LEARNING WHEN TO COMMIT: TOKEN SCORES, STOPPING RULES, AND STABILITY IN DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Parallel diffusion decoding trades task accuracy for fewer backbone evaluations, and the accuracy retained depends on both token scoring and the rule that converts scores into commitments. We introduce a $5{,}121$-parameter predictor of token stability, defined as agreement with a one-token-per-step decoding reference. Confidence determines candidate order, while a cumulative product of predicted stability scores controls how many tokens are committed, without additional backbone evaluations. Heads trained on at most 50 problems are reused across tasks without retraining, with low measured inference overhead. Across three backbones and three benchmarks, the decoder has lower relative accuracy loss than its paired comparator in 71% of the measured comparisons at approximately matched mean cost. Two observations clarify what the learned score contributes. First, controlled comparisons show that a token score's observed accuracy advantage can depend on the stopping rule and decoding budget. Second, local stability and task accuracy differ. At approximately matched mean commitment counts, learned stability reduces reference disagreement relative to raw confidence and margin, but not significantly relative to a learned confidence-only control. Validation-selected evaluations and single-decision interventions do not establish consistent gains in final task accuracy. Our contribution combines lightweight, transferable stability-guided decoding with a controlled analysis of how token scores and stopping rules jointly shape the accuracy–cost trade-off.

1 INTRODUCTION

Masked diffusion language models (DLLMs) generate text by iteratively refining a partially masked sequence (Shi et al., 2024; Sahoo et al., 2024; Gong et al., 2022; Li et al., 2022), rather than decoding strictly from left to right (Yang et al., 2025; Touvron et al., 2023). At each denoising step, a DLLM predicts token distributions at all masked positions in parallel, allowing a decoder to commit multiple predictions from one backbone evaluation. Committing only one prediction per step leaves much of this parallelism unused. Accelerated decoding (Sun et al., 2026b; Kong et al., 2025) therefore seeks to identify predictions that can be committed together while preserving task accuracy (Wu & Zhang, 2025; Luo et al., 2026; Ayoub et al., 2026).

Existing acceleration methods reduce the number of backbone evaluations (Zhang et al., 2025) by committing multiple predictions at each denoising step. They decide which predictions to commit using output confidence or margin (Wu et al., 2026; Zoabi et al., 2026), attention-derived signals (Sahin et al., 2026; Zhang et al., 2026b), local sequence consistency (Hersche et al., 2026), or policies learned from decoding trajectories (Jazbec et al., 2025; Bao et al., 2026). One central challenge is therefore to commit enough predictions per step to reduce the number of backbone evaluations (NFE) (Zhang et al., 2026a; Peng et al., 2025), which largely determines decoding cost, while retaining task accuracy (Chang et al., 2026; Qian et al., 2026).

To address this challenge, we introduce a lightweight predictor of *token stability*: whether a current proposal agrees with the token produced by a fixed one-token-per-step reference. Related learned decoders, including Learn2PD and TraceLock (Bao et al., 2026; Sun et al., 2026a), use future decoding outcomes to train token filters or commitment decisions. We instead use predicted stability as a

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.

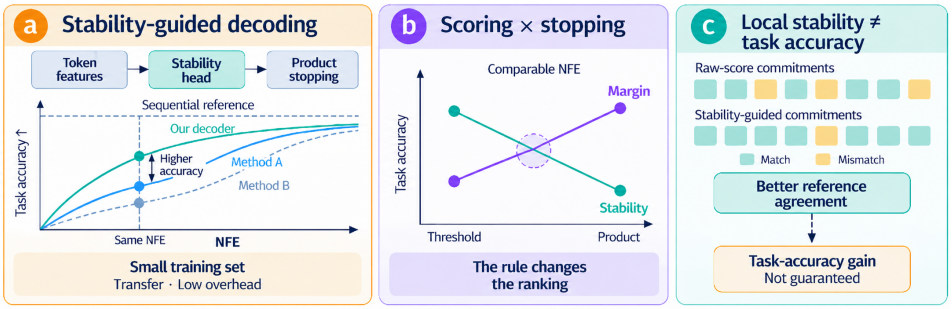

Figure 1: **Problem, method, and findings.** (a) Our lightweight stability predictor guides how many confidence-ordered tokens to commit, reducing backbone evaluations while preserving task accuracy and transferring across tasks without retraining. (b) Our comparisons show that changing the stopping rule can reverse the accuracy ranking of token scores. (c) At approximately matched mean commitment counts across states, learned stability selects more reference-consistent tokens, but this local improvement does not consistently translate into higher task accuracy. Curves are illustrative.

modular token score, separate from the stopping rule. Our predictor uses scalar output-distribution statistics and decoding context. During decoding, confidence orders the candidate positions; after the highest-confidence proposal is committed, the cumulative product of the remaining stability scores determines how far the committed prefix extends. The predictor uses information already produced by the backbone, adds no backbone evaluations, and can be reused across tasks on the same backbone without retraining.

In addition, we study how the accuracy–cost trade-off depends on two components of a commitment policy: **(i) a token score**, which assesses each proposed token, and **(ii) a stopping rule**, which converts the scores into the number of tokens committed. This decomposition lets us compare stability, token confidence, and margin—the probability gap between the most likely and second-most likely tokens—under a common stopping rule, and compare stopping rules while holding the score fixed, at approximately matched decoding costs.

Problem statement. Parallel decoding creates an accuracy–cost trade-off: committing more tokens per step reduces backbone evaluations (NFE) but can lower task accuracy. For a fixed 256-token continuation, let A_{seq} denote the accuracy of the one-token-per-step reference, and let A_π and N_π denote the accuracy and mean NFE of a parallel policy π. We measure its relative accuracy loss as: $\ell_\pi = (A_{\text{seq}} - A_\pi)/A_{\text{seq}}$, where a lower value indicates better accuracy retention at a given decoding cost. **Our goal** is therefore to retain as much task accuracy as possible for a given NFE budget. Figure 1 provides an overview of our method and main findings.

Our contributions can be summarized as follows:

1. **Lightweight, transferable stability-guided decoding.** We introduce a 5,121-parameter stability predictor that controls parallel commitments without additional backbone evaluations. Across three backbones and three benchmarks, our decoder achieves lower relative accuracy loss than the paired comparator in 71% of measured comparisons at approximately matched mean NFE. Heads trained on at most 50 problems are reused on unseen tasks without retraining; we further demonstrate robustness to training-sample selection and training-set size, with low inference overhead.
2. **Interactions between scoring and stopping.** Controlled comparisons show that the effect of a token score depends on the stopping rule and operating point. Across 13 task–backbone–budget settings, two MBPP interactions remain significant. A separate evaluation gives a smaller interaction estimate in the same direction, with a confidence interval that includes zero.
3. **A gap between local stability and task accuracy.** When compared at the same average token-commitment budget, our stability head selects more tokens that match the sequential reference than raw confidence or margin in every tested setting. Changing one commit-

ment decision often changes the final output, but does not consistently change final-answer accuracy. This shows that local stability and task accuracy are distinct objectives.

