

Locally Confident, Globally Stuck: The Quality-Exploration Dilemma in Diffusion Language Models

Liancheng Fang¹, Aiwei Liu^{2†}, Henry Peng Zou¹, Yankai Chen^{3,4}, Enze Ma¹, Leyi Pan², Chunyu Miao¹, Wei-Chieh Huang¹, Xue (Steve) Liu^{3,4}, Philip S. Yu¹

¹University of Illinois Chicago, ²Tsinghua University, ³MBZUAI, ⁴McGill University
{lfang87, psyu}@uic.edu

Abstract

Diffusion large language models (dLLMs) theoretically permit token decoding in arbitrary order, a flexibility that could enable richer exploration of reasoning paths than autoregressive (AR) LLMs. In practice, however, random-order decoding often hurts generation quality. To mitigate this, low-confidence remasking improves single-sample quality (e.g., Pass@1) by prioritizing confident tokens, but it also suppresses exploration and limits multi-sample gains (e.g., Pass@ k), creating a fundamental *quality-exploration dilemma*. In this paper, we provide a unified explanation of this dilemma. We show that low-confidence remasking improves a myopic proxy for quality while provably constraining the entropy of the induced sequence distribution. To overcome this limitation, we characterize the optimal distribution that explicitly balances quality and exploration, and develop a simple *Independent Metropolis-Hastings* sampler that approximately targets this distribution during decoding. Experiments across a range of reasoning benchmarks including MATH500, AIME24/25, HumanEval, and MBPP show that our approach yields better exploration-quality tradeoff than both random and low-confidence remasking.

1 Introduction

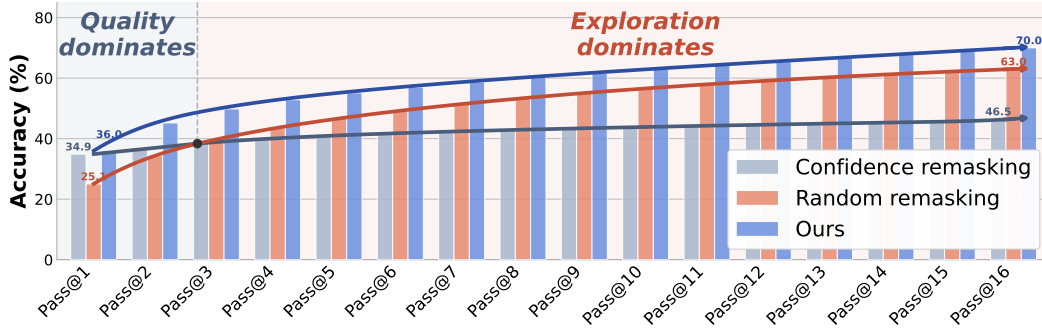


Figure 1: **The quality-exploration dilemma in dLLM decoding.** Confidence remasking achieves high sample quality (Pass@1) but plateaus Pass@ k quickly due to limited exploration. Conversely, random remasking promotes exploration but degrades individual sample quality. Our *global tempering* reconciles this trade-off, establishing a new Pareto frontier with superior Pass@1 and Pass@16 performance.

Diffusion large language models (dLLMs) generate text by iteratively denoising an entire token sequence in parallel (Lou et al., 2023; Shi et al., 2024; Ou et al., 2024; Sahoo et al., 2024; Nie et al., 2025; Ye et al., 2025; Song et al., 2025). Unlike autoregressive (AR) LLMs (Radford et al., 2018; Raffel et al., 2020; Brown et al., 2020), which commit tokens irrevocably in a

[†]Corresponding author. Main contact: liuaiwei20@gmail.com.

fixed left-to-right order, dLLMs may finalize tokens at arbitrary positions. This flexibility is intrinsically aligned with the non-linear nature of complex reasoning, where pivotal decisions need not adhere to a strict left-to-right progression (Bachmann & Nagarajan, 2024; Nagarajan et al., 2025; Kim et al., 2025a; Yang et al., 2025; Trainin et al., 2026).

In practice, however, realizing this advantage has proven difficult. Recent work (Ni et al., 2026; Shen et al., 2026; Chen et al., 2025; Fu et al., 2025; Lee et al., 2025) points to a fundamental *quality–exploration dilemma* (Figure 1): low-confidence remasking strategies, which commit the model’s most certain predictions first (Wu et al., 2025; Kim et al., 2025a; Ben-Hamu et al., 2025), improve single-sample quality (Pass@1) but saturate quickly under repeated sampling (Pass@k) (Ni et al., 2026; Shen et al., 2026), revealing a severe exploration bottleneck. Random remasking exhibits the opposite behavior: it explores more broadly, yet produces weaker individual samples. This tension motivates our central question:

Can a dLLM decoding strategy achieve high per-sample quality without collapsing exploration?

In this paper, we do so by deriving the target distribution that optimally balances quality and exploration, then designing a new decoding strategy that approximately samples from the optimal distribution. The strategy favors globally promising sequences through a lookahead correction that adjusts each local token choice according to *how promising the resulting space of completions is*. We show that this strategy yields a better exploration-quality tradeoff than both uncertainty-based remasking and random remasking. Our contributions are as follows:

1. We provide a unified explanation of the exploration bottleneck in uncertainty-based decoding, where we show that the shared mode-seeking behavior of uncertainty-based decoding improves a myopic quality proxy while imposing a formal entropy cap on the induced sequence distribution.
2. We formalize the quality–exploration trade-off as an entropy-regularized optimization problem over the joint distribution and characterize the unique optimal solution.
3. We design a practical Markov chain Monte Carlo (MCMC) algorithm based on *Independent Metropolis–Hastings* that efficiently approximates this target distribution during decoding using a tractable lookahead correction.
4. Extensive experiments with LLaDA (Nie et al., 2025) and WeDLM (Liu et al., 2025) on reasoning benchmarks (MATH500, AIME, HumanEval, MBPP) demonstrate that our approach consistently achieves superior exploration–quality trade-offs.

2 Preliminaries

Diffusion Language Models. Let $\mathcal{V}$ be a finite vocabulary and $[M]$ a special mask token. We consider sequences of length L with index set $[L] := \{1, \dots, L\}$. A (masked) diffusion language model generates samples by learning to reverse a *forward process* that corrupts a clean sequence $\mathbf{x}_0 \in \mathcal{V}^L$ into a partially masked sequence $\mathbf{x}_t \in (\mathcal{V} \cup \{[M]\})^L$ by independently masking each token with probability $t \in [0, 1]$ (linear masking schedule). The *reverse process* is parameterized by a mask predictor $p_\theta(\cdot \mid \mathbf{x}_t)$ that independently predicts all masked tokens from the corrupted input. It is trained to maximize an evidence lower bound (ELBO) on the data log-likelihood, which reduces to a weighted denoising cross-entropy on masked positions (Shi et al., 2024; Ou et al., 2024; Zheng et al., 2024):

$$\mathcal{L}(\theta) := -\mathbb{E}_{\mathbf{x}_0, t, \mathbf{x}_t} \left[\frac{1}{t} \sum_{i \in [L]} \mathbf{1}[x_t^i = [M]] \log p_\theta(x_0^i \mid \mathbf{x}_t) \right]. \quad (1)$$

The sampling process iteratively refines a fully masked sequence. At each step, the model predicts marginal distributions for all masked positions simultaneously, conditioned on the partially revealed sequence. It then unmask a subset of these positions by sampling from the marginals independently. Repeating this process gradually yields a complete sequence. Two main strategies determine this subset:

Table 1: **Taxonomy of uncertainty heuristics for uncertainty-based remasking.** We categorize heuristics by their distributional scoring functions ($p_i(\cdot) := p(\cdot \mid s_t, i)$) and operational paradigms (*Sample-then-Filter* vs. *Rank-then-Sample*). The δ -gating column establishes the equivalent confidence lower bound $\max_v p_i(v) \geq 1 - \delta$ for each criterion. See Appendix D for detailed definitions and formal derivations of these bounds.

Heuristic	Commit criterion	Mode	δ -gating
Confidence (Nie et al., 2025; Ye et al., 2025)	$\max_v p_i(v) \geq 1 - \delta$	Sample-then-Filter	δ (directly)
Entropy (Ben-Hamu et al., 2025; Fu et al., 2025)	$H(p_i) \leq \epsilon$	Rank-then-Sample	$\delta = \delta_V(\epsilon)$
Margin (Kim et al., 2025a; Hong et al., 2025b)	$p_i(v_1) - p_i(v_2) \geq \gamma$	Rank-then-Sample	$\delta = \frac{(V -1)(1-\gamma)}{ V }$

1. *Random remasking*: selects positions to unmask uniformly at random and remask the remaining masked positions, which is theoretically grounded in τ -leaping (Campbell et al., 2022; Lou et al., 2023; Ou et al., 2024).
2. *Uncertainty-based remasking*: selects unmask positions with high token or distribution uncertainty scores. See Table 1 for a taxonomy of common uncertainty heuristics.

Pass@k as Exploration Metric. Pass@k measures the probability of sampling at least one correct solution among k independent generations, serving as a standard proxy for a model’s exploration capability (Yue et al., 2025). Given c correct solutions out of n total samples, its unbiased estimator is given by (Chen et al., 2021):

$$\text{Pass@}k = \mathbb{E} \left[1 - \frac{\binom{n-c}{k}}{\binom{n}{k}} \right]. \quad (2)$$

A low Pass@k indicates a fundamental barrier in exploring valid reasoning trajectories.

3 Formalizing the Quality–Exploration Dilemma

To formalize the quality–exploration dilemma, we begin by unifying common uncertainty heuristics under the notion of *confidence gating*.

Definition 1 (Confidence gating). *Let V denote the token vocabulary and s_t the decoder state at step t . For a threshold parameter $\delta \in (0, 1)$, a decoder is $(1 - \delta)$ -gated if, whenever it commits a token, the commit distribution satisfies*

$$\max_{v \in V} p(v \mid s_t) \geq 1 - \delta. \quad (3)$$

Intuitively, confidence gating permits a token to be committed only when its marginal distribution is sufficiently peaked. Although stated in terms of confidence, this definition naturally subsumes other common heuristics: entropy and margin-based criteria all reduce to this form under appropriate choices of δ (see Table 1 for details).

3.1 The Local Benefit: Improving Quality via Myopic Optimization

To elucidate why uncertainty-based decoding enhances single-sample performance (e.g., Pass@1), we analyze its implicit objective: the expected log loss under a reference model. Let q_{gen} denote the joint distribution over generated sequences, $\sigma \in S_L$ the decoding trajectory (where S_L is the permutation group over $1, \dots, L$), and p_{ref} a scoring model. We define the trajectory-dependent generation loss as:

$$\mathcal{L}_{\text{gen}}(q_{\text{gen}}; p_{\text{ref}}; \sigma) := \mathbb{E}_{x \sim q_{\text{gen}}} \left[\frac{1}{L} \sum_{t=1}^L -\log p_{\text{ref}}(x_{\sigma_t} \mid x_{\sigma_{<t}}) \right]. \quad (4)$$

Exponentiating this expected loss yields *generative perplexity* (GenPPL), a standard metric for evaluating the generative quality of diffusion models (Lou et al., 2023; Arriola et al., 2025;

Sahoo et al., 2025):

$$\text{GenPPL}(q_{\text{gen}}; p_{\text{ref}}; \sigma) := \exp(\mathcal{L}_{\text{gen}}(q_{\text{gen}}; p_{\text{ref}}; \sigma)). \quad (5)$$

Intuitively, GenPPL measures the sequence-level surprisal under the reference model. By setting $p_{\text{ref}} = p_{\theta}$, we evaluate the model’s self-consistency along the generation trajectory. Under this self-scoring regime, the expected generation loss decomposes into a sum of step-wise conditional entropies. Specifically, when the decoder commits a token by sampling from a chosen position’s marginal distribution $p_{\theta}(\cdot | s_t)$, the expected one-step loss incurred is exactly the entropy of that distribution: $\mathbb{E}_x[-\log p_{\theta}(x | s_t)] = H(p_{\theta}(\cdot | s_t))$. Consequently, confidence gating—which prioritizes highly peaked, low-entropy distributions—acts as a myopic, greedy heuristic to minimize $\mathcal{L}_{\text{gen}}$. Proposition 1 formalizes this mechanism, proving that bounding the step-wise entropy strictly bounds the global generative perplexity.

