UNCERTAINTY SHAPING FOR RISK-AWARE TOKEN FIXATION IN DISCRETE DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Discrete diffusion language models (dLLMs) enable parallel generation, but their quality and efficiency depend on which masked tokens are committed at each denoising step: fixing uncertain predictions too early can propagate errors. We study token fixation from a risk-aware perspective and derive a local robustness criterion favoring low-entropy-first fixation under bounded model mismatch and propagation risk. Rather than merely using uncertainty as a decoding score, our framework jointly shapes it during training and exploits it at inference. Differential corruption and structure-aware loss reweighting over a deterministic structural partition encourage structure-associated uncertainty separation, while a dynamic entropy-gated decoder prioritizes low-entropy predictions with per-step fixation constraints. The framework requires no architectural modification or additional model forward pass within a denoising step. On GSM8K, the full framework achieves a **23.5%** relative EM improvement and a **9.28**× speedup over the LLaDA baseline. Isolating the inference-time component, entropy-gated decoding alone yields a **6.1%** relative EM improvement with a **5.64**× speedup. Matched controls distinguish entropy ranking from maximum-probability confidence and meaningful structural partitions from length-matched random spans, while cross-task evaluations show improvements beyond GSM8K. Code is anonymously available at `https://anonymous.4open.science/r/EATO-LLM-4DDD/`.

1 INTRODUCTION

Large language models (LLMs) have achieved substantial progress in natural language understanding and generation (Zhao et al., 2023; Brown et al., 2020; Chowdhery et al., 2022). However, autoregressive models (Touvron et al., 2023; Vaswani et al., 2023) generate tokens sequentially, causing inference latency to increase with output length (Pope et al., 2022; Khoshnoodi et al., 2024). Discrete diffusion language models (dLLMs) (Austin et al., 2023; Sahoo et al., 2024) provide an alternative by iteratively predicting multiple masked positions in parallel. This parallelism introduces an additional decoding problem: determining which predictions should be committed earlier during denoising.

Token fixation directly affects subsequent generation because committed predictions become part of the context for later denoising steps. Prematurely fixing uncertain predictions can propagate errors, whereas overly conservative fixation reduces the efficiency advantage of parallel decoding. Predictive entropy provides a natural measure of model uncertainty (Gal & Ghahramani, 2016), and recent work has begun to exploit uncertainty- or information-based signals for adaptive token selection (Yang et al., 2026). We therefore do not treat entropy itself as the contribution. Instead, we formulate token fixation as a risk-sensitive ordering problem and derive, under bounded model mismatch and propagation risk, a local robustness criterion favoring low-entropy-first fixation. In this formulation, entropy serves as a proxy for fixation risk rather than a guarantee of semantic correctness.

A further limitation of inference-only ranking is that the uncertainty used for token selection is inherited from a fixed predictive landscape. We therefore couple training-time uncertainty shaping with inference-time risk-aware fixation. During training, a deterministic structural partition of the original response tokens is used to assign different corruption rates and loss weights. Differential corruption and structure-aware loss reweighting encourage structure-associated separation in predictive uncertainty. During inference, a dynamic entropy-gated decoder prioritizes low-entropy predictions using a time-dependent threshold together with minimum and maximum fixation constraints. The

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.

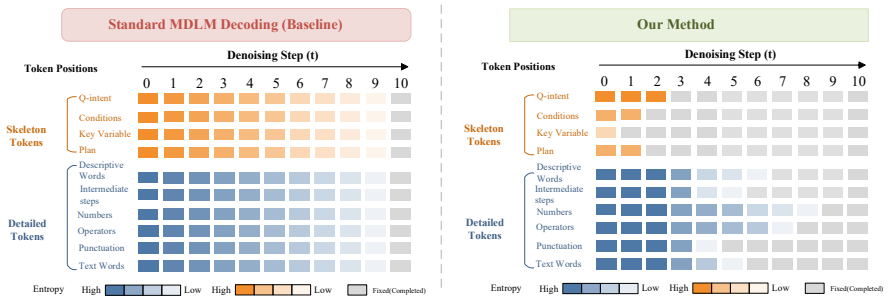

Figure 1: Comparison of token-fixation trajectories under the baseline decoder and the entropy-gated decoder. **Left:** baseline decoding, where structural and detail tokens are finalized in a relatively interleaved manner. **Right:** entropy-gated decoding, where lower-entropy structural tokens tend to be fixed earlier. Orange and blue denote structural and detail tokens, respectively; darker shading indicates higher token-level entropy, and gray denotes finalized tokens.

resulting decoder requires no structural annotations at inference time, no auxiliary policy or reward model, and no additional model forward pass within a denoising step. Figure 1 illustrates the resulting difference in token-fixation trajectories between baseline decoding and entropy-gated fixation. Our contributions are summarized as follows:

- We formulate token fixation in dLLM decoding as a risk-sensitive ordering problem and derive a local robustness criterion favoring low-entropy-first fixation under model mismatch and propagation risk.
- We introduce a coupled framework for training-time uncertainty shaping and inference-time risk-aware fixation. Differential corruption and structure-aware loss reweighting modify the predictive uncertainty associated with a deterministic structural partition, while dynamic entropy gating exploits this uncertainty through per-step fixation constraints.
- We conduct controlled evaluations that separate the effects of entropy ranking, structural partitioning, training-time uncertainty shaping, and fixation constraints. On GSM8K, entropy-gated decoding alone yields a **6.1%** relative EM improvement with a **5.64**$\times$ speedup over the original LLaDA decoder, while the full framework achieves a **23.5%** relative EM improvement and a **9.28**$\times$ speedup. Additional scientific and commonsense reasoning benchmarks show improvements beyond GSM8K.

2 Related Work

Discrete Diffusion Language Models. Diffusion models have been extended to discrete text generation as an alternative to sequential autoregressive decoding. Early continuous-diffusion approaches (Li et al., 2022) motivated discrete-state formulations such as D3PM (Austin et al., 2023) and SEDD (Lou et al., 2024). Recent work scales masked diffusion through MDLM (Sahoo et al., 2024) and LLaDA (Nie et al., 2025; Zhu et al., 2025), or combines diffusion with block-wise generation (Arriola et al., 2025). Their flexible generation order motivates explicit study of token-selection and commitment strategies.

Generation Order and Uncertainty-Guided Decoding. Recent work studies adaptive token selection during diffusion decoding. EB-Sampler uses entropy-bounded unmasking (Ben-Hamu et al., 2025); denoising-entropy methods optimize decoding paths using accumulated uncertainty (Chen et al., 2025); and Info-Gain accounts for the downstream effect of current decisions on remaining uncertainty (Yang et al., 2026). Other work analyzes generation-order sensitivity and confidence-based decoding (Zhang et al., 2026; Cai & Li, 2026), exploits cross-step prediction stability (Wang et al., 2026), or learns ordering policies and reward-guided scores (Mohamud et al., 2026; Bu et al., 2026). Our work therefore does not treat entropy or token ordering alone as novel. Instead, we study how the predictive uncertainty used for fixation can itself be shaped during training, and couple this training-time intervention with risk-aware low-entropy fixation at inference.

Training-Time Uncertainty Shaping and Structural Priors. Explicit reasoning and hierarchical generation have long exploited intermediate structure, including Chain-of-Thought (Wei et al., 2023), Plan-and-Solve (Wang et al., 2023), Skeleton-of-Thought (Ning et al., 2024), and tree-based generation (Fan et al., 2018; Yao et al., 2023). In our framework, a deterministic structural partition serves only as a task-dependent training prior and is not required at inference. Related diffusion work modifies corruption according to token properties (He et al., 2022; Chen et al., 2023), incorporates token importance into training (Wang et al., 2025), or addresses train–inference mismatch (He et al., 2025). Our method does not treat any of these ingredients in isolation as the contribution; rather, differential corruption and structure-aware loss reweighting are used to reshape the predictive uncertainty that is subsequently exploited by the risk-aware fixation rule.

3 PRELIMINARIES

Masked Diffusion Language Models. Given a token sequence $x_0 = (x^1, \ldots, x^L)$, masked diffusion language models progressively corrupt tokens into a special `[MASK]` token according to a noise schedule α_t (Austin et al., 2023; Sahoo et al., 2024), and learn a reverse process to recover the original sequence. At inference, generation starts from a fully masked response region and proceeds through iterative denoising. Parallel decoding commonly approximates the reverse transition as

$$p_\theta(x_{t-1} \mid x_t) \approx \prod_{i \in \mathcal{M}_t} p_\theta(x^i_{t-1} \mid x_t), \tag{1}$$

where $\mathcal{M}_t$ denotes the masked positions at step t. This approximation enables multiple tokens to be updated in parallel, while making the choice of which tokens to fix at each step consequential for subsequent predictions. Our goal is therefore to study the denoising order under this parallel-decoding setting.

Predictive Entropy as an Ordering Signal. For each masked position i at step t, the model predicts a distribution $p_\theta(x^i \mid x_t)$ with entropy

$$H_i(t) = -\sum_k p_\theta(x^i = k \mid x_t) \log p_\theta(x^i = k \mid x_t). \tag{2}$$

Predictive entropy measures the model's instantaneous uncertainty over a token (Shannon, 1948; Badger & Neligeorge, 2025). We use it as an intrinsic signal for determining which masked positions to fix during denoising, with the corresponding risk-aware motivation developed in Section 4.

Task Setting. We consider conditional generation, where a response y is generated given a prompt c, and write the full sequence as $x = [c; y]$. The prompt remains unmasked throughout generation, while only the response region participates in denoising. All masking and entropy statistics are computed over the response region.

4 THEORETICAL FRAMEWORK

We analyze token fixation from three perspectives. Section 4.1 provides an error-free information-gain reference, Section 4.2 derives a local risk-aware criterion for practical decoding, and Section 4.3 isolates the effect of differential corruption on conditional uncertainty. All distributions are implicitly conditioned on the input prompt c.

