RAPD: REPULSION-ATTRACTION ENERGY FOR PARALLEL DECODING OF DIFFUSION LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Parallel decoding accelerates masked diffusion large language models by committing multiple positions per forward pass, but selecting a set of mutually compatible positions remains challenging. Existing methods exploit inter-position cues such as locality and dependency, but most of them model these interactions in only one direction, either promoting or suppressing joint commitment. We propose repulsion–attraction parallel decoding (RAPD), a training-free strategy that combines individual commitment evidence and signed pairwise interactions in a single energy function for commit-set selection. Nearby positions attract each other and promote joint commitment, whereas positions that compete for the same candidate tokens repel each other and suppress conflicting commitment. RAPD iteratively refines commitment probabilities using variational updates, without additional model forward passes. On LLaDA-8B-Instruct, RAPD uses a single configuration across GSM8K, MATH, HumanEval, and MBPP, achieving the fewest forward passes among the compared methods while maintaining competitive accuracy. On HumanEval, RAPD matches the pass@1 of Fast-dLLM with 27.8% fewer forward passes and achieves 4.73$\times$ the throughput of sequential decoding. On the same benchmark, threshold sweeps further show that RAPD matches or exceeds the accuracy of Fast-dLLM and LocalLeap with fewer forward passes across all tested baseline thresholds. Our analysis shows that attraction reduces decoding steps, while repulsion mitigates the accuracy loss from attraction-only decoding.

1 INTRODUCTION

Diffusion large language models (dLLMs) are gaining attention as an alternative to autoregressive models, with parallel decoding offering an opportunity for faster generation (Nie et al., 2025; Ye et al., 2025; Lou et al., 2024). Starting from a fully masked sequence, a dLLM iteratively predicts tokens at masked positions and reveals a subset of tokens at each step; throughout this paper, we refer to fixing such predicted tokens as *committing* them. Unlike autoregressive decoding (Brown et al., 2020), which requires N forward passes for N tokens, a dLLM produces predictive distributions for every remaining masked position in each forward pass and can commit multiple positions before the next step. Since model forward passes dominate decoding cost, reducing the number of function evaluations (NFE) is central to efficient dLLM generation (Wu et al., 2025).

The difficulty is that generation quality generally degrades as more positions are committed together. A single forward pass provides only the marginal distribution at each masked position, so jointly committing several positions relies on a factorized approximation across them. Predictions that appear reliable individually can therefore conflict when committed simultaneously (Figure 1(a)). How many positions to commit determines the speed; which positions to group together largely determines the quality. Existing methods exploit cues such as confidence (Wu et al., 2025), locality (Kong et al., 2025), or dependency (Luo et al., 2026) to guide this decision, but most interaction mechanisms act primarily in one direction, either promoting or suppressing joint commitment. This motivates a unified commit-set criterion that represents attraction and repulsion jointly.

We propose **RAPD (Repulsion–Attraction Parallel Decoding)**, a training-free decoding strategy that formulates commit-set selection using an energy function over binary commitment indicators.

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.

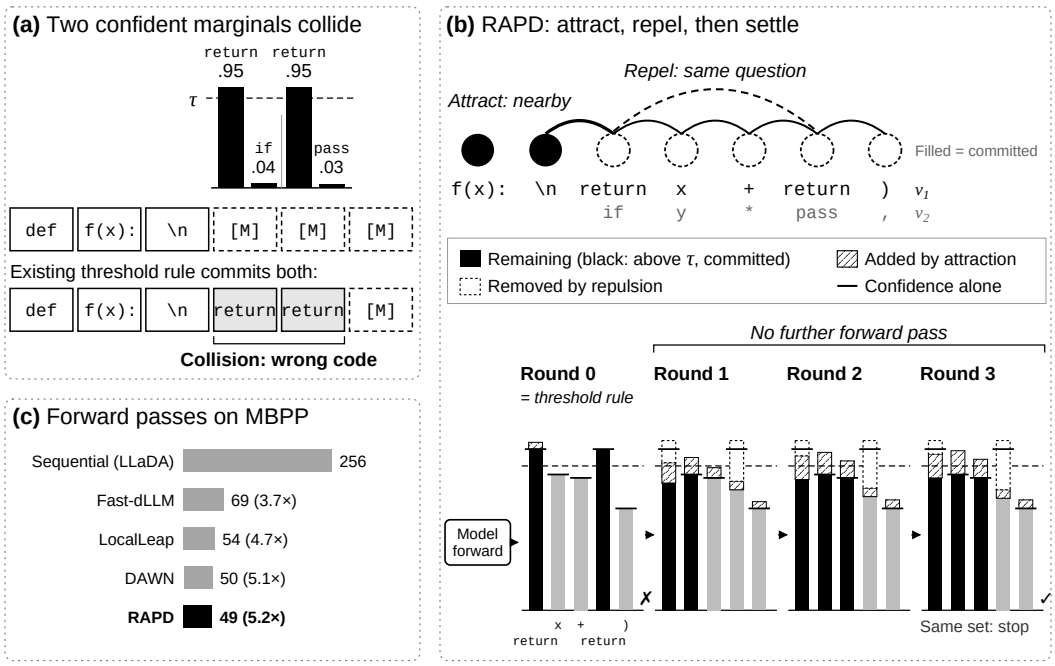


Figure 1: Illustration of the prediction collision in parallel decoding and how RAPD revises the commit set through signed pairwise interactions. (a) Two masked positions assign high confidence to the same token, and a position-wise threshold commits both, producing an incorrect sequence. (b) RAPD couples the positions through attraction and repulsion and iteratively refines their commit probabilities. (c) Number of forward passes required to decode 256 tokens on MBPP.

The energy scores a candidate commit set by combining the evidence for committing each position individually with signed pairwise interactions between positions. Nearby positions attract each other to promote parallel commitment, whereas positions that compete over similar choices repel each other to suppress conflicting commitments. We measure this similarity using the unembedding directions that distinguish each position's two leading candidates. The shared energy provides a common objective for balancing these opposing interactions and deriving the refinement updates. Attraction can bring positions below the confidence threshold into the commit set, while repulsion can defer positions above it. Furthermore, these interactions are computed from quantities already available after the model forward pass and require no additional model evaluations.

To implement this formulation efficiently, we derive iterative updates for commitment probabilities from a variational approximation of the energy-based distribution. The final commit set is obtained through a few lightweight refinement rounds in which positions iteratively update their commitment probabilities in response to one another. RAPD thresholds the refined probabilities to select positions for commitment (Figure 1(b)). In addition, we analyze the convergence of the refinement updates and establish a sufficient coupling-budget condition under which they converge to a unique fixed point.

Our contributions are as follows.

- **Signed energy formulation.** We formulate commit-set selection for parallel decoding as an energy-based problem with unary evidence and signed interactions. The formulation also recovers confidence-threshold and fixed-count commitment rules as special cases.

- **Efficient iterative refinement.** We derive variational updates that refine commitment probabilities without additional model forward passes. We also establish a sufficient coupling-budget condition under which the refinement converges to a unique fixed point.

- **Empirical validation.** Experiments on two backbones and four benchmarks demonstrate decoding efficiency gains with competitive accuracy. Ablations further characterize the accuracy–efficiency trade-off and the complementary roles of attraction and repulsion.

2 Background and Related Work

2.1 Preliminaries

Discrete diffusion language models define generation over a sequence $\boldsymbol{x} = (x_1, \ldots, x_L) \in \mathcal{V}^L$ of length L by reversing a discrete-state noising process (Austin et al., 2021a; Sahoo et al., 2024; Shi et al., 2024). The forward process replaces each token independently with a mask token `[M]`:

$$q_{t|0}(\boldsymbol{x}_t \mid \boldsymbol{x}_0) = \prod_{i=1}^{L} \mathrm{Cat}\big(x_{t,i};\ (1-t)\,\delta_{x_{0,i}} + t\,\delta_{\texttt{[M]}}\big), \qquad t \in [0,1] \tag{1}$$

Here, $\boldsymbol{x}_0 = \boldsymbol{x}$ is the original sequence and $\boldsymbol{x}_1$ is fully masked. The model p_θ learns to predict the original token at each masked position.

Let $\mathcal{U}_t = \{\, i : x_{t,i} = \texttt{[M]} \,\}$ denote the positions that remain masked at decoding step t. A single forward pass produces a predictive distribution π_i over the vocabulary at every position in $\mathcal{U}_t$:

$$p_\theta(x_i \mid \boldsymbol{x}_t) = \mathrm{Cat}(x_i;\, \pi_i), \qquad i \in \mathcal{U}_t \tag{2}$$

At each step, the decoder selects a commit set $\mathcal{S}_t \subseteq \mathcal{U}_t$ from these distributions. The positions in $\mathcal{S}_t$ decide their tokens and become fixed, while the remaining positions stay masked and carry over to the next step. The number of function evaluations (NFE) equals the number of steps until $\mathcal{U}$ empties, not the sequence length.

Following prior work, we adopt semi-autoregressive block decoding (Wang et al., 2018; Nie et al., 2025), splitting the sequence into blocks of size B and completing one block before opening the next. Throughout the paper, $\mathcal{U}$ and $n = |\mathcal{U}|$ refer to the masked positions inside the current block. Parallel decoding gains efficiency by committing more than one position per step ($|\mathcal{S}_t| > 1$). However, because a forward pass provides only per-position predictive distributions, a joint commitment relies on the factorized approximation:

$$p_\theta\big(x_{\mathcal{S}_t} \mid \boldsymbol{x}_t\big) \;\approx\; \prod_{i \in \mathcal{S}_t} p_\theta\big(x_i \mid \boldsymbol{x}_t\big). \tag{3}$$

This approximation is exact only when the tokens at the selected positions are mutually independent given the revealed context. Violations of this assumption have been identified as a problem in non-autoregressive translation (Gu et al., 2018), and their cost on parallel decoding has been empirically studied across tasks (Kang et al., 2026).