Proposition 1 (Generative perplexity upper bound under confidence gating). *Assume the decoder is $(1 - \delta)$ -gated, the committed token at each step is sampled from the decoder’s commit distribution, and the scoring model is the decoder itself, i.e., $p_{\text{ref}} = p_{\theta}$. Then for any decoding trajectory σ ,*

$$\text{GenPPL}(q_{\text{gen}}; p_{\theta}; \sigma) \leq \exp(h_V(\delta)),$$

where $h_V(\delta) = h_b(\delta) + \delta \log(|V| - 1)$ and $h_b(\delta) = -\delta \log \delta - (1 - \delta) \log(1 - \delta)$.

The bound $\exp(h_V(\delta))$ corresponds precisely to the maximum entropy of any vocabulary distribution whose largest probability mass is at least $1 - \delta$. By controlling the worst-case one-step uncertainty, confidence gating effectively bounds the accumulated self-cross-entropy along the generation trajectory. This provides a principled explanation for why uncertainty-based heuristics improve single-sample quality. Nevertheless, because each commit decision fundamentally alters the context for future steps, this approach remains a *myopic* surrogate rather than a direct optimization of the global sequence-level objective.

3.2 The Global Cost: Suppressing Exploration via Entropy Collapse

While the mode-seeking behavior of confidence gating improves single-sample quality, it inherently restricts sequence-level diversity, which ultimately hurts multi-sample performance (e.g., Pass@k). Proposition 2 formalizes this limitation by bounding the sequence entropy $H(X)$ and the effective branching factor (Jurafsky & Martin, 2026).

Proposition 2 (Entropy cap under confidence gating). *Assume the dLLM decoder is $(1 - \delta)$ -gated. Then the sequence entropy satisfies*

$$H(X) \leq L \cdot h_V(\delta),$$

where $h_V(\delta) = h_b(\delta) + \delta \log(|V| - 1)$ and $h_b(\delta) = -\delta \log \delta - (1 - \delta) \log(1 - \delta)$ is the binary entropy function. Equivalently, the effective branching factor, defined as the per-token perplexity of the induced sequence distribution,

$$B_{\text{eff}} := \exp(H(X)/L),$$

is upper bounded as

$$B_{\text{eff}} \leq \exp(h_V(\delta)).$$

Proposition 2 establishes a hard *entropy budget* that depends only on the gating threshold δ and the vocabulary size $|V|$. The induced quantity B_{eff} , which measures the geometric-mean number of effective tokens available at each decoding step, is sharply constrained under strong gating. As a concrete instantiation, setting $|V| = 5 \times 10^4$ and $\delta = 0.05$ yields $B_{\text{eff}} < 2.1$, indicating that the decoder concentrates almost all probability mass on fewer than three candidates per step. This tight ceiling explains the poor Pass@k scaling: even if the model internally supports multiple correct reasoning paths, a strongly gated decoder cannot simultaneously explore these valid modes.

4 Principled Exploration via Global Tempering

4.1 The Optimal Target Distribution

Motivated by the limitations of local heuristics discussed in Section 3, we now formalize the quality–exploration trade-off at the level of complete-sequence distributions. We define generation quality as the expected log-likelihood under the base model, $\mathbb{E}_{x \sim p}[\log q(x)]$, and exploration as the Shannon entropy $H(p)$ of the sequence distribution. Rather than relying on step-wise proxies, we optimize directly over the joint distribution p , yielding the entropy-regularized objective:

$$\max_{p \in \Delta(\mathcal{X})} \underbrace{\alpha \mathbb{E}_{x \sim p}[\log q(x)]}_{\text{quality}} + \underbrace{H(p)}_{\text{exploration}}, \quad (6)$$

where $\alpha \geq 0$ controls the trade-off: larger α places more mass on high-likelihood sequences, while the entropy term discourages mode collapse.

Proposition 3 (Optimality of the power distribution). *For any $\alpha \geq 0$, the unique optimizer of equation 6 is the power distribution*

$$p_\alpha^*(x) = \frac{q(x)^\alpha}{Z_\alpha}, \quad Z_\alpha = \sum_{x \in \mathcal{X}} q(x)^\alpha.$$

Formally, p_α^* applies a *global* temperature scaling to the joint sequence distribution $q(x)$, flattening ($\alpha < 1$) or sharpening ($\alpha > 1$) the sequence-level distribution. Unlike conventional local logit tempering, p_α^* operates over the entire sequence. We now address the resulting algorithmic challenge: efficiently sampling from this globally tempered distribution.

4.2 Corrected Conditionals: Exact Form and Tractable Approximation

Sampling from the globally tempered distribution $p_\alpha^*(x) \propto q(x)^\alpha$ presents two major obstacles in the dLLM setting. First, the normalizing constant Z_α is intractable to compute over the exponentially large sequence space $\mathcal{X} = \mathcal{V}^L$. While autoregressive LLMs can circumvent this issue using Markov chain Monte Carlo (MCMC) samplers that only require evaluating unnormalized sequence likelihoods (Karan & Du, 2025), dLLMs face a second, more severe hurdle: their exact sequence likelihood $q(x)$ is itself intractable (Shi et al., 2024; Pan et al., 2025). Consequently, standard sequence-level MCMC techniques cannot be directly applied.

To bypass the need for evaluating intractable sequence-level likelihoods, we shift our perspective from the global joint distribution to the local one-step generation. Fundamentally, the generative process of a dLLM samples this joint distribution by iteratively sampling from a sequence of marginal conditionals. A natural question thus arises: for a global tempered joint distribution, what is the exact form of the resulting marginal conditionals at each decoding step? We can analytically derive this marginal conditional below:

Proposition 4 (Corrected conditional for global tempering). *Let q be a distribution over $\mathcal{X} = \mathcal{V}^L$ with full support, and fix $\alpha > 0$. Let $s = (A, x_A)$ denote a partially decoded state with committed positions $A \subseteq [L]$ and remaining positions $R(s) := [L] \setminus A$. For any uncommitted position $i \in R(s)$ and token $v \in \mathcal{V}$, let $\ell_i(s)$ be the logit vector such that $q(x_i = v | s) = \text{softmax}(\ell_i(s))_v$. Letting $s' = s \oplus (i, v)$ denote the updated state after committing token v at position i , define*

$$\Delta_{i,v}(s) := \log \sum_{x_{R(s')} \in \mathcal{V}^{|R(s')|}} q(x_{R(s')} | s')^\alpha. \quad (7)$$

Then the conditional of $p_\alpha^*(x) \propto q(x)^\alpha$ at position i is

$$p_\alpha^*(x_i = v | s) = \frac{\exp(\alpha \ell_{i,v}(s) + \Delta_{i,v}(s))}{\sum_{u \in \mathcal{V}} \exp(\alpha \ell_{i,u}(s) + \Delta_{i,u}(s))}. \quad (8)$$

Equation (8) features two interpretable terms: a **local tempering** term $\alpha \ell_{i,v}(s)$ and a **suffix lookahead correction** $\Delta_{i,v}(s)$, which measures the total tempered probability mass of the completions reachable after committing v . Crucially, $\Delta_{i,v}(s)$ does not simply favor the most locally confident token. Instead, it favors tokens that preserve more *globally promising continuations*. This is why it can mitigate the entropy cap of confidence-based decoding: confidence gating suppresses exploration by forcing low entropy at the current step, whereas the lookahead correction evaluates each token by the value of the continuation space it leaves available, avoiding premature collapse onto a single reasoning path.

Tractable approximation via mean-field factorization. Evaluating the exact lookahead correction $\Delta_{i,v}(s)$ is computationally prohibitive due to the exponential cardinality of the suffix space. However, the mask-prediction objective of dLLMs inherently models the sequence via conditionally independent marginals given the current state (Equation (1)). This structural property motivates a mean-field approximation (Zhao et al., 2025), allowing us to factorize the intractable suffix joint as $q(x_{R(s')} | s') \approx \prod_{j \in R(s')} q(x_j | s')$. This factorization decouples the α -power summation along the sequence length:

$$\hat{\Delta}_{i,v}(s) := \sum_{j \in R(s')} \log \left(\sum_{u \in V} q(x_j = u | s')^\alpha \right), \quad s' = s \oplus (i, v). \quad (9)$$

Consequently, the surrogate correction $\hat{\Delta}_{i,v}(s)$ becomes readily computable. It requires only a *single* network evaluation at the proposed state s' to obtain the requisite marginals. Plugging this approximation back into equation 8 yields a one-step sampling target:

$$\pi_i(v | s) \propto \exp(\alpha \ell_{i,v}(s) + \hat{\Delta}_{i,v}(s)), \quad v \in V. \quad (10)$$

A controlled analysis of the mean-field approximation error is deferred to Appendix C.2.

4.3 Independent Metropolis–Hastings with Batched Lookahead

Given the pointwise evaluable one-step target $\pi_i(\cdot | s)$, we employ Markov Chain Monte Carlo (MCMC) to sample from this unnormalized distribution. Specifically, we propose an independent Metropolis–Hastings (IMH) sampler (Hastings, 1970). For a fixed decoding state s and position i , we define the independence proposal as the locally tempered logits without correction:

$$r_i(v | s) := \text{softmax}(\alpha \ell_i(s))_v. \quad (11)$$

This proposal distribution is chosen for two primary reasons. First, we empirically find that it captures the dominant probability mass of the target distribution (see Table 3 for validation). Second, because the target factorizes as $\pi_i(v | s) \propto r_i(v | s) \exp(\hat{\Delta}_{i,v}(s))$, the local $\alpha \ell$ terms exactly cancel within the Metropolis–Hastings ratio (see Appendix E for the detailed derivation), yielding a highly streamlined acceptance probability:

$$A(x \rightarrow y) = \min\{1, \exp(\hat{\Delta}_{i,y}(s) - \hat{\Delta}_{i,x}(s))\}. \quad (12)$$

Consequently, the IMH sampler generates candidates from r_i and accepts them via Equation (12), relying solely on the difference in suffix corrections without requiring vocabulary-wide normalization. The full algorithm is presented in Algorithm 1.

Algorithm 1 Batched IMH for one-token corrected sampling

Require: State s ; position $i \in R(s)$; number of proposals T ; exponent α
Ensure: A token approximately distributed as $\pi_i(\cdot | s)$ in equation 10

- 1: Sample $x \sim r_i(\cdot | s)$ using equation 11
- 2: Draw i.i.d. proposals $y_1, \dots, y_T \sim r_i(\cdot | s)$
- 3: Compute $\hat{\Delta}_{i,x}(s)$ and $\hat{\Delta}_{i,y_t}(s)$ for $t \in \{1, \dots, T\}$ in one batch
- 4: **for** $t = 1, \dots, T$ **do**
- 5: $a \leftarrow \min\{1, \exp(\hat{\Delta}_{i,y_t}(s) - \hat{\Delta}_{i,x}(s))\}$
- 6: Sample $u \sim \text{Uniform}(0, 1)$
- 7: **if** $u < a$ **then**
- 8: $x \leftarrow y_t$
- 9: $\hat{\Delta}_{i,x}(s) \leftarrow \hat{\Delta}_{i,y_t}(s)$
- 10: **end if**
- 11: **end for**
- 12: **return** x

Table 2: **Pass@1 and Pass@k across benchmarks and models.** Bold indicates the best result for a given metric. LLaDA-8B results on AIME24/25 are omitted as they are close to 0.