2 Related work

Diffusion language models. Autoregressive language models (Touvron et al., 2023; Yang et al., 2025; Zhu et al., 2025) generate one token at a time, conditioning each new prediction on the prefix already produced. Masked diffusion language models (Nie et al., 2025; Bie et al., 2026) instead begin with a partially or fully masked continuation and iteratively predict the masked positions. Because a single backbone evaluation produces distributions for multiple positions, several tokens can potentially be unmasked in parallel. This parallelism can reduce the number of backbone evaluations required for generation (Kong et al., 2025), but committing an incorrect token can alter the context used for later predictions (Ringel et al., 2026). DLLM decoding must therefore balance computational savings against the accuracy lost through aggressive parallel commitments (Sun et al., 2026b; Israel et al., 2025).

Parallel decoding methods. Existing DLLM decoders determine which proposals to commit using confidence thresholds (Wu et al., 2026), probability margins (Zoabi et al., 2026), entropy (Ben-Hamu et al., 2025) or divergence (Kim et al., 2025), dependency (Luo et al., 2026) and attention-derived signals (Sahin et al., 2026), or local sequence consistency (Hersche et al., 2026). These approaches are largely training-free and construct commitment decisions directly from signals available during inference. Learned approaches instead train token filters or commitment policies from decoding trajectories (Jazbec et al., 2025). Their supervision can indicate whether a current proposal agrees with a later output (Agrawal et al., 2025), remains unchanged during subsequent denoising, or should be included in the current commitment. This allows a learned decoder to combine several distributional and contextual signals rather than relying on a single hand-designed statistic.

Stability prediction and commitment. Learn2PD (Bao et al., 2026) predicts agreement between current proposals and completed outputs, while TraceLock (Sun et al., 2026a) learns token-commitment decisions from future-stability labels. Our target is related to these forms of future-outcome supervision: we predict whether a current proposal agrees with the token produced by a fixed one-token-per-step reference. However, we treat predicted stability as a modular token score rather than a complete commitment policy. Confidence determines the candidate order, while a separately specified stopping rule determines how many candidates are committed. This separation allows the stability predictor to be evaluated as part of the complete decoder and also compared with raw confidence and margin under the same stopping rule.

Stopping rules and decoding budgets. A token score does not by itself specify a decoder (Bertolani et al., 2026). Existing methods convert scores into commitments through independent thresholds (Kohut et al., 2026), adaptive commitment counts or block sizes (Lu et al., 2026; Xia et al., 2026), dependency-aware selection of compatible tokens or token clusters (Azangulov et al., 2026; Qi et al., 2026), and cumulative criteria. Cumulative confidence products have previously been applied in positional order (Liu et al., 2025); we instead apply the product to confidence-ordered candidates. We do not claim this rule as new. Our contribution is to pair it with learned stability, margin, and raw confidence and compare the resulting policies at approximately matched decoding costs.

3 Token stability for budgeted unmasking

Target and features. We predict whether a token proposed at the current decoding step agrees with the token ultimately produced at the same position by a one-token-per-step reference decode.

Given a prompt x and a partially filled continuation z_t, the frozen backbone produces a distribution $P_t^{(i)}(v)$ over the vocabulary at each masked position i. The proposed token and its confidence are $\hat{y}_{t,i} = \arg\max_v P_t^{(i)}(v)$ and $c_{t,i} = \max_v P_t^{(i)}(v)$. After each backbone evaluation, the reference decoder commits only the highest-confidence proposal in the active block, eventually producing the continuation $y^\star$. We define the stability target as; $s_{t,i} = \mathbf{1}[\hat{y}_{t,i} = y_i^\star]$. Stability therefore measures agreement with the model's own sequential reference, not task correctness.

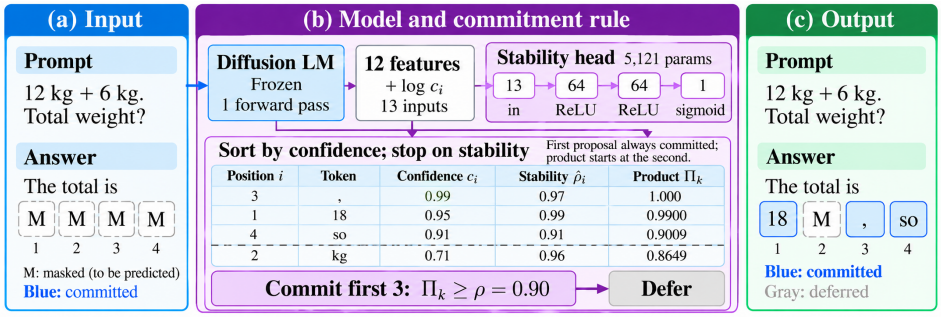

Figure 2: **Stability-guided unmasking.** A 5,121-parameter head estimates proposal stability from the frozen backbone's output statistics and decoding context. Candidates are ordered by confidence. The first is always committed; subsequent candidates are accepted while the cumulative product of their stability scores remains at least ρ. The first candidate is excluded from this product. No additional backbone evaluations are required.

The predictor receives 13 scalar features. Eight summarize the **output distribution**: the three largest token probabilities, the gaps between the largest and the next two probabilities, the log-ratio of the two largest probabilities, entropy, and log-confidence. Five describe the **decoding state**: the replay offset, the fraction and number of masked positions remaining, the position within the active block, and the block index. The replay offset is used during reference-trajectory collection and is set to zero during parallel decoding. An MLP with two hidden layers of 64 units each maps these features to a sigmoid output $\hat{p}_{t,i}$, which estimates the stability of the proposal at position i, as shown in Figure 2. The predictor has 5,121 parameters, operates on statistics already produced by the frozen backbone, and requires no additional backbone evaluations.

Training. We collect labels while generating the sequential reference $y^\star$. A base backbone pass records the proposals and features of all masked positions in the active block and commits its highest-confidence proposal. Each subsequent probe pass identifies the next position and token that the sequential decoder would commit. The proposal stored for that position by the base pass is labelled stable if it matches the probe's token and unstable otherwise. If stable, we commit the probe's token and continue probing. At the first unstable label, we retain the label, leave that position masked, and begin a new base pass from the current state. This procedure generates the reference and labels together, without a separate reference decode. Given N labelled proposals, we train the predictor by minimizing binary cross-entropy: $\mathcal{J}(\psi) = -\frac{1}{N}\sum_{r=1}^{N}\left[s_r \log \hat{p}_r + (1-s_r)\log(1-\hat{p}_r)\right]$.

We use Adam with learning rate 3×10^{-3} for 15 full-batch epochs. Fitting the predictor takes less than one minute on CPU; collecting the labels is a separate, one-time offline cost. Appendix A.2 describes the collection procedure in detail.