2.2 Position-wise Decoding

Since the model provides only the per-position predictive distribution π_i, the simplest commit rules reduce each π_i to a scalar score and select $\mathcal{S}_t$ by ranking or thresholding (Chang et al., 2022). LLaDA (Nie et al., 2025) predicts every masked position and re-masks the positions with low probability, following the rule introduced by Mask-Predict (Ghazvininejad et al., 2019) for non-autoregressive translation. On the other hand, the entropy-based decoder of Dream (Ye et al., 2025) selects low-entropy positions, with the number of commitments determined by a predefined schedule. Fast-dLLM (Wu et al., 2025) instead commits every position whose $\max_v \pi_i(v)$ exceeds a global threshold τ, so the number of committed positions, $|\mathcal{S}_t|$, varies across decoding steps. Other methods incorporate additional position-wise information. EB-Sampler (Ben-Hamu et al., 2025) bounds the number of revealed positions using marginal entropy, and KLASS (Kim et al., 2025) combines confidence with the stability of each prediction across denoising steps. These approaches differ in their scoring criteria and schedules, but their commit scores are primarily determined from each position independently. In other words, they cannot explicitly account for whether two individually plausible commitments interact favorably or conflict when selected together.

2.3 Interaction-aware Decoding

Recent methods incorporate inter-position information when selecting the commit set. Dependency-aware approaches estimate dependency between positions using an auxiliary model (Liu et al., 2025), a lightweight predictor over backbone representations (Ringel et al., 2026), or attention patterns

already produced by the model during the forward pass (Luo et al., 2026). Such dependency estimates are primarily used to suppress simultaneous commitment of strongly coupled or conflicting positions. In contrast, LocalLeap (Kong et al., 2025) exploits locality in the opposite direction; positions near a high-confidence anchor receive a relaxed commitment threshold, promoting additional local commitments. These approaches show that inter-position information can influence parallel decoding in two directions, either suppressing or promoting joint commitment.

DAWN (Luo et al., 2026) incorporates both effects using a dependency graph, but realizes them through two separate decoding rules: one relaxes the confidence threshold for positions connected to committed anchors, while the other prevents conflicting positions from being committed together. RAPD instead combines attraction and repulsion within a single energy function and derives commitment updates from the resulting objective.

3 RAPD: Repulsion-Attraction Energy for Parallel Decoding

3.1 Energy-based Commit-set Formulation

Each denoising step calls the model once to obtain a predictive distribution π_i at every masked position $i \in \mathcal{U}$ in the current block (of size B). Rather than scoring each position independently, RAPD represents the joint commitment decision using binary indicators $\boldsymbol{\sigma} \in \{0,1\}^n$ ($n = |\mathcal{U}|$), where $\sigma_i = 1$ indicates that position i is committed. We assign each commitment configuration the energy:

$$E(\boldsymbol{\sigma}) \;=\; -\sum_i c_i\,\sigma_i \;-\; \frac{1}{2}\sum_{i\neq j} J_{ij}\,\sigma_i\sigma_j \tag{4}$$

Lower energy configurations are preferred. The unary term c_i represents the evidence for committing position i on its own, whereas the pairwise term J_{ij} represents how committing positions i and j together changes their preference for commitment. Note that $J_{ij} > 0$ promotes (i.e., *attraction*) joint commitment, whereas $J_{ij} < 0$ suppresses (i.e., *repulsion*) it between i and j.

3.2 Unary Evidence and Signed Pairwise Interaction

Unary evidence. The unary term c_i measures the evidence for committing position i independently of the other positions. Common criteria include the margin between the two most likely tokens and the absolute confidence of the leading token. Both can be generalized as the log ratio between the leading probability and the probability mass assigned to competing candidates. We therefore define the unary evidence as:

$$c_i^{(k)} \;=\; \log p_{(1)}(i) \;-\; \mathrm{logsumexp}\big(\log p_{(2)}(i),\ldots,\log p_{(k)}(i)\big) \;=\; \log\frac{p_{(1)}(i)}{\sum_{r=2}^{k} p_{(r)}(i)} \tag{5}$$

where $p_{(r)}(i)$ is the r-th largest probability in π_i, and $v_r(i)$ is its corresponding token. Setting the cutoff rank $k = 2$ recovers the top-two log-probability margin, and $k = V$ gives the absolute confidence $\mathrm{logit}(p_{(1)})$. Increasing k includes more competing probability mass and decreases the unary evidence.

Signed pairwise interaction. The pairwise term combines an attractive interaction based on positional proximity with a repulsive interaction based on similarity between token decisions:

$$J_{ij} \;=\; \underbrace{w_{\text{loc}}\exp\Big(-\frac{d_{ij}^2}{2s^2}\Big)}_{\text{attraction}} \;-\; \underbrace{w_{\text{sim}}\big(\mathcal{T}_\kappa[S]\big)_{ij}}_{\text{repulsion}}, \qquad S_{ij} = \big[\cos(\boldsymbol{u}_i,\boldsymbol{u}_j)\big]_+, \quad J_{ii} = 0 \tag{6}$$

The attraction term is a Gaussian kernel over the within-block distance d_{ij}, where the width s controls its spatial range. It assigns stronger positive interactions to nearby positions.

For repulsion, we define the decision direction of each position i from its two leading token candidates. The difference between their unembedding vectors gives the gradient of the logit margin with respect

to the hidden state. Because the logit of token v is determined by the inner product of the hidden state with the corresponding unembedding vector $W_U[v]$, the *decision direction*

$$\boldsymbol{u}_i = \text{Normalize}\Big(W_U[v_1(i)] - W_U[v_2(i)]\Big) \tag{7}$$

represents the local decision between the leading candidate and its closest competitor. Positions with aligned directions receive a larger interaction score, S_{ij}, producing a stronger negative contribution to J_{ij} and making them less likely to be committed together. We use positive cosine similarity between decision directions as a proxy for competing token decisions.

Truncation. Unlike the locality kernel, whose contribution decays with distance, decision-direction similarity is not inherently local, so its total repulsive contribution can grow with the number of candidate positions. For each position i, let

$$\mathcal{N}_\kappa(i) = \text{TopK}_{j \neq i}(S_{ij}, \kappa) \tag{8}$$

denote the indices of the κ largest off-diagonal similarities in row i of S, with ties broken deterministically. We define the mutual top-κ truncation operator as

$$\big(\mathcal{T}_\kappa[S]\big)_{ij} = S_{ij}\, \mathbb{I}[j \in \mathcal{N}_\kappa(i)]\, \mathbb{I}[i \in \mathcal{N}_\kappa(j)]\,. \tag{9}$$

Thus, (i, j) is retained only when the two positions lie within each other's top-κ similarities; otherwise its value is set to zero. This mutual truncation preserves symmetry and leaves at most κ nonzero repulsive interactions in each row, yielding

$$\sum_j w_{\text{sim}} \big(\mathcal{T}_\kappa[S]\big)_{ij} \leq \kappa w_{\text{sim}}, \tag{10}$$

independently of the block size (Appendix Proposition 1). This bound is used in Section 3.4 to derive the coupling-budget condition for convergence.

3.3 Iterative Commitment Refinement

Interactions with the positions already committed in the current block, $\mathcal{A}$, contribute a fixed field $\text{static}_i = \sum_{j \in \mathcal{A}} J_{ij}$. Adding this field to Eq. (4) gives $E_{\mathcal{A}}(\boldsymbol{\sigma}) = E(\boldsymbol{\sigma}) - \sum_{i \in \mathcal{U}} \text{static}_i\, \sigma_i$, which induces a distribution over commitment configurations,

$$P(\boldsymbol{\sigma}) = \frac{1}{Z} \exp\left(-\frac{E_A(\boldsymbol{\sigma})}{T}\right), \tag{11}$$

where $T > 0$ is the temperature and Z is the partition function. Under this distribution, the commitment probability of position i is the marginal $P(\sigma_i = 1)$. We commit positions whose commitment probabilities exceed a threshold τ, allowing the number of commitments to adapt to the current decoding state.

Computing the marginals of $P(\boldsymbol{\sigma}) \propto e^{-E_A(\boldsymbol{\sigma})/T}$ exactly requires summing over exponentially many configurations, which is intractable. We approximate P with a fully factorized distribution

$$Q(\boldsymbol{\sigma}) = \prod_i q_i^{\sigma_i} (1 - q_i)^{1 - \sigma_i}, \tag{12}$$

where q_i approximates the marginal commitment probability $P(\sigma_i = 1)$. Importantly, the factorization applies only to the commitment indicators; interactions between positions remain through the dependence of each q_i on the others.

The stationary points of $\text{KL}(Q \| P)$ satisfy the fixed-point condition (Appendix Proposition 3):

$$h_i(\mathbf{q}) = c_i + \text{static}_i + \sum_{j \neq i} J_{ij} q_j, \qquad q_i = \text{sigmoid}\left(\frac{h_i(\mathbf{q})}{T}\right) \tag{13}$$

We solve Eq. (13) using damped iterative refinement with $\alpha \in (0, 1]$,

$$\mathbf{q}^{(r+1)} = (1 - \alpha)\mathbf{q}^{(r)} + \alpha\ \text{sigmoid}\left(\frac{\mathbf{h}(\mathbf{q}^{(r)})}{T}\right). \tag{14}$$

We initialize $\mathbf{q}^{(0)} = \mathrm{sigmoid}((\mathbf{c} + \mathbf{static})/T)$, perform at most K refinement rounds, and stop early once the committed set $\{ i : q_i > \tau \}$ remains unchanged for two consecutive rounds. We select the final commit set by thresholding the refined probabilities and assign each selected position its greedy token $\arg\max_v \pi_i(v)$. If thresholding selects no position, we commit the position with the largest unary score; when the block step cap is reached, we commit all remaining positions using their latest greedy predictions.

The interaction matrix is constructed once per model forward pass and reused throughout refinement, which requires no additional model evaluations. Moreover, each pairwise interaction is computed in the decision-direction dimension d rather than the vocabulary dimension V; for LLaDA-8B-Instruct, $d = 4096$ vs. $V = 126{,}464$ ($d \ll V$).

3.4 Coupling Budget and Convergence

The strength of the pairwise interactions determines the stability of the iterative refinement in Eq. (14). A sufficient condition for stable refinement can be expressed in terms of the maximum absolute row sum $R = \max_i \sum_j |J_{ij}|$. Since the derivative of the sigmoid is bounded by $1/4$, the refinement becomes a contraction when $R < 4T$, in which case it converges geometrically to a unique fixed point (Appendix Proposition 2). Thus, the total attraction and repulsion assigned to each position should remain within a finite coupling budget.