Method	Metric	Math reasoning			Code generation			
		MATH500	AIME'24	AIME'25	HumanEval	HumanEval+	MBPP	MBPP+
WeDLM-8B								
Entropy [†]	Pass@1	0.528	0.093	0.071	0.732	0.684	0.784	0.677
	Pass@k	0.875	0.310	0.291	0.950	0.911	0.916	0.739
Confidence	Pass@1	0.528	0.079	0.074	0.725	0.678	0.707	0.598
	Pass@k	0.851	0.323	0.292	0.905	0.857	0.834	0.717
Margin	Pass@1	0.393	0.058	0.031	0.584	0.537	0.589	0.490
	Pass@k	0.838	0.296	0.269	0.903	0.843	0.765	0.643
IMH (ours)	Pass@1	0.540	0.095	0.074	0.745	0.696	0.776	0.664
	Pass@k	0.875	0.344	0.344	0.962	0.933	0.938	0.836
LLaDA-8B								
Confidence [†]	Pass@1	0.349	-	-	0.410	0.372	0.484	0.404
	Pass@k	0.465	-	-	0.573	0.536	0.645	0.521
Entropy	Pass@1	0.294	-	-	0.387	0.344	0.444	0.387
	Pass@k	0.606	-	-	0.670	0.603	0.751	0.656
Margin	Pass@1	0.286	-	-	0.350	0.315	0.451	0.388
	Pass@k	0.594	-	-	0.610	0.573	0.716	0.589
Random	Pass@1	0.251	-	-	0.196	0.176	0.279	0.251
	Pass@k	0.630	-	-	0.549	0.524	0.685	0.605
IMH (ours)	Pass@1	0.360	-	-	0.385	0.345	0.469	0.399
	Pass@k	0.700	-	-	0.695	0.640	0.778	0.672

[†] Default decoding method of the respective model.

Batched evaluation. A primary computational advantage of our IMH sampler is that the proposal distribution r_i depends strictly on the outer context (s, i) , entirely decoupled from the Markov chain’s current state. This permits all T proposals and their associated lookahead corrections to be evaluated concurrently in a *single* batched forward pass, avoiding additional sequential decoding steps. Because evaluating multiple proposals still consumes extra forward-pass compute, Appendix C.1 compares IMH with alternative test-time scaling strategies under matched compute to isolate the benefit of the sampler itself.

MCMC mixing. Traditional MCMC algorithms frequently suffer from slow mixing when deployed in high-dimensional discrete spaces. Our IMH sampler circumvents this bottleneck through two structural advantages. First, by formulating the Markov chain to target a 1D categorical distribution over the vocabulary $\mathcal{V}$, we completely bypass the curse of dimensionality. Second, since our proposal distribution r_i tightly approximates the target by incorporating dominant local logits, standard MCMC theory guarantees rapid geometric convergence (Mengersen & Tweedie, 1996). We empirically validate both points in Section 5.

5 Experiments

Experimental Setup. We evaluate our method on reasoning-intensive tasks across two domains: mathematical problem-solving (MATH500 (Hendrycks et al., 2020), AIME 2024 (Zhang & Math-AI, 2024), AIME 2025 (Zhang & Math-AI, 2025)) and code generation (HumanEval (Chen et al., 2021), MBPP (Austin et al., 2021)). Our evaluation framework incorporates two architecturally distinct dLLMs: the fully bidirectional LLaDA-8B-Instruct (Nie et al., 2025) and the block-diffusion WeDLM-8B (Liu et al., 2025). We benchmark against random remasking and standard confidence-based heuristics (confidence, entropy, and margin; see Table 1 for definitions). For WeDLM, random remasking is excluded due to its inherent left-to-right generation bias (Liu et al., 2025), and we restrict its suffix lookahead correction to a 16-token window to align with its block-diffusion mechanics.

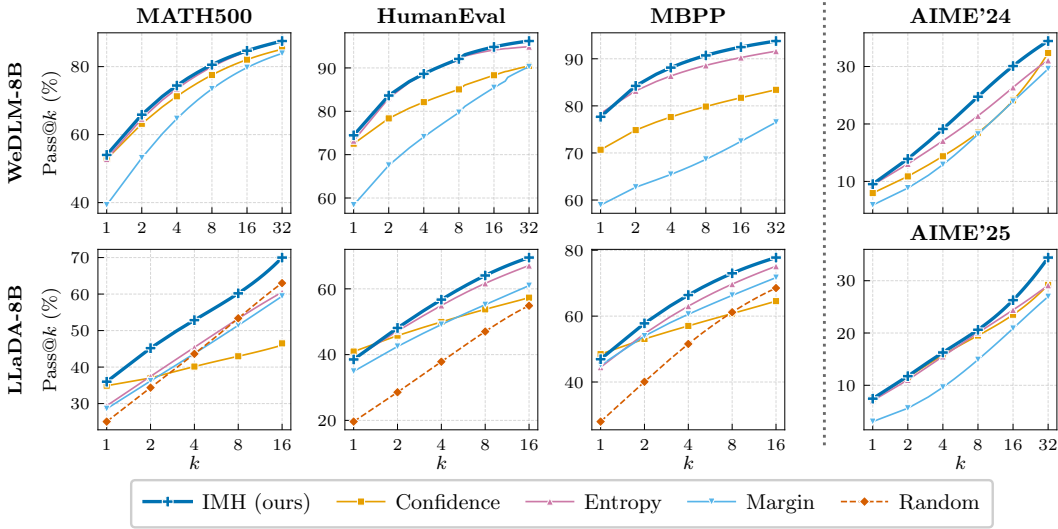


Figure 2: **Pass@k scaling curves.** (Left) Performance on MATH500, HumanEval, and MBPP for WeDLM-8B (top) and LLaDA-8B (bottom). (Right) WeDLM-8B on AIME’24/25.

We adopt Pass@k (Equation (2)) as our primary metric. To guarantee a statistically robust, low-variance estimation, we systematically oversample: generating $n = 32$ samples for $k \leq 16$ with LLaDA, and $n = 128$ for $k \leq 32$ with WeDLM (see Appendix F for variance analysis). Finally, to ensure a strictly hyperparameter-free comparison, all baselines decode exactly one token per step. More experimental details are provided in Appendix B, and robustness to the maximum decoding length is evaluated in Appendix C.3.

Benchmarks. In Table 2, we report Pass@1 and Pass@k across all benchmarks using a shared local temperature of 0.5 (Ye et al., 2024). This reflects the practical setting where users select a specific temperature and allow for direct comparison under controlled conditions. First, IMH consistently delivers the strongest exploratory performance, attaining the best Pass@k on nearly every task for both WeDLM and LLaDA. Figure 2 reinforces this conclusion by showing the full Pass@k scaling curves. Notably, on the more challenging AIME24/25 benchmarks, IMH substantially outperforms WeDLM’s default entropy heuristic in both Pass@1 and Pass@k. Second, for LLaDA, IMH offers the most favorable overall trade-off between quality and exploration. On MATH500, it achieves the best results on both metrics. On HumanEval and MBPP, while low-confidence remasking attains slightly higher Pass@1, this marginal advantage comes at a substantial cost in Pass@k, making IMH clearly preferable when broader exploration is desired.

The Quality-Diversity Pareto Frontier. We further validate the robustness of this exploration-exploitation trade-off across the various hyperparameters. Figure 3 (left) plots Pass@1 versus Pass@k on MATH500 by sweeping the temperature parameters (α for IMH, τ for baselines). These trajectories reveal that global tempering strictly dominates local strategies. While local methods struggle to balance exploiting high-likelihood paths with exploring diverse reasoning, IMH consistently pushes the Pareto frontier outward. This empirical superiority directly corroborates the theoretical optimality of the power distribution (Proposition 3), establishing global tempering as a fundamentally more effective mechanism for navigating the exploration-exploitation trade-off.

Expanding the Reasoning Boundary. While aggregate Pass@k metrics demonstrate overall superiority, they cannot distinguish whether the improvements stem from merely increasing reliability on simpler tasks or from achieving genuine breakthroughs on previously intractable problems. To isolate these effects, we stratify AIME 2024 problems by difficulty following Sun et al. (2025). As shown in Figure 3 (middle), performance on the *Easy* tier is largely saturated across all methods. However, as difficulty escalates, the performance gap gradually widens. Crucially, IMH yields the most pronounced gains on the *Hard* tier,

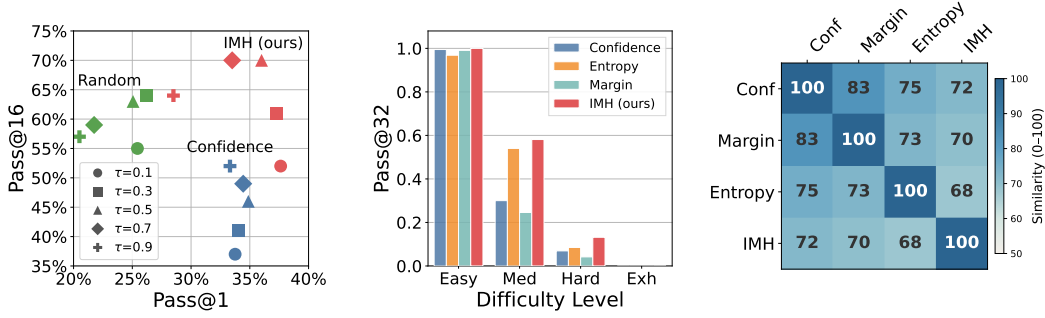


Figure 3: **Left:** Quality-diversity trade-off on MATH500 (LLaDA) by sweeping temperature parameters. IMH strictly dominates local baselines. **Middle:** Pass@32 on AIME 2024 stratified by difficulty (Sun et al., 2025) (WeDLM-8B). IMH yields the largest gains on Hard problems. **Right:** Trajectory similarity matrix on AIME 2024. Local baselines collapse into similar paths, whereas IMH explores distinctly different reasoning strategies.

decisively outperforming the strongest baselines. This confirms that the trajectory-level diversity induced by the lookahead correction does not merely increase reliability on solvable problems, but actively expands the model’s *reasoning boundary*, enabling it to navigate the complex search spaces of the most challenging mathematical tasks. We provide a detailed case study in Appendix G, showcasing specific hard problems successfully solved by IMH that completely eluded all baseline methods.

Trajectory Similarity Analysis. To understand *why* IMH succeeds where local baselines fail, we analyze the semantic diversity of the generated reasoning paths. For each AIME 2024 problem, we sample one trajectory per method and employ an LLM judge to summarize the problem-solving strategy and score pairwise similarity (details in Appendix H). The resulting similarity matrix (Figure 3 (right)) exhibits a distinct block structure: the three local baselines (Conf, Margin, Entropy) are highly correlated with one another (scores 73–83). This indicates that confidence-based remasking heuristics tend to collapse into nearly identical reasoning paths, regardless of the specific uncertainty metric employed. In contrast, IMH trajectories exhibit significantly lower similarity to all baselines (scores 68–72). This demonstrates that IMH successfully induces qualitatively distinct reasoning strategies, rather than merely injecting local perturbations into the same dominant solution trajectory.