4.1 ORACLE INFORMATION-GAIN REFERENCE

Let A denote the revealed token positions and $\mathcal{R} = \{1, \ldots, N\} \setminus A$ the remaining masked positions. If an oracle reveals the true value of X_i, its expected information gain under the realized context $X_A = x_A$ is

$$\Delta_{\text{oracle}}(i \mid x_A) = H(X_{\mathcal{R}} \mid X_A = x_A) - H(X_{\mathcal{R}\setminus\{i\}} \mid X_A = x_A, X_i) = H(X_i \mid X_A = x_A). \tag{3}$$

For an ordering $\pi = (i_1, \ldots, i_N)$, with $A_{k-1} = \{i_1, \ldots, i_{k-1}\}$, define

$$\Gamma_K(\pi) = \sum_{k=1}^{K} H(X_{i_k} \mid X_{A_{k-1}}). \tag{4}$$

For $K = N$, $\Gamma_N(\pi) = H(X_{1:N})$ for every ordering, so ordering only affects the truncated objective.

Assume that the unrevealed variables are conditionally independent given every realized context:

$$P(X_{\mathcal{R}} \mid X_A = x_A) = \prod_{i \in \mathcal{R}} P(X_i \mid X_A = x_A). \tag{5}$$

Theorem 1 (Oracle Greedy Optimality)**.** *Under this assumption, selecting at each step the unrevealed position with largest conditional entropy maximizes $\Gamma_K(\pi)$ for every $K \in \{1, \ldots, N\}$.*

Conditional independence makes revealing one token leave the conditional entropy of the others unchanged; the truncated objective therefore reduces to selecting the K largest-entropy positions. The proof is given in Appendix A.1.

4.2 Robust Decoding under Predictive Risk

Practical fixation is based on predictions that may be incorrect. At a fixed decoding context, let

$$\hat{x}_i = \arg\max_k p_i(k), \qquad p_i(k) = p_\theta(X_i = k \mid x_A), \tag{6}$$

with predictive entropy $H_i = H(\mathbf{p}_i)$. Let q_i denote the unknown true conditional distribution and $\epsilon_i = 1 - q_i(\hat{x}_i)$ the commitment error rate.

Assume bounded model mismatch,

$$D_{\mathrm{KL}}(q_i \| \mathbf{p}_i) \le \kappa, \tag{7}$$

and let $\Delta > 1$ upper-bound the downstream penalty associated with an incorrect commitment.

Lemma 2 (Error-Rate Upper Bound). Under the bounded model-mismatch assumption in Eq. (7), the commitment error rate at position i satisfies

$$\epsilon_i \le 1 - e^{-H_i} + \sqrt{\frac{\kappa}{2}}.$$

This follows from Pinsker's inequality and $p_i^{\max} \ge e^{-H_i}$.

Define the worst-case risk-adjusted margin

$$\Psi(i) = \inf_{q_i : D_{\mathrm{KL}}(q_i \| \mathbf{p}_i) \le \kappa} \left[H(q_i) - \Delta \epsilon_i \right]. \tag{8}$$

Entropy continuity and Lemma 2 yield

$$\Psi(i) \ge L(H_i) = H_i - \Delta(1 - e^{-H_i}) - C_{\kappa,\mathcal{V}}, \tag{9}$$

where $C_{\kappa,\mathcal{V}}$ is independent of the candidate position. The explicit bound is given in Appendix A.2.

Define the local candidate set

$$\mathcal{R}_\Delta = \{i \in \mathcal{R} : H_i < \log \Delta\}. \tag{10}$$

Theorem 2 (Local Robustness Certificate for Low-Entropy-First)**.** *For $i \in \mathcal{R}_\Delta$, $L(H_i)$ is strictly decreasing in predictive entropy. Hence, if $\mathcal{R}_\Delta \neq \emptyset$,*

$$i^*_{\mathrm{robust}} \in \arg\min_{i \in \mathcal{R}_\Delta} H_i$$

maximizes the derived robustness lower bound within this regime.

Indeed,

$$L'(H) = 1 - \Delta e^{-H} < 0 \qquad \text{for } H < \log \Delta. \tag{11}$$

Thus, the result motivates low-entropy-first fixation locally; it does not claim global optimality of the exact margin $\Psi(i)$.

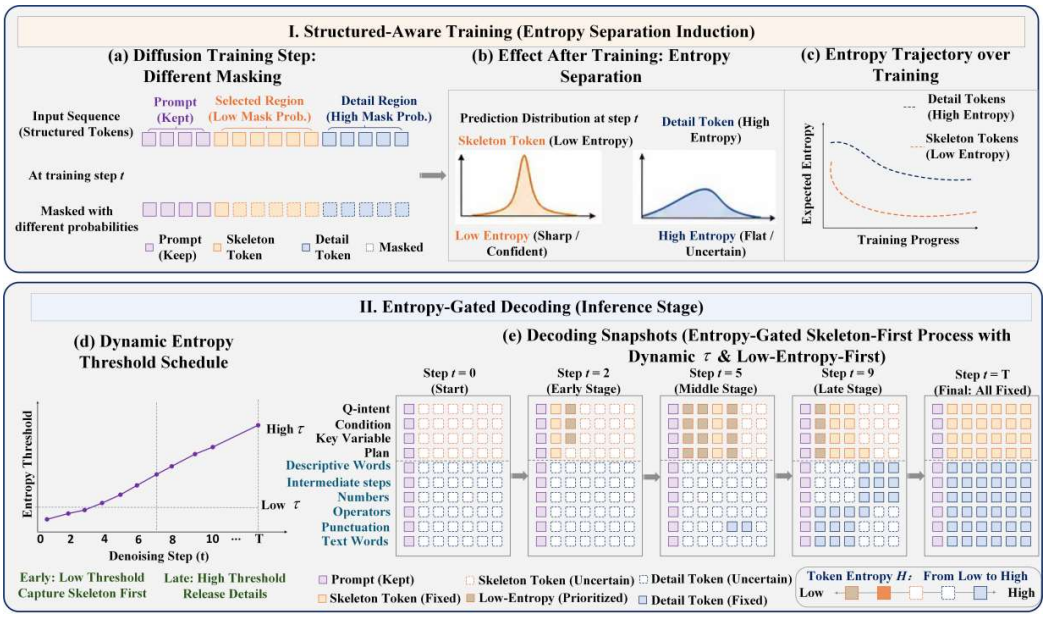

Figure 2: Overview of the proposed framework. (a) Region-dependent corruption during training. (b–c) The resulting separation and evolution of predictive entropy between structural and detail tokens. (d) Dynamic entropy-threshold schedule. (e) Entropy-gated decoding, which prioritizes low-entropy masked positions under per-step fixation constraints.

4.3 Differential Corruption and Conditional Entropy

The preceding analysis treats predictive uncertainty as given. We next isolate how differential corruption can affect conditional uncertainty during training. Consider the deterministic structural/detail partition $\mathcal{S} \cup \mathcal{T}$ defined in Section 5.1. For the same underlying token–context distribution, compare two masking probabilities

$$0 \le p_{\mathcal{S}} < p_{\mathcal{T}} < 1, \tag{12}$$

and let Z_p denote the corresponding masked observation.

Proposition 1 (Conditional-Entropy Monotonicity under Masking). *Under this matched-target setting,*

$$H(X \mid Z_{p_{\mathcal{S}}}) \le H(X \mid Z_{p_{\mathcal{T}}}).$$

Because $Z_{p_{\mathcal{T}}}$ can be obtained by applying additional independent masking to $Z_{p_{\mathcal{S}}}$,

$$X \to Z_{p_{\mathcal{S}}} \to Z_{p_{\mathcal{T}}}$$

forms a Markov chain, and the result follows from the data-processing inequality. The full derivation is given in Appendix A.3. This proposition isolates the effect of corruption rate while holding the underlying target distribution fixed. It does not imply a pointwise entropy ordering between structural and detail tokens in actual decoding trajectories. In the complete objective, differential corruption and structure-aware loss reweighting jointly shape the predictive uncertainty used at inference.

5 Method

Guided by the analyses in Section 4, our framework combines training-time uncertainty shaping with inference-time risk-aware fixation, as illustrated in Figure 2.

5.1 Identification of Structural Tokens

We apply a fixed deterministic regex-based rule to identify step-level structural spans in the original responses. Matched tokens form the structural set $\mathcal{S}$, while the remaining response tokens form

the detail set $\mathcal{T}$. This partition only derives token-level metadata without modifying the original training examples or target responses, and introduces no additional synthetic data. It serves as a task-dependent structural prior for region-dependent corruption and loss weighting, rather than a universal semantic decomposition. Neither structural annotations nor the partition are required at inference time.

5.2 Training-Time Uncertainty Shaping

Let the global masking probability at diffusion time t be

$$p_{\text{global}}(t) = 1 - \alpha_t. \tag{13}$$

We assign region-dependent masking probabilities:

$$p_{\text{mask}}(i,t) = \begin{cases} \max(p_{\text{global}}(t)s_{\mathcal{S}}, p_{\text{floor}}), & i \in \mathcal{S}, \\ \min(p_{\text{global}}(t)s_{\mathcal{T}}, 1), & i \in \mathcal{T}, \end{cases} \tag{14}$$

where $s_{\mathcal{S}} < 1$, $s_{\mathcal{T}} > 1$, and p_{floor} is a minimum masking probability for structural tokens. The floor maintains reconstruction supervision for structural positions when the global corruption level is low. The matched-target analysis in Section 4.3 isolates the effect of differential corruption when the resulting rates satisfy $p_{\mathcal{S}} < p_{\mathcal{T}}$.