Using the attraction and repulsion terms in Eq. (6), this condition is satisfied if

$$w_{\mathrm{loc}} \left(\sqrt{2\pi}\, s\, \vartheta(s) - 1 \right) + w_{\mathrm{sim}} \kappa < 4T\eta, \qquad \eta < 1, \tag{15}$$

where η is a safety factor and $\vartheta(s)$ is defined in Appendix Lemma 1. The derivation and convergence analysis are provided in the appendix (Corollary 1). This bound is independent of block size and can be checked before decoding. We select the coupling parameters to satisfy Eq. (15), and use the resulting configuration across two backbones and all four benchmarks in our main evaluation.

4 Experiments

4.1 Setup

We evaluate all methods on LLaDA-8B-Instruct (Nie et al., 2025) and Dream-7B-Instruct (Ye et al., 2025). Unless stated otherwise, all runs use greedy decoding, block size 32, generation length 256, and batch size 1 on a single NVIDIA RTX 4090 GPU (24 GB). We compare RAPD with sequential decoding, which decodes one token per step for 256 steps, and with three training-free parallel decoders: Fast-dLLM (Wu et al., 2025), LocalLeap (Kong et al., 2025), and DAWN (Luo et al., 2026). We run all baselines ourselves without tuning, using the settings that the evaluation scripts released with DAWN specify for each backbone. RAPD uses a single configuration per backbone across all benchmarks (Appendix B and C).

4.2 Main Results

On LLaDA-8B-Instruct, RAPD achieves the lowest NFE and highest measured TPS across all four benchmarks with a single configuration (Table 1). While NFE counts model evaluations, TPS also reflects refinement overhead. RAPD limits this overhead by computing unembedding-based similarities once per forward pass and reusing the interaction matrix in lightweight refinement rounds. On HumanEval, RAPD matches the pass@1 of Fast-dLLM with 27.8% fewer forward passes and achieves 4.73× the throughput of sequential decoding. On Dream-7B-Instruct, RAPD retains the same coupling parameters, adjusting only the commitment threshold and per-block step cap (Table 2). RAPD also achieves both the lowest NFE and the highest TPS on three of the four benchmarks, extending the efficiency gains to another backbone.

4.3 Accuracy–NFE Trade-off

RAPD defines the empirical accuracy–NFE Pareto frontier among the compared methods on HumanEval (Figure 2). We sweep the thresholds of RAPD, Fast-dLLM, and LocalLeap over

Table 1: **Performance on LLaDA-8B-Instruct.** Numbers are measured relative to sequential decoding on the same benchmark. Both NFE reductions and TPS speedups are relative to sequential decoding. Bold and underline indicate the best and second-best results, respectively.

Benchmark	Method	NFE ↓	Acc. ↑	TPS ↑
GSM8K (5-shot)	LLaDA	256.0 1.00×	77.56	4.89 1.00×
	+ DAWN	<u>56.0</u> <u>4.57×</u>	77.18	<u>21.46</u> <u>4.39×</u>
	+ LocalLeap	58.9 4.35×	**78.39**	20.90 4.27×
	+ Parallel (Fast-dLLM)	78.5 3.26×	**78.39**	15.78 3.23×
	+ RAPD (ours)	**55.9 4.58×**	<u>77.79</u>	**24.08 4.92×**
MATH (4-shot)	LLaDA	256.0 1.00×	**33.00**	7.55 1.00×
	+ DAWN	<u>72.5</u> <u>3.53×</u>	32.30	<u>26.01</u> <u>3.44×</u>
	+ LocalLeap	75.7 3.38×	32.14	25.16 3.33×
	+ Parallel (Fast-dLLM)	97.1 2.64×	<u>32.98</u>	19.84 2.63×
	+ RAPD (ours)	**69.5 3.68×**	31.56	**30.26 4.01×**
HumanEval (0-shot)	LLaDA	256.0 1.00×	<u>39.63</u>	15.23 1.00×
	+ DAWN	64.1 3.99×	37.80	60.82 3.99×
	+ LocalLeap	<u>61.0</u> <u>4.20×</u>	<u>39.63</u>	<u>61.21</u> <u>4.02×</u>
	+ Parallel (Fast-dLLM)	78.0 3.28×	**40.24**	50.05 3.29×
	+ RAPD (ours)	**56.3 4.55×**	**40.24**	**72.06 4.73×**
MBPP (3-shot)	LLaDA	256.0 1.00×	40.40	4.93 1.00×
	+ DAWN	<u>50.0</u> <u>5.12×</u>	40.60	<u>24.18</u> <u>4.91×</u>
	+ LocalLeap	54.3 4.72×	**42.20**	23.53 4.77×
	+ Parallel (Fast-dLLM)	69.1 3.70×	**42.20**	18.59 3.77×
	+ RAPD (ours)	**49.4 5.18×**	<u>41.00</u>	**28.09 5.70×**

Table 2: **Performance on Dream-7B-Instruct.** Notation follows Table 1.

Benchmark	Method	NFE ↓	Acc. ↑	TPS ↑
GSM8K (5-shot)	Dream	256.0 1.00×	<u>77.10</u>	3.58 1.00×
	+ DAWN	**51.9 4.93×**	73.77	**16.60 4.64×**
	+ LocalLeap	<u>54.1</u> <u>4.73×</u>	74.75	<u>16.33</u> <u>4.56×</u>
	+ Parallel (Fast-dLLM)	62.4 4.10×	75.36	14.31 4.00×
	+ RAPD (ours)	56.4 4.54×	**77.41**	16.30 <u>4.56×</u>
MATH (4-shot)	Dream	256.0 1.00×	<u>42.84</u>	5.74 1.00×
	+ DAWN	<u>94.6</u> <u>2.71×</u>	42.08	<u>15.00</u> <u>2.61×</u>
	+ LocalLeap	97.9 2.62×	42.46	14.39 2.51×
	+ Parallel (Fast-dLLM)	115.4 2.22×	**43.22**	12.42 2.16×
	+ RAPD (ours)	**77.7 3.30×**	41.44	**19.88 3.46×**
HumanEval (0-shot)	Dream	256.0 1.00×	52.44	10.43 1.00×
	+ DAWN	<u>80.1</u> <u>3.20×</u>	53.66	<u>27.96</u> <u>2.68×</u>
	+ LocalLeap	81.3 3.15×	54.27	26.54 2.54×
	+ Parallel (Fast-dLLM)	94.9 2.70×	<u>56.10</u>	23.55 2.26×
	+ RAPD (ours)	**69.5 3.68×**	**59.76**	**33.82 3.24×**
MBPP (3-shot)	Dream	256.0 1.00×	53.40	1.70 1.00×
	+ DAWN	<u>43.0</u> <u>5.96×</u>	55.20	<u>9.10</u> <u>5.37×</u>
	+ LocalLeap	43.6 5.87×	<u>55.40</u>	8.98 5.29×
	+ Parallel (Fast-dLLM)	62.7 4.08×	**55.60**	6.38 3.76×
	+ RAPD (ours)	**30.9 8.29×**	53.80	**13.12 7.74×**

$\{0.80, 0.85, 0.90, 0.95\}$ and vary RAPD's attraction–repulsion balance at each threshold under a fixed coupling budget. The main RAPD configuration ($\tau = 0.85$) achieves 40.2% accuracy at 56.3 NFE, matching or exceeding the accuracy of every tested Fast-dLLM and LocalLeap setting with fewer forward passes. RAPD also reaches 40.9% accuracy at 76.7 NFE, extending the frontier beyond the highest baseline accuracy.

Higher thresholds make commitment more conservative but do not consistently improve accuracy: with the main coupling weights, increasing τ from 0.85 to 0.95 raises NFE while accuracy remains at 40.2%. At each threshold, allocating more of the coupling budget to attraction reduces NFE, although excessive attraction sharply degrades accuracy at $\tau = 0.80$. Lower thresholds therefore benefit from stronger repulsion, whereas higher thresholds permit more attraction without sacrificing accuracy in general (Appendix G).

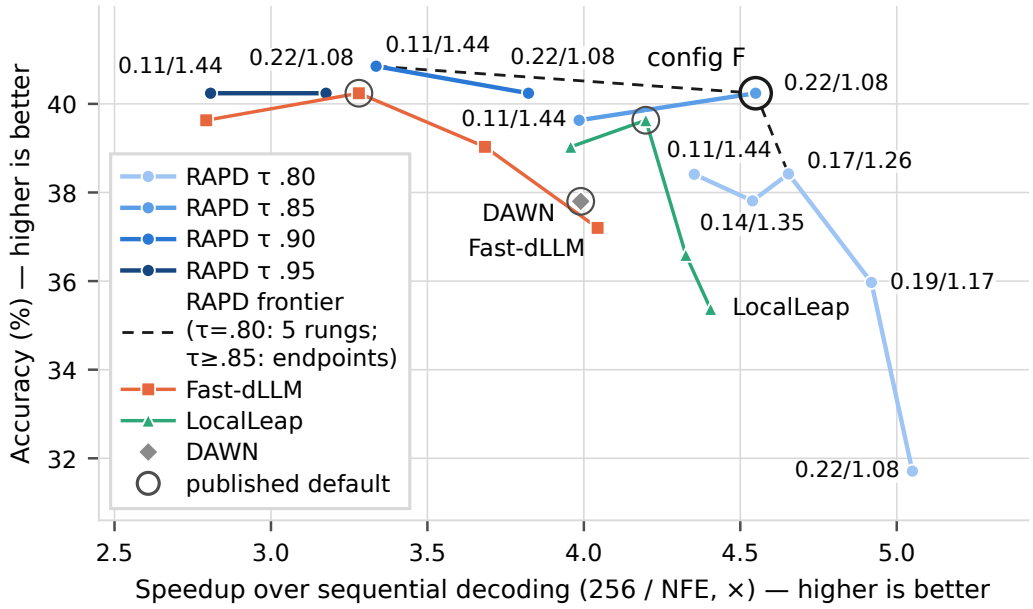


Figure 2: Accuracy–NFE trade-off on HumanEval. Each RAPD curve fixes τ and varies the attraction–repulsion balance under a fixed coupling budget; point labels list $w_{\text{loc}}/w_{\text{sim}}$. The dashed line marks the Pareto frontier of the plotted settings. Circles mark baseline defaults and RAPD's main configuration (labeled config F); DAWN is shown only at its published setting.

Table 3: Effects of repulsion allocation and the unary criterion. (a) Repulsion allocation at similar coupling budgets. (b) Absolute confidence ($k = V$) versus the top-two margin ($k = 2$). All other settings are fixed within each comparison.