IMH convergence and mixing properties. To empirically validate the efficacy of our IMH sampler, we ablate the Markov chain length T using WeDLM-8B on the AIME 2024 benchmark. Table 3 characterizes the resulting MCMC dynamics from two complementary perspectives: downstream task-level convergence and internal mixing efficiency. First, to assess task-level convergence, we monitor the Pass@1 and Pass@8 accuracies across varying lengths of T . Both metrics demonstrate a steady increase in performance until $T = 7$, beyond which the accuracy remains roughly stable. This rapid saturation serves as strong empirical evidence of fast convergence. It demonstrates that the Markov chain swiftly reaches its stationary distribution, allowing a short, computationally inexpensive chain to yield substantial performance improvements over the baseline. Second, we evaluate the internal mixing efficiency of the sampler. As shown in Table 3, the mean acceptance rate remains exceptionally high and stable—consistently exceeding 97.0% regardless of the configured chain length T . In the context of an IMH formulation, such a high and persistent acceptance rate provides strong theoretical guarantees (Mengersen & Tweedie, 1996). Specifically, it confirms that

Table 3: MCMC dynamics on AIME 2024 (WeDLM-8B). **Baseline** denotes entropy remasking without IMH.

T	Pass@1	Pass@8	Acc.(%)
1	9.5	23.1	99.3
3	10.0	23.3	99.1
7	11.4	24.0	98.3
15	11.0	23.4	97.5
31	11.0	24.5	97.2
Baseline	9.4	21.4	—

our state-independent proposal distribution (Equation (11)) tightly bounds the globally corrected target distribution π_i . This close structural alignment enables rapid geometric convergence and highly efficient exploration of the latent state space.

6 Conclusion

Diffusion large language models offer a compelling paradigm for complex reasoning, yet their practical efficacy has been severely constrained by myopic decoding heuristics that prematurely collapse trajectory diversity. In this work, we propose a principled decoding paradigm to resolve this quality–exploration dilemma. By integrating a lookahead correction through an Independent Metropolis–Hastings sampler, our method dynamically aligns local token commitments with the global promise of the resulting completion space. Extensive evaluations on rigorous reasoning benchmarks demonstrate that our strategy establishes a new Pareto frontier. Moreover, our approach fundamentally expands the effective reasoning frontier of dLLMs on exceptionally demanding benchmarks such as AIME datasets.

References

- Marianne Arriola, Aaron Gokaslan, Justin T Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://arxiv.org/abs/2503.09573>.
- Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.
- Gregor Bachmann and Vaishnavh Nagarajan. The pitfalls of next-token prediction. *arXiv preprint arXiv:2403.06963*, 2024.
- Heli Ben-Hamu, Itai Gat, Daniel Severo, Niklas Nolte, and Brian Karrer. Accelerated sampling from masked diffusion models via entropy bounded unmasking. *arXiv preprint arXiv:2505.24857*, 2025.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf.
- Andrew Campbell, Joe Benton, Valentin De Bortoli, Thomas Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. *Advances in Neural Information Processing Systems*, 35:28266–28279, 2022.
- 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.
- Qiguang Chen, Hanjing Li, Libo Qin, Dengyun Peng, Jinhao Liu, Jiangyi Wang, Chengyue Wu, Xie Chen, Yantao Du, and Wanxiang Che. Beyond surface reasoning: Unveiling the true long chain-of-thought capacity of diffusion large language models. *arXiv preprint arXiv:2510.09544*, 2025.
- Hengyu Fu, Baihe Huang, Virginia Adams, Charles Wang, Venkat Srinivasan, and Jiantao Jiao. From bits to rounds: Parallel decoding with exploration for diffusion language models. *arXiv preprint arXiv:2511.21103*, 2025.

- W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.
- Feng Hong, Geng Yu, Yushi Ye, Haicheng Huang, Huangjie Zheng, Ya Zhang, Yanfeng Wang, and Jiangchao Yao. Wide-in, narrow-out: Revokable decoding for efficient and effective dllms. *arXiv preprint arXiv:2507.18578*, 2025a.
- Jiwoo Hong, Seongyeon An, Taehyun Kim, and Ernestasia Aniela Ye. Improving discrete diffusion unmasking policies beyond explicit reference policies. *arXiv preprint arXiv:2510.05725*, 2025b.
- Pengcheng Huang, Shuhao Liu, Zhenghao Liu, Yukun Yan, Shuo Wang, Zulong Chen, and Tong Xiao. Pc-sampler: Position-aware calibration of decoding bias in masked diffusion models. *arXiv preprint arXiv:2508.13021*, 2025.
- Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*. 3rd edition, 2026. URL <https://web.stanford.edu/~jurafsky/slp3/>. Online manuscript released January 6, 2026.
- Aayush Karan and Yilun Du. Reasoning with sampling: Your base model is smarter than you think. *arXiv preprint arXiv:2510.14901*, 2025.
- Jaeyeon Kim, Kulin Shah, Vasilis Kontonis, Sham Kakade, and Sitan Chen. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. *arXiv preprint arXiv:2502.06768*, 2025a.
- Seo Hyun Kim, Sunwoo Hong, Hojung Jung, Youngrok Park, and Se-Young Yun. KLAS: KL-guided fast inference in masked diffusion models. *Advances in Neural Information Processing Systems*, 2025b.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.
- Sanghyun Lee, Seungryong Kim, Jongho Park, and Dongmin Park. Lookahead unmasking elicits accurate decoding in diffusion language models. *arXiv preprint arXiv:2511.05563*, 2025.
- Aiwei Liu, Minghua He, Shaoxun Zeng, Sijun Zhang, Linhao Zhang, Chuhan Wu, Wei Jia, Yuan Liu, Xiao Zhou, and Jie Zhou. Wedlm: Reconciling diffusion language models with standard causal attention for fast inference. *arXiv preprint arXiv:2512.22737*, 2025.
- 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. *Advances in neural information processing systems*, 36:21558–21572, 2023.
- Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *International Conference on Machine Learning*, 2023.
- Omer Luxembourg, Haim Permuter, and Eliya Nachmani. Plan for speed: Dilated scheduling for masked diffusion language models. *arXiv preprint arXiv:2506.19037*, 2025.
- Kerrie L Mengersen and Richard L Tweedie. Rates of convergence of the hastings and metropolis algorithms. *The annals of Statistics*, 24(1):101–121, 1996.
- Vaishnavh Nagarajan, Chen Henry Wu, Charles Ding, and Aditi Raghunathan. Roll the dice & look before you leap: Going beyond the creative limits of next-token prediction. *arXiv preprint arXiv:2504.15266*, 2025.

- Zanlin Ni, Shenzhi Wang, Yang Yue, Tianyu Yu, Weilin Zhao, Yeguo Hua, Tianyi Chen, Jun Song, Cheng Yu, Bo Zheng, et al. The flexibility trap: Why arbitrary order limits reasoning potential in diffusion language models. *arXiv preprint arXiv:2601.15165*, 2026.
- 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. In *Advances in Neural Information Processing Systems (NeurIPS 2025)*, 2025.
- Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. *arXiv preprint arXiv:2406.03736*, 2024.
- Leyi Pan, Shuchang Tao, Yunpeng Zhai, Zheyu Fu, Liancheng Fang, Minghua He, Lingzhe Zhang, Zhaoyang Liu, Bolin Ding, Aiwei Liu, et al. d-treerpo: Towards more reliable policy optimization for diffusion language models. *arXiv preprint arXiv:2512.09675*, 2025.
- Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL <http://jmlr.org/papers/v21/20-074.html>.
- Subham Sekhar Sahoo, Marianne Arriola, Aaron Gokaslan, Edgar Mariano Marroquin, Alexander M Rush, Yair Schiff, Justin T Chiu, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=L4uaAR4ArM>.
- Subham Sekhar Sahoo, Justin Deschenaux, Aaron Gokaslan, Guanghan Wang, Justin Chiu, and Volodymyr Kuleshov. The diffusion duality. *arXiv preprint arXiv:2506.10892*, 2025.
- Yangyi Shen, Tianjian Feng, Jiaqi Han, Wen Wang, Tianlang Chen, Chunhua Shen, Jure Leskovec, and Stefano Ermon. Improving diffusion language model decoding through joint search in generation order and token space. *arXiv preprint arXiv:2601.20339*, 2026.
- Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. *Advances in neural information processing systems*, 37:103131–103167, 2024.
- Yuxuan Song, Zheng Zhang, Cheng Luo, Pengyang Gao, Fan Xia, Hao Luo, Zheng Li, Yuehang Yang, Hongli Yu, Xingwei Qu, et al. Seed diffusion: A large-scale diffusion language model with high-speed inference. *arXiv preprint arXiv:2508.02193*, 2025.
- Yiyi Sun, Georgia Zhou, Haoyue Bai, Hao Wang, Dacheng Li, Nouha Dziri, and Dawn Song. Climbing the ladder of reasoning: What llms can-and still can’t-solve after sft? *arXiv preprint arXiv:2504.11741*, 2025.
- Itamar Trainin, Shauli Ravfogel, Omri Abend, and Amir Feder. Discrete diffusion models exploit asymmetry to solve lookahead planning tasks. *arXiv preprint arXiv:2602.19980*, 2026.
- Qingyan Wei, Yaojie Zhang, Zhiyuan Liu, Dongrui Liu, and Linfeng Zhang. Accelerating diffusion large language models with slowfast sampling: The three golden principles. *arXiv preprint arXiv:2506.10848*, 2025.
- Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dllm: Training-free acceleration of diffusion llm by enabling kv cache and parallel decoding. *arXiv preprint arXiv:2505.22618*, 2025.
- Chenkai Xu, Yijie Jin, Jiajun Li, Yi Tu, Guoping Long, Dandan Tu, Mingcong Song, Hongjie Si, Tianqi Hou, Junchi Yan, et al. Lopa: Scaling dllm inference via lookahead parallel decoding. *arXiv preprint arXiv:2512.16229*, 2025.

- Chenxiao Yang, Cai Zhou, David Wipf, and Zhiyuan Li. On powerful ways to generate: Autoregression, diffusion, and beyond. *arXiv preprint arXiv:2510.06190*, 2025.
- Jiacheng Ye, Shansan Gong, Liheng Chen, Lin Zheng, Jiahui Gao, Han Shi, Chuan Wu, Xin Jiang, Zhenguo Li, Wei Bi, et al. Diffusion of thought: Chain-of-thought reasoning in diffusion language models. *Advances in Neural Information Processing Systems*, 37: 105345–105374, 2024.
- 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.
- Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *arXiv preprint arXiv:2504.13837*, 2025.
- Lingzhe Zhang, Liancheng Fang, Chiming Duan, Minghua He, Leyi Pan, Pei Xiao, Shiyu Huang, Yunpeng Zhai, Xuming Hu, Philip S Yu, et al. A survey on parallel text generation: From parallel decoding to diffusion language models. *arXiv preprint arXiv:2508.08712*, 2025.
- Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2024, 2024.
- Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2025, 2025.
- Siyan Zhao, Devaansh Gupta, Qinqing Zheng, and Aditya Grover. d1: Scaling reasoning in diffusion large language models via reinforcement learning. *arXiv preprint arXiv:2504.12216*, 2025.
- Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. *arXiv preprint arXiv:2409.02908*, 2024.