For the masked response positions $\mathcal{M}$, let

$$\ell_i = -\log p_\theta(x_0^i \mid \mathbf{x}_t). \tag{15}$$

We optimize the normalized region-weighted reconstruction objective

$$\mathcal{L} = \frac{1}{\sum_{i\in\mathcal{M}} w_i} \sum_{i\in\mathcal{M}} w_i \ell_i, \qquad w_i = \begin{cases} \lambda, & i \in \mathcal{S}, \\ 1, & i \in \mathcal{T}, \end{cases} \tag{16}$$

where $\lambda > 1$. The normalization keeps the overall loss scale comparable across different values of λ. Prompt tokens remain unmasked and are excluded from the reconstruction loss. Together, differential corruption and structure-aware loss reweighting are designed to shape the predictive uncertainty subsequently used by the decoder.

5.3 Risk-Aware Entropy-Gated Decoder

Motivated by the local robustness criterion in Theorem 2, the decoder prioritizes low-entropy predictions while limiting the number of simultaneous commitments. Let $r \in \{0, \ldots, T-1\}$ denote the decoding iteration and $\mathbf{x}^{(r)}$ the current partially denoised sequence. A single model forward pass produces

$$H_i^{(r)} = -\sum_k p_\theta(x^i = k \mid \mathbf{x}^{(r)}) \log p_\theta(x^i = k \mid \mathbf{x}^{(r)}) \tag{17}$$

for each currently masked position $i \in \mathcal{M}_r$.

We use a linearly increasing entropy threshold

$$\tau_r = \tau_{\text{start}} + \frac{r}{T-1}(\tau_{\text{end}} - \tau_{\text{start}}), \tag{18}$$

and define the candidate set

$$\mathcal{C}_r = \left\{ i \in \mathcal{M}_r : H_i^{(r)} < \tau_r \right\}. \tag{19}$$

The maximum and minimum fixation budgets are

$$N_{\max} = \max(\lfloor |\mathcal{M}_r|\eta \rfloor, 1), \qquad N_{\min} = 1, \tag{20}$$

where $\eta \in (0,1)$ is the maximum fixation ratio. The upper budget limits aggressive parallel commitments, whereas the minimum budget prevents decoding from stalling.

Let $\mathrm{LowEnt}_m(\mathcal{A})$ denote the m positions in $\mathcal{A}$ with the lowest predictive entropy. The fixation set is

$$\mathcal{F}_r = \begin{cases} \mathrm{LowEnt}_{N_{\max}}(\mathcal{C}_r), & |\mathcal{C}_r| > N_{\max}, \\ \mathcal{C}_r, & N_{\min} \le |\mathcal{C}_r| \le N_{\max}, \\ \mathcal{C}_r \cup \mathrm{LowEnt}_{N_{\min}-|\mathcal{C}_r|}(\mathcal{M}_r \setminus \mathcal{C}_r), & |\mathcal{C}_r| < N_{\min}. \end{cases} \tag{21}$$

Selected positions are committed using

$$\hat{x}^i = \arg\max_k p_\theta(x^i = k \mid \mathbf{x}^{(r)}), \qquad i \in \mathcal{F}_r, \tag{22}$$

and remain fixed in subsequent iterations. Thus, the decoder commits sufficiently low-entropy predictions when available, caps the number of simultaneous commitments, and otherwise fixes the lowest-entropy position to ensure progress. Because entropy is computed from the same predictive distribution used for token prediction, the procedure requires no additional model forward pass within a denoising iteration.

6 Experiments

We organize the experiments to evaluate the effectiveness and underlying mechanisms of our framework. We first isolate the contributions of inference-time risk-aware fixation and training-time uncertainty shaping. We then examine how training reshapes predictive uncertainty and influences subsequent fixation dynamics. Finally, we evaluate cross-task transfer beyond GSM8K and assess the role of minimum and maximum fixation constraints. Additional diagnostic and robustness analyses are provided in Appendix C, with implementation details in Appendix D.

6.1 Main Results

Table 1: Main results on GSM8K (Cobbe et al., 2021) using LLaDA-8B (Nie et al., 2025).

Model & Decoding Strategy	EM (%)	Inference Time(s)
LLaDA-8B Base + Original Decoder	66.0	227.4
LLaDA-8B Base + Entropy-Gated Decoder	70.0	40.3
Structure-Aware FT ($\lambda = 2$) + Entropy-Gated Decoder	**81.5**	**24.5**

Table 1 reports EM accuracy and inference time on GSM8K (Cobbe et al., 2021) using LLaDA-8B (Nie et al., 2025). For all main evaluations, we set `block_length` = `gen_length` = 128 with 128 denoising steps, so the original LLaDA baseline and our decoder operate over the same full generation region. Keeping the pretrained model fixed, entropy-gated decoding improves EM from 66.0 to 70.0 and reduces inference time from 227.4 to 40.3, corresponding to a 6.1% relative improvement and a 5.64× speedup. Combining entropy-gated decoding with structure-aware fine-tuning further improves EM to 81.5 and reduces inference time to 24.5. Overall, the full framework achieves a 23.5% relative EM improvement and a 9.28× speedup over the original baseline.

6.2 Analysis of Learned Uncertainty Dynamics

Entropy Trajectory Analysis. We examine whether the proposed training procedure induces the intended separation in predictive uncertainty during denoising. Tokens are divided into structural and detail groups using the same deterministic token-level partition employed during training. We compare the fine-tuned model with entropy-gated decoding against the base model with the original decoder. For the trajectory diagnostics in Figure 3, we use a 512-step denoising trajectory to obtain fine-grained temporal measurements.

Fig. 3 reports the average entropy trajectories. Two observations stand out. First, in the fine-tuned model, structural tokens exhibit lower entropy than detail tokens from early denoising steps, with an initial gap of approximately 2.0 nats. In contrast, the base model shows little separation during the first 100 steps. This behavior is consistent with the intended uncertainty-shaping effect of the

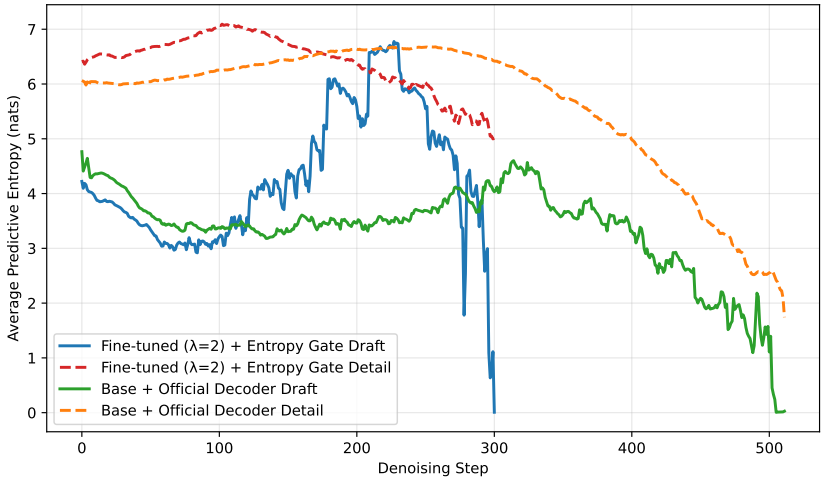

Figure 3: Predictive-entropy trajectories of structural and detail tokens during denoising. Blue/red lines denote the fine-tuned model with entropy-gated decoding, while green/orange lines denote the base model with the original decoder. The fine-tuned model exhibits a pronounced early entropy separation between structural and detail tokens.

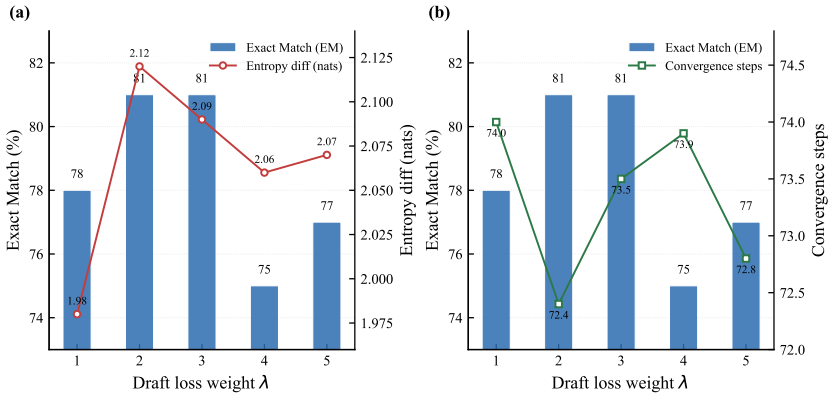

Figure 4: Effect of structural loss weight λ on EM, entropy separation, and structural-token convergence. (a) Blue bars denote EM scores, and the red curve shows the structural–detail entropy gap. (b) The red curve shows the number of steps required for structural-token entropy to fall below 1.0 nat.

proposed training procedure, although it is not a direct empirical verification of Proposition 1. Second, structural-token entropy in the fine-tuned model reaches near-zero values around step 300, whereas the base model requires approximately 400 steps. This trajectory is associated with earlier stabilization of structural positions under the full framework. A detailed discussion of the transient entropy increase is provided in Appendix B.2.

Effect of Loss Weight and Convergence. We further examine how the structural loss weight λ affects entropy separation and convergence. We train models with $\lambda \in \{1, 2, 3, 4, 5\}$ and record two diagnostics: (1) the entropy gap between structural and detail tokens at step 10, and (2) the number of steps required for structural-token entropy to fall below 1.0 nat. Fig. 4 shows that $\lambda = 2$ achieves the largest early entropy gap (2.12 nats) and the best EM among the tested settings. Increasing λ further reduces the entropy gap and does not improve EM, suggesting that stronger structural weighting is not monotonically beneficial. The measured structural convergence point remains within 72–74 steps across all tested values of λ, indicating that this diagnostic is relatively stable once the uncertainty separation emerges. Additional robustness analysis is provided in Appendix C.1.

Table 2: Cross-task transfer results. Dream-7B (Ye et al., 2025) is included as an external diffusion-model reference rather than a matched controlled baseline.