Benchmark	Condition	NFE ↓	Acc. ↑
(a) Repulsion allocation			
HumanEval	$w_{\text{loc}} = 0.22$, $w_{\text{sim}} = 1.08$, $\kappa = 2$ (main result)	56.3 4.55×	**40.24**
	$w_{\text{loc}} = 0.22$, $w_{\text{sim}} = 0.50$, $\kappa = 4$ (weaker suppression)	55.5 4.61×	35.98
(b) Unary criterion			
GSM8K	$k = V$ (absolute confidence)	55.9 4.58×	**77.79**
	$k = 2$ (margin)	35.5 **7.22**×	70.13
MATH	$k = V$ (absolute confidence)	69.5 3.68×	**31.56**
	$k = 2$ (margin)	42.7 **6.00**×	28.14
HumanEval	$k = V$ (absolute confidence)	56.3 4.55×	**40.24**
	$k = 2$ (margin)	33.1 **7.73**×	25.61
MBPP	$k = V$ (absolute confidence)	49.4 5.18×	**41.00**
	$k = 2$ (margin)	26.6 **9.62**×	27.60

5 Analysis

5.1 Effects of Design Choices

Table 3(a) examines how repulsion is allocated under the coupling-budget constraint in Eq. (15). We fix $w_{\text{loc}} = 0.22$ and jointly vary κ and w_{sim}, which enter the repulsive contribution to the bound through their product κw_{sim}. This trades the number of repulsive neighbors against the strength of each interaction while maintaining coupling budgets of similar magnitude (3.594 vs. 3.434), both below the prescribed limit of 3.600. In contrast, Figure 2 fixes κ and varies w_{loc} and w_{sim} to redistribute a fixed budget between attraction and repulsion. Concentrating stronger repulsion on fewer pairs ($\kappa = 2$, $w_{\text{sim}} = 1.08$) improves HumanEval accuracy by 4.27 percentage points over spreading weaker repulsion across more pairs ($\kappa = 4$, $w_{\text{sim}} = 0.50$), with nearly unchanged NFE

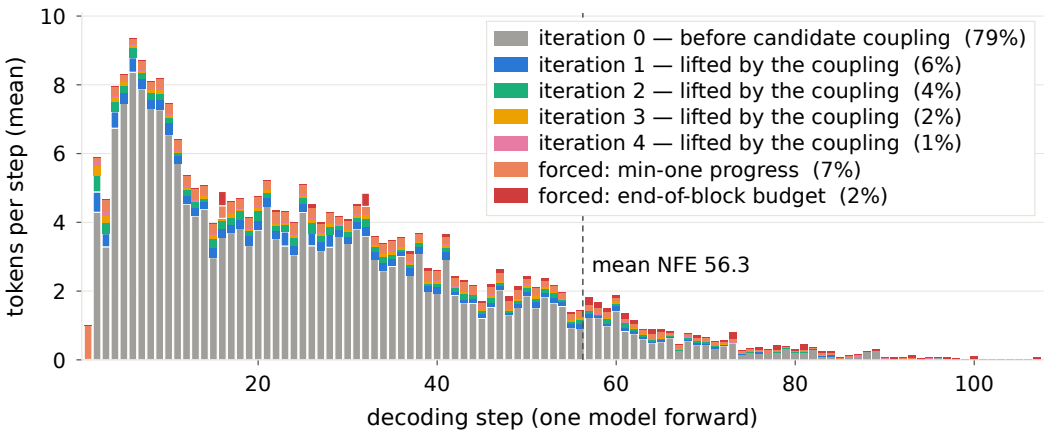


Figure 3: Commitment breakdown by decoding step on HumanEval, averaged over all 164 problems. Stacked bars distinguish positions already above threshold at initialization, positions first crossing it during refinement, and positions committed by safeguards. The dashed line marks mean NFE.

(56.3 vs. 55.5). This suggests that selectively suppressing the most strongly competing positions can be more effective than distributing weaker suppression across a broader set of pairs.

Table 3(b) compares absolute confidence ($k = V$) with the top-two margin ($k = 2$), using the same coupling parameters and commitment threshold. Absolute confidence accounts for the total probability mass of competing tokens, providing more conservative commitment evidence when several alternatives remain plausible. The top-two margin compares only the two most probable tokens, producing more aggressive commitment and lower NFE. Absolute confidence achieves higher accuracy across all four benchmarks, supporting its use in the main RAPD configuration.

5.2 Commitment Refinement

Figure 3 traces the contribution of refinement to the final commit sets on HumanEval. Of the final commitments, 79.5% are already above the threshold at initialization, while 11.9% first cross it during refinement, predominantly within the first two rounds. The remaining 8.6% are committed by the progress and block-completion safeguards.

Successive refinement rounds contribute progressively fewer new threshold crossings, decreasing from approximately 6% of all commitments in the first round to 1% in the fourth. This diminishing contribution is consistent with the convergence analysis in Section 3.4, which guarantees geometric convergence of the commitment probabilities under the coupling-budget condition. Most observed additional commitments arise within the first two rounds, with later rounds making smaller contributions. RAPD's iterative refinement thus uncovers additional opportunities for parallel unmasking beyond the initial threshold-based selection.

6 Conclusion

We presented RAPD, a training-free parallel decoding strategy for dLLMs that jointly models attractive and repulsive interactions between commitment decisions. By formulating commit-set selection with signed pairwise interactions, RAPD promotes local commitments while suppressing conflicting ones within a single formulation. We derived lightweight variational updates that refine commitment probabilities without additional model forward passes and established a sufficient, block-size-independent coupling-budget condition for convergence to a unique fixed point. On LLaDA-8B-Instruct, RAPD required the fewest forward passes and achieved the highest measured throughput on all four benchmarks; on Dream-7B-Instruct, it did so on three of the four benchmarks. Its accuracy stayed within 1.8 points of the most accurate compared method on every benchmark. These results suggest that explicitly modeling both directions of inter-position interaction is an effective way to improve the efficiency of parallel dLLM decoding.

AI USE STATEMENT

We used generative AI tools only for language editing to improve the readability of the manuscript. We did not use generative AI tools for any other task requiring disclosure, such as designing or implementing methods, conducting experiments, or analyzing results. We have reviewed and revised all AI-assisted text, and we take responsibility for the final content of this work.

ETHICS STATEMENT

This work focuses primarily on the theoretical analysis of inference in diffusion language models. It uses only publicly available models, LLaDA-8B-Instruct (Nie et al., 2025) and Dream-7B-Instruct (Ye et al., 2025), and publicly available benchmarks, GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), HumanEval (Chen et al., 2021), and MBPP (Austin et al., 2021b). Within this scope, we do not expect specific ethical concerns or direct negative societal impacts beyond the general risks associated with language models.

REPRODUCIBILITY STATEMENT

We will release the code that can reproduce the reported values. We provide hyperparameters and baseline settings in Appendix C, the full algorithm in Appendix D, and the proofs in Appendix A.

REFERENCES

Marianne Arriola, Aaron Gokaslan, Justin Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *International Conference on Learning Representations*, 2025.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 2021a.

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*, 2021b.

Heli Ben-Hamu, Itai Gat, Daniel Severo, Niklas S Nolte, and Brian Karrer. Accelerated sampling from masked diffusion models via entropy bounded unmasking. *Advances in Neural Information Processing Systems*, 2025.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11305–11315. IEEE, 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.

Xinhua Chen, Sitao Huang, Cong Guo, Chiyue Wei, Yintao He, Jianyi Zhang, Hai Li, and Yiran Chen. Dpad: Efficient diffusion language models with suffix dropout. In *International Conference on Learning Representations*, 2026a.

Zigeng Chen, Gongfan Fang, Xinyin Ma, Ruonan Yu, and Xinchao Wang. dparallel: Learnable parallel decoding for dllms. In *International Conference on Learning Representations*, 2026b.

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

R. L. Dobrushin. The description of a random field by means of conditional probabilities and conditions of its regularity. *Theory of Probability and Its Applications*, 13(2):197–224, 1968.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. Mask-predict: Parallel decoding of conditional masked language models. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, 2019.

Andrzej Granas and James Dugundji. *Fixed Point Theory*. Springer, 2003.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O.K. Li, and Richard Socher. Non-autoregressive neural machine translation. In *International Conference on Learning Representations*, 2018.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2nd edition, 2012.

Metod Jazbec, Theo X. Olausson, Louis Béthune, Pierre Ablin, Michael Kirchhof, Joao Monteiro, Victor Guilherme Turrisi da Costa, Jason Ramapuram, and marco cuturi. Learning unmasking policies for diffusion language models. In *Forty-third International Conference on Machine Learning*, 2026.

Wonjun Kang, Kevin Galim, Seunghyuk Oh, Minjae Lee, Yuchen Zeng, Shuibai Zhang, Coleman Hooper, Yuezhou Hu, Hyung Koo, Nam Ik Cho, et al. Parallelbench: Understanding the trade-offs of parallel decoding in diffusion llms. In *International Conference on Learning Representations*, 2026.

Seo Hyun Kim, Sunwoo Hong, Hojung Jung, Youngrok Park, and Se-Young Yun. Klass: Kl-guided fast inference in masked diffusion models. *Advances in Neural Information Processing Systems*, 2025.

Fanheng Kong, Jingyuan Zhang, Yahui Liu, Zirui Wu, Yu Tian, and Guorui Zhou. Accelerating diffusion llm inference via local determinism propagation. *arXiv preprint arXiv:2510.07081*, 2025.

Anji Liu, Oliver Broadrick, Mathias Niepert, and Guy Van den Broeck. Discrete copula diffusion. In *International Conference on Learning Representations*, 2025.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*, 2024.

Lizhuo Luo, Zhuoran Shi, Jiajun Luo, Zhi Wang, Shen Ren, Wenya Wang, and Tianwei Zhang. Dawn: Dependency-aware fast inference for diffusion llms. *arXiv preprint arXiv:2602.06953*, 2026.

Joris M. Mooij and Hilbert J. Kappen. Sufficient conditions for convergence of the sum-product algorithm. *IEEE Transactions on Information Theory*, 53(12):4422–4437, 2007.

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*, volume 38, pp. 50608–50646, 2025.

Manfred Opper and David Saad. Advanced mean field methods: Theory and practice. *MIT Press*, 2001.