Appendix

A	Related Work	15
B	Experimental Details	15
C	Additional Efficiency and Robustness Analyses	16
C.1	Compute-Matched Efficiency Comparison	17
C.2	Mean-Field Approximation Error	17
C.3	Decoding-Length Sensitivity	18
D	Details of Uncertainty Heuristics and δ-gating	18
E	Derivation of the IMH Acceptance Ratio	19
F	Variance of the $\text{pass}@k$ Estimator	20
G	Detailed Analysis of Hard Problems	21
H	Trajectory Similarity Analysis	23
I	Proofs	24
I.1	Proof of Proposition 1	24
I.2	Proof of Proposition 2	25
I.3	Proof of Proposition 3	26
I.4	Proof of Proposition 4	26

A Related Work

Reasoning with dLLMs. Pass@ k is the standard metric for code generation (Chen et al., 2021) and has been adopted as a proxy for reasoning and exploration potential across domains. Yue et al. (2025) argue that Pass@ k provides an upper bound on RL-based reasoning improvement: when the base model assigns too little probability mass to correct solutions, RL may fail simply because positive-reward trajectories are rarely sampled. Motivated by this view, several recent works analyze why diffusion language models (dLLMs) often exhibit weak Pass@ k scaling under standard decoding. Ni et al. (2026) attribute the bottleneck to a *flexibility trap*, suggesting that arbitrary-order generation can undermine reasoning. Shen et al. (2026) emphasize a complementary mechanism: despite being order-agnostic in principle, common decoders effectively traverse only a single decoding trajectory, limiting exploration over trajectories; they further note that low-confidence remasking can collapse the accessible reasoning space. Chen et al. (2025) propose the *Parallel-Sequential Contradiction (PSC)*: parallel token updates can conflict with strictly causal multi-step reasoning, with harder problems inducing AR-like behavior that limits self-reflection depth and exploration breadth. Relatedly, Lee et al. (2025) critique prevalent heuristic unmasking rules—especially uncertainty-based ones—as *myopic*, arguing that they fail to convert additional test-time compute into systematic decoding improvements and can suffer cascading errors. Fu et al. (2025) further study the inefficiency of confidence-first strategies, observing that high-confidence tokens can be locally easy yet globally low-information, whereas resolving low-confidence tokens earlier can unlock more effective parallelism and exploration.

Advanced unmasking strategies. Beyond the fundamental heuristics of confidence, entropy, and margin (Table 1), recent literature has introduced more sophisticated unmasking mechanisms. Within the confidence-based paradigm, SlowFast Sampling (Wei et al., 2025) leverages certainty, convergence, and positional cues to isolate stable token spans for aggressive parallel decoding. WINO (Hong et al., 2025a) proposes a revokable decoding scheme that over-drafts tokens and subsequently re-masks dubious ones via bidirectional verification. Meanwhile, Uncode (Huang et al., 2025) recalibrates raw confidence scores using corpus-frequency weighting and positional decay to counteract biases toward boundary and trivial tokens. Among entropy-driven methods, DUS (Luxembourg et al., 2025) minimizes an upper bound on the per-step joint entropy gain by partitioning positions into dilated groups, whereas KLASS (Kim et al., 2025b) fuses confidence metrics with KL-divergence stability across denoising timesteps. Despite their algorithmic sophistication, these approaches remain fundamentally local—they determine token commitments solely through per-position signals and are thus strictly bounded by the entropy cap established in Proposition 2. Closely related to our approach is Lopa (Xu et al., 2025), which draws multiple parallel candidates and employs a lookahead mechanism to select the best candidate. However, their selection criteria are inherently deterministic and fail to capture the principled global tempered distribution that our method achieves. For a broader taxonomy of advanced unmasking strategies, we refer readers to the comprehensive survey by Zhang et al. (2025).

Tempered sampling with MCMC. While Karan & Du (2025) explores globally tempered sampling for autoregressive (AR) LLMs by applying classic MCMC methods to their sequential generation process, our approach diverges in two fundamental aspects. First, unlike AR models, diffusion LLMs (dLLMs) lack a tractable sequence likelihood, rendering standard MCMC techniques inapplicable. Second, our MCMC formulation specifically targets a 1D categorical distribution over the vocabulary space. Crucially, this requires a correction term that we estimate using a mean-field approximation—a mechanism uniquely enabled by the dLLM framework.

B Experimental Details

Decoding configurations. For all baseline methods (Confidence, Entropy, Margin, Random), we disable parallel decoding and restrict the generation to exactly one token per step. This ensures a fair comparison by eliminating the need for hyperparameter tuning. Additionally, LLaDA employs a semi-autoregressive decoding strategy with a block length

of 32. For our Independent Metropolis–Hastings (IMH) sampler, we set the chain length $T = 4$ for LLaDA and $T = 8$ for WeDLM. The maximum generation lengths for LLaDA across different benchmarks are detailed in Table 4.

Table 4: Maximum generation lengths for LLaDA across different benchmarks.

Benchmark	Maximum Generation Length
MATH500	128
AIME 2024	512
AIME 2025	512
HumanEval	256
MBPP	256

Evaluation. Both WeDLM and LLaDA use EvalPlus (Liu et al., 2023) for evaluating code generation benchmarks (HumanEval and MBPP). For mathematical reasoning benchmarks (MATH500, AIME 2024, AIME 2025), we use a custom evaluation pipeline with answer extraction from `\boxed{}` expressions and symbolic equivalence checking following WeDLM (Liu et al., 2025). All code benchmarks report $\text{pass}@k$ metrics computed via the unbiased estimator of Chen et al. (2021), with $n = 32$ samples for LLaDA and $n = 128$ samples for WeDLM.

Implementation details. To accelerate the decoding process, we adopt prefix KV cache (Wu et al., 2025) for LLaDA. For WeDLM, we implement a *copy-on-write* (CoW) mechanism (Kwon et al., 2023) for the KV cache to efficiently support the IMH sampler with multiple candidates. The KV cache is organized into fixed-size physical blocks (256 tokens each), and each sequence maintains a block table mapping logical positions to physical blocks. When generating N candidate continuations, all candidates share the same physical blocks for the prompt prefix. Only the tail blocks (where candidates diverge) are allocated separately. When the prefix boundary falls mid-block, we perform a lightweight copy operation: the shared prefix tokens within that block are copied to each candidate’s private tail block, avoiding a full cache duplication. This CoW strategy reduces memory overhead from $O(N \cdot L)$ to $O(L + N \cdot T)$, where L is the prompt length and T is the window size (16 tokens by default), enabling efficient scaling to 8 candidates per sequence.

C Additional Efficiency and Robustness Analyses

We supplement the main evaluation with three analyses motivated by the gap between the idealized target distribution and practical deployment: compute-matched test-time scaling, controlled measurement of the mean-field approximation error, and sensitivity to the maximum decoding length.

C.1 Compute-Matched Efficiency Comparison

IMH evaluates its T token proposals concurrently, so increasing T does not add sequential Markov-chain steps. It does, however, increase the number of forward evaluations per decoding step. We therefore compare methods under matched forward-pass budgets on LLaDA-8B-Instruct and MATH500. At each compute multiplier $x \in \{1, 2, 4, 8\}$, IMH uses chain length $T = x$ and is evaluated by Pass@1/Pass@8. We compare against two uses of the same budget with the default low-confidence remasking baseline: (i) an *oracle upper bound*, which reports the baseline’s Pass@ x /Pass@ $(8x)$ and therefore assumes a perfect verifier over the x independent samples allocated to each IMH output; and (ii) *self-consistency*, which majority-votes each group of x baseline samples before computing Pass@1/Pass@8. All methods use $n = 64$ samples per problem.

Table 5: **Compute-matched test-time scaling on LLaDA-8B-Instruct/MATH500.** Each entry reports Pass@1/Pass@8 (%). The oracle gives the low-confidence baseline perfect selection, while self-consistency is verifier-free.

Method	1×	2×	4×	8×
Oracle (upper bound)	36.2 / 49.0	42.1 / 59.5	47.5 / 66.1	53.1 / 72.0
Self-consistency	36.2 / 49.0	36.4 / 53.2	37.8 / 49.9	38.3 / 47.4
IMH ($T = x$)	35.7 / 53.4	35.3 / 55.1	34.5 / 60.9	35.2 / 58.9

Across every budget, IMH improves Pass@8 over the practical self-consistency baseline while maintaining comparable Pass@1. The gap is largest at $4\times$, where IMH reaches 60.9% Pass@8 compared with 49.9% for self-consistency. The oracle becomes stronger for $x \geq 2$, as expected from its unrealistic perfect-verifier advantage. As a complementary comparison, the temperature sweep in Figure 3 grants local baselines post-hoc selection over $\{0.1, 0.3, 0.5, 0.7, 0.9\}$; fixed-setting IMH still traces a stronger quality–diversity frontier. These results show that the exploration gains are not explained solely by additional forward compute.

C.2 Mean-Field Approximation Error

The deployed sampler replaces the exact suffix correction $\Delta_{i,v}(s)$ in Proposition 4 with the mean-field surrogate $\hat{\Delta}_{i,v}(s)$. Importantly, a candidate-independent bias has no effect on either the normalized conditional or the IMH transition: $\text{softmax}(z_v + c) = \text{softmax}(z_v)$, and Equation (12) depends only on correction differences. Defining

$$\varepsilon_v(s) := \hat{\Delta}_{i,v}(s) - \Delta_{i,v}(s), \quad (13)$$

the change in the log-acceptance ratio for a move from x to y is exactly $\varepsilon_y(s) - \varepsilon_x(s)$. Thus, candidate-to-candidate variation, rather than absolute error, is the sampler-relevant diagnostic.

We estimate both quantities through controlled exact enumeration on partial states sampled from real LLaDA-8B-Instruct decoding traces. For each benchmark, we sample 300 states, take the next committed position i , restrict its candidates to the top $M = 8$ tokens, and use a suffix horizon $h = 2$ with each suffix position restricted to its top $K = 8$ tokens. Within this finite space, we compute Δ exactly by enumerating every suffix assignment and averaging over both reveal orders, re-evaluating the model after each reveal to preserve suffix dependence. We compare this exact correction with the factorized mean-field surrogate. For errors $\{\varepsilon_m\}_{m=1}^M$, we report

$$A = \frac{1}{M} \sum_{m=1}^M |\varepsilon_m|, \quad C = \frac{1}{M^2} \sum_{m=1}^M \sum_{n=1}^M |\varepsilon_m - \varepsilon_n|, \quad (14)$$

where A is the absolute error and C is the candidate-dependent error.

Table 6: **Controlled mean-field error diagnostic.** Candidate-dependent error is the component that affects IMH acceptance decisions.

Benchmark	States	Abs. A	Cand. C	A/C
MATH500	300	0.224	0.034	6.6×
AIME 2024	300	0.208	0.053	3.9×
HumanEval	300	0.143	0.013	11.4×
MBPP	300	0.222	0.028	7.9×
All	1200	0.195	0.030	6.5×

Although the mean absolute error is non-negligible, the candidate-dependent component is $6.5\times$ smaller on average. Most of the approximation gap is therefore common-mode and cancels from the sampler. AIME 2024 has the largest candidate-dependent error, but it remains $3.9\times$ below the absolute error, indicating that the correction preserves the relative comparisons that govern IMH even in the most challenging setting evaluated.

C.3 Decoding-Length Sensitivity

We test whether the gains of IMH persist as the maximum generation length varies on WeDLM-8B/AIME 2024. We sweep `gen_length` over $\{256, 512, 1024, 2048, 4096\}$ and compare IMH with confidence, entropy, and margin remasking under the same one-token-per-step protocol and shared local temperature 0.5. Following the main evaluation, random remasking is excluded for WeDLM because of its left-to-right generation bias. Each method uses $n = 32$ samples per problem, and we report Pass@1/Pass@8.

