Koo, Nam Ik Cho, and Kangwook Lee. Parallelbench: Understanding the trade-offs of parallel decoding in diffusion LLMs. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=OsZr5T7Cd0.

Bumjun Kim, Dongjae Jeon, Moongyu Jeon, and Albert No. Dependency-aware parallel decoding via attention for diffusion LLMs. *arXiv preprint arXiv:2603.12996*, 2026. URL https://arxiv.org/abs/2603.12996.

Jaeyeon Kim, Kulin Shah, Vasilis Kontonis, Sham M. Kakade, and Sitan Chen. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 30749–30768. PMLR, 2025a.

Seo Hyun Kim, Sunwoo Hong, Hojung Jung, Youngrok Park, and Se-Young Yun. KLASS: KL-guided fast inference in masked diffusion models. In *Advances in Neural Information Processing Systems*, 2025b. URL https://arxiv.org/abs/2511.05664.

Ian Li, Zilei Shao, Benjie Wang, Rose Yu, Guy Van den Broeck, and Anji Liu. Breaking the factorization barrier in diffusion language models. *arXiv preprint arXiv:2603.00045*, 2026. URL https://arxiv.org/abs/2603.00045.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023. URL https://arxiv.org/abs/2305.20050.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? rigorous evaluation of large language models for code generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0943.

Lizhuo Luo, Zhuoran Shi, Jiajun Luo, Zhi Wang, Shen Ren, Wenya Wang, and Tianwei Zhang. DAWN: Dependency-aware fast inference for diffusion LLMs. *arXiv preprint arXiv:2602.06953*, 2026. URL https://arxiv.org/abs/2602.06953.

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. URL https://arxiv.org/abs/2502.09992.

Liran Ringel, Ameen Ali, and Yaniv Romano. Dependency-guided parallel decoding in discrete diffusion language models. *arXiv preprint arXiv:2604.02560*, 2026. URL https://arxiv.org/abs/2604.02560.

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*, volume 37, 2024. doi: 10.52202/079017-4135.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3277.

Anej Svete and Ashish Sabharwal. On the reasoning abilities of masked diffusion language models. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=BVnIsh4Nz1. Oral.

Satosi Watanabe. Information theoretical analysis of multivariate correlation. *IBM Journal of Research and Development*, 4(1):66–82, 1960. doi: 10.1147/rd.41.0066.

Lin Yao. Don't commit alone: Joint token commitment in diffusion large language models. *arXiv preprint arXiv:2607.04469*, 2026. URL https://arxiv.org/abs/2607.04469.

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. URL https://arxiv.org/abs/2508.15487.

Jewon Yeom, Jaewon Sok, Seonghyeon Park, Jeongjae Park, Hwiyeong Lee, and Taesup Kim. Answer first, reason later: Commitment order in diffusion LLMs. *arXiv preprint arXiv:2608.05687*, 2026. URL https://arxiv.org/abs/2608.05687.

Shaorong Zhang, Longxuan Yu, Rob Brekelmans, Luhan Tang, Salman Asif, and Greg Ver Steeg. Generation order and parallel decoding in masked diffusion models: An information-theoretic perspective. *arXiv preprint arXiv:2602.00286*, 2026. URL https://arxiv.org/abs/2602.00286.

Yangyang Zhong, Yanmei Gu, Zhengqing Zang, Xiaomeng Li, Yuqi Ding, Xibei Jia, Yuting Shen, Zhenzhong Lan, Liwang Zhu, Weiping Liu, Junlin Zhou, Haisheng Liu, Zhong Xin Yu, Pengxin Luo, Donglian Qi, Yunfeng Yan, and Junbo Zhao. Parallelism and generation order in masked diffusion language models: Limits today, potential tomorrow. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 7178–7218. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.357. URL https://aclanthology.org/2026.findings-acl.357/.

Xueyu Zhou, Yangrong Hu, and Jian Huang. DOS: Dependency-oriented sampler for masked diffusion language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 17404–17419. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.861. URL https://aclanthology.org/2026.findings-acl.861/.