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

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

Subham Sekhar Sahoo, Zhihan Yang, Yash Akhauri, Johnna Liu, Deepansha Singh, Zhoujun Cheng, Zhengzhong Liu, Eric P. Xing, John Thickstun, and Arash Vahdat. Esoteric language models: A family of any-order diffusion LLMs. In *International Conference on Machine Learning*, 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*, 2024.

Yuerong Song, Xiaoran Liu, Ruixiao Li, Zhigeng Liu, Zengfeng Huang, Qipeng Guo, Ziwei He, and Xipeng Qiu. Sparse-dllm: Accelerating diffusion llms with dynamic cache eviction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

Elias M. Stein and Rami Shakarchi. *Fourier Analysis: An Introduction*. Princeton University Press, 2003.

Sekhar C. Tatikonda and Michael I. Jordan. Loopy belief propagation and Gibbs measures. In *Uncertainty in Artificial Intelligence (UAI)*, pp. 493–500, 2002.

Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008.

Chunqi Wang, Ji Zhang, and Haiqing Chen. Semi-autoregressive neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

Zeqing Wang, Gongfan Fang, Xinyin Ma, Xingyi Yang, and Xinchao Wang. Sparsed: Sparse attention for diffusion language models. In *International Conference on Learning Representations*, 2026.

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.

Chengyue Wu, Hao Zhang, Shuchen Xue, Shizhe Diao, Yonggan Fu, Zhijian Liu, Pavlo Molchanov, Ping Luo, Song Han, and Enze Xie. Fast-dllm v2: Efficient block-diffusion llm. *International Conference on Learning Representations*, 2026.

Zhongyu Xiao, Zhiwei Hao, Jianyuan Guo, Yong Luo, Jia Liu, Jie Xu, and Han Hu. Streaming-dllm: Accelerating diffusion llms via suffix pruning and dynamic decoding. *arXiv preprint arXiv:2601.17917*, 2026.

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.

Zijian Zhu, Fei Ren, Zhanhong Tan, and Kaisheng Ma. Es-dllm: Efficient inference for diffusion large language models by early-skipping. In *International Conference on Learning Representations*, 2026.

A PROPOSITIONS AND PROOFS

Proposition 1 (Mutual truncation). *Let $\mathcal{N}_\kappa(i)$ be the neighborhood set of Equation* (8)*. Under any fixed tie-breaking rule, $|\mathcal{N}_\kappa(i)| = \min(\kappa,\, n-1)$. For every i,*

$$\sum_j w_{\mathrm{sim}}\big(\mathcal{T}_\kappa[S]\big)_{ij} \;\le\; \kappa\, w_{\mathrm{sim}},$$

and this bound depends neither on n nor on how ties are broken. A cluster of $\kappa+1$ positions competing over the same ordered pair attains equality, so the bound cannot be improved. The union variant and the value-based threshold, by contrast, do not guarantee this bound: for both variants there exist configurations whose row sum reaches $(n-1)w_{\mathrm{sim}}$.

Proof. Upper bound. For (i,j) to survive, it is necessary in particular that $j \in \mathcal{N}_\kappa(i)$, so row i retains at most $|\mathcal{N}_\kappa(i)| \le \kappa$ entries, and since $[\,\cdot\,]_+ \le 1$, the row sum is at most κw_{sim}. The argument uses neither n nor the tie-breaking rule.

Attainment. The decision direction $\boldsymbol{u}_i$ is a function of the ordered pair $(v_1(i), v_2(i))$ alone, so positions competing over the same pair are exactly tied at $\cos = 1$. Suppose $n \ge \kappa+2$, that $\kappa+1$ positions compete over one pair, and that every other pair has $\cos \le 0$. In each row of the cluster, $\mathcal{N}_\kappa$ coincides exactly with the other κ cluster members and the relation is mutual, so all κ entries survive with value 1 and the row sum equals κw_{sim}.

Union variant. Retaining (i,j) when $j \in \mathcal{N}_\kappa(i)$ *or* $i \in \mathcal{N}_\kappa(j)$ produces the symmetric k-nearest-neighbor graph instead of the mutual k-nearest-neighbor graph (von Luxburg, 2007). If every candidate in the block competes over one pair, all pairs are tied at $\cos = 1$, and under index-order tie-breaking the position with the smallest index enters the top κ of every other row, so its row sum is $(n-1)w_{\mathrm{sim}}$.

Value-based threshold. "Retain everything at least as large as the κ-th largest value" keeps every entry tied with the κ-th, so in the same configuration one row retains the entire block. The rule sets a separate threshold for each row, so the mask is asymmetric. The symmetry of J is therefore lost, and with it Proposition 3, which relies on that symmetry.

Both variants attain a row sum linear in n, so no fixed $(w_{\mathrm{sim}}, \kappa)$ can maintain the condition of Proposition 2 as the block grows. □

Lemma 1 (Attraction row sum). *Independently of the block size,*

$$\sum_{j\neq i} \exp\Big(-\frac{d_{ij}^2}{2s^2}\Big) \;\le\; \sqrt{2\pi}\, s\, \vartheta(s) - 1, \qquad \vartheta(s) = 1 + 2\sum_{m\ge 1} e^{-2\pi^2 m^2 s^2}. \tag{16}$$

The theta factor contributes $\sqrt{2\pi}\, s\,\big(\vartheta(s)-1\big)$ to the bound. This quantity decreases in s for $s \ge 1$ and stays below 1.5×10^{-8} on that range; at $s=3$ it falls below 2×10^{-76}, so the bound is indistinguishable from $\sqrt{2\pi}\, s - 1$ in double precision.

Proof. The positions lie on the integer lattice and $j \mapsto p_j - p_i$ is injective into $\mathbb{Z}\setminus\{0\}$, so the row sum for any finite block is bounded above by the sum over the whole lattice. Since $f(x) = e^{-x^2/2s^2}$ is a Schwartz function with Fourier transform $\hat f(\xi) = \sqrt{2\pi}\, s\, e^{-2\pi^2 s^2 \xi^2}$, the Poisson summation formula (Stein & Shakarchi, 2003) gives

$$\sum_{d\in\mathbb{Z}} f(d) \;=\; \sum_{m\in\mathbb{Z}} \hat f(m) \;=\; \sqrt{2\pi}\, s\Big(1 + 2\sum_{m\ge 1} e^{-2\pi^2 m^2 s^2}\Big) \;=\; \sqrt{2\pi}\, s\, \vartheta(s),$$

and removing the $d=0$ term $f(0)=1$ produces Equation (16).

For $s \ge 1$ we have $m^2 \ge 2m-1$, so $\vartheta(s) - 1 \le 2e^{-2\pi^2 s^2}\big(1-e^{-4\pi^2 s^2}\big)^{-1} < 2.1\, e^{-2\pi^2 s^2}$. Moreover, each term of $s\big(\vartheta(s)-1\big) = 2\sum_{m\ge1} s\, e^{-2\pi^2 m^2 s^2}$ has derivative $2e^{-2\pi^2 m^2 s^2}\big(1-4\pi^2 m^2 s^2\big)$, which is negative for $s \ge 1$, so the whole sum decreases and attains its maximum at $s=1$: $\sqrt{2\pi}\big(\vartheta(1) - 1\big) < 2.1\sqrt{2\pi}\, e^{-2\pi^2} < 1.5\times 10^{-8}$. At $s=3$, $2\pi^2 s^2 = 177.653$, so $\sqrt{2\pi}\cdot 3\,\big(\vartheta(3)-1\big) < 6.3\sqrt{2\pi}\, e^{-177.653} < 2\times 10^{-76}$. □

Remark 1 (Why the -1 matters)**.** *Since f decreases monotonically on $[0,\infty)$, the integral comparison gives $\sum_{d\neq 0} f(d) \le \int_{-\infty}^{\infty} f = \sqrt{2\pi}\, s$. This is true but loses the -1 that removes the self-coupling term. At $s = 3$ the looser bound is 7.5199, and with the coupling coefficients of Appendix C the row-sum bound becomes $0.22\times 7.5199 + 2\times 1.08 = 3.8144$, which exceeds the limit 3.600. Lemma 1, by contrast, gives $0.22 \times 6.5199 + 2 \times 1.08 = 3.5944$. Whether the budget holds depends on the sharpness of this bound, so we state it in the Poisson form. Numerically, the worst-case row sum is 6.519884 at block size 32 and 6.519885 at block size 128, both below the bound 6.519885, and it does not grow with the block size.*

Proposition 2 (Contraction)**.** *Let $\alpha \in (0, 1]$. If $R < 4T$, the update of Equation* (14) *maps $[0, 1]^n$ into itself and is an ℓ_∞ contraction with modulus*

$$L \;=\; 1 - \alpha\Big(1 - \frac{R}{4T}\Big) \;<\; 1.$$

The fixed point is therefore unique and is reached geometrically from any initial value.

Proof. Let $\Phi(\boldsymbol{q}) = (1-\alpha)\boldsymbol{q} + \alpha\,\mathrm{sigmoid}(\boldsymbol{h}(\boldsymbol{q})/T)$. The terms c_i and static_i in h_i do not depend on $\boldsymbol{q}$, so $\partial h_i/\partial q_j = J_{ij}$, and $0 < s'(z) \le 1/4$ for $s(z) = (1+e^{-z})^{-1}$. Therefore

$$\frac{\partial \Phi_i}{\partial q_j} = (1-\alpha)\delta_{ij} + \frac{\alpha}{T}\, s'\big(h_i/T\big)\, J_{ij}, \qquad \sum_j \Big|\frac{\partial \Phi_i}{\partial q_j}\Big| \;\le\; (1-\alpha) + \frac{\alpha}{4T}\sum_j |J_{ij}| \;\le\; L.$$

Since $J_{ii} = 0$, the two terms never occupy the same entry, and since $\alpha \le 1$ gives $1 - \alpha \ge 0$, the absolute value splits cleanly into the two terms. Because $[0, 1]^n$ is convex, the mean value inequality gives $\|\Phi(\boldsymbol{q}) - \Phi(\boldsymbol{q}')\|_\infty \le L\|\boldsymbol{q} - \boldsymbol{q}'\|_\infty$, and $\Phi(\boldsymbol{q})$ is a convex combination of $\boldsymbol{q}$ and a point in $[0, 1]^n$, so it does not leave $[0, 1]^n$. With $\alpha > 0$ and $R < 4T$ we have $L < 1$, so the Banach fixed-point theorem (Granas & Dugundji, 2003) applies. □