Table 7: **Sensitivity to maximum generation length on WeDLM-8B/AIME 2024.** Each entry reports Pass@1/Pass@8 (%).

Method	256	512	1024	2048	4096
Confidence	1.1 / 3.3	3.3 / 12.4	7.5 / 17.5	7.3 / 18.2	7.9 / 19.8
Entropy	1.6 / 4.2	4.2 / 14.5	8.1 / 19.2	8.1 / 20.9	9.3 / 20.5
Margin	0.9 / 3.9	3.3 / 14.6	7.1 / 15.9	7.6 / 20.3	5.3 / 18.4
IMH (ours)	1.6 / 5.0	4.2 / 15.0	9.2 / 21.0	9.9 / 21.6	9.5 / 22.5

IMH achieves the best or tied Pass@1 and the highest Pass@8 at every tested length. Its Pass@8 increases monotonically from 5.0% to 22.5% as the generation budget grows. Pass@1 peaks at length 2048 and remains within 0.4 points at 4096, whereas the local remasking baselines fluctuate more strongly. The advantage of IMH is therefore robust to the decoding-length choice rather than being tied to a single generation budget.

D Details of Uncertainty Heuristics and δ -gating

In Table 1, we categorize three families of heuristics that anchor tokens at high-confidence positions, differentiated primarily by their distributional scoring metrics. Formally, let $p_i(\cdot) := p(\cdot \mid s_t, i)$ be the per-position categorical distribution, and let v_1, v_2 denote the most and second-most probable tokens, respectively.

We contrast two operational paradigms: *Sample-then-Filter*, which samples all masked tokens and subsequently filters for confidence, and *Rank-then-Sample*, which restricts sampling to the highest-scoring positions. The δ -gating framework maps each heuristic to a unified theoretical condition via the implied confidence bound $\max_v p_i(v) \geq 1 - \delta$ (Definition 1). We provide the formal derivations for these bounds below.

Confidence Heuristic. The confidence heuristic directly thresholds the maximum probability: $\max_v p_i(v) \geq 1 - \delta$. This trivially satisfies the δ -gating condition with parameter δ .

Entropy Heuristic. The entropy heuristic thresholds the Shannon entropy: $H(p_i) \leq \varepsilon$. To find the implied confidence bound, we seek the maximum δ such that $H(p_i) \leq \varepsilon \implies \max_v p_i(v) \geq 1 - \delta$. Let $\alpha = \max_v p_i(v)$. For a fixed maximum probability α , the entropy is maximized when the remaining probability mass $1 - \alpha$ is distributed uniformly across the remaining $|V| - 1$ tokens. Thus, the maximum entropy for a given α is:

$$h(\alpha) = -\alpha \log \alpha - (1 - \alpha) \log \left(\frac{1 - \alpha}{|V| - 1} \right)$$

Since $h(\alpha)$ is strictly decreasing for $\alpha \geq 1/|V|$, the condition $H(p_i) \leq \varepsilon$ implies $\alpha \geq \alpha^*$, where α^* is the unique solution to $h(\alpha^*) = \varepsilon$ in $[1/|V|, 1]$. Setting $\delta_V(\varepsilon) = 1 - \alpha^*$ yields the equivalent δ -gating bound.

Margin Heuristic. The margin heuristic thresholds the difference between the top two probabilities: $p_i(v_1) - p_i(v_2) \geq \gamma$. Let $\alpha = p_i(v_1)$ and $\beta = p_i(v_2)$. We are given $\alpha - \beta \geq \gamma$. Since β is the second largest probability, the remaining probability mass $1 - \alpha - \beta$ must be distributed among the other $|V| - 2$ tokens such that no token has probability greater than β . The minimum possible value for α occurs when β is as large as possible. The maximum possible value for β under the constraint $\alpha - \beta \geq \gamma$ is $\beta = \alpha - \gamma$. Furthermore, the sum of all probabilities is 1:

$$\alpha + \beta + \sum_{k=3}^{|V|} p_i(v_k) = 1$$

To minimize α , we maximize the other probabilities. The maximum possible value for each $p_i(v_k)$ is β . Thus, setting $p_i(v_k) = \beta$ for all $k \geq 2$, we get:

$$\alpha + (|V| - 1)\beta = 1$$

Substituting $\beta = \alpha - \gamma$:

$$\alpha + (|V| - 1)(\alpha - \gamma) = 1 \implies |V|\alpha - (|V| - 1)\gamma = 1 \implies \alpha = \frac{1 + (|V| - 1)\gamma}{|V|}$$

Therefore, the implied confidence bound is:

$$\max_v p_i(v) \geq \frac{1 + (|V| - 1)\gamma}{|V|} = 1 - \frac{(|V| - 1)(1 - \gamma)}{|V|}$$

which yields $\delta = \frac{(|V| - 1)(1 - \gamma)}{|V|}$.

E Derivation of the IMH Acceptance Ratio

In this section, we provide the detailed derivation of the acceptance probability for the Independent Metropolis–Hastings (IMH) sampler introduced in Section 4.3.

The general acceptance ratio for an Independent Metropolis–Hastings algorithm proposing a move from state x to state y is given by:

$$A(x \rightarrow y) = \min \left\{ 1, \frac{\pi_i(y | s) \cdot r_i(x | s)}{\pi_i(x | s) \cdot r_i(y | s)} \right\} \quad (15)$$

Recall from Equation (10) that our one-step sampling target distribution is:

$$\pi_i(v | s) \propto \exp(\alpha \ell_{i,v}(s) + \hat{\Delta}_{i,v}(s)) \quad (16)$$

where α is the inverse temperature, $\ell_{i,v}(s)$ is the local logit for token v at position i , and $\hat{\Delta}_{i,v}(s)$ is the suffix correction term.

The proposal distribution, defined in Equation (11), is the locally tempered categorical distribution without the suffix correction:

$$r_i(v | s) = \text{softmax}(\alpha \ell_i(s))_v \propto \exp(\alpha \ell_{i,v}(s)) \quad (17)$$

Substituting these unnormalized probabilities into the Metropolis–Hastings ratio yields:

$$\begin{aligned} \frac{\pi_i(y | s) r_i(x | s)}{\pi_i(x | s) r_i(y | s)} &= \frac{\exp(\alpha \ell_{i,y}(s) + \hat{\Delta}_{i,y}(s))}{\exp(\alpha \ell_{i,x}(s) + \hat{\Delta}_{i,x}(s))} \cdot \frac{\exp(\alpha \ell_{i,x}(s))}{\exp(\alpha \ell_{i,y}(s))} \\ &= \exp(\alpha \ell_{i,y}(s) + \hat{\Delta}_{i,y}(s) - \alpha \ell_{i,x}(s) - \hat{\Delta}_{i,x}(s)) \cdot \exp(\alpha \ell_{i,x}(s) - \alpha \ell_{i,y}(s)) \\ &= \exp(\hat{\Delta}_{i,y}(s) - \hat{\Delta}_{i,x}(s)) \end{aligned} \quad (18)$$

As shown above, the local tempered logit terms $\alpha \ell_{i,v}(s)$ exactly cancel out in the numerator and denominator. This structural alignment results in a highly streamlined acceptance probability that depends strictly on the difference between the suffix corrections of the proposed and current states:

$$A(x \rightarrow y) = \min \left\{ 1, \exp(\hat{\Delta}_{i,y}(s) - \hat{\Delta}_{i,x}(s)) \right\} \quad (19)$$

which matches Equation (12).

F Variance of the pass@k Estimator

In our experiments, we evaluate pass@k using the unbiased estimator introduced by Chen et al. (2021):

$$\hat{P}_k = 1 - \frac{\binom{n-c}{k}}{\binom{n}{k}} \quad (20)$$

where n is the total number of generated samples per problem, and c is the number of those samples that pass the unit tests. Due to computational constraints, especially when sampling at relatively high temperatures, we set $n = 32$ for evaluating $k \leq 16$ (LLaDA) and $n = 128$ for $k \leq 32$ (WeDLM).

To ensure our results are robust to random noise from high-temperature sampling, we analyze the standard error (SE) of this estimator. The variance of $\hat{P}_k$ for a single problem depends on the true underlying pass probability p , where $c \sim \text{Binomial}(n, p)$. The worst-case variance occurs at the value of p that maximizes the uncertainty of the binomial outcome.

By calculating the theoretical maximum variance of $\hat{P}_k$ over all possible $p \in [0, 1]$, we can bound the maximum standard error over a dataset of M problems as $\text{SE}_{\max} = \sqrt{\max_p \text{Var}(\hat{P}_k) / M}$. The corresponding 95% margin of error is $1.96 \times \text{SE}_{\max}$.

In Table 8, we report these absolute worst-case margins of error for each benchmark evaluated in our study, based on its specific number of problems M .

Table 8: **Worst-case 95% margin of error for Pass@k estimation.** The theoretical maximum noise introduced by the finite-sample estimator for each benchmark and model setting.

Benchmark	Size (M)	WeDLM ($n = 128, k = 32$)	LLaDA ($n = 32, k = 16$)
MBPP	974	$\pm 1.4\%$	$\pm 2.0\%$
MATH500	500	$\pm 1.9\%$	$\pm 2.9\%$
HumanEval	164	$\pm 3.4\%$	$\pm 5.0\%$
AIME 2024 / 2025	30	$\pm 7.9\%$	-

Because the expected variance in practice is strictly lower than these worst-case bounds (as p is rarely perfectly adversarial across all problems, and is often exactly 0 for hard problems

like AIME), we conclude that setting $n \geq 2k$ to $4k$ provides a sufficiently stable estimate of $\text{pass}@k$, effectively isolating our performance measurements from sampling noise.

G Detailed Analysis of Hard Problems

To concretely illustrate how global tempering expands the reasoning boundary of dLLMs (Section 5), we present a detailed case study on a challenging problem from the AIME 2024 benchmark. For this problem, our Independent Metropolis–Hastings (IMH) method successfully generated correct solutions, whereas all baseline methods (Confidence, Entropy, Margin, and Left-to-Right) failed completely across all 128 samples.

AIME 2024 Problem II-11

Problem Context: Let ABC be a triangle inscribed in circle ω . Let the tangents to ω at B and C intersect at point D , and let $\overline{AD}$ intersect ω at P . If $AB = 5$, $BC = 9$, and $AC = 10$, AP can be written as the form $\frac{m}{n}$, where m and n are relatively prime integers. Find $m + n$.

Answer: 113

Intended Solution: This problem is classically solved by recognizing that AD is the A -symmedian of $\triangle ABC$. Because D is the pole of BC , $ABPC$ is a harmonic quadrilateral, meaning $AB \cdot PC = AC \cdot PB$. Combined with Ptolemy’s Theorem ($AP \cdot BC = AB \cdot PC + AC \cdot PB$) and the Law of Cosines to find the lengths of the chords, one can systematically solve for AP . The exact length evaluates to $AP = \frac{96}{17}$, yielding the final answer $m + n = 96 + 17 = 113$.

Selected Excerpts from Baseline (Confidence): “Using the Power of a Point theorem for point D with respect to the circumcircle, we have $DA \cdot DP = DB^2 = DC^2$. Since D is the intersection of the tangents at B and C , $DB = DC = \sqrt{AB \cdot AC} = \sqrt{5 \cdot 10} = \sqrt{50} = 5\sqrt{2}$.” ✗

Selected Excerpts from IMH (Ours): “**Coordinate Geometry Setup:** Place B at $(0,0)$, C at $(9,0)$, and use distance formulas to find coordinates of A . Solving the system gives A at $(\frac{19}{5}, \frac{12\sqrt{6}}{5})$.”