Decoding. At each step, the backbone produces a token proposal for every masked position in the active block. We order these positions by decreasing proposal confidence, denoting them by $(1), \ldots, (m_t)$, and let $\hat{p}_{t,(j)}$ be the predicted stability of the proposal at position (j). The highest-confidence proposal (1) is always committed, ensuring that every step makes progress. Beginning with the second proposal, we multiply the stability scores in confidence order and extend the committed prefix while the running product remains at least ρ:

$$k_t = \max\left\{k \in \{1,\ldots,m_t\} : \prod_{j=2}^{k} \hat{p}_{t,(j)} \geq \rho\right\}, \qquad \mathcal{C}_t = \{(1),\ldots,(k_t)\}. \tag{1}$$

The empty product for $k = 1$ equals one. The selected proposals are then written simultaneously at their original positions. Thus, confidence determines which positions are considered first, while predicted stability determines how far the committed prefix extends. A lower ρ permits longer commitments and typically fewer backbone evaluations; a higher ρ is more conservative.

We use the cumulative product as an acceptance heuristic, rather than as a calibrated probability that every committed proposal agrees with the reference. The criterion is related to that of sequential

diffusion language models, which multiply raw confidences in positional order and include the first proposal in the product. Our decoder instead multiplies predicted stability scores in confidence order after the always-committed first proposal. For more details about product transformation see Section D.4.

Score and stopping rule. A per-token threshold rule accepts proposals whose individual scores exceed a cutoff. In contrast, the product rule accumulates scores along the confidence-ordered prefix, so its decisions depend on their numerical magnitudes. The same product cutoff can therefore produce different decoding costs for different scores. We compare raw confidence, margin (the gap between the two largest token probabilities), and learned stability under a common product rule. We also compare threshold and product stopping while holding the score fixed, at approximately matched achieved costs. Appendix A provides the complete feature definitions, training provenance, and implementation details.

4 Experimental setup

We evaluate three backbones: LLaDA-1.5-Instruct and LLaDA-8B-Instruct (Zhu et al., 2026; Nie et al., 2025), and Dream-v0-Instruct-7B (Ye et al., 2025). Each backbone is evaluated on GSM8K, MBPP, and HumanEval (Cobbe et al., 2021; Austin et al., 2021; Chen et al., 2021) benchmarks. Prompts use 5-shot GSM8K with exemplars sampled from a predefined pool, 3-shot MBPP with fixed demonstrations, and 0-shot HumanEval. These few-shot demonstrations are solved examples provided in the prompt, rather than samples used to train the stability head. A chat template is applied only to LLaDA-8B-Instruct. Generation is greedy, with a continuation length of 256 positions and block size $B = 32$, and decoding cost is measured by mean NFE.

The LLaDA experiments use a head trained with LLaDA-1.5 on 32 HumanEval problems, denoted H_L, and reuse it unchanged across tasks and both LLaDA backbones. Unless stated otherwise, the Dream experiments use a head trained with Dream-7B on 50 GSM8K problems, denoted H_D, and reuse it unchanged across tasks. We compare our decoder with tuned confidence thresholding (Wu et al., 2026), DAWN (Luo et al., 2026), ADAS (Sahin et al., 2026), DAPD (Kim et al., 2026), Learn2PD (Bao et al., 2026), and LocalLeap (Kong et al., 2025).

We use three complementary evaluation protocols. First, exploratory sweeps compare complete decoders and different score–rule combinations. Second, cross-fitted diagnostics compare token scores on the same recorded decoding states and at approximately matched commitment volumes. Third, validation-selected evaluations choose each setting using NFE on separate calibration problems, without inspecting accuracy on the evaluation problems. For approximately matched-cost comparisons, we pair measured operating points whose mean NFE differs by at most 3.5. This window identifies nearby measured configurations but does not imply equal computational cost, especially at low NFE, so both achieved costs are reported. Accuracy is compared on identical problems using two-sided exact McNemar tests. Each run uses one NVIDIA A100-SXM4-40GB GPU. Appendix A provides generation, scoring, training, and comparator details; Appendix C.1 describes matching and statistical procedures.

5 Decoder performance, transfer, and robustness

Accuracy loss at comparable decoding cost. Figure 3 asks a direct question: at approximately the same decoding cost, which policy retains more task accuracy? Each measured comparator configuration is paired with the nearest configuration of our decoder on the same task and backbone when their mean NFE differs by at most 3.5. Points below the diagonal indicate that our decoder achieves higher accuracy at comparable cost. Each point compares the deployed complete decoder with an evaluated method: H_L is used on both LLaDA backbones and H_D on Dream-7B.

Across the nine task–backbone settings, our decoder has lower relative accuracy loss in 117 of 164 matched pairs (71%), equal loss in 4, and higher loss in 43. The advantage is strongest at aggressive decoding budgets. For pairs with mean NFE below 30, our decoder has lower loss in 31 of 35 comparisons (89%); between 30 and 40 NFE, it has lower loss in 44 of 55 (80%). The corresponding proportions are 19 of 33 (58%) between 40 and 50 NFE and 23 of 41 (56%) at 50

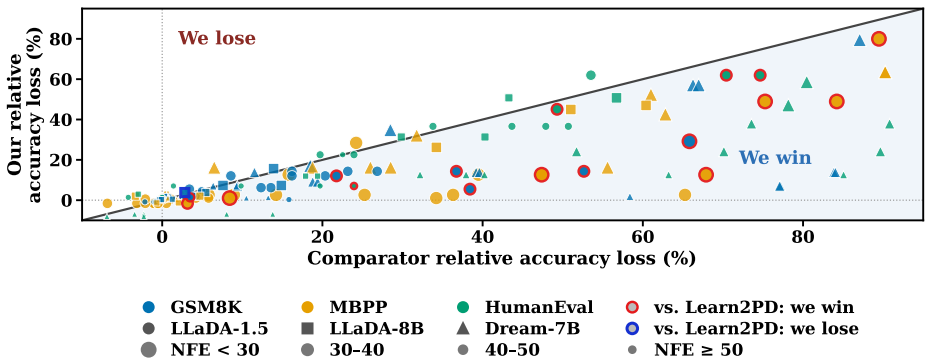

Figure 3: **Relative accuracy loss at approximately matched decoding cost.** Each point compares our decoder with a measured comparator configuration within 3.5 mean NFE. Points below the diagonal favour our decoder; points below zero indicate accuracy above the one-token-per-step reference. Color denotes the task, marker shape denotes the backbone, and marker size denotes the pair's mean NFE, with larger markers representing lower NFE. Red and blue rings identify comparisons with Learn2PD, the only supervised comparator.

NFE or above. Thus, the decoder is most consistently beneficial where few backbone evaluations are available and the commitment policy must be most aggressive.

In addition, our decoder has negative relative loss in 22 matched pairs, meaning that its task accuracy exceeds that of the one-token-per-step reference. Sixteen of these points also fall below the diagonal, so our decoder exceeds both the sequential reference and the paired decoding method at similar cost. The majority of matched comparisons continue to favour our decoder under both tighter and looser NFE-matching windows; Appendix B.1 reports the full sensitivity analysis.