Tamim Zoabi, Ameen Ali, Liran Ringel, and Lior Wolf. Mean-field parallel decoding for discrete diffusion language models. *arXiv preprint arXiv:2606.15805*, 2026. URL https://arxiv.org/abs/2606.15805.

A Additional Experimental Tables and Diagnostics

A.1 Implementation and evaluation protocol

The HOR baseline regressor f_ψ and residual predictor g_ϕ are trained only on development trajectories; f_ψ is frozen before residual targets are generated for g_ϕ. Reasoning development questions come from the official non-test training partitions associated with GSM8K and MATH, while code development uses non-test MBPP training questions. HumanEval+ and MBPP+ test items are not used for predictor fitting or early stopping. We use MBPP+ v0.1.0, the 399-task release. Table 2 reports the primary Dream-7B predictor; the LLaDA predictor is trained independently with the same hyperparameters but its backbone-specific validation metrics are not pooled into Table 2.

The online thresholds τ_1, τ_2, and τ_{token} are fixed after development validation and are shared across downstream test benchmarks and random seeds. $R_{\text{pair}}(S)$ is the cumulative dependency score emitted by the original DEMASK greedy selector for candidate set S. End-to-end latency is averaged over the same three decoding seeds, uses the same A100-80GB environment and batch configuration for every method, and includes auxiliary scheduling/predictor/coordination overhead where applicable. NFE counts full dLLM denoising forward evaluations; CoCommit's auxiliary partial coordination pass is accounted for in latency rather than counted as one full NFE.

Additional diagnostics. The tables below provide the complete backbone/task comparisons, risk scaling summaries, and scheduling ablations.

Table 6: Additional Phase-0 diagnostics, including mean HOR by model and benchmark. Blind and conditional error percentages are repeated for comparison.

Model	Task	Bundles	Blind%	In%	Out%	Mean HOR
Dream	GSM8K	12,450	14.8	68.5	12.2	.42
Dream	MATH500	15,200	18.2	74.2	18.5	.49
Dream	HumanEval+	11,800	16.5	71.0	14.0	.46
Dream	MBPP+	11,200	15.2	66.8	13.1	.43
LLaDA	GSM8K	13,100	13.9	65.2	11.8	.39
LLaDA	MATH500	16,000	17.5	72.0	17.2	.47
LLaDA	HumanEval+	12,200	15.8	68.8	13.5	.44
LLaDA	MBPP+	11,600	14.5	64.5	12.5	.41

A.2 Scaling diagnostics

The bundle-size and diffusion-stage analyses in this subsection use GSM8K, MATH500, and HumanEval+; MBPP+ is not included in these scaling diagnostics.

Table 7: Complete main comparison, averaged over seeds $\{42, 77, 123\}$. NFE counts full dLLM denoising forward evaluations; speedup is the ratio of vanilla to method end-to-end latency; Commit denotes average commit size. CoCommit partial coordination overhead is reflected in latency.