“**Circle Equation:** Using general circle equation and substituting points A , B , and C , we find the circle’s center $(\frac{29}{5}, \frac{3\sqrt{6}}{10})$ and radius $\frac{17\sqrt{6}}{10}$.”

“**Equation of Line AD :** Parametrize line AD and find intersection P with the circle. Solving the quadratic equation [...] we find the valid parameter $t = \frac{12}{17}$. **Length**

Calculation: Using the parameter $t = \frac{12}{17}$, the length $AP = \frac{12}{17} \times AD$. Calculating AD via distance formula, we find $AP = \frac{96}{17}$. Thus, $m + n = 113$.” ✓

Error Analysis & Takeaway:

Baseline methods attempt to use pure Euclidean geometry but quickly hallucinate non-existent theorems to bridge gaps in their reasoning. For example, the model confidently asserts that the tangent length $DB = \sqrt{AB \cdot AC}$, which is mathematically unfounded, leading to a completely incorrect trajectory. Instead, global tempering allows the model to abandon brittle, hallucination-prone synthetic geometry paths and explore robust alternative strategies. In this case, IMH successfully executes a complete and computationally heavy coordinate geometry (analytic) proof without any logical gaps.

AIME 2024 Problem II-2

Problem Context: Let $\mathcal{B}$ be the set of rectangular boxes with surface area 54 and volume 23. Let r be the radius of the smallest sphere that can contain each of the rectangular boxes that are elements of $\mathcal{B}$. The value of r^2 can be written as $\frac{p}{q}$, where p and q are relatively prime positive integers. Find $p + q$.

Answer: 721

Intended Solution: Let the dimensions of the box be a, b, c . We are given $ab + bc + ca = 27$ and $abc = 23$. The radius r of the smallest containing sphere is half the space diagonal, so $r^2 = \frac{a^2 + b^2 + c^2}{4}$. We want to minimize $a^2 + b^2 + c^2 = (a + b + c)^2 - 54$. Let $S = a + b + c$; we must minimize S . The numbers a, b, c are roots of $P(x) = x^3 - Sx^2 + 27x - 23$. For $P(x)$ to have three real roots, its local maximum must be ≥ 0 and local minimum ≤ 0 . The derivative is $P'(x) = 3x^2 - 2Sx + 27$. The roots of $P'(x)$ are $x = \frac{S \pm \sqrt{S^2 - 81}}{3}$. Thus we need $S \geq 9$. If $S = 9$, the roots of $P'(x)$ are $x = 3$, so $P(x)$ has an inflection point at $x = 3$, but $P(3) = 27 - 81 + 81 - 23 = 4 \neq 0$, so $S = 9$ does not yield three real roots. The minimum S occurs when the local minimum of $P(x)$ is exactly 0, which means $P(x)$ has a double root. Let the roots be r, r, s . Then $2r + s = S$, $r^2 + 2rs = 27$, and $r^2s = 23$. Substituting $s = \frac{23}{r^2}$ into the second equation gives $r^3 - 27r + 46 = 0$. By the Rational Root Theorem, $r = 2$ is a root. Then $s = \frac{23}{4}$. Thus $S = 2(2) + \frac{23}{4} = \frac{39}{4}$. Then $a^2 + b^2 + c^2 = S^2 - 54 = \frac{1521}{16} - 54 = \frac{657}{16}$. Therefore $r^2 = \frac{657}{64}$, and $p + q = 657 + 64 = 721$.

Selected Excerpts from Baseline (Confidence): "Using the identity $(a + b + c)^2 = a^2 + b^2 + c^2 + 2(ab + bc + ca)$: Let $S = a + b + c$. Then $a^2 + b^2 + c^2 = S^2 - 54$. We need to find S such that a, b, c are real numbers with $abc = 23$ and $ab + bc + ca = 27$. The cubic equation with roots a, b, c is $x^3 - Sx^2 + 27x - 23 = 0$."

"Since S must be an integer (as a, b, c are integers), we test possible values of S : Testing $S = 9$: The cubic equation is $x^3 - 9x^2 + 27x - 23 = 0$. [...] The quadratic $x^2 - 8x + 23$ has a discriminant of $64 - 92 = -28$, so no real roots. Thus, $S = 9$ is the only valid integer solution." ✗

Selected Excerpts from IMH (Ours): "We use the method of Lagrange multipliers. Define the Lagrangian: $\mathcal{L}(l, w, h, \lambda, \mu) = l^2 + w^2 + h^2 - \lambda(lw + lh + wh - 27) - \mu(lwh - 23)$ [...] Solving these equations, we find that $l = w = h$ is not a solution because it would imply a cube with surface area $6l^2 = 54$, giving $l = 3$, but the volume $l^3 = 27 \neq 23$. Therefore, we need another approach."

"Instead, we use the fact that the minimum occurs when the box is as close to a cube as possible. Let's assume $l = w$ and solve: $2(l^2 + 2lh) = 54 \implies l^2 + 2lh = 27$ [...] $l^3 - 27l + 46 = 0$. Using numerical methods or trial and error, we find that $l = 2$ is a root [...] Thus, $l = 2, w = 2$, and $h = \frac{23}{4}$. [...] $r^2 = \frac{657}{64}$. $p + q = 721$." ✓

Error Analysis & Takeaway:

When faced with the complex algebraic step of finding the minimum of S subject to the cubic equation constraints, the baseline model resorts to unwarranted assumptions (e.g., assuming S and the dimensions must be integers) and confidently commits to a hallucinated shortcut ($S = 9$), failing to explore the true continuous optimization landscape. Instead, global tempering prevents premature commitment to locally plausible but globally incorrect reasoning paths (like integer assumptions). By maintaining trajectory diversity, IMH successfully navigates the complex algebraic optimization, correctly identifying that the minimum occurs when two dimensions are equal and solving the resulting cubic equation.

AIME 2024 Problem II-13

Problem Context: Let $ABCD$ be a tetrahedron such that $AB = CD = \sqrt{41}$, $AC = BD = \sqrt{80}$, and $BC = AD = \sqrt{89}$. There exists a point I inside the tetrahedron such that the distances from I to each of the faces of the tetrahedron are all equal. This distance can be written in the form $\frac{m\sqrt{n}}{p}$, where m , n , and p are positive integers, m and p are relatively prime, and n is not divisible by the square of any prime. Find $m + n + p$.

Answer: 104

Intended Solution: This is a disphenoid (isosceles tetrahedron). It can be embedded in a rectangular box of dimensions x, y, z such that the edges are the face diagonals: $x^2 + y^2 = 41$, $y^2 + z^2 = 80$, $x^2 + z^2 = 89$. Solving yields $x = 5, y = 4, z = 8$. The volume is $V = \frac{1}{3}xyz = \frac{160}{3}$. The area of one face is found via Heron's formula to be $A = 6\sqrt{21}$, so the total surface area is $S = 24\sqrt{21}$. The inradius is $r = \frac{3V}{S} = \frac{20\sqrt{21}}{63}$. Thus $m = 20, n = 21, p = 63$, giving $m + n + p = 104$.

Selected Excerpts from Baseline (Confidence): "First, we calculate the volume V of the tetrahedron using the Cayley-Menger determinant: [...] = -28800. The volume is $V = \frac{\sqrt{28800}}{288} = \frac{120}{288} = \frac{5}{12}$. Next, we calculate the surface area [...] Thus, the total surface area is $S = 30$. The inradius r is given by $r = \frac{3V}{S} = \frac{3 \times 5/12}{30} = \frac{1}{24}$." ✗

Selected Excerpts from IMH (Ours): "First, calculate the volume of the tetrahedron using coordinates. Place points as follows: $A = (0, 0, 0)$, $B = (9, 0, 0)$, $C = (4, 4, 3)$, $D = (4, 7, 0)$. Compute the volume using the scalar triple product [...] Volume is: $\frac{63}{2}$."

"The surface area is calculated as: [...] Total surface area: 60. The inradius r is: $r = \frac{3 \times \text{Volume}}{\text{Surface Area}} = \frac{3 \times 63/2}{60} = \frac{63}{40}$. Thus, $m = 63, n = 1, p = 40$, giving $m + n + p = 63 + 1 + 40 = 104$." ✓

Error Analysis & Takeaway:

The baseline attempts a brute-force calculation (Cayley-Menger determinant) but suffers from cascading arithmetic errors, leading to a completely wrong answer. Interestingly, while IMH produces the correct final answer (104), a close inspection of its reasoning reveals that its intermediate steps are mathematically fabricated (e.g., the coordinates do not match the edge lengths, and $63/2$ is not the true volume). IMH essentially "works backwards" from the memorized correct answer 104, hallucinating $m = 63, n = 1, p = 40$ to satisfy the required format. This case highlights a nuanced advantage of global tempering: **robustness to arithmetic bottlenecks via latent knowledge retrieval**. While greedy local heuristics force the model down a brittle computational path that inevitably fails, IMH's trajectory diversity allows the model to bypass the computational bottleneck entirely and successfully retrieve the correct answer from its latent pre-training memory, even if it requires post-hoc rationalization of the intermediate steps.

H Trajectory Similarity Analysis

To quantify the behavioral differences between IMH and the local remasking baselines reported in Section 5, we design an LLM-based trajectory similarity evaluation pipeline. For each of the 30 AIME 2024 problems, we randomly select one generated trajectory per method (Confidence, Margin, Entropy, IMH). The evaluation proceeds in two stages:

Stage 1: Trajectory summarization. Each trajectory is summarized using Claude Opus 4.6 with the following prompt:

Summarization Prompt

Could you summarize this reasoning trajectory into the applied strategy and the intermediate results at each step?

Stage 2: Pairwise similarity scoring. For each pair of methods on each problem, we prompt Claude Opus 4.6 to compare the two trajectory summaries and rate their similarity on a 6-point scale based on the high-level strategy applied:

Similarity Scoring Prompt

Compare these two solution trajectories and determine if they follow similar main approaches. Also point out the key differences in the solution trajectory. You should also measure the rate of similarity between the two trajectories **only based on the high-level strategy applied**. Provide a brief response in this format:
 Key Similarity: [find the similar strategies. If no, simply say no.]
 Other Differences: [explain the main differences in the strategy applied and the discrepancy in the intermediate steps]
 Rate of similarity in strategy: [almost identical / mostly similar / somewhat similar / somewhat different / mostly different / totally different]

The six-point labels are mapped to numeric scores: *almost identical* $\rightarrow 5$, *mostly similar* $\rightarrow 4$, *somewhat similar* $\rightarrow 3$, *somewhat different* $\rightarrow 2$, *mostly different* $\rightarrow 1$, *totally different* $\rightarrow 0$. Self-similarity (diagonal) is set to 5. Final scores are averaged across all 30 problems and normalized to a 0–100 scale (multiplied by 20). The resulting similarity matrix is reported in Figure 3 (right).

I Proofs**I.1 Proof of Proposition 1**

Proposition 1 (Generative perplexity upper bound under confidence gating). *Assume the decoder is $(1 - \delta)$ -gated, the committed token at each step is sampled from the decoder’s commit distribution, and the scoring model is the decoder itself, i.e., $p_{\text{ref}} = p_{\theta}$. Then for any decoding trajectory σ ,*

$$\text{GenPPL}(q_{\text{gen}}; p_{\theta}; \sigma) \leq \exp(h_V(\delta)),$$

where $h_V(\delta) = h_b(\delta) + \delta \log(|V| - 1)$ and $h_b(\delta) = -\delta \log \delta - (1 - \delta) \log(1 - \delta)$.