Transfer. Table 1 evaluates cross-task transfer separately on each backbone. For each backbone, a stability head trained on a small GSM8K subset is reused on MBPP and HumanEval with unchanged weights, feature normalization, and cutoff $\rho = 0.99$. The transferred heads retain 96.3–98.7% of sequential accuracy on MBPP while achieving 5.42–9.21× speedup. On HumanEval, they retain 85.1–106.9% at 3.45–4.71× speedup. The GSM8K cells include the problems used to train each head and provide the in-domain reference; MBPP and HumanEval are unseen tasks. These results demonstrate cross-task reuse on all three backbones, with retained accuracy depending on the destination task. Appendix C.4 provides additional comparisons between native and ported Dream heads.

Table 1: **Cross-task transfer without head retraining.** For each backbone, a stability head trained on a small GSM8K subset (30 problems for each LLaDA backbone and 50 for Dream-7B) is reused unchanged on MBPP and HumanEval with $\rho = 0.99$. Acc. is task accuracy; retained is accuracy as a percentage of the corresponding sequential accuracy; speedup is $256/\overline{\text{NFE}}$. Colors indicate head-training task and task transfer. †GSM8K evaluations include the problems used to train each head.

	GSM8K†			MBPP			HumanEval		
Backbone	Acc.	Retained	Speedup	Acc.	Retained	Speedup	Acc.	Retained	Speedup
LLaDA-1.5	80.59	100.1%	5.07×	36.60	96.3%	9.21×	40.24	93.0%	4.25×
LLaDA-8B	76.88	98.6%	4.92×	29.40	98.7%	5.42×	34.76	85.1%	4.71×
Dream-7B	73.69	96.6%	4.99×	53.60	98.2%	5.97×	56.71	106.9%	3.45×

Training-sample robustness and data efficiency. We retain the original experimental grids: $\rho \in \{0.95, 0.99\}$ for training-subset sensitivity and $\rho \in \{0.99, 0.999\}$ for training-set-size sensitivity. These settings cover distinct measured decoding speeds, from faster commitment at $\rho = 0.95$ to more conservative acceptance at $\rho = 0.999$. Within each study, cutoffs remain fixed across heads to assess training sensitivity without individually retuning the stopping rule. Appendix B.2 details these settings, and illustrates their effects on accuracy and speedup.

Table 2: **Training-sample robustness and data efficiency on LLaDA-1.5.** Panel A reports mean $\pm$ standard deviation across five heads trained on independent 32-problem HumanEval samples. Panel B evaluates heads trained on increasing numbers of GSM8K problems and transferred to HumanEval. NFE-based speedup is $256/\overline{\mathrm{NFE}}$. All evaluations use fixed cutoffs and full task splits. †HumanEval evaluations in Panel A include each head's own 32 training problems.

A. Five randomly sampled 32-problem HumanEval training sets			
Task	Metric	$\rho = 0.95$	$\rho = 0.99$
GSM8K	Accuracy (%)	78.18 ± 1.25	80.96 ± 0.46
	Speedup	$6.02\times$	$4.57\times$
MBPP	Accuracy (%)	31.76 ± 0.46	37.24 ± 1.02
	Speedup	$11.13\times$	$8.59\times$
HumanEval†	Accuracy (%)	33.17 ± 1.91	41.95 ± 1.58
	Speedup	$5.14\times$	$3.75\times$

B. GSM8K training-set size; evaluation on HumanEval					
Training problems	15	37	75	112	142
$\rho = 0.99$					
Accuracy (%)	39.02	40.85	41.46	43.29	42.68
Speedup	$4.06\times$	$4.24\times$	$3.71\times$	$3.87\times$	$3.93\times$
$\rho = 0.999$					
Accuracy (%)	42.07	42.07	42.07	42.07	42.07
Speedup	$3.14\times$	$2.99\times$	$2.37\times$	$2.42\times$	$2.53\times$

We evaluate sensitivity to both the selected training problems and the amount of training data. Across five heads trained on independent 32-problem HumanEval samples, accuracy standard deviations range from 0.46 to 1.91 percentage points across the six evaluated task–cutoff settings as shown in Table 2. This limited variation shows that the learned decoder is not strongly tied to one particular small training sample in these settings. Increasing the GSM8K training set from 15 to 142 problems does not consistently improve HumanEval accuracy. At $\rho = 0.99$, accuracy increases from 39.02% with 15 problems to 43.29% with 112, then decreases to 42.68% with 142. At $\rho = 0.999$, all five training sizes obtain 42.07%. Even the 15-problem head reaches 39.02% accuracy at $4.06\times$ speedup or 42.07% at $3.14\times$ speedup. These results show that useful stability predictors can be trained from a few dozen problems, while additional training data alone does not reliably improve the resulting decoder.

Efficiency. During decoding, the stability predictor uses statistics already produced by the backbone and adds no backbone evaluations. Across the measured LLaDA runs, evaluating the predictor accounts for only 0.4–1.4% of decoding time, and the observed wall-clock speedup remains within 4% of the NFE-based speedup, $256/\overline{\mathrm{NFE}}$. Fitting the predictor takes less than one minute on CPU. Collecting its supervision is a one-time offline cost of approximately 284 backbone evaluations per HumanEval training problem. Appendix C.5 details the training-draw and sample-size protocols, and Appendix A.5 reports the runtime measurements.

6 Scoring and stopping rules interact

We use a factorial experiment crossing three token scores—raw confidence, margin, and learned stability—with two stopping rules. Margin is the gap between the two largest token probabilities. Threshold stopping commits proposals whose individual scores exceed a cutoff; product stopping orders proposals by confidence and extends the committed prefix while their cumulative score product remains above a cutoff.

For each task, backbone, and target cost, every score–rule combination receives its own cutoff. Cutoffs are selected using only NFE on the first 50 problems, then evaluated on the full split. This yields approximately cost-matched comparisons that vary either the score or the stopping rule while holding the other fixed.

Table 3: **Score–stopping interaction on MBPP.** Accuracy and mean NFE for the 2×2 comparison of stability head versus margin under threshold and product stopping. $I = (\text{head} - \text{margin})_{\text{threshold}} - (\text{head} - \text{margin})_{\text{product}}$; positive values mean that the head is relatively stronger under threshold stopping. Bold interactions survive Holm correction over the 13 exploratory blocks. The validation-selected follow-up uses its prespecified exact test. †The adjusted value is based on a raw bootstrap p at the resolution limit of 20,000 resamples.