Model	GSM8K	ARC-C	ARC-E	HellaSwag
LLaDA Base (Nie et al., 2025) + Original Decoder	66.0	46.0	70.5	50.6
Dream-7B (Ye et al., 2025)	70.3	56.3	82.3	73.3
Ours ($\lambda = 2$ **+ Entropy-Gated)**	**81.5**	**82.0**	**91.0**	**66.0**

Table 3: Ablation study on key components of the entropy-gated decoder on GSM8K (Cobbe et al., 2021).

Decoder Configuration	EM (%)
Full Entropy-Gated Decoder	81.5
w/o Maximum Fixation Constraint	79.2
w/o Minimum Fixation Guarantee	78.8

6.3 Cross-Task Transfer Analysis

We evaluate the cross-task transfer of the model fine-tuned on GSM8K (Cobbe et al., 2021). The model is directly evaluated on ARC-Challenge (Clark et al., 2018), ARC-Easy (Clark et al., 2018), and HellaSwag (Zellers et al., 2019) without additional task-specific fine-tuning. We compare against the LLaDA-8B (Nie et al., 2025) base model with the original decoder, while Dream-7B (Ye et al., 2025) is included as an external diffusion-model reference. As shown in Table 2, our method improves over the original LLaDA pipeline on all evaluated tasks. Performance increases from 66.0 to 81.5 on GSM8K, from 46.0 to 82.0 on ARC-C, and from 70.5 to 91.0 on ARC-E. On HellaSwag, the score improves from 50.6 to 66.0, although it remains below the Dream-7B reference. These results provide evidence that the learned training and decoding behavior transfers beyond the GSM8K fine-tuning domain. The magnitude of improvement varies across tasks, indicating task-dependent transfer rather than broad task-independent generalization. We additionally conduct an independent training-side validation on MATH (Hendrycks et al., 2021) to examine whether the uncertainty-shaping mechanism extends beyond the GSM8K training distribution; detailed results are reported in Appendix D.6.

6.4 Ablation Study

We evaluate two key constraints of the entropy-gated decoder, motivated by the local robustness analysis in Theorem 2. Removing the maximum fixation constraint reduces EM by 2.8% relative, consistent with the role of the upper budget in limiting aggressive simultaneous commitments. Removing the minimum fixation guarantee results in a 3.3% relative drop, supporting its role in preventing early-stage decoding stagnation. These results demonstrate the practical importance of controlling both maximum fixation size and minimum decoding progress. Additional analyses of structural-token fixation dynamics and predictive-entropy ranking stability are provided in Appendix C.2 and Appendix C.3. We further compare entropy ranking with maximum-probability ranking under an identical per-step fixation budget. The matched comparison, reported in Appendix D.5, shows that entropy provides an ordering signal distinct from maximum-probability confidence.

7 Conclusion

We study risk-aware token fixation for parallel decoding in discrete diffusion language models. Our analysis distinguishes an error-free information-gain reference from practical decoding under prediction and propagation risk, motivating low-entropy-first fixation in a local robustness regime. Based on this insight, we combine training-time uncertainty shaping with entropy-gated decoding. On GSM8K, entropy-gated decoding improves EM from 66.0 to 70.0 with a **5.64**$\times$ speedup, while the full framework reaches **81.5** EM with a **9.28**$\times$ speedup. Cross-task results further show transfer beyond GSM8K. Overall, our results support jointly shaping predictive uncertainty during training and exploiting it for risk-aware fixation at inference.

AI USE STATEMENT

Generative AI tools were used to aid or polish writing and for research ideation or execution. No synthetic training examples or target responses were generated for the experiments reported in this paper. The original training instances and response texts were retained unchanged, and the structural partition was derived through deterministic preprocessing. All AI-assisted outputs were reviewed and verified by the authors, who take responsibility for the final content of this work.

REFERENCES

Marianne Arriola, Aaron Gokaslan, et al. Block diffusion: Interpolating between autoregressive and diffusion language models, 2025.

Jacob Austin, Daniel D. Johnson, et al. Structured denoising diffusion models in discrete state-spaces, 2023.

Benjamin L. Badger and Matthew Neligeorge. Know your limits: Entropy estimation modeling for compression and generalization, 2025.

Heli Ben-Hamu, Itai Gat, et al. Accelerated sampling from masked diffusion models via entropy bounded unmasking. In *Advances in Neural Information Processing Systems*, 2025.

Tom B. Brown, Benjamin Mann, et al. Language models are few-shot learners, 2020.

Dake Bu, Wei Huang, et al. DPRM: A plug-in doob h transform-induced token-ordering module for diffusion language models, 2026.

Changxiao Cai and Gen Li. Confidence-based decoding is provably efficient for diffusion language models, 2026.

Jiaao Chen, Aston Zhang, et al. A cheaper and better diffusion language model with soft-masked noise, 2023.

Ziyu Chen, Xinbei Jiang, et al. Optimizing decoding paths in masked diffusion models by quantifying uncertainty, 2025.

Aakanksha Chowdhery, Sharan Narang, et al. Palm: Scaling language modeling with pathways, 2022.

Peter Clark, Isaac Cowhey, et al. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *arXiv e-prints*, art. arXiv:1803.05457, March 2018. doi: 10.48550/arXiv.1803.05457.

Karl Cobbe, Vineet Kosaraju, et al. Training Verifiers to Solve Math Word Problems. *arXiv e-prints*, art. arXiv:2110.14168, October 2021. doi: 10.48550/arXiv.2110.14168.

Angela Fan, Mike Lewis, et al. Hierarchical neural story generation, 2018.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning, 2016.

Haoyu He, Katrin Renz, et al. Mdpo: Overcoming the training-inference divide of masked diffusion language models, 2025.

Zhengfu He, Tianxiang Sun, et al. Diffusionbert: Improving generative masked language models with diffusion models, 2022.

Dan Hendrycks, Collin Burns, et al. Measuring mathematical problem solving with the math dataset, 2021.

Mahsa Khoshnoodi, Vinija Jain, et al. A comprehensive survey of accelerated generation techniques in large language models, 2024.

Xiang Lisa Li, John Thickstun, et al. Diffusion-lm improves controllable text generation, 2022.

Aaron Lou, Chenlin Meng, et al. Discrete diffusion modeling by estimating the ratios of the data distribution, 2024.

Jama Hussein Mohamud, Mohsin Hasan, et al. Adaptive order policies for masked diffusion, 2026.

Shen Nie, Fengqi Zhu, et al. Large Language Diffusion Models. *arXiv e-prints*, art. arXiv:2502.09992, February 2025. doi: 10.48550/arXiv.2502.09992.

Xuefei Ning, Zinan Lin, et al. Skeleton-of-thought: Prompting llms for efficient parallel generation, 2024.

Reiner Pope, Sholto Douglas, et al. Efficiently scaling transformer inference, 2022.

Subham Sekhar Sahoo, Marianne Arriola, et al. Simple and effective masked diffusion language models, 2024.

Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Hugo Touvron, Thibaut Lavril, et al. Llama: Open and efficient foundation language models, 2023.

Ashish Vaswani, Noam Shazeer, et al. Attention is all you need, 2023.

Chengcheng Wang, Tingzhang Luo, et al. TACG: Trajectory-aware commit gating for diffusion language model decoding, 2026.

Lei Wang, Wanyu Xu, et al. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models, 2023.

Shenzhi Wang, Le Yu, et al. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning, 2025.

Jason Wei, Xuezhi Wang, et al. Chain-of-thought prompting elicits reasoning in large language models, 2023.

Kaisen Yang, Jayden Teoh, et al. Improving sampling for masked diffusion models via information gain, 2026.

Shunyu Yao, Dian Yu, et al. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.

Jiacheng Ye, Zhihui Xie, et al. Dream 7B: Diffusion Large Language Models. *arXiv e-prints*, art. arXiv:2508.15487, August 2025. doi: 10.48550/arXiv.2508.15487.

Rowan Zellers, Ari Holtzman, et al. HellaSwag: Can a Machine Really Finish Your Sentence? *arXiv e-prints*, art. arXiv:1905.07830, May 2019. doi: 10.48550/arXiv.1905.07830.

Shaorong Zhang, Longxuan Yu, et al. Generation order and parallel decoding in masked diffusion models: An information-theoretic perspective, 2026.

Wayne Xin Zhao, Kun Zhou, et al. A Survey of Large Language Models. *arXiv e-prints*, art. arXiv:2303.18223, March 2023. doi: 10.48550/arXiv.2303.18223.

Fengqi Zhu, Zebin You, et al. LLaDA-MoE: A Sparse MoE Diffusion Language Model. *arXiv e-prints*, art. arXiv:2509.24389, September 2025. doi: 10.48550/arXiv.2509.24389.

A Full Proofs

This appendix provides complete proofs for all theoretical results presented in Section 4. Unless otherwise specified, all logarithms are in the natural base e. We assume that all random variables are defined on a common probability space and that the relevant joint distributions are well-defined.

A.1 Details for Section 4.1: Oracle-Optimal Benchmark

We provide complete proofs of the claims stated in Section 4.1 of the main text. All notation follows the main text. Let $X_{1:N}$ denote a sequence of N discrete random variables with joint distribution $P_{X_{1:N}}$. Let A be the set of already revealed indices with observed values x_A, and let $\mathcal{R} = \{1, \dots, N\} \setminus A$ denote the set of still-masked indices. An oracle reveals the ground-truth value x_i of X_i for any requested $i \in \mathcal{R}$.