Task	Method	Quality	NFE	Sec	×	Commit
GSM8K	Vanilla	78.2	256	4.82	1.00	1.00
	Confidence	77.8	128.5	2.51	1.92	2.12
	DAWN	78.5	102.4	2.05	2.35	2.65
	DAPD	78.9	94.2	1.91	2.52	2.88
	DEMASK	79.4	88.0	1.80	2.68	3.10
	DEMASK+CoCommit	81.2	86.5	1.88	2.56	3.15
	HOR	83.1	81.2	1.62	2.98	3.38
	HOR+CoCommit	84.8	79.0	1.65	2.92	3.45
MATH500	Vanilla	46.2	256	5.12	1.00	1.00
	Confidence	45.1	135.2	2.80	1.83	1.98
	DAWN	46.5	112.0	2.38	2.15	2.42
	DAPD	47.0	104.5	2.25	2.28	2.60
	DEMASK	47.8	96.0	2.10	2.44	2.85
	DEMASK+CoCommit	49.5	94.2	2.18	2.35	2.90
	HOR	52.3	88.5	1.88	2.72	3.12
	HOR+CoCommit	54.1	86.0	1.90	2.69	3.18
HumanEval+	Vanilla	64.4	256	4.65	1.00	1.00
	Confidence	63.8	122.0	2.32	2.00	2.20
	DAWN	64.8	98.5	1.92	2.42	2.75
	DAPD	65.2	91.0	1.80	2.58	2.98
	DEMASK	66.1	84.5	1.68	2.77	3.22
	DEMASK+CoCommit	67.9	83.0	1.74	2.67	3.28
	HOR	70.1	77.5	1.50	3.10	3.52
	HOR+CoCommit	72.0	75.0	1.52	3.06	3.60
MBPP+	Vanilla	68.2	256	4.50	1.00	1.00
	Confidence	67.5	120.0	2.25	2.00	2.25
	DAWN	68.6	96.0	1.85	2.43	2.80
	DAPD	69.1	88.5	1.72	2.62	3.05
	DEMASK	69.8	82.0	1.60	2.81	3.30
	DEMASK+CoCommit	71.5	80.5	1.66	2.71	3.35
	HOR	73.8	75.2	1.44	3.12	3.62
	HOR+CoCommit	75.4	73.5	1.46	3.08	3.68

Table 8: GSM8K comparison of set selection and within-set coordination. Wrong is the rate of erroneous joint commits (percent).

Selector	Coordination	Quality	NFE	Sec	Wrong (%)
Vanilla	None	78.2	256.0	4.82	14.8
DEMASK	None	79.4	88.0	1.80	11.2
Vanilla	CoCommit	80.1	248.0	4.88	10.5
DEMASK	CoCommit	81.2	86.5	1.88	8.2
HOR	None	83.1	81.2	1.62	4.1
HOR	CoCommit	84.8	79.0	1.65	2.3

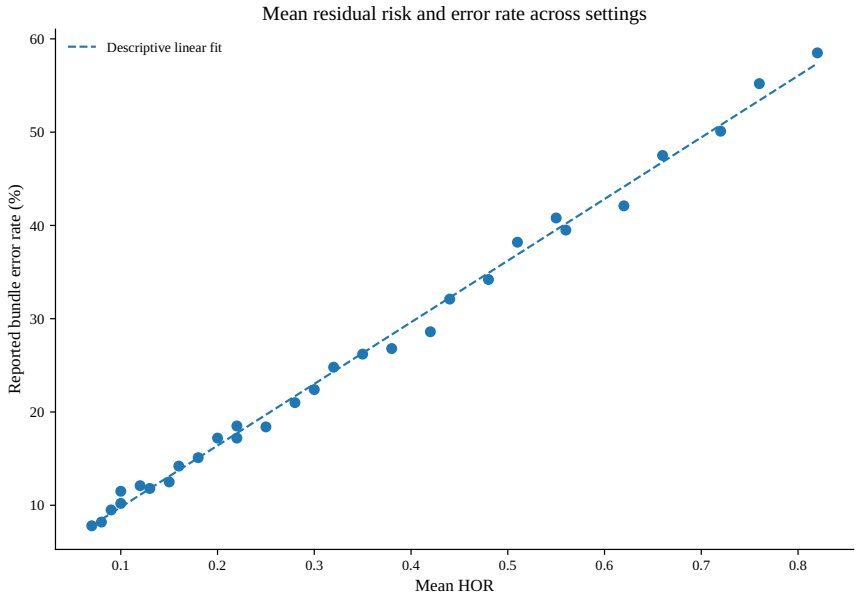


Figure 3: Association between average HOR and error rate for 30 model/task/bundle-size aggregates. The fitted line summarizes aggregate association rather than token-level calibration.