	Exploratory				Validation-selected follow-up	
	LLaDA-1.5 ($n = 500$)		LLaDA-8B ($n = 500$)		LLaDA-8B ($n = 374$)	
Score and stopping rule	Accuracy (%)	NFE	Accuracy (%)	NFE	Accuracy (%)	NFE
Head, threshold	35.8	25.6	21.6	33.0	24.06	30.83
Margin, threshold	27.2	22.4	16.2	32.5	22.99	33.32
Head, product	33.2	22.8	16.4	33.6	17.91	33.78
Margin, product	31.8	24.4	21.8	34.0	20.05	33.44
Interaction I (pp)	**+7.20**		**+10.80**		+3.21	
95% CI	[+2.60, +11.80]		[+6.20, +15.40]		[−2.14, +8.56]	
Test	Holm $p = 0.024$		Holm $p = 0.0013^{\dagger}$		Exact $p = 0.291$	

To test whether the effect of a score depends on the stopping rule, we focus on the 2×2 comparison between learned stability (H) and margin (M) under threshold and product stopping. We measure the interaction as: $I = (A_{H,\text{thr}} - A_{M,\text{thr}}) - (A_{H,\text{prod}} - A_{M,\text{prod}})$. A positive I means that the learned score performs better relative to margin under threshold stopping than under product stopping. If the head is better under one rule and worse under the other, their ranking reverses. We evaluate this interaction in 13 task–backbone–budget settings where both scores are tested with both stopping rules, with Holm correction across the 13 settings. Table 3 presents the two MBPP interactions that remain significant after correction and a separate evaluation on different problems. Appendix D reports the complete exploratory score–rule comparisons, additional block-size checks, and product-transform analysis.

Validation-selected follow-up. Because the reversal was found after examining many blocks, we repeated the LLaDA-8B MBPP comparison with each arm's setting chosen on 90 validation problems by mean NFE alone and evaluated once on 374 problems not used by the exploratory blocks. The follow-up preserves the signs of the within-rule differences but does not resolve the interaction: $I = +3.21$ points with 95% interval $[-2.14, +8.56]$. Its evaluation problems had previously been used with other backbones, so it is not a blind confirmation.

Stopping rule alone. Changing only the rule can also change accuracy. On LLaDA-1.5 GSM8K, replacing thresholding with product stopping raises raw-confidence accuracy by 4.7 points at the same mean NFE (36.8), while the learned score adds no detected improvement at nearby cost (+0.2 points, $p = 0.91$). Product stopping depends on score magnitudes as well as candidate order. A common numerical cutoff therefore does not imply a common computational cost. Appendix D provides all interaction blocks and their achieved costs, the factorial arm registry and contrasts, the controlled LLaDA-1.5 comparisons, the follow-up protocol, and details on product transformations.

7 A LOCAL-TO-GLOBAL GAP

A better local stability score. We compare learned stability, raw confidence, and margin on the same recorded states, using a shared confidence ordering and approximately matched mean commitment counts. Learned stability commits fewer proposals that disagree with the sequential reference in all eight task–backbone–volume settings; all 16 comparisons with raw confidence and margin survive Holm correction. This advantage also persists on states reached by raw-confidence-product and head-product decoders on LLaDA-1.5 GSM8K. Thus, the learned score consistently improves local reference agreement under the evaluated conditions. Differences from a predictor trained on the same labels using only confidence and log-confidence are at most 0.17 percentage points; none remains significant after Holm correction.

Task-level accuracy. We then evaluate whether this local improvement leads to higher task accuracy, using cutoffs selected on separate calibration problems by NFE alone. At the primary GSM8K

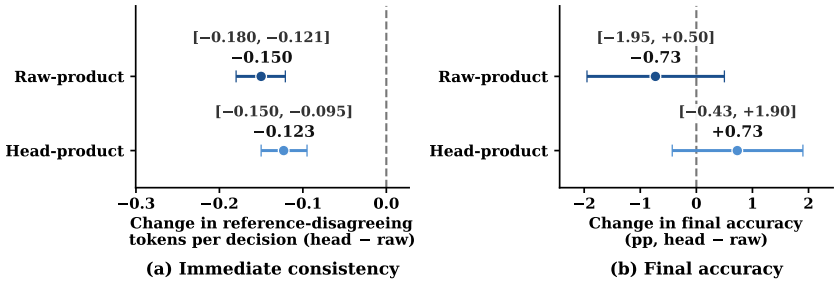

Figure 4: **A locally better commitment does not necessarily improve final correctness.** From the same parallel-decoding state, one continuation uses the prefix selected by the stability head and the other uses the raw-confidence prefix; both then resume the same base policy. The head reduces immediate reference disagreement, while the final-accuracy intervals include zero. Error bars are 95% bootstrap intervals clustered over 150 problems.

budgets, the head's accuracy difference from the raw-confidence product ranges from -4.22 to $+0.22$ percentage points across the three backbones, with none of the comparisons surviving Holm correction. The strongest positive estimate occurs on LLaDA-1.5 MBPP, where the head exceeds the raw-confidence product by 4.81 points at the secondary budget (nominal $p = 0.0096$; Holm $p = 0.057$). The direction and magnitude of the task-level effect therefore vary across tasks, backbones, and budgets, despite the consistent improvement in local reference agreement.

Following one decision to completion. To observe the consequence of one commitment decision directly, we start two continuations from the same recorded parallel-decoding state. One commits the prefix selected by the stability head, while the other commits the prefix selected by raw confidence; both are then completed using the same base policy. Across 150 LLaDA-1.5 GSM8K problems. The stability-guided decision commits fewer reference-disagreeing tokens; at least one branch differs from the unmodified run in 36–41% of tested states.. Nevertheless, the final-accuracy differences are -0.7 points $[-2.0, +0.5]$ on raw-product states and $+0.7$ points $[-0.4, +1.9]$ on head-product states, as shown in Figure 4.

Together, these experiments reveal a **local-to-global gap**: learned stability selects more reference-consistent commitments, and those decisions frequently change the generated completion, yet a single improved decision produces no measurable change in final-answer correctness. Appendix E provides the state construction, confidence-only control, complete validation-selected results, and rollout protocol.

8 Limitations and Conclusion

Limitations. Decoding is greedy and irreversible, with 256-token continuations and primarily block size 32; Appendix D.3 reports three block-64 checks with retrained heads. Longer generations remain untested. Methods differ in their observed accuracy–cost ranges, with some degrading more sharply at low NFE. Comparisons use exploratory sweeps at approximately matched NFE; the majority advantage persists across matching windows, although sweep coverage is unequal. Three settings include head-training problems, and Dream and ported LLaDA heads differ in training task and sample size. Local diagnostics do not establish an advantage over a learned confidence-only control. Validation-selected evaluations share problems across backbones; the interaction follow-up remains unresolved, and rollouts change one decision on one task and backbone.

Conclusion. We introduce a lightweight stability predictor that controls parallel commitments without additional backbone evaluations and supports reuse across tasks and LLaDA backbones. The decoder achieves higher accuracy in 71% of measured comparisons at approximately matched NFE. Our analyses show that a score's accuracy effect can depend on the stopping rule, while improved local reference agreement does not consistently translate into higher task accuracy. Token scores should therefore be evaluated through the commitment policies they induce and the task accuracy those policies achieve.