Proof of the Single-Step Oracle Gain Lemma

Lemma 1. *For any $i \in \mathcal{R}$,*

$$\Delta_{\text{oracle}}(i \mid A) := H(X_{\mathcal{R}} \mid x_A) - H(X_{\mathcal{R}\setminus\{i\}} \mid x_A, X_i) = H(X_i \mid x_A). \tag{23}$$

Proof. By the chain rule for conditional entropy,

$$\begin{aligned} H(X_{\mathcal{R}} \mid x_A) &= H(X_i, X_{\mathcal{R}\setminus\{i\}} \mid x_A) \\ &= H(X_i \mid x_A) + H(X_{\mathcal{R}\setminus\{i\}} \mid x_A, X_i). \end{aligned} \tag{24}$$

Substituting this expansion into the definition gives

$$\begin{aligned} \Delta_{\text{oracle}}(i \mid A) &= H(X_i \mid x_A) + H(X_{\mathcal{R}\setminus\{i\}} \mid x_A, X_i) \\ &\quad - H(X_{\mathcal{R}\setminus\{i\}} \mid x_A, X_i) \\ &= H(X_i \mid x_A). \end{aligned} \tag{25}$$

□

Proposition A.1 (Total gain is constant over orderings)

We now consider the expected oracle gain before observing the value of each newly revealed token. Let

$$\pi = (i_1, \dots, i_N)$$

be a permutation of $\{1, \dots, N\}$, and define

$$A_0 = \varnothing, \qquad A_k = A_{k-1} \cup \{i_k\}.$$

At step k, the expected oracle gain conditioned on the previously revealed random variables is

$$\Delta_{\text{oracle}}(i_k \mid A_{k-1}) = H(X_{i_k} \mid X_{A_{k-1}}).$$

Accordingly, the cumulative expected oracle gain over all N reveals is

$$\Gamma_N(\pi) = \sum_{k=1}^{N} \Delta_{\text{oracle}}(i_k \mid A_{k-1}) = \sum_{k=1}^{N} H(X_{i_k} \mid X_{A_{k-1}}). \tag{26}$$

Proposition A.1 (Total expected gain is order-invariant). For any permutation $\pi = (i_1, \dots, i_N)$ of the N token indices,

$$\Gamma_N(\pi) = H(X_{1:N}). \tag{27}$$

Proof. For a fixed permutation $\pi = (i_1, \dots, i_N)$, we have

$$A_{k-1} = \{i_1, \dots, i_{k-1}\}.$$

Therefore,

$$H(X_{i_k} \mid X_{A_{k-1}}) = H(X_{i_k} \mid X_{i_1}, \dots, X_{i_{k-1}}).$$

By the chain rule for entropy,

$$\begin{aligned} H(X_{i_1}, \dots, X_{i_N}) &= \sum_{k=1}^{N} H(X_{i_k} \mid X_{i_1}, \dots, X_{i_{k-1}}) \\ &= \sum_{k=1}^{N} H(X_{i_k} \mid X_{A_{k-1}}) \\ &= \Gamma_N(\pi). \end{aligned} \tag{28}$$

Since π is a permutation of $\{1, \ldots, N\}$, reordering the components of a random vector does not change its joint entropy:

$$H(X_{i_1}, \ldots, X_{i_N}) = H(X_{1:N}).$$

Hence,

$$\Gamma_N(\pi) = H(X_{1:N}),$$

for every permutation π. □

PROOF OF THEOREM 1: GREEDY OPTIMALITY UNDER CONDITIONAL INDEPENDENCE

We restate the assumption and theorem for completeness. Throughout this subsection, all distributions and entropies are understood to be conditioned on the fixed initial context (e.g., the prompt and the initial diffusion state), which is suppressed from the notation.

Conditional-independence condition. For every revealed index set $A \subseteq \{1, \ldots, N\}$ and every realization x_A with positive probability, let

$$\mathcal{R} = \{1, \ldots, N\} \setminus A.$$

As in Section 4.1, we assume

$$P(X_{\mathcal{R}} \mid X_A = x_A) = \prod_{i \in \mathcal{R}} P(X_i \mid X_A = x_A)\,. \tag{29}$$

Restatement of Theorem 1. Under the conditional-independence condition, define the greedy reveal rule at step k as

$$i_k^{\text{greedy}} \in \arg\max_{i \in \mathcal{R}_{k-1}} H\big(X_i \mid X_{A_{k-1}}\big)\,, \tag{30}$$

where A_{k-1} denotes the set of previously revealed indices and $\mathcal{R}_{k-1}$ the remaining indices.

Then, for every truncation horizon $K \in \{1, \ldots, N\}$, the greedy ordering maximizes the K-step cumulative expected oracle gain:

$$\Gamma_K^{\text{greedy}} = \max_{\pi} \Gamma_K(\pi), \qquad \forall K \in \{1, \ldots, N\}. \tag{31}$$

Proof. First, applying the conditional-independence condition at $A = \varnothing$ gives

$$P(X_{1:N}) = \prod_{i=1}^{N} P(X_i), \tag{32}$$

where the fixed initial context is suppressed from the notation. Thus, under the oracle benchmark assumption, the token random variables are mutually independent conditional on that initial context.

Consequently, for any index set $A \subseteq \{1, \ldots, N\} \setminus \{j\}$ and any realization x_A with positive probability,

$$P(X_j \mid X_A = x_A) = P(X_j). \tag{33}$$

Hence revealing any subset of the other tokens does not change the distribution, and therefore does not change the entropy, of X_j:

$$H(X_j \mid X_A) = H(X_j). \tag{34}$$

Define

$$h_i := H(X_i), \qquad i = 1, \ldots, N, \tag{35}$$

again with the fixed initial context implicitly conditioned upon. For any reveal order

$$\pi = (i_1, \ldots, i_N),$$

the expected oracle gain at step k is therefore

$$\Delta_{\text{oracle}}(i_k \mid A_{k-1}) = H(X_{i_k} \mid X_{A_{k-1}}) = h_{i_k}. \tag{36}$$

Accordingly, the cumulative expected gain after the first K reveals is

$$\Gamma_K(\pi) = \sum_{k=1}^{K} h_{i_k}. \tag{37}$$

Let

$$h_{(1)} \geq h_{(2)} \geq \cdots \geq h_{(N)}$$

denote the values $\{h_i\}_{i=1}^N$ sorted in non-increasing order. Because the entropy of every unrevealed token remains unchanged throughout the process, the greedy strategy selects a token with entropy $h_{(1)}$ at the first step, then one with entropy $h_{(2)}$, and so on, with arbitrary tie-breaking. Therefore,

$$\Gamma_K^{\text{greedy}} = \sum_{k=1}^{K} h_{(k)}. \tag{38}$$

Now consider any competing permutation

$$\pi' = (j_1, \ldots, j_N).$$

Its first K reveals correspond to some subset of K distinct entropy values from $\{h_1, \ldots, h_N\}$, so

$$\Gamma_K(\pi') = \sum_{k=1}^{K} h_{j_k} \leq \sum_{k=1}^{K} h_{(k)} = \Gamma_K^{\text{greedy}}. \tag{39}$$

The inequality follows because the sum of the K largest elements of a finite collection is at least as large as the sum of any other K distinct elements from that collection.

Since the same non-increasing ordering of $\{h_i\}$ realizes this maximum for every $K = 1, \ldots, N$, the greedy ordering is simultaneously optimal for all truncation horizons. □

Remark. This result is deliberately restricted to the counterfactual oracle setting. The conditional independence assumption implies that revealing one token does not alter the uncertainty of any remaining token; it is not intended as a model of the dependency structure of natural language. The theorem therefore isolates the information-gain objective in the absence of prediction error and inter-token dependency effects, rather than establishing the optimality of high-entropy-first decoding in practical diffusion generation.

A.2 Details for Section 4.2: Local Robustness of Low-Entropy-First Fixation

We provide the derivation of the local robustness certificate used in Section 4.2. All quantities are conditioned on a fixed decoding context x_A, which is omitted from the notation for clarity.

For a masked position i, let

$$\mathbf{p}_i(k) = p_\theta(X_i = k \mid x_A), \qquad H_i = H(\mathbf{p}_i),$$

and let

$$\hat{x}_i = \arg\max_k \mathbf{p}_i(k)$$

be the model prediction. Let q_i denote the unknown true conditional distribution and define the corresponding commitment error probability as

$$\epsilon_i = 1 - q_i(\hat{x}_i).$$

We assume bounded model mismatch,

$$D_{\text{KL}}(q_i \| \mathbf{p}_i) \leq \kappa, \tag{40}$$

and a bounded downstream penalty $\Delta > 1$ for an incorrect commitment.

Distributional proximity

Distributional proximity. Under Eq. (40),

$$\|q_i - p_i\|_1 \leq \sqrt{2\kappa}.$$

Proof. Pinsker's inequality gives

$$D_{\mathrm{KL}}(q_i \| \mathbf{p}_i) \geq \frac{1}{2} \|q_i - \mathbf{p}_i\|_1^2.$$

Combining this inequality with $D_{\mathrm{KL}}(q_i \| \mathbf{p}_i) \leq \kappa$ yields the result. □

Define

$$\rho_\kappa = \min\left\{\sqrt{\frac{\kappa}{2}},\, 1\right\}. \tag{41}$$

ERROR-RATE BOUND

Proof of Lemma 2. The true commitment error probability satisfies

$$\epsilon_i \leq 1 - e^{-H_i} + \rho_\kappa.$$

Proof. By the total-variation bound implied by Pinsker's inequality,

$$q_i(\hat{x}_i) \geq \mathbf{p}_i(\hat{x}_i) - \rho_\kappa = p_i^{\max} - \rho_\kappa,$$

where

$$p_i^{\max} = \max_k \mathbf{p}_i(k).$$

Moreover,

$$H_i = -\sum_k \mathbf{p}_i(k) \log \mathbf{p}_i(k) \geq -\log p_i^{\max},$$

which implies

$$p_i^{\max} \geq e^{-H_i}.$$

Therefore,

$$\begin{aligned} \epsilon_i &= 1 - q_i(\hat{x}_i) \\ &\leq 1 - p_i^{\max} + \rho_\kappa \\ &\leq 1 - e^{-H_i} + \rho_\kappa. \end{aligned}$$

□

ENTROPY CONTINUITY UNDER MODEL MISMATCH

We next lower-bound the true conditional entropy $H(q_i)$ in terms of the predictive entropy H_i.