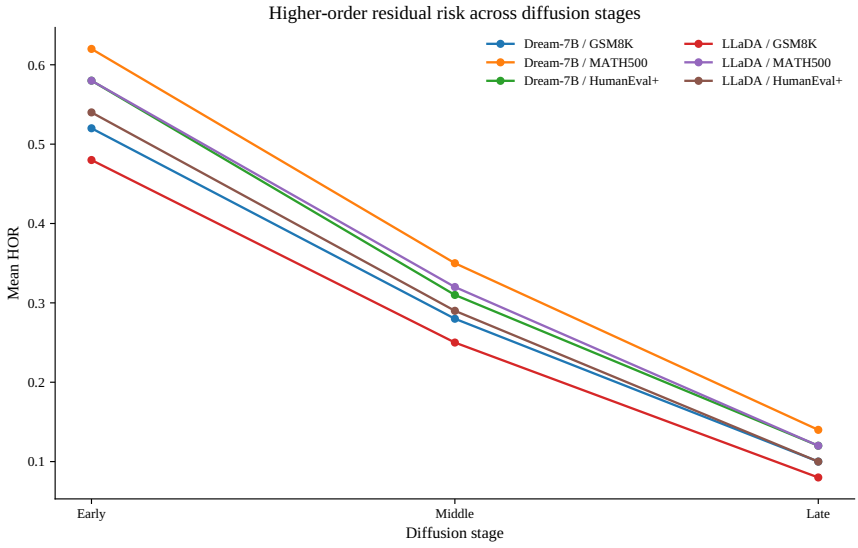


Figure 4: Mean HOR by denoising stage across model/task settings.

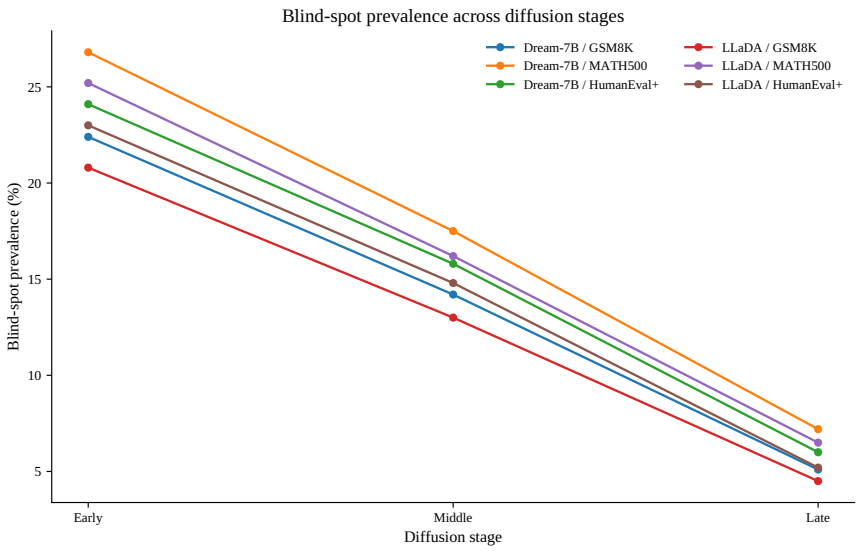


Figure 5: Blind-spot prevalence by denoising stage across model/task settings.