Remark 2 (The condition is not new)**.** *The condition $R < 4T$ is the form the Dobrushin uniqueness condition (Dobrushin, 1968) takes for this model, and row-sum conditions of the same family are also used to guarantee convergence of message-passing algorithms (Tatikonda & Jordan, 2002; Mooij & Kappen, 2007). What is specific here is not the condition itself but its use. Because both kernels admit closed-form row-sum bounds, the condition becomes something to verify at design time rather than to monitor at run time.*

Proposition 3 (Variational free energy)**.** *For $Q = \prod_i \mathrm{Bern}(\sigma_i; q_i)$, $T\,\mathrm{KL}(Q\|P)$ equals*

$$F(\boldsymbol{q}) = -\sum_i (c_i + \mathrm{static}_i)\, q_i - \tfrac{1}{2}\boldsymbol{q}^\top J \boldsymbol{q} + T\sum_i \big[q_i \log q_i + (1-q_i)\log(1-q_i)\big] \qquad (17)$$

up to an additive constant, and the stationary points of F on $(0, 1)^n$ are exactly the solutions of Equation (13)*. All such solutions lie in the interior of $(0, 1)^n$.*

Proof. The derivation below is the standard naive mean-field computation (Opper & Saad, 2001; Wainwright & Jordan, 2008); the only part specific to this model is the absorption of the anchor term. Equation (4) is written over candidates only, but E also includes couplings with the already committed positions. Since J is symmetric, the coupling between a candidate i and a committed position j appears twice, as (i, j) and (j, i), which cancels the factor $-\frac{1}{2}$, and its contribution is absorbed into the unary term as $-\sum_i \mathrm{static}_i \sigma_i$. Couplings among the committed positions are constants independent of $\boldsymbol{q}$ and drop out as well. The derivation below uses E with these adjustments.

We have $\mathrm{KL}(Q\|P) = \mathbb{E}_Q[E(\boldsymbol{\sigma})]/T + \mathbb{E}_Q[\log Q] + \log Z$, and since Q factorizes, $\mathbb{E}_Q[\sigma_i] = q_i$ and $\mathbb{E}_Q[\sigma_i\sigma_j] = q_i q_j$ for $i \neq j$. Because $J_{ii} = 0$, $\mathbb{E}_Q[E] = -\sum_i (c_i + \mathrm{static}_i) q_i - \frac{1}{2}\boldsymbol{q}^\top J\boldsymbol{q}$, and multiplying by T produces Equation (17) up to the constant $T \log Z$.

Since J is symmetric with $J_{ii} = 0$, $\partial\big(\frac{1}{2}\boldsymbol{q}^\top J\boldsymbol{q}\big)/\partial q_i = \sum_{j\neq i} J_{ij} q_j$, and therefore $\partial F/\partial q_i = -h_i(\boldsymbol{q}) + T\log\frac{q_i}{1-q_i}$. Setting this to 0 gives $q_i = \mathrm{sigmoid}(h_i(\boldsymbol{q})/T)$, which is Equation (13). For $\alpha > 0$, the damping in Equation (14) does not change this solution set. Since the range of sigmoid is $(0, 1)$, every solution lies in the interior. □

Corollary 1 (Design-time budget)**.** *If $\eta < 1$ and Equation* (15) *holds, then $R < 4T\eta < 4T$, so Proposition 2 applies. Moreover, $\mathrm{KL}(Q\|P)$ is then strictly convex in $\boldsymbol{q}$, so the fixed point of Proposition 3 is the unique global minimum.*

Proof. Both kernels are nonnegative, so $|J_{ij}| \le w_{\mathrm{loc}} \exp\left(-d_{ij}^2/2s^2\right) + w_{\mathrm{sim}}\big(\mathcal{T}_\kappa[S]\big)_{ij}$. Summing over a row and applying Proposition 1 and Lemma 1, R is at most the left-hand side of Equation (15), which by assumption is below $4T\eta$.

Convexity follows from $\mathrm{KL}(Q\|P) = F(\boldsymbol{q})/T + \mathrm{const}$ in Proposition 3. Its Hessian is $-J/T + \mathrm{diag}\big(1/(q_i(1-q_i))\big)$; since $q_i(1-q_i) \le 1/4$, the diagonal entries are at least 4, whereas the off-diagonal absolute row sums are $R/T < 4\eta < 4$. By the Gershgorin circle theorem (Horn & Johnson, 2012), every eigenvalue is therefore at least $4(1-\eta) > 0$. By Proposition 3 the fixed point lies in the interior of $(0,1)^n$, and an interior stationary point of a convex function is a global minimum, so that fixed point is the unique global minimum. □

B Benchmarks and Metrics

For mathematical reasoning, we use the full GSM8K (Cobbe et al., 2021) (5-shot, 1319 problems) and MATH (Hendrycks et al., 2021) `minerva_math` (4-shot, 5000 problems); for code generation, the full HumanEval (Chen et al., 2021) (0-shot, 164 problems) and MBPP (Austin et al., 2021b) (3-shot, 500 problems). We report accuracy, number of function evaluations (NFE), and directly measured tokens per second (TPS). All methods use the same post-processing and scoring pipeline following DAWN (Luo et al., 2026).

C Hyperparameters

RAPD used the configuration below unchanged on all four benchmarks of each backbone, without per-benchmark retuning; the two backbones differ only in τ and $t_{\max}$.

Symbol	LLaDA	Dream	Notes
Unary index k	V	V	
Commitment threshold τ	0.85	0.90	
Block length B	32	32	
Number of candidates k_{cand}	32	32	$= B$
Step cap per block $t_{\max}$	16	32	commit by argmax when reached
Attraction weight w_{loc}	0.22	0.22	
Attraction width s	3.0	3.0	
Repulsion weight w_{sim}	1.08	1.08	
Truncation κ	2	2	rank-based intersection
Decision direction $\boldsymbol{u}_i$	final layer	final layer	top two candidates of the output logits
Temperature T	1	1	
Damping α	0.5	0.5	
Round cap K	4	4	early stop when unchanged for 2 consecutive rounds
Safety factor η	0.9	0.9	limit $4T\eta = 3.600$

In a step where the committed set is empty, the masked position with the largest unary term is committed to guarantee progress.

How the configuration was selected The coupling weights were selected on HumanEval (164 problems), the same set we report. (1) On an attraction ladder $w_{\mathrm{loc}} \in \{0.10, \ldots, 0.30\}$ at $w_{\mathrm{sim}} = 0.5$ and $\kappa = 4$, $w_{\mathrm{loc}} = 0.22$ was chosen as the largest value before accuracy fell. (2) At $w_{\mathrm{loc}} = 0.22$, three ways of concentrating repulsion ($\kappa \in \{1, 2\}$) were compared with the $\kappa = 4$ reference; $\kappa = 2$, $w_{\mathrm{sim}} = 1.08$ was kept. (3) The readout layer was moved from $L-3$ to $L-1$ after the two gave indistinguishable results on HumanEval and MBPP. (4) The unary ($k = V$ over $k = 2$) was chosen on GSM8K and MATH (Table 3). In total about ten coupling arms were compared on HumanEval, so HumanEval accuracy at the chosen point is optimistically biased. GSM8K, MBPP, and MATH were not used to choose the coupling weights and serve as held-out tasks for them: MBPP entered only step (3), which found no difference between the two readout layers, and GSM8K and MATH informed only the unary choice in step (4). The method and configuration were then frozen for all experiments in this paper.

Budget check. Since $\vartheta(s) - 1 < 10^{-8}$ for $s \geq 1$, the left-hand side of Equation (15) is effectively $w_{\mathrm{loc}}(\sqrt{2\pi}\, s - 1) + w_{\mathrm{sim}}\kappa$, which for the configuration above is $0.22 \times 6.5199 + 2 \times 1.08 = 3.5944$, below the limit $4T\eta = 3.600$. The -1 removes the self-coupling term and is not decoration: the looser bound obtained by integral approximation loses this term, gives 3.8144, and breaks the budget (Remark 1). The run-time check that rescales the coupling cannot fire under this configuration, and the measured firing rate was 0 in every run reported. This inequality determines w_{sim} for the chosen w_{loc} and κ (see above). We fix $T = 1$ so that, without coupling, $\mathbf{q}^{(0)}$ equals the model confidence (Lemma 2) and τ keeps the meaning of the Fast-dLLM threshold.

Baselines. All baselines used their publicly released default settings without per-benchmark retuning. Sequential decoding ("LLaDA") commits one token per step and uses 256 steps. Fast-dLLM (Wu et al., 2025) commits every position above the confidence threshold 0.9, uses low-confidence remasking, and imposes no step cap per block. LocalLeap (Kong et al., 2025) uses an anchor threshold of 0.9, a relaxed threshold of 0.75, and a radius of 4 (the released LLaDA defaults), also with no step cap. DAWN (Luo et al., 2026) follows the dependency-graph configuration of its public implementation.

D ALGORITHM

Algorithm 1 gives one block of RAPD decoding.