Let

$$\bar{\rho}_\kappa = \min\left\{\rho_\kappa,\, 1 - \frac{1}{|\mathcal{V}|}\right\}, \tag{42}$$

and define

$$C_{\mathrm{ent}}(\kappa, \mathcal{V}) = h_2(\bar{\rho}_\kappa) + \bar{\rho}_\kappa \log(|\mathcal{V}| - 1), \tag{43}$$

where

$$h_2(u) = -u \log u - (1-u) \log(1-u)$$

is the binary entropy function.

By the Fannes–Audenaert continuity bound for entropy,

$$H(q_i) \geq H_i - C_{\mathrm{ent}}(\kappa, \mathcal{V}). \tag{44}$$

LOCAL ROBUSTNESS CERTIFICATE

Recall the worst-case risk-adjusted margin

$$\Psi(i) = \inf_{q_i : D_{\mathrm{KL}}(q_i \| \mathbf{p}_i) \le \kappa} [H(q_i) - \Delta \epsilon_i] . \tag{45}$$

For every feasible q_i, combining the error-rate bound above with the entropy-continuity bound gives

$$H(q_i) - \Delta\epsilon_i \ge H_i - C_{\mathrm{ent}}(\kappa, \mathcal{V}) - \Delta \left(1 - e^{-H_i} + \rho_\kappa\right) \tag{46}$$

$$= H_i - \Delta(1 - e^{-H_i}) - C_{\kappa,\mathcal{V},\Delta}, \tag{47}$$

where

$$C_{\kappa,\mathcal{V},\Delta} = C_{\mathrm{ent}}(\kappa, \mathcal{V}) + \Delta\rho_\kappa \tag{48}$$

is independent of the candidate position i.

Taking the infimum over all feasible q_i therefore preserves the bound:

$$\Psi(i) \ge L(H_i) := H_i - \Delta(1 - e^{-H_i}) - C_{\kappa,\mathcal{V},\Delta}. \tag{49}$$

Restatement of Theorem 2. Define the local candidate set

$$\mathcal{R}_\Delta = \{i \in \mathcal{R} : H_i < \log \Delta\} . \tag{50}$$

For every $i \in \mathcal{R}_\Delta$, the lower bound $L(H_i)$ in Eq. (49) is strictly decreasing in H_i. Hence, if $\mathcal{R}_\Delta \neq \varnothing$,

$$i^*_{\mathrm{robust}} \in \arg\min_{i \in \mathcal{R}_\Delta} H_i \tag{51}$$

maximizes this entropy-dependent robustness lower bound within the local regime.

Proof. The terms collected in $C_{\kappa,\mathcal{V},\Delta}$ do not depend on H. Therefore,

$$L'(H) = 1 - \Delta e^{-H}. \tag{52}$$

For $H < \log \Delta$,

$$\Delta e^{-H} > 1,$$

and consequently

$$L'(H) < 0.$$

Thus $L(H)$ is strictly decreasing throughout the local candidate regime. Among candidates in $\mathcal{R}_\Delta$, the smallest predictive entropy therefore yields the largest value of the derived robustness certificate. □

Scope of the result. The theorem concerns the derived lower bound $L(H_i)$ rather than the exact worst-case margin $\Psi(i)$, and only within the local regime $H_i < \log \Delta$. It therefore provides a risk-aware motivation for low-entropy-first fixation, not a claim of global optimality over all predictive entropy values.

A.3 DETAILS FOR SECTION 4.3: CONDITIONAL-ENTROPY MONOTONICITY UNDER MASKING

We provide the derivation of Proposition 1 in Section 4.3. The purpose of this result is only to isolate the effect of the corruption rate under a matched target–context distribution. It does not establish a pointwise entropy ordering between structural and detail tokens in actual decoding trajectories.

Let U denote the clean conditioning signal available to the denoiser and let X denote the target token, with joint distribution $P_{X,U}$. For a masking probability $p \in [0, 1)$, let

$$Z_p = \mathcal{M}_p(U)$$

denote the corrupted observation obtained by independently replacing each relevant component of U with the special `[MASK]` symbol with probability p.

Consider two masking probabilities

$$0 \le p_{\mathcal{S}} < p_{\mathcal{T}} < 1. \tag{53}$$

Define

$$r = \frac{p_{\mathcal{T}} - p_{\mathcal{S}}}{1 - p_{\mathcal{S}}}. \tag{54}$$

Starting from $Z_{p_{\mathcal{S}}}$, independently remask every currently unmasked component with probability r. Let the resulting observation be $\widetilde{Z}$.

For each component, the probability of being masked in $\widetilde{Z}$ is

$$p_{\mathcal{S}} + (1 - p_{\mathcal{S}})r = p_{\mathcal{S}} + (1 - p_{\mathcal{S}})\frac{p_{\mathcal{T}} - p_{\mathcal{S}}}{1 - p_{\mathcal{S}}} \tag{55}$$

$$= p_{\mathcal{T}}. \tag{56}$$

Hence $\widetilde{Z}$ has the same distribution as $Z_{p_{\mathcal{T}}}$. Moreover, the additional masking operation depends only on $Z_{p_{\mathcal{S}}}$ and independent masking noise, so

$$X \longrightarrow Z_{p_{\mathcal{S}}} \longrightarrow Z_{p_{\mathcal{T}}} \tag{57}$$

forms a Markov chain.

By the data-processing inequality,

$$I(X; Z_{p_{\mathcal{S}}}) \geq I(X; Z_{p_{\mathcal{T}}}). \tag{58}$$

Since

$$H(X \mid Z_p) = H(X) - I(X; Z_p),$$

and the marginal distribution of X is identical in the two matched comparisons, we obtain

$$H(X \mid Z_{p_{\mathcal{S}}}) \leq H(X \mid Z_{p_{\mathcal{T}}}). \tag{59}$$

This proves Proposition 1. The same argument holds conditional on any uncorrupted prompt or context that is shared by the two corruption processes.

Scope of the result. This proposition compares two corruption levels applied to the same underlying target–context distribution. It therefore isolates the effect of masking probability alone. It does not imply that structural tokens have lower predictive entropy than detail tokens at every step of an actual decoding trajectory, since the two token groups may have different underlying distributions and the composition of the remaining masked set changes during decoding.

In the full training objective, differential corruption is combined with structure-aware loss reweighting. The proposition above characterizes only the corruption component; the effect of loss reweighting on predictive uncertainty is evaluated empirically rather than claimed as a separate theoretical guarantee. In addition, the argument applies to training states for which the realized masking probabilities satisfy $p_{\mathcal{S}} < p_{\mathcal{T}}$.

B Qualitative Dynamics of Entropy Inversion

This appendix provides a qualitative interpretation of two phenomena observed in Fig. 3: a transient increase in structural-token entropy during the middle stage of denoising and a temporary crossover between the structural and detail entropy curves. The discussion is descriptive rather than a formal consequence of the theoretical results in Section 4.

B.1 Observed Phases

From Fig. 3, the entropy trajectories of our method exhibit three distinct phases. In Phase I (0∼150 steps), the entropy of structural tokens drops rapidly from about 4.5 to below 3 nats, while the entropy of detail tokens decreases more slowly. This corresponds to a stronger early fixation preference for structural positions under entropy-gated decoding. In Phase II (150∼250 steps), structural-token entropy temporarily increases, reaching a peak of approximately 6.5 nats, while detail-token entropy decreases more rapidly. The two curves approach each other and cross around step 200.

In Phase III (after 250 steps), structural-token entropy decreases again toward near-zero values, while detail-token entropy continues to decrease as denoising proceeds.

The base model does not exhibit the same degree of early separation: the two entropy curves remain closer during the early stage, and structural positions stabilize later.

B.2 Interpreting the Observed Bump and Crossover

Structural-Entropy Bump in Phase II. A plausible explanation for the temporary increase involves both remaining-set composition and evolving contextual support. After many relatively easy, low-entropy structural positions have been fixed, the remaining structural set becomes increasingly concentrated on more difficult positions, including those involving longer-range dependencies or multi-step reasoning. At the same time, some of these positions may still lack sufficient support from unresolved detail tokens, leading to more ambiguous predictive distributions. As additional context is gradually fixed, this ambiguity may be reduced and structural-token entropy decreases again. This interpretation is consistent with the observed trajectory and with the self-conditioning nature of iterative denoising, but it is not intended as a causal derivation.

Entropy Crossover. During the middle stage, detail-token entropy decreases rapidly while the remaining structural set becomes smaller and more difficult. One plausible interpretation is that previously fixed, relatively reliable structural positions provide additional contextual constraints that help reduce uncertainty over detail tokens. Together with the changing composition of the remaining structural set, this can produce a temporary crossover between the two entropy curves. We emphasize that this observation does not establish a causal effect of structural fixation on detail-token entropy.

Relation to Proposition 1. Proposition 1 in Section 4.3 compares different masking probabilities under the same underlying target–context distribution. It therefore does not imply a pointwise ordering between the empirical entropy trajectories of structural and detail tokens. In actual decoding, the two groups correspond to different token populations, and the composition of the remaining masked positions changes dynamically as fixation proceeds. The observed bump and crossover should therefore be interpreted as empirical trajectory phenomena rather than as either a violation or a direct verification of Proposition 1.

B.3 Summary

The entropy trajectories in Fig. 3 exhibit an early structural–detail separation followed by a transient bump and crossover. These observations are compatible with changes in available context and in the composition of the remaining masked positions during iterative denoising. This appendix provides a qualitative interpretation of the observed behavior and does not introduce an additional theoretical guarantee beyond the results established in Section 4.

C Additional Experiments

C.1 Hyperparameter Sensitivity

We evaluate the sensitivity of the proposed framework to key training hyperparameters, including the structural noise-scaling coefficient $s_{\mathcal{S}}$, the structural loss weight λ, and the learning rate. All experiments are conducted on GSM8K using the entropy-gated decoder.

Sensitivity to Noise Scaling. We vary the structural noise-scaling coefficient $s_{\mathcal{S}} \in \{0.30, 0.45, 0.65\}$ while fixing the learning rate at 2×10^{-6}, the structural loss weight at $\lambda = 2$, and all other training settings unchanged. The corresponding EM scores are 80, 81, and 80, respectively. Performance therefore remains similar across the tested values of $s_{\mathcal{S}}$.