AI USE STATEMENT

Generative AI tools assisted manuscript revision and organization, figure preparation, mathematical exposition, bibliographic checking, code implementation, and experimental analysis. This included running and analyzing the validation-selected MBPP evaluations on LLaDA-1.5 and Dream-7B in Section 7 under author-specified settings. The authors retain responsibility for the research design, verification of the code and experimental results, accuracy of references, and all scientific claims. AI assistance does not constitute independent validation of the experimental record.

ETHICS STATEMENT

This work evaluates existing language models on established benchmarks without collecting new human-subject data. Agreement with a sequential decoding reference does not guarantee task correctness or safety. Generated programs require isolated execution during evaluation, and reducing inference cost does not remove the limitations or risks of the underlying models.

REPRODUCIBILITY STATEMENT

The main text and appendices document the predictor architecture and features, label-collection procedure, training provenance, generation settings, comparator implementations, and software and hardware configurations. They also specify the evaluation splits, cutoff-selection procedures, approximate cost matching, cross-fitted diagnostics, and statistical tests. Results from decoding sweeps are distinguished from evaluations with settings selected on separate validation data. Training–evaluation overlap and reuse of evaluation problems are documented.

REFERENCES

Sudhanshu Agrawal, Risheek Garrepalli, Raghavv Goel, Christopher Lott, Fatih Porikli, and Mingu Lee. Structuring the future: Diffusion llm speculative decoding via calibrated draft graphs. *arXiv preprint arXiv:2509.18085*, 2025.

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.

Giries Abu Ayoub, Mario Barbara, Lluís Pastor-Pérez, Tanja Bien, Aneesh Barthakur, Alaa Maalouf, and Loay Mualem. Supportive token revealing for fast diffusion language model decoding. *arXiv preprint arXiv:2606.04236*, 2026.

Iskander Azangulov, Teodora Pandeva, Niranjani Prasad, Javier Zazo, and Sushrut Karmalkar. Parallel sampling from masked diffusion models via conditional independence testing. In *International Conference on Learning Representations*, volume 2026, pp. 140485–140521, 2026.

Wenrui Bao, Zhiben Chen, Dan Xu, and Yuzhang Shang. Learning to parallel: Accelerating diffusion large language models via learnable parallel decoding. In *International Conference on Learning Representations*, volume 2026, pp. 83376–83392, 2026.

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*, 38:55981–56007, 2025.

Thomas Bertolani, Davide Bucciarelli, Leonardo Zini, Marcella Cornia, and Lorenzo Baraldi. Diffusion language models: An experimental analysis. *arXiv preprint arXiv:2606.19475*, 2026.

Tiwei Bie, Maosong Cao, Xiang Cao, Bingsen Chen, Fuyuan Chen, Kun Chen, Lun Du, Daozhuo Feng, Haibo Feng, Mingliang Gong, et al. Llada2. 1: Speeding up text diffusion via token editing. *arXiv preprint arXiv:2602.08676*, 2026.

Tzu-Tao Chang, Benjamin Yuanyang Hong, Kiet Pham, and Shivaram Venkataraman. Dilaserve: High slo attainment serving for diffusion language models. *arXiv preprint arXiv:2606.29094*, 2026.

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.

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

Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and LingPeng Kong. Diffuseq: Sequence to sequence text generation with diffusion models. *arXiv preprint arXiv:2210.08933*, 2022.

Michael Hersche, Nicolas Menet, Ronan Tanios, and Abbas Rahimi. Locally coherent parallel decoding in diffusion language models. *arXiv preprint arXiv:2603.20216*, 2026.

Daniel Israel, Guy Van den Broeck, and Aditya Grover. Accelerating diffusion llms via adaptive parallel decoding. *Advances in neural information processing systems*, 38:52870–52888, 2025.

Metod Jazbec, Theo X Olausson, Louis Béthune, Pierre Ablin, Michael Kirchhof, João Monteiro, Victor Turrisi, Jason Ramapuram, and Marco Cuturi. Learning unmasking policies for diffusion language models. *arXiv preprint arXiv:2512.09106*, 2025.

Bumjun Kim, Dongjae Jeon, Moongyu Jeon, and Albert No. Dapd: Dependency-aware parallel decoding via attention for diffusion llms. *arXiv preprint arXiv:2603.12996*, 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*, 38:92267–92301, 2025.

Zakhar Kohut, Severyn Shykula, Mykola Vysotskyi, Serhii Dmytryshyn, Dmytro Khamula, Michal Zakrzewski, Damian Rynczak, Jacek Małecki, Taras Rumezhak, and Volodymyr Karpiv. Just on time: Token-level early stopping for diffusion language models. *arXiv preprint arXiv:2602.11133*, 2026.

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

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in neural information processing systems*, 35: 4328–4343, 2022.

Yangzhou Liu, Yue Cao, Hao Li, Gen Luo, Zhe Chen, Weiyun Wang, Xiaobo Liang, Biqing Qi, Lijun Wu, Changyao Tian, Yanting Zhang, Yuqiang Li, Tong Lu, Yu Qiao, Jifeng Dai, and Wenhai Wang. Sequential diffusion language models. *arXiv preprint arXiv:2509.24007*, 2025.

Guanxi Lu, Hao Chen, Yuto Karashima, Zhican Wang, Daichi Fujiki, and Hongxiang Fan. Adablock-dllm: Semantic-aware diffusion llm inference via adaptive block size. In *International Conference on Learning Representations*, volume 2026, pp. 140018–140033, 2026.

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.

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. *Advances in Neural Information Processing Systems*, 38:50608–50646, 2025.

Han Peng, Peiyu Liu, Zican Dong, Daixuan Cheng, Junyi Li, Yiru Tang, Shuo Wang, and Wayne Xin Zhao. How efficient are diffusion language models? a critical examination of efficiency evaluation practices. *arXiv preprint arXiv:2510.18480*, 2025.

Heqiang Qi, Wei Huang, Mingyuan Bai, and Xiangming Meng. Cluster-level attention-guided parallel decoding for masked diffusion language models. *arXiv preprint arXiv:2605.29607*, 2026.

Yu-Yang Qian, Junda Su, Lanxiang Hu, Peiyuan Zhang, Zhijie Deng, Peng Zhao, and Hao Zhang. d3llm: Ultra-fast diffusion llm using pseudo-trajectory distillation. *arXiv preprint arXiv:2601.07568*, 2026.

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

Yusuf Sahin, Ahmed Rockey Saikia, Volkan Cevher, and Paolo Favaro. Attention-discounted adaptive sampler for masked diffusion language models. *arXiv preprint arXiv:2606.10829*, 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*, 37:130136–130184, 2024.

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.

Bohang Sun, Max Zhu, Francesco Caso, Jindong Gu, Junchi Yu, Philip Torr, Pietro Liò, and Jialin Yu. The path matters: Learning a token-commitment policy for diffusion language models. *arXiv preprint arXiv:2605.24697*, 2026a.