Algorithm 1 RAPD: decoding one block

Require: canvas x, block $\mathcal{B}$, model, unembedding W_U
Require: $\tau,\ t_{\max},\ k_{\text{cand}},\ K,\ \alpha,\ T,\ w_{\mathrm{loc}},\ s,\ w_{\mathrm{sim}},\ \kappa,\ \eta$

Leave all positions in $\mathcal{B}$ masked
for $t = 1, \ldots, t_{\max}$ **do**
  $\pi \leftarrow \mathrm{softmax}\big(\mathrm{model}(x)|_{\mathcal{B}}\big)$ ▷ one forward pass; NFE += 1
  Unary term $c_i \leftarrow$ Eq. (5), $\boldsymbol{u}_i \leftarrow \mathrm{normalize}\big(W_U[v_1(i)] - W_U[v_2(i)]\big)$
  $\mathcal{C} \leftarrow$ top-k_{cand} masked positions by c_i, $\mathcal{A} \leftarrow$ the remaining committed positions
  $J \leftarrow$ Eq. (6) on $\mathcal{C} \cup \mathcal{A}$ ▷ $J_{ii} = 0$
  $\mathrm{static}_i \leftarrow \sum_{j \in \mathcal{A}} J_{ij}$, $J \leftarrow J|_{\mathcal{C}\times\mathcal{C}}$
  if $R := \max_i \sum_j |J_{ij}| \geq 4T\eta$ **then** $J \leftarrow (4T\eta/R)\, J$ ▷ safeguard
  $\boldsymbol{q} \leftarrow \mathrm{sigmoid}\big((\boldsymbol{c} + \mathrm{static})/T\big)$
  for $r = 1, \ldots, K$ **do**
    $\boldsymbol{h} \leftarrow \boldsymbol{c} + \mathrm{static} + J\boldsymbol{q}$
    $\boldsymbol{q} \leftarrow (1 - \alpha)\, \boldsymbol{q} + \alpha\, \mathrm{sigmoid}(\boldsymbol{h}/T)$
    if $\{\, i : q_i > \tau \,\}$ unchanged for two consecutive rounds **then break**
  end for
  $\mathcal{S} \leftarrow \{\, i \in \mathcal{C} : q_i > \tau \,\}$
  if $\mathcal{S} = \emptyset$ **then**
    $\mathcal{S} \leftarrow \{\, \arg\max_i c_i \,\}$ ▷ guarantee progress
  end if
  Commit each position in $\mathcal{S}$ to $\arg\max \pi_i$ and write it into x
  if no masked positions remain **then break**
end for
Commit all remaining masked positions to the final $\arg\max \pi_i$

E EXISTING COMMIT RULES AS SPECIAL CASES OF THE ENERGY

This section specifies which existing decoders are instances of the energy family of §3 and under which conditions. One decoding step of the family (Algorithm 1) is the map $\mathcal{S} =$

$\mathrm{Commit}(\pi;\, k, b, J, T, \alpha, K, \tau)$ given by

$$\begin{aligned} \mathbf{q}^{(0)} &= \mathrm{sigmoid}\big((\mathbf{c}+\mathbf{static})/T\big), \qquad c_i = c_i^{(k)} - b, \qquad \mathrm{static}_i = \textstyle\sum_{j\in\mathcal{A}} J_{ij}, \\ \mathbf{q}^{(r+1)} &= (1-\alpha)\,\mathbf{q}^{(r)} + \alpha\,\mathrm{sigmoid}\big((\mathbf{c}+\mathbf{static}+J\mathbf{q}^{(r)})/T\big), \qquad r = 0,\dots,K-1, \\ \mathcal{S} &= \{i \in \mathcal{C} : q_i^{(K)} > \tau\}, \quad \text{or } \{\arg\max_i c_i\} \text{ if this set is empty,} \end{aligned} \tag{18}$$

with $b \geq 0$, $J = J^\top$, and $J_{ii} = 0$. The main configuration is $(k, b, T, K, \alpha) = (V, 0, 1, 4, 0.5)$ with J from (6). We write $p_i := p_{(1)}(i)$ and assume $p_i \notin \{\tau, \tau_a, \tau_r\}$; the rules below differ only on such ties.

Lemma 2 (Uncoupled marginals). *If $J \equiv 0$, then for every $r \geq 0$ and $\alpha \in (0, 1]$,*

$$\mathbf{q}^{(r)} = \mathrm{sigmoid}(\mathbf{c}/T), \qquad \textit{and} \qquad q_i^{(r)} = p_i \;\; \textit{if } (k, b, T) = (V, 0, 1).$$

Proof. With $J \equiv 0$, $\mathbf{static} \equiv 0$, and the update maps $\mathbf{q}^{(0)} = \mathrm{sigmoid}(\mathbf{c}/T)$ to $(1-\alpha)\,\mathbf{q}^{(0)} + \alpha\,\mathrm{sigmoid}(\mathbf{c}/T) = \mathbf{q}^{(0)}$. For $k = V$, $\sum_{r\geq 2} p_{(r)}(i) = 1 - p_i$, so $c_i^{(V)} = \mathrm{logit}\, p_i$ and $\mathrm{sigmoid}(\mathrm{logit}\, p_i) = p_i$. □

Proposition 4 (Confidence thresholding: Fast-dLLM). *Let $(k, b, T, J) = (V, 0, 1, 0)$, $k_{\mathrm{cand}} = |\mathcal{U}|$, and $t_{\max} \geq B$. Then, at every step, the commit set equals that of Fast-dLLM with threshold τ,*

$$\mathcal{S} = \mathcal{S}_{\mathrm{F}} := \{i : p_i \geq \tau\} \cup \{\arg\max_i p_i\},$$

so, with the same model, prompt, and greedy tokens, the two decoders produce identical sequences with identical NFE.

Proof. By Lemma 2 and the no-tie assumption,

$$\{i : q_i > \tau\} = \{i : p_i > \tau\} = \{i : p_i \geq \tau\}, \qquad \arg\max_i c_i = \arg\max_i \mathrm{logit}\, p_i = \arg\max_i p_i.$$

A non-empty threshold set contains $\arg\max_i p_i$, and an empty one triggers the fallback; in both cases $\mathcal{S} = \mathcal{S}_{\mathrm{F}}$. Since $|\mathcal{S}| \geq 1$, a block finishes within $B \leq t_{\max}$ steps, so the end-of-block commit never fires, and induction over steps and blocks gives identical sequences and NFE. □

Proposition 5 (Margin thresholding). *If $(k, b, T, J) = (2, 0, 1, 0)$, then, with the same fallback,*

$$q_i = \mathrm{sigmoid}\Big(\log \frac{p_{(1)}(i)}{p_{(2)}(i)}\Big) = \frac{p_{(1)}(i)}{p_{(1)}(i) + p_{(2)}(i)}, \qquad \mathcal{S} = \Big\{i : \frac{p_{(1)}(i)}{p_{(2)}(i)} > \frac{\tau}{1-\tau}\Big\}.$$

Proof. Apply Lemma 2 with $c_i^{(2)} = \log\big(p_{(1)}(i)/p_{(2)}(i)\big)$; then $q_i > \tau \iff (1-\tau)\,p_{(1)}(i) > \tau\,p_{(2)}(i)$. □

Proposition 6 (Fixed-count schedules: low-confidence remasking). *Let $J \equiv 0$, let the threshold τ_t depend on the step, and let the c_i be distinct. For a schedule n_t, with $q_{[m]}$ the m-th largest q_i,*

$$\tau_t \in \big[q_{[n_t+1]},\, q_{[n_t]}\big) \implies \mathcal{S} = \mathrm{Top}_{n_t}(\mathbf{c}),$$

the set of the n_t positions with the largest c_i; for $k = V$, this is LLaDA's low-confidence remasking (Nie et al., 2025).

Proof. By Lemma 2, $q_i = \mathrm{sigmoid}(c_i/T)$ is strictly increasing in c_i; for $k = V$, $c_i = \mathrm{logit}\, p_i$ orders positions as p_i does. □

Proposition 7 (Anchor-relaxed thresholding: LocalLeap). *LocalLeap (Kong et al., 2025) with anchor threshold τ_a, relaxed threshold $\tau_r < \tau_a$, and radius r commits*

$$\mathcal{S}_{\mathrm{L}} = \{i : p_i \geq \tau_a\} \cup \{i \in N : p_i \geq \tau_r\} \cup \{\arg\max_i p_i\}, \qquad N = \bigcup_{i:\, p_i \geq \tau_a} \{j : |i - j| \leq r\}.$$

Take the family with $K = \alpha = 1$, $T \to 0$, $\mathbf{static} \equiv 0$ (LocalLeap's anchors are those of the current step), and

$$k = V, \qquad b = \operatorname{logit} \tau_a, \qquad J_{ij} = \Delta\, \mathbb{I}[0 < |i-j| \le r], \qquad \Delta = \operatorname{logit} \tau_a - \operatorname{logit} \tau_r > 0.$$

Let $m_j = \big|\{i : p_i > \tau_a,\ 0 < |i-j| \le r\}\big|$ count the anchors near a non-anchor j. Then $\mathcal{S}_{\mathrm{L}} \subseteq \mathcal{S}$ at every step, and $\mathcal{S} = \mathcal{S}_{\mathrm{L}}$ if no non-anchor j satisfies

$$m_j \ge 2 \qquad \textit{and} \qquad \operatorname{logit} \tau_a - m_j \Delta \le \operatorname{logit} p_j < \operatorname{logit} \tau_r,$$

in particular whenever $\max_j m_j \le 1$.

Proof. As $T \to 0$, $\operatorname{sigmoid}(x/T) \to \mathbb{I}[x > 0]$ for $x \neq 0$, so $q_i > \tau \iff h_i > 0$ for any $\tau \in (0,1)$. With $\mathbf{static} \equiv 0$ and $K = \alpha = 1$,

$$q_i^{(0)} = \mathbb{I}[p_i > \tau_a], \qquad h_i \ge \operatorname{logit} p_i - \operatorname{logit} \tau_a > 0 \ \ (\text{anchor } i),$$
$$h_j = \operatorname{logit} p_j - \operatorname{logit} \tau_a + m_j \Delta \ \ (\text{non-anchor } j).$$

Every anchor is therefore committed, and for a non-anchor j,

$$j \in \mathcal{S} \iff \begin{cases} p_j > \tau_a, & m_j = 0, \\ p_j > \tau_r, & m_j = 1, \\ \operatorname{logit} p_j > \operatorname{logit} \tau_a - m_j \Delta, & m_j \ge 2, \end{cases} \qquad j \in \mathcal{S}_{\mathrm{L}} \iff m_j \ge 1 \text{ and } p_j > \tau_r.$$

The first case never holds for a non-anchor, the second is LocalLeap's relaxed test, and in the third $\operatorname{logit} \tau_a - m_j \Delta < \operatorname{logit} \tau_r$, so the two sets differ only on the stated interval. Without anchors, both commit $\arg\max_i p_i$ (the family through its fallback). □

F Additional Related Works

A complementary line of work reduces the cost of a single step rather than the number of decoding steps. These methods shrink what each forward pass computes: they evict low-importance cache entries from both the prefix and the suffix (Song et al., 2026), skip low-importance token positions in the early layers (Zhu et al., 2026), compute only part of the attention matrix (Wang et al., 2026), or drop far-future mask positions from the input (Chen et al., 2026a; Xiao et al., 2026). Block diffusion models decode blocks autoregressively so that KV caching becomes possible (Arriola et al., 2025; Wu et al., 2026), while Eso-LM (Sahoo et al., 2026) uses causal attention over the whole sequence for the same purpose. Other work trains the model so that more positions reach high confidence together (Chen et al., 2026b), or learns the unmasking policy directly (Jazbec et al., 2026). These approaches are complementary to RAPD, which is training-free and modifies only the commit-set selection after each model forward pass.