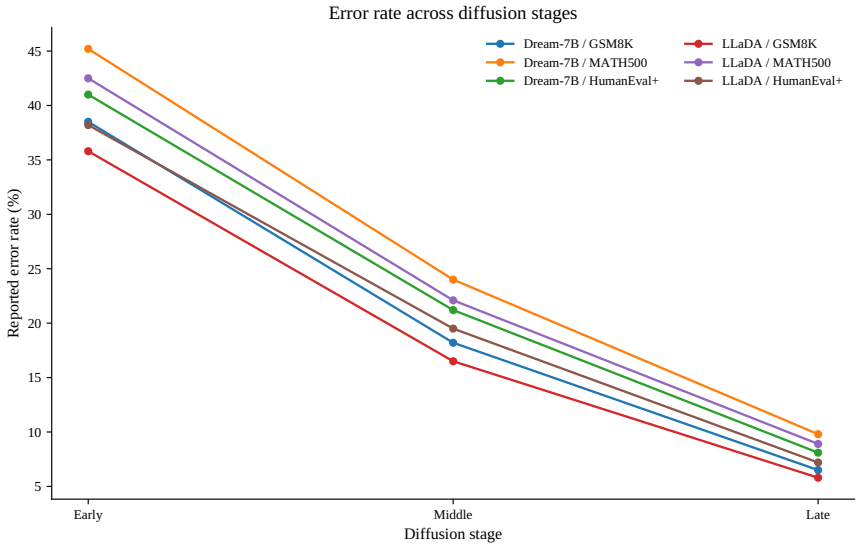


Figure 6: Error rate across denoising stages across model/task settings.

Table 9: Bundle-size scaling for average pairwise risk, joint risk, HOR, and error rate.

Model	Task	k	Pair	Joint	HOR	Err%
Dream	GSM8K	2	.12	.16	.08	8.2
		3	.18	.28	.15	12.5
		4	.25	.42	.25	18.4
		6	.38	.68	.42	28.6
		8	.52	.92	.62	42.1
Dream	MATH500	2	.15	.22	.12	12.1
		3	.22	.38	.22	18.5
		4	.30	.55	.35	26.2
		6	.45	.85	.55	40.8
		8	.60	1.15	.82	58.5
Dream	HumanEval+	2	.14	.20	.10	10.2
		3	.20	.32	.18	15.1
		4	.28	.48	.30	22.4
		6	.42	.75	.48	34.2
		8	.55	1.02	.72	50.1
LLaDA	GSM8K	2	.11	.15	.07	7.8
		3	.17	.26	.13	11.8
		4	.24	.39	.22	17.2
		6	.36	.63	.38	26.8
		8	.50	.86	.56	39.5
LLaDA	MATH500	2	.14	.20	.10	11.5
		3	.21	.35	.20	17.2
		4	.29	.51	.32	24.8
		6	.43	.80	.51	38.2
		8	.58	1.08	.76	55.2
LLaDA	HumanEval+	2	.13	.18	.09	9.5
		3	.19	.30	.16	14.2
		4	.27	.45	.28	21.0
		6	.40	.70	.44	32.1
		8	.53	.95	.66	47.5

Table 10: Diffusion-stage scaling using disjoint intervals: early $t \in (0.7, 1.0]$, middle $t \in (0.3, 0.7]$, and late $t \in [0, 0.3]$.

Model	Task	Stage	HOR	Blind%	Err%
Dream	GSM8K	Early	.52	22.4	38.5
		Middle	.28	14.2	18.2
		Late	.10	5.1	6.5
Dream	MATH500	Early	.62	26.8	45.2
		Middle	.35	17.5	24.0
		Late	.14	7.2	9.8
Dream	HumanEval+	Early	.58	24.1	41.0
		Middle	.31	15.8	21.2
		Late	.12	6.0	8.1
LLaDA	GSM8K	Early	.48	20.8	35.8
		Middle	.25	13.0	16.5
		Late	.08	4.5	5.8
LLaDA	MATH500	Early	.58	25.2	42.5
		Middle	.32	16.2	22.1
		Late	.12	6.5	8.9
LLaDA	HumanEval+	Early	.54	23.0	38.2
		Middle	.29	14.8	19.5
		Late	.10	5.2	7.2

B PROOF OF PAIRWISE-INDEPENDENCE COUNTEREXAMPLE

For $k \geq 3$, let $X_1, \ldots, X_{k-1}$ be i.i.d. fair Bernoulli variables, and set $X_k = \oplus_{j=1}^{k-1} X_j$. Each pair has the uniform distribution on $\{0,1\}^2$, but only 2^{k-1} of the 2^k full assignments occur, so $p(x_1, \ldots, x_k) \neq \prod_i p(x_i)$.