Shengyin Sun, Yiming Li, Renxi Liu, Xinqi Li, Hui-Ling Zhen, Weizhe Lin, Chen Chen, Xianzhi Yu, Mingxuan Yuan, and Chen Ma. Psd: Pushing the pareto frontier of diffusion llms via parallel speculative decoding. *arXiv preprint arXiv:2605.15609*, 2026b.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

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. In *International Conference on Learning Representations*, volume 2026, pp. 57027–57051, 2026.

Shutong Wu and Jiawei Zhang. Free draft-and-verification: Toward lossless parallel decoding for diffusion large language models. *arXiv preprint arXiv:2510.00294*, 2025.

Xiang Xia, Wuyang Zhang, Jiazheng Liu, Cheng Yan, and Yanyong Zhang. Depcap: Adaptive block-wise parallel decoding for efficient diffusion lm inference. *arXiv preprint arXiv:2604.15750*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

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.

Hengxiang Zhang, Jiaxi Ren, Renchunzi Xie, and Hongxin Wei. Understanding evaluation illusion in diffusion large language models. *arXiv preprint arXiv:2606.29228*, 2026a.

Jintao Zhang, Pengle Zhang, Jun Zhu, Jianfei Chen, et al. Sageattention: Accurate 8-bit attention for plug-and-play inference acceleration. In *International Conference on Learning Representations*, volume 2025, pp. 71566–71585, 2025.

Tianao Zhang, Zhiteng Li, Xianglong Yan, Haotong Qin, Yong Guo, and Yulun Zhang. Quant-dllm: Post-training extreme low-bit quantization for diffusion large language models. In *International Conference on Learning Representations*, volume 2026, pp. 79492–79503, 2026b.

Fengqi Zhu, Rongzhen Wang, Shen Nie, Xiaolu Zhang, Chunwei Wu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Llada 1.5: Variance-reduced preference optimization for large language diffusion models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11425–11460, 2026.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

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

A METHOD AND EXPERIMENTAL DETAILS

A.1 FEATURES, PREDICTOR, AND DECODING

For each masked position, let $p_1 \geq p_2 \geq p_3$ denote the three largest output probabilities and $c = p_1$ the confidence of the proposed token. The 13 inputs, in order, are c, $p_1 - p_2$, $\log(p_1/p_2)$, p_2, p_3, $p_1 - p_3$, the entropy of the full vocabulary distribution, the replay offset, the fraction of the active block that remains masked, the number of masked positions, the position within the block, the block index, and $\log c$. Each input is standardized with the mean and standard deviation of the head's own training labels, which are stored with the checkpoint. The predictor is a $13 \to 64 \to 64 \to 1$ network with ReLU hidden layers and a sigmoid output, for a total of $5{,}121$ parameters. Raw confidence and margin use c and $p_1 - p_2$, respectively, and contain no trainable parameters. Algorithm 1 gives the decoding procedure.

Algorithm 1 Confidence-ordered stability decoding

Input: prompt, frozen backbone, fitted stability predictor, continuation length L, block length B, cutoff ρ.
Initialize L masked continuation positions and divide them into blocks of length B.
for each block **do**
  while the active block contains masked positions M **do**
  Run the backbone once; obtain proposals $\hat{y}_i$, confidences c_i, and distributional features.
  Set replay offset to zero; construct and standardize the 13 features; compute $\hat{p}_i$.
  Order M by descending c_i; set $r \leftarrow 1$ and $k \leftarrow 1$ (the first proposal is always committed).
  for $j = 2, \ldots, |M|$ **do**
  if $r\hat{p}_{(j)} < \rho$ **then break**.
  $r \leftarrow r\hat{p}_{(j)}$; $k \leftarrow j$.
  Commit the first k proposals at their original positions.
Return the completed continuation.

A.2 LABEL COLLECTION

The reference $y^\star$ is the model's own sequential continuation, generated during label collection by committing one highest-confidence proposal at a time within the active block. Base passes record proposals and features, while probe passes identify subsequent reference tokens and compare them with the stored proposals. The reference and training labels are therefore produced together, without a separate reference decode.

As shown in Algorithm 2, each collection step begins with a base pass that records the proposals, confidences, and features of all masked positions in the active block. The replay then advances along the reference: the highest-confidence position of the base pass is filled with its reference token and is not labelled. After each insertion the backbone is re-run on the partially filled state (a *probe pass*), which recomputes all proposals. The position with the highest recomputed confidence, that is, the position the sequential decoder fills next, is labelled stable if the proposal stored from the base pass equals its reference token $y^\star_i$ and unstable otherwise; a stable position is then filled with $y^\star_i$, whereas an unstable position ends the collection from this base pass and stays masked. The replay offset of a label is the number of reference tokens inserted since the base pass, so it is at least 1 during collection. Collection from a base pass stops at the first unstable label, which is kept, or when the block is full, and the next base pass starts from the state filled with the accepted reference tokens. Because the candidates follow the sequential decoder's order, which changes as tokens are inserted, and not the order of the base-pass confidences, the labelled positions are those up to and including the first disagreement in that order, rather than every masked position. At deployment, candidates are ordered by base-pass confidence and the replay offset is set to zero, a value that does not occur in training; Appendix A.7 examines this difference.

A.3 HEAD REGISTRY AND TRAINING

Table 4 summarizes the heads used in this paper. The complete-decoder comparisons use H_L on both LLaDA backbones and H_D on Dream-7B; H_L is also used in the separate ported-head experiments on Dream. The task-transfer evaluation in Table 1 uses a GSM8K-trained head for each backbone:

Algorithm 2 Offline reference-consistency label harvesting

Input: prompt x, frozen backbone P_θ, and lengths L and B.
Initialize z with L masks and $\mathcal{D} \leftarrow \varnothing$.
for each continuation block **do**
  while its masked set M is nonempty **do**
    Run a base pass and store $(\hat{y}_{t,i}, c_{t,i}, \phi_{t,i})$ for all $i \in M$.
    Set $i \leftarrow \arg\max_{j \in M} c_{t,j}$ and write $\hat{y}_{t,i}$ into z_i.
    Set $q \leftarrow 1$.
    while masked positions remain in the block **do**
      Run a probe pass; select its highest-confidence masked position i.
      Let $\tilde{y}_i$ be the probe's proposal at position i.
      Set $s \leftarrow \mathbf{1}[\hat{y}_{t,i} = \tilde{y}_i]$.
      Add $(\phi_{t,i}$ with offset $q, s)$ to $\mathcal{D}$.
      if $s = 0$ **then break**, leaving position i masked.
      Write $\tilde{y}_i$ into z_i and set $q \leftarrow q + 1$.
The completed sequence z is the sequential reference $y^\star$.
Return $\mathcal{D}$.

Table 4: **Stability heads used at block size 32.** Training backbone and task, number of training problems, distinct labelled proposals, number of inputs, and parameter count. Counts are per head; ranges describe the heads in the training-size study. The source package lists each head's training-problem identifiers.