G Sign balance and single-weight curves

On both tasks, accuracy stays flat up to a knee in ρ and then falls. On HumanEval, where $\rho = .25$, .3, and .35 were also run (Figure 2), accuracy at $\tau = .80$ is significantly higher than at the main configuration's share ($\rho = .4$) for every $\rho \le .3$ and still higher, though not significantly, at $\rho = .35$ (36.0%); the knee therefore narrows from $[.2, .4]$ on the grid of Table 4 to $[.35, .4]$. MBPP has no setting between $\rho = .2$ and $\rho = .4$, so at the grid resolution its knee for $\tau = .80$ lies in $[.2, .4]$; at this threshold, $\rho = .4$ is significantly less accurate than $\rho = 0$ on both tasks (HumanEval -7.3, MBPP -3.8 points). At the other three thresholds, the additional HumanEval settings leave every knee unchanged: the knee stays at $\rho = .4$ for $\tau = .85$, at .4/.6 (HumanEval/MBPP) for $\tau = .90$, and at .6/.8 for $\tau = .95$, and no cell at $\rho = .25$, .3, or .35 differs significantly from the cell at $\rho = .4$. On MBPP, the grid does not close the gap to LocalLeap: in the NFE range of the LocalLeap points (51.8–57.3), no setting of the 24-setting grid lies above the segment joining the $\tau = .85$ and $\tau = .90$ settings of the τ sweep.

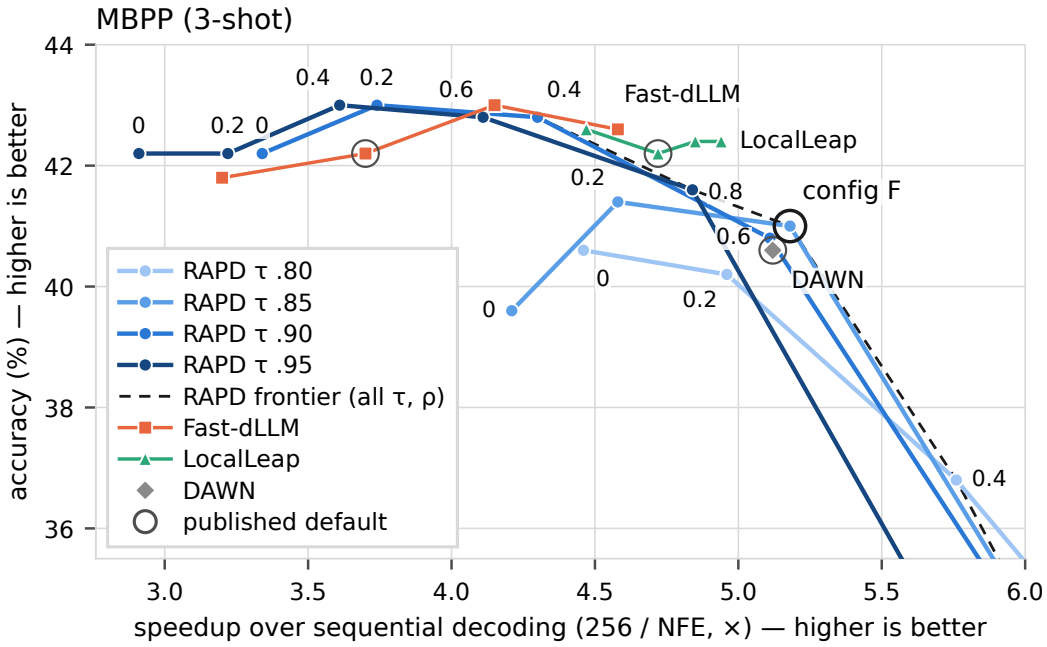


Figure 4: The counterpart of Figure 2 on MBPP. Point labels give the attractive share ρ ($\rho = 0.4$ is config F's split); settings below 35.5% lie outside the plot (all values in Table 4).

Table 4: Fixed-budget sign balance: NFE / accuracy at attractive share ρ ($6.52\,w_{\text{loc}} + 2\,w_{\text{sim}} = 3.594$; $\rho = .4$ is the main configuration's split). HumanEval was also run at $\rho = .25, .3, .35$ (Figure 2; NFE / accuracy at $\tau = .80, .85, .90, .95$: $\rho = .25$: 56.4/37.80, 61.1/40.24, 74.2/40.85, 88.2/39.63; $\rho = .3$: 55.0/38.41, 60.5/39.63, 71.0/39.02, 86.4/39.02; $\rho = .35$: 52.0/35.98, 58.1/38.41, 69.2/39.63, 83.4/39.02); MBPP was not. The safeguard (Algorithm 1) never rescales J in any cell.

	τ	$\rho = 0$	0.2	0.4	0.6	0.8	1.0
HumanEval	.80	66.4 / 39.02	58.8 / 38.41	50.7 / 31.71	44.4 / 22.56	37.9 / 13.41	34.6 / 6.10
	.85	69.6 / 35.37	64.2 / 39.63	56.3 / 40.24	47.7 / 25.00	41.0 / 15.85	37.6 / 10.98
	.90	87.2 / 40.24	76.7 / 40.85	67.0 / 40.24	56.2 / 33.54	47.8 / 26.83	42.7 / 10.98
	.95	101.2 / 39.63	91.2 / 40.24	80.6 / 40.24	69.8 / 39.63	60.5 / 34.76	50.1 / 29.88
MBPP	.80	57.3 / 40.60	51.6 / 40.20	44.4 / 36.80	37.0 / 30.20	31.2 / 21.20	25.9 / 12.60
	.85	60.8 / 39.60	55.9 / 41.40	49.4 / 41.00	42.4 / 34.40	34.6 / 25.00	28.8 / 14.40
	.90	76.6 / 42.20	68.5 / 43.00	59.5 / 42.80	50.1 / 40.80	41.2 / 32.80	33.9 / 20.60
	.95	87.9 / 42.20	79.5 / 42.20	71.0 / 43.00	62.3 / 42.80	52.9 / 41.60	44.0 / 33.40

Controlled sign ablation. All arms use the main configuration ($\tau = 0.85$, $t_{\max} = 16$) and differ only in $(w_{\text{loc}}, w_{\text{sim}})$ (NFE / accuracy).

Task	No coupling (0, 0)	Attraction only (0.22, 0)	Repulsion only (0, 1.08)	RAPD (0.22, 1.08)
GSM8K	67.7 / 75.82	49.4 / 77.41	73.6 / 74.75	55.9 / 77.79
HumanEval	64.4 / 35.98	52.0 / 33.54	68.4 / 36.59	56.3 / 40.24
MBPP	56.5 / 40.20	46.3 / 38.20	59.2 / 40.00	49.4 / 41.00

At the same threshold and step cap, the signed coupling uses 13–17% fewer forward passes than no coupling and is more accurate on GSM8K (+1.97) and HumanEval (+4.27). Repulsion recovers the accuracy that attraction alone loses on HumanEval (+6.71) and MBPP (+2.80), but not on GSM8K (+0.38).

Single-weight curves. Raising one weight from 0 to its main value while the other stays at its main value (every setting stays within the coupling budget) gives, on HumanEval, NFE 68.4 → 56.3

and accuracy $36.6 \to 40.2$ for $w_{\mathrm{loc}} : 0 \to 0.22$, and NFE $52.0 \to 56.3$ and accuracy $33.5 \to 40.2$ for $w_{\mathrm{sim}} : 0 \to 1.08$; on MBPP, it gives NFE $59.2 \to 49.4$ with flat accuracy (40.0–41.0) for w_{loc}, and NFE $46.3 \to 49.4$ and accuracy $38.2 \to 41.0$ for w_{sim}. On both tasks, attraction lowers NFE, and repulsion raises accuracy while increasing NFE by 3–4; on HumanEval, attraction also raises accuracy.

Per-block step cap. At $\tau = .85$, raising $t_{\max}$ from 16 to 32 changes one of the 164 HumanEval outcomes (39.63% at 58.0 NFE, against 40.24% at 56.3 NFE) and raises MBPP accuracy from 41.00% at 49.4 NFE to 42.00% at 51.5 NFE (5 discordant problems, all in favor of $t_{\max} = 32$); neither difference is significant.

H BLOCK SIZE AND GENERATION LENGTH

Table 5: NFE / accuracy at other block sizes and generation lengths (LLaDA-8B-Instruct, published baseline settings, RAPD main configuration with $t_{\max} = 32$ at block 64 and 64 at block 128). Block 32 rows are from Table 1.

Task	Shape	RAPD	Fast-dLLM	LocalLeap	DAWN
HumanEval	block 32	56.3 / 40.24	78.0 / 40.24	61.0 / 39.63	64.1 / 37.80
	block 64	54.2 / 37.80	77.2 / 40.85	59.5 / 36.59	60.3 / 34.76
	block 128	52.7 / 37.20	74.8 / 40.24	58.1 / 35.98	55.7 / 35.37
	length 512	117.2 / 40.24	164.9 / 45.12	124.3 / 45.73	130.0 / 43.90
GSM8K	block 32	55.9 / 77.79	78.5 / 78.39	58.9 / 78.39	56.0 / 77.18
	block 64	51.7 / 78.24	75.8 / 77.79	56.3 / 77.41	51.6 / 77.03
	block 128	51.2 / 72.48	78.7 / 73.31	57.3 / 73.24	51.1 / 71.72
	length 512	70.0 / 77.86	95.1 / 78.17	72.8 / 77.79	70.4 / 76.50

RAPD uses significantly fewer NFE than Fast-dLLM and LocalLeap in every shape on both tasks and than DAWN on HumanEval; on GSM8K, its NFE difference from DAWN is not significant in any shape. On GSM8K, it is non-inferior to every baseline in every shape, and all four methods lose 5.1–5.5 points at block 128 relative to block 32. On HumanEval, it is 3.05 points less accurate than Fast-dLLM at blocks 64 and 128 and less accurate than all three baselines at length 512 (significantly only against LocalLeap, by 5.49 points); at length 512, every baseline gains 4.9–6.1 points over length 256, whereas RAPD gains none. The safeguard (Algorithm 1) never rescales the coupling in any RAPD run of Table 5.