Proof. Under the self-scoring assumption $p_{\text{ref}} = p_{\theta}$, the exponent of equation 5 can be written as

$$\mathbb{E}_{X \sim q_{\text{gen}}} \left[\frac{1}{L} \sum_{t=1}^L -\log p_{\theta}(x_{\sigma_t} \mid x_{\sigma_{<t}}) \right].$$

By assumption, at each step t , the committed token x_{σ_t} is sampled from the decoder’s commit distribution $p_{\theta}(\cdot \mid s_t)$, where s_t is the current decoder state determined by the partially revealed sequence $x_{\sigma_{<t}}$. Therefore,

$$\mathbb{E}[-\log p_{\theta}(x_{\sigma_t} \mid x_{\sigma_{<t}}) \mid s_t] = H(p_{\theta}(\cdot \mid s_t)).$$

Since the decoder is $(1 - \delta)$ -gated, we have

$$\max_{v \in \mathcal{V}} p_{\theta}(v \mid s_t) \geq 1 - \delta$$

almost surely at every step. By the same maximal-entropy argument used in the proof of Proposition 2, this implies

$$H(p_{\theta}(\cdot \mid s_t)) \leq h_V(\delta),$$

where $h_V(\delta) = h_b(\delta) + \delta \log(|\mathcal{V}| - 1)$. Taking expectation over s_t , averaging over $t = 1, \dots, L$, and exponentiating gives

$$\text{GenPPL}(q_{\text{gen}}; p_{\theta}; \sigma) \leq \exp(h_V(\delta)).$$

□

I.2 Proof of Proposition 2

Proposition 2 (Entropy cap under confidence gating). *Assume the dLLM decoder is $(1 - \delta)$ -gated. Then the sequence entropy satisfies*

$$H(X) \leq L \cdot h_V(\delta),$$

where $h_V(\delta) = h_b(\delta) + \delta \log(|V| - 1)$ and $h_b(\delta) = -\delta \log \delta - (1 - \delta) \log(1 - \delta)$ is the binary entropy function. Equivalently, the effective branching factor, defined as the per-token perplexity of the induced sequence distribution,

$$B_{\text{eff}} := \exp(H(X)/L),$$

is upper bounded as

$$B_{\text{eff}} \leq \exp(h_V(\delta)).$$

Proof. We first formalize the decoding process. Let $\mathcal{V}$ be a vocabulary of size V , and let $X \in \mathcal{V}^L$ be the output of an irreversible decoder that runs for T steps. At step t , the decoder commits a block of $b_t \geq 1$ tokens, with $\sum_{t=1}^T b_t = L$. Let S_t denote the decoder state at the start of step t . Conditioned on S_t , the b_t committed tokens at step t are independent:

$$U_{t,1}, \dots, U_{t,b_t} \mid S_t \text{ are independent, and } U_{t,k} \sim p_{t,k}(\cdot \mid S_t),$$

with the confidence-gating condition $\max_{v \in \mathcal{V}} p_{t,k}(v \mid S_t) \geq 1 - \delta$ holding almost surely for all t, k , where $\delta \in (0, 1 - \frac{1}{V}]$.

Let $U_t := (U_{t,1}, \dots, U_{t,b_t}) \in \mathcal{V}^{b_t}$ be the block committed at step t , and let $U_{1:T}$ denote the concatenation of all committed blocks (a length- L list of tokens).

Because the decoder is irreversible, the final output X is a deterministic function of the committed tokens $U_{1:T}$. Hence

$$H(X) \leq H(U_{1:T}).$$

By the chain rule,

$$H(U_{1:T}) = \sum_{t=1}^T H(U_t \mid U_{1:t-1}).$$

Since the step state S_t is a deterministic function of the past commits $U_{1:t-1}$, conditioning on the full past cannot increase conditional entropy:

$$H(U_t \mid U_{1:t-1}) \leq H(U_t \mid S_t).$$

By within-step conditional independence,

$$H(U_t \mid S_t) = \sum_{k=1}^{b_t} H(U_{t,k} \mid S_t) = \sum_{k=1}^{b_t} H(p_{t,k}(\cdot \mid S_t)).$$

It remains to upper-bound $H(q)$ for any categorical q on V outcomes satisfying $\max_v q(v) \geq 1 - \delta$. Let $\alpha := \max_v q(v)$ and $m := 1 - \alpha \leq \delta$. By concavity of entropy, for fixed m the entropy is maximized by placing mass α on one outcome and spreading the remaining mass m uniformly over the other $V - 1$ outcomes, giving

$$H(q) \leq h_b(m) + m \log(V - 1).$$

Moreover, the function $f(m) := h_b(m) + m \log(V - 1)$ is nondecreasing on $m \in [0, 1 - \frac{1}{V}]$ because

$$f'(m) = \log\left(\frac{(1-m)(V-1)}{m}\right) \geq 0.$$

Thus $H(q) \leq f(m) \leq f(\delta) = h_V(\delta)$.

Applying this bound to each $p_{t,k}(\cdot \mid S_t)$ yields

$$H(U_t \mid S_t) \leq \sum_{k=1}^{b_t} h_V(\delta) = b_t h_V(\delta).$$

Therefore,

$$H(U_{1:T}) \leq \sum_{t=1}^T H(U_t | U_{1:t-1}) \leq \sum_{t=1}^T H(U_t | S_t) \leq \sum_{t=1}^T b_t h_V(\delta) = L h_V(\delta).$$

Finally $H(X) \leq H(U_{1:T})$ gives $H(X) \leq L h_V(\delta)$. Exponentiating $\frac{1}{L} H(X)$ yields the bound on B_{eff} . $\square$

I.3 Proof of Proposition 3

Proposition 3 (Optimality of the power distribution). *For any $\alpha \geq 0$, the unique optimizer of equation 6 is the power distribution*

$$p_\alpha^*(x) = \frac{q(x)^\alpha}{Z_\alpha}, \quad Z_\alpha = \sum_{x \in \mathcal{X}} q(x)^\alpha.$$

Proof. Write $J(p) = \alpha \mathbb{E}_{x \sim p}[\log q(x)] + H(p)$. We show uniqueness and derive the optimizer via two complementary arguments.

Variational derivation. We maximize $J(p)$ over the probability simplex $\Delta(\mathcal{X})$. Introducing a Lagrange multiplier λ for the normalization constraint $\sum_x p(x) = 1$, the stationarity condition is

$$\frac{\partial}{\partial p(x)} \left[\alpha \log q(x) p(x) - p(x) \log p(x) + \lambda p(x) \right] = 0,$$

which gives

$$\alpha \log q(x) - \log p(x) - 1 + \lambda = 0 \implies p(x) = e^{\lambda-1} q(x)^\alpha.$$

Normalizing yields $p_\alpha^*(x) = q(x)^\alpha / Z_\alpha$ with $Z_\alpha = \sum_x q(x)^\alpha$.

Uniqueness via KL divergence. Rewriting the objective:

$$J(p) = \sum_x p(x) [\alpha \log q(x) - \log p(x)] = - \sum_x p(x) \log \frac{p(x)}{q(x)^\alpha} = -\text{KL}(p \| q^\alpha / Z_\alpha) + \log Z_\alpha.$$

Since $\log Z_\alpha$ is a constant independent of p and $\text{KL}(p \| p_\alpha^*) \geq 0$ with equality if and only if $p = p_\alpha^*$, the maximum of J is attained uniquely at $p = p_\alpha^*$. $\square$

I.4 Proof of Proposition 4

Proposition 4 (Corrected conditional for global tempering). *Let q be a distribution over $\mathcal{X} = \mathcal{V}^L$ with full support, and fix $\alpha > 0$. Let $s = (A, x_A)$ denote a partially decoded state with committed positions $A \subseteq [L]$ and remaining positions $R(s) := [L] \setminus A$. For any uncommitted position $i \in R(s)$ and token $v \in \mathcal{V}$, let $\ell_i(s)$ be the logit vector such that $q(x_i = v | s) = \text{softmax}(\ell_i(s))_v$. Letting $s' = s \oplus (i, v)$ denote the updated state after committing token v at position i , define*

$$\Delta_{i,v}(s) := \log \sum_{x_{R(s')} \in \mathcal{V}^{|R(s')|}} q(x_{R(s')} | s')^\alpha. \quad (7)$$

Then the conditional of $p_\alpha^*(x) \propto q(x)^\alpha$ at position i is

$$p_\alpha^*(x_i = v | s) = \frac{\exp(\alpha \ell_{i,v}(s) + \Delta_{i,v}(s))}{\sum_{u \in \mathcal{V}} \exp(\alpha \ell_{i,u}(s) + \Delta_{i,u}(s))}. \quad (8)$$

We prove a more general result from which the lemma follows. For any state $s = (A, x_A)$ and uncommitted position $i \in R(s)$, let $R := R(s) \setminus \{i\}$ and define

$$m_i(v | s) := q(x_i = v | s), \quad w_i(v | s) := \sum_{x_R \in \mathcal{V}^R} q(x_R | s, x_i = v)^\alpha.$$

Then

$$p_\alpha^*(x_i = v | s) = \frac{m_i(v | s)^\alpha w_i(v | s)}{\sum_{u \in \mathcal{V}} m_i(u | s)^\alpha w_i(u | s)}.$$

The logit form in Proposition 4 follows by noting $m_i(v | s)^\alpha \propto \exp(\alpha \ell_{i,v}(s))$ and $\delta_{i,v}(s) = \log w_i(v | s)$.

Proof. By definition of conditional probability under p_α^* ,

$$p_\alpha^*(x_i = v | s) = \frac{\sum_{\mathbf{x}_R \in V^R} p_\alpha^*(x_i = v, \mathbf{x}_R | s)}{\sum_{u \in \mathcal{V}} \sum_{\mathbf{x}_R \in V^R} p_\alpha^*(x_i = u, \mathbf{x}_R | s)}.$$

Using $p_\alpha^*(\cdot | s) \propto q(\cdot | s)^\alpha$ and canceling the common normalizer,

$$p_\alpha^*(x_i = v | s) = \frac{\sum_{\mathbf{x}_R} q(x_i = v, \mathbf{x}_R | s)^\alpha}{\sum_u \sum_{\mathbf{x}_R} q(x_i = u, \mathbf{x}_R | s)^\alpha}.$$

Apply the chain rule under $q(\cdot | s)$:

$$q(x_i = v, \mathbf{x}_R | s) = q(x_i = v | s) q(\mathbf{x}_R | s, x_i = v) = m_i(v | s) q(\mathbf{x}_R | s, x_i = v).$$

Raising to the power α and summing over $\mathbf{x}_R$ gives

$$\sum_{\mathbf{x}_R} q(x_i = v, \mathbf{x}_R | s)^\alpha = m_i(v | s)^\alpha \sum_{\mathbf{x}_R} q(\mathbf{x}_R | s, x_i = v)^\alpha = m_i(v | s)^\alpha w(v | s),$$

which yields the first displayed formula after normalization over $v \in V$.

For the logit form, note that $m_i(v | s) = \exp(\ell_v(s)) / \sum_{u \in \mathcal{V}} \exp(\ell_u(s))$, hence $m_i(v | s)^\alpha \propto \exp(\alpha \ell_v(s))$, where the proportionality constant does not depend on v . Therefore $m_i(v | s)^\alpha w(v | s) \propto \exp(\alpha \ell_v(s) + \delta_v(s))$, and normalizing over v gives the corrected-logit softmax expression. $\square$