Sensitivity to Structural Loss Weight. We further evaluate the effect of the structural loss weight by training models with $\lambda \in \{1, 2, 3, 4, 5\}$. The corresponding EM scores are 78.2, 81.5, 80.1, 75.2, and 77.8, respectively. All tested settings train successfully, while $\lambda = 2$ achieves the highest EM among the evaluated values. The associated entropy-gap and convergence behavior is further analyzed in Section 6.2.

Sensitivity to Learning Rate. We train models with $s_{\mathcal{S}} = 0.65$ and evaluate four learning rates: 1×10^{-6}, 2×10^{-6}, 4×10^{-6}, and 1×10^{-5}. The corresponding EM scores are 78, 77, 80, and 76, respectively. All tested configurations converge without training divergence, with moderate variation in final performance.

Summary. Across the tested hyperparameter ranges, training remains stable and no configuration exhibits optimization failure. Performance is relatively insensitive to the structural noise-scaling

coefficient, while larger variation is observed for the structural loss weight and learning rate. These results suggest that the proposed framework does not rely on a narrowly tuned training configuration, although its final performance remains affected by hyperparameter choice.

C.2 Skeleton Fixation Ratio Analysis in Newly Fixed Tokens

This experiment further examines the effect of the entropy-gated decoder and structured training on the preferential fixation of skeleton tokens. Unlike the entropy trajectory analysis in Section 6.2, we directly analyze the proportion of skeleton tokens among newly fixed tokens at each denoising step.

We define the Skeleton Fixation Ratio (SFR) as:

$$R(t) = \frac{|F_t \cap \mathcal{S}|}{|F_t|}, \tag{60}$$

where F_t denotes the set of tokens newly fixed at step t, and S denotes the set of skeleton tokens. A higher value indicates a stronger preference for skeleton tokens during decoding.

Effect of Entropy-Gated Decoding. The baseline model exhibits consistently low SFR (below 0.33), indicating no preference for skeleton tokens. This behavior is consistent across all denoising stages and reflects the absence of structural bias in the original decoding strategy.

In contrast, adding the entropy-gated decoder significantly increases early-stage skeleton selection. At 10% progress, SFR rises sharply to 0.9027. However, this advantage decays rapidly as decoding proceeds, dropping to 0.4775 at 20% and 0.3694 at 30%. This suggests that entropy alone is insufficient to maintain stable structural selection, likely due to unreliable uncertainty estimates in the absence of structured training. As a result, early skeleton fixation may include incorrectly selected tokens, limiting long-term benefit.

Effect of Structured Training. Our method achieves slightly lower early SFR (0.8117 at 10%) compared to the entropy-only variant, reflecting a more calibrated selection behavior. However, its key advantage lies in stability: the decline of SFR is significantly slower, remaining at 0.6754 at 20% and 0.5909 at 30%, and retaining a non-trivial level even at later stages.

This indicates that structured training is associated with a more persistent preference for structural-token fixation over the evaluated trajectory. As a result, the decoder is able to continuously prioritize skeleton tokens rather than only in early steps. This behavior is consistent with the strong performance gains reported in Table 1, where our method achieves both higher accuracy and faster inference.

Summary. Overall, this experiment provides token-level evidence complementing the entropy trajectory analysis in Section 6.2. The results show that: entropy-gated decoding introduces strong but unstable early skeleton preference, structured training stabilizes this preference across the entire denoising process, and the baseline fails to exhibit meaningful structural bias at any stage. These findings are consistent with the observed entropy separation and further support the role of training-time uncertainty shaping in inducing preferential fixation of structural tokens.

C.3 Stability of Predictive-Entropy Rankings Across Denoising States

This experiment examines whether the predictive-entropy ranking of unresolved tokens remains locally stable as the denoising context evolves. Such stability is useful for entropy-based fixation because token selection at one iteration is then not based on a ranking that changes arbitrarily after a small update to the partially denoised sequence.

Experimental Setup. We analyze 100 GSM8K examples generated by the fine-tuned model with the entropy-gated decoder. For each pair of consecutive denoising states t and $t + 1$, we identify the token positions that remain masked in both states. Using the predictive entropies recorded from the corresponding denoising trajectories, we compute Kendall's τ between the entropy rankings of these common unresolved positions.

No additional model forward pass is introduced for this analysis: the entropy values are taken directly from the saved denoising trajectories. We compute τ for every valid consecutive-state pair and aggregate the resulting correlations across examples and denoising steps.

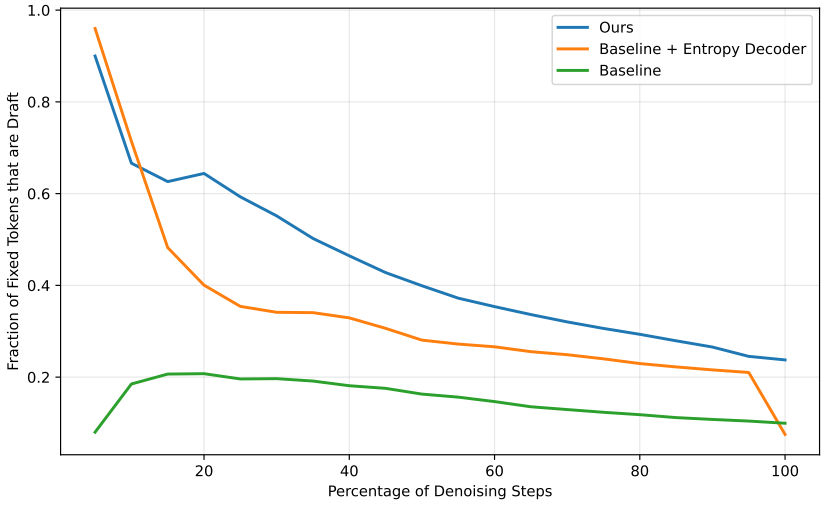

Figure 5: Skeleton fixation ratio over denoising steps.

Results. Figure 6 shows the distribution of Kendall's τ across all valid consecutive-state pairs. The average rank correlation is $\tau = 0.761$, with most correlations concentrated in the positive, high-agreement regime.

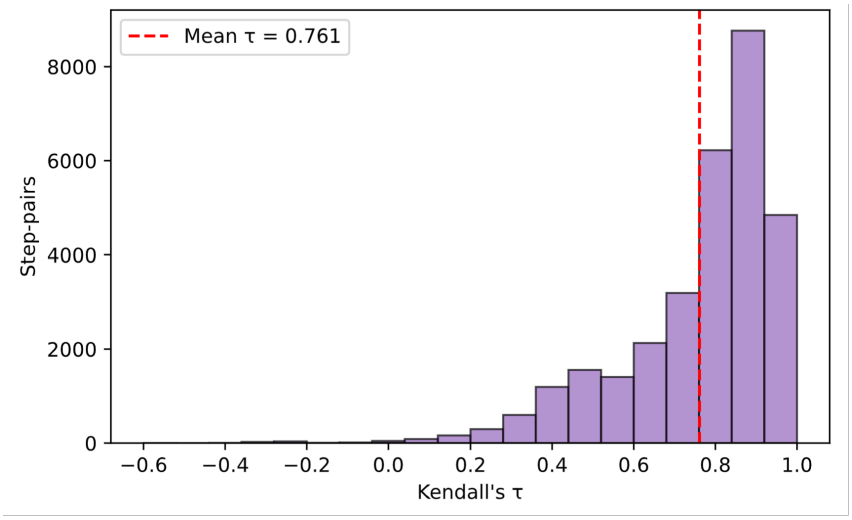

Figure 6: Distribution of Kendall's τ rank correlations between predictive-entropy rankings at consecutive denoising states. The dashed line denotes the mean correlation, $\tau = 0.761$.

Analysis. The positive average correlation indicates substantial local consistency in the entropy ordering of unresolved tokens as the denoising state changes. Although individual rankings can change as new tokens are fixed and the available context evolves, the ordering is generally preserved between adjacent states rather than being completely reshuffled.

This result complements the trajectory-level analyses by showing that the predictive-entropy signal used for fixation exhibits local temporal stability. It should not be interpreted as an equivalence between different notions of entropy, nor as evidence that the ranking remains unchanged over long denoising intervals.

Table 4: Configuration of LLaDA baseline used for the main end-to-end comparison.

Setting	Configuration
Checkpoint	LLaDA-8B-Instruct base, no adapter
Prompting	Zero-shot, no demonstrations
Generation / block length	128 / 128 tokens
Denoising steps	128
Remasking	Random
Prediction	Greedy argmax
Temperature / CFG	0 / none
EOS handling	Native tokenizer stopping and original post-processing
Batch size	1
Timing	Total wall-clock time over the same evaluation set

Conclusion. Across consecutive denoising states, predictive-entropy rankings exhibit an average Kendall correlation of 0.761. This provides empirical evidence that entropy-based token ordering is locally stable under the evolving denoising context.

D Implementation Details

D.1 Base Model and Training Data

All experiments in this work use LLaDA-8B-Instruct as the base model. Structured fine-tuning is conducted on the GSM8K training set, which contains approximately 7,500 elementary math word problems. Each sample includes a question, a numerical answer, and human-annotated solution steps. During training, model parameters are stored in bfloat16 precision. We use the AdamW optimizer with a weight decay of 0.01 and a gradient clipping threshold of 1.0. Training is performed on four NVIDIA A100 GPUs with an effective batch size of 32 for 8 epochs. No synthetic or LLM-generated training data are used. Structured fine-tuning is conducted primarily on the original GSM8K training set, with an additional independent training-side validation performed on MATH.

D.2 Structural and Detail Token Construction

Section 5.1 describes the deterministic structural partition used during training. Here we provide additional implementation details for the main GSM8K experiments. For each original solution text, we apply the regular expression `Step \d+ [-:]` to identify step-heading spans. The continuous token span from each matched heading through the first period of the corresponding sentence is assigned to the structural set $\mathcal{S}$, while all remaining response tokens outside the prompt are assigned to the detail set $\mathcal{T}$.

The original response sequence is retained unchanged: the rule only derives token-level metadata used to determine region-dependent corruption probabilities and loss weights. Prompt tokens remain fixed and are excluded from the response partition. The rule is applied automatically to the existing GSM8K solutions and requires no additional human annotation. Structural annotations are used only during training and are not required by the entropy-gated decoder at inference time.

As a concrete illustration, consider the following GSM8K solution: "Step 1 - Calculate the total cost. The cassettes cost $9 each. Step 2 - Add the additional items. The headphones cost $25." The structural spans are "Step 1 - Calculate the total cost." and "Step 2 - Add the additional items.", while the remaining response tokens belong to the detail set.

D.3 Key Hyperparameters and Training Configuration

For the main GSM8K structured fine-tuning experiments, the structural noise-scaling coefficient is set to $s_{\mathcal{S}} = 0.45$, the detail noise-scaling coefficient to $s_{\mathcal{T}} = 1.5$, and the minimum masking probability for structural tokens to $p_{\text{floor}} = 0.15$. The structural loss weight is set to $\lambda = 2$, based on the parameter analysis in Section 6.2.

Table 5: Controlled comparison of entropy and maximum-probability ranking on GSM8K under identical per-step fixation budgets.

Method	# Samples	Strict EM	Δ EM	95% CI / p
MaxProb ranking	100	32.0%	–	–
Entropy ranking	100	**70.0%**	**+38.0 pp**	$[+26.0, +49.0]$ pp, $p = 1.37 \times 10^{-8}$

We use a learning rate of 2×10^{-6} with cosine annealing and a warmup ratio of 5% of the total training steps. The training diffusion schedule is linear, with 128 diffusion steps and a lower time bound of $\epsilon = 10^{-3}$. These settings are selected through a small-scale search on the GSM8K validation set and are used throughout the main GSM8K experiments unless otherwise stated. Hyperparameter sensitivity analyses are reported in Appendix C.1.

D.4 Inference Configuration and Controlled Baseline

For entropy-gated decoding, we set $\tau_{\text{start}} = 0.1$, $\tau_{\text{end}} = 2.0$, the maximum fixation ratio to $\eta = 0.4$, and the minimum fixation count to $N_{\min} = 1$. Unless otherwise stated, entropy-gated inference uses 128 denoising steps. Fixed token values are selected greedily using $\arg\max$, with temperature 0 and no classifier-free guidance.

The entropy-gated decoder operates in full-sequence mode without block boundaries. At each denoising step, all currently masked response positions are considered jointly for entropy-based candidate selection and fixation. The main inference experiments are conducted on four GPUs.

All compared configurations use the same prompts, formatting, preprocessing, answer format, sample order, parser, and EOS/post-processing pipeline, without method-specific prompting or truncation. Decoding-specific parameters are varied only as explicitly described for each configuration.

The original LLaDA decoder is blockwise. For the controlled baseline used in the main end-to-end comparison, we set `block_length` = `gen_length` = 128 with 128 denoising steps, so that the entire response region is processed as a single generation block. This places the controlled baseline and our entropy-gated decoder over the same full generation region while preserving their respective fixation schedules. Accordingly, this baseline should not be interpreted as an unchanged reproduction of the original blockwise LLaDA decoding procedure. Its complete configuration is summarized in Table 4.

D.5 Controlled Ranking Comparison: Entropy vs. MaxProb

To isolate the effect of the ranking criterion, we compare entropy ranking against maximum-probability (MaxProb) ranking under an identical per-step fixation schedule. We first run the entropy-based decoder and record the number of positions fixed at each denoising step, denoted by $\{k_t\}$. The MaxProb baseline reuses exactly the same $\{k_t\}$ and differs only in the ranking score used to select which positions are fixed. Thus, both methods commit the same number of tokens at every step.

Under this matched schedule, entropy ranking improves strict EM from 32.0% to 70.0%. The paired-bootstrap 95% confidence interval for the difference is $[+26.0, +49.0]$ percentage points, and an exact McNemar test gives $p = 1.37 \times 10^{-8}$. Because the per-step fixation counts are held fixed, this comparison isolates the ranking criterion under the evaluated schedule and shows that predictive entropy provides an ordering signal distinct from maximum-probability confidence. This result does not imply universal superiority across alternative fixation schedules.

D.6 Independent Training-Side Validation on MATH

To examine whether the proposed training-time uncertainty-shaping mechanism is specific to the GSM8K training distribution, we conduct an independent fine-tuning experiment on MATH. This experiment is performed separately from the GSM8K training pipeline and does not affect any result reported in the main GSM8K or cross-task evaluations.

Table 6: Independent training-side validation on MATH.

Configuration	Accuracy (%)
LLaDA-8B Base + Controlled Baseline Decoder	26.1
LLaDA-8B Base + Entropy-Gated Decoder	28.3
Structure-Aware FT ($\lambda = 2$) + Entropy-Gated Decoder	**37.4**

Table 7: Aligned comparison with EB-Sampler on 100 GSM8K examples. Wall-clock time is measured over the same evaluation set under the aligned four-GPU setting.

Method	Strict EM (%)	Wall-Clock Time (s)	Relative Speed
EB-Sampler (Ben-Hamu et al., 2025)	75.0	4325.43	1.00×
Ours	**82.0**	**1693.29**	**2.55**×

We retain the original MATH training examples and target solutions without generating or rewriting response text. A deterministic token-level structural partition is applied to the existing solutions, and the same differential-corruption and structure-aware loss-reweighting objective is used for fine-tuning. Unless otherwise stated, the training and decoding configurations follow those used in the main experiments.

As shown in Table 6, replacing the controlled baseline decoder with entropy-gated decoding improves MATH accuracy from 26.1% to 28.3%, corresponding to an 8.4% relative improvement with the pretrained model held fixed. Applying the same training-time uncertainty-shaping procedure on MATH further increases accuracy to 37.4%, yielding a 32.2% relative improvement over entropy-gated decoding alone and a 43.3% relative improvement over the controlled baseline.

These results provide an independent validation of the training-side intervention on a second training distribution. In particular, the additional gain after applying structure-aware fine-tuning suggests that the benefit of uncertainty shaping is not restricted to the GSM8K fine-tuning setting. We treat this experiment as cross-dataset evidence for the proposed training–inference coupling rather than as a claim of universal generalization across tasks or structural partitions.

D.7 Aligned Comparison with EB-Sampler

To provide a controlled external comparison with an existing entropy-based adaptive unmasking method, we evaluate EB-Sampler (Ben-Hamu et al., 2025) under an aligned setting using the same LLaDA-8B backbone, the same 100 GSM8K examples, the same evaluation pipeline, and the same four-GPU hardware setting. The sampler-specific decoding rules are retained for each method.

Under this aligned evaluation, our framework improves strict EM from 75.0% to 82.0%, corresponding to a gain of 7.0 percentage points, while reducing total wall-clock time from 4325.43 s to 1693.29 s ($2.55\times$ faster).

This comparison distinguishes the proposed training–inference framework from an existing entropy-based adaptive unmasking method under a matched evaluation setting. In particular, our method does not only use uncertainty as an inference-time selection signal: the predictive uncertainty used for fixation is additionally shaped during training through differential corruption and structure-aware loss reweighting. We treat this experiment as an aligned comparison with the closest entropy-based decoding baseline, rather than as a broad ranking across diffusion samplers.

D.8 Structural-Partition Control

To test whether the benefit of the structural partition can be explained simply by preferentially treating an arbitrary subset of response tokens, we construct a length-matched random-span control. For each reasoning step, the structural label is reassigned from the original structural span to a continuous span of equal length sampled from the corresponding step body. The original response text remains unchanged.

Table 8: Controlled comparison between the deterministic structural partition and length-matched random body spans on 500 GSM8K examples.

Training Partition	Strict EM (%)	Δ EM	95% CI / p
Length-matched random body span	72.0	–	–
Structural partition	**79.4**	**+7.4 pp**	$[+2.1, +12.7]$ pp, $p = 0.006$

Table 9: Inference-time seed sensitivity on GSM8K.

Configuration	Strict EM (%)
LLaDA decoder, remasking seed 1	66.0
LLaDA decoder, remasking seed 2	66.0
LLaDA decoder, remasking seed 3	67.0
LLaDA decoder mean $\pm$ std.	66.3 ± 0.58
Entropy-Gated Decoder	70.0

Apart from the assignment of structural metadata, the training data, number of selected tokens, corruption scheme, loss weighting, optimization, generation, and evaluation settings are kept identical. The structural partition improves strict EM by 7.4 percentage points over the length-matched random control. The approximate 95% confidence interval for the difference is $[+2.1, +12.7]$ percentage points, and a two-proportion test gives $p = 0.006$.

Because the control matches the number of preferentially treated tokens and their membership within the same reasoning steps, the observed gain cannot be explained solely by token count or step membership. This experiment supports the usefulness of the task-dependent structural prior, but does not establish that the particular partition used here is universally optimal.

D.9 Inference-Time Variance

The LLaDA baseline uses stochastic remasking, so we additionally evaluate its inference-time variability across three independent remasking seeds. In contrast, the entropy-gated decoder uses greedy prediction, deterministic entropy ranking, and deterministic fixation; with the checkpoint, prompt, and input fixed, it has no analogous inference-time sampling variation.

Across the three remasking seeds, the LLaDA baseline achieves 66.0%, 66.0%, and 67.0% strict EM, with a mean of 66.3 ± 0.58. The entropy-gated decoder obtains 70.0% under its deterministic inference configuration. The decoder-only improvement therefore exceeds the observed variation caused by stochastic remasking in the baseline.

This analysis characterizes inference-time randomness only. The 81.5% result obtained after structure-aware fine-tuning comes from a single training seed and therefore does not establish robustness across independent training runs.