Head	Training backbone, task	Problems	Labels	Inputs	Parameters
Deployed LLaDA head (H_L)	LLaDA-1.5, HumanEval	32	7,936	13	5,121
Deployed Dream head (H_D)	Dream-7B, GSM8K (test)	50	12,400	13	5,121
Task-transfer head	LLaDA-1.5, GSM8K	30	7,440	13	5,121
Task-transfer head	LLaDA-8B, GSM8K	30	7,440	13	5,121
Confidence-only control	LLaDA-1.5, HumanEval	32	7,936	2	4,417
Feature-ablation heads	LLaDA-1.5, HumanEval	32	7,936	7/8/13	4,737/4,801/5,121
Training-draw heads (5)	LLaDA-1.5, HumanEval	32	7,936	13	5,121
Training-size heads (5)	LLaDA-1.5, GSM8K (test)	15–142	3,720–35,216	13	5,121

30 training problems for each LLaDA head and 50 for H_D. The confidence-only control is used in the local diagnostics and feature ablation.

All heads are trained with binary cross-entropy using Adam with learning rate 3×10^{-3} for 15 full-batch epochs and training seed 0. The five training-draw heads use seeds 1–5 to sample their 32-problem training sets. The training-size study uses 15, 37, 75, 112, and 142 GSM8K problems.

For 256-token continuation positions and block length 32, label collection produces 31 labels per block and 248 per problem at block size $B = 32$, and 252 per problem at block size $B = 64$. Unless otherwise stated, methods generate 256 continuation tokens with block length 32. The label file for H_D contains two exact copies of each of its 12,400 distinct labels; Table 4 reports distinct counts. Uniform duplication leaves the full-batch mean loss unchanged. Fitting a head takes less than one minute on CPU.

A.4 Generation, scoring, comparators, and software

Generation and scoring. All methods generate 256 continuation tokens with block length 32, greedy proposals, and no classifier-free guidance. Prompts use five-shot GSM8K, three-shot MBPP, and zero-shot HumanEval; a chat template is applied only to LLaDA-8B-Instruct. HumanEval outputs are sanitized before execution. MBPP generations are not sanitized: the harness truncates each generation at the task's `[DONE]` stop sequence, removes special tokens, and executes the result against the task's three test assertions. All methods are evaluated on identical problems through the same task-specific scoring pipeline. Product cutoffs and confidence thresholds are swept over task-specific grids to obtain the exploratory operating points; the validation-selected evaluations use separate calibration grids (see Appendix E.4).

Table 5: **Measured inference speedup.** $S_{\mathrm{NFE}} = 256/\overline{\mathrm{NFE}}$, S_{time} is measured wall-clock speedup, and $\eta = S_{\mathrm{time}}/S_{\mathrm{NFE}}$. Head share is the fraction of decoding time spent evaluating the predictor.

Backbone	Task	NFE	S_{NFE}	S_{time}	η	Head share
LLaDA-1.5	GSM8K	33.2	7.70	7.83	1.02	0.4%
		37.7	6.80	6.96	1.02	0.4%
	MBPP	22.8	11.21	11.37	1.01	0.5%
		28.0	9.15	9.17	1.00	0.5%
	HumanEval	39.7	6.45	6.41	0.99	1.4%
		42.0	6.09	6.07	1.00	1.1%
LLaDA-8B	GSM8K	37.8	6.78	7.04	1.04	0.4%
		42.7	5.99	6.22	1.04	0.4%
	MBPP	33.6	7.61	7.69	1.01	0.5%
		39.0	6.57	6.64	1.01	0.4%
	HumanEval	40.5	6.33	6.31	1.00	1.1%
		44.5	5.76	5.79	1.00	1.1%
Dream-7B	HumanEval	29.1	8.80	8.19	0.93	–
		48.1	5.32	5.02	0.94	–

Comparators. Tuned confidence thresholding follows Fast-dLLM (Wu et al., 2026). DAWN uses its published dependency-based decoding rule, sweeping τ_{low} while fixing $\tau_{\mathrm{sink}} = 0.01$, $\tau_{\mathrm{edge}} = 0.07$, and $\tau_{\mathrm{induce}} = 0.7$. DAPD reproduces its reported block-256 result within 2.1 accuracy points in our harness before evaluation at block 32. LocalLeap is evaluated in the settings for which runs are available. For Learn2PD, the released GSM8K filter reaches 76.72% accuracy at $4.08\times$ speedup in our harness, close to the reported 78.62% at $4.13\times$; for LLaDA-1.5 we use a filter trained with the authors' recipe on 300 GSM8K training prompts. ADAS's released Dream configuration uses temperature and top-p sampling, which saturates its stopping statistic in our setting, so we evaluate an aligned greedy version (temperature 0, no top-p, no chat template, explicit BOS) after a 20-problem parity check that matched contexts, NFE, and extracted answers. Comparator coverage varies by task and backbone. KLASS (single-block only) and TraceLock (Sun et al., 2026a) (its release could not be run) were not evaluated.

Software and hardware. Experiments use `lm-evaluation-harness` 0.4.11, `transformers` 4.46.3, `torch` 2.7.0, and `evaluate` 0.4.6.

A.5 Runtime and supervision cost

Wall-clock speedup is sequential decoding time divided by policy decoding time under the same task, backbone, hardware, and batch size. On LLaDA, wall-clock and NFE-based speedups differ by at most 4%, and the predictor accounts for 0.4–1.4% of decoding time, as shown in Table 5. The timings were taken on shared nodes and are not isolated profiling; Dream-7B sequential timings were unavailable for GSM8K and MBPP. Across the 164 HumanEval problems, label collection averages 284.35 backbone evaluations per problem: 36.35 base passes and 248 probe passes. The first base pass of each block and the probe passes together evaluate the 256 states of sequential decoding. The remaining 28.35 base passes repeat states after disagreements. Thus, collection includes reference generation and costs approximately 11% more than one sequential decode. This one-time offline cost is excluded from decoding NFE.

A.6 Feature ablation

Table 6 varies the predictor's inputs while holding the labels and the product rule fixed. The distribution-only predictor stays within one accuracy point of the full predictor at comparable cost on both tasks, whereas adding decoding context to confidence alone lowers accuracy by 5.0 points on GSM8K and 0.6 on MBPP; the two confidence-only MBPP points show how strongly the comparison depends on cost. The ablation supports the chosen inputs but is a single-operating-point

Table 6: **Feature ablation under product stopping.** LLaDA-1.5 accuracy and cost. All predictors use the same labels; the second confidence-only MBPP point shows cost-matching sensitivity. Stability AUC is a local diagnostic.

Features	Accuracy (%)	NFE	Stability AUC
MBPP			
Confidence only	30.2	25.5	0.9539
Confidence only (2nd point)	34.2	27.5	0.9539
Confidence + context	29.6	25.8	0.9512
Distributional	36.8	26.2	0.9676
Full deployed head (ours)	37.0	28.0	0.9656
GSM8K			
Confidence only	75.0	37.3	0.9152
Confidence + context	70.0	37.5	0.9117
Distributional	76.1	37.0	0.9306
Full deployed head (ours)	75.5	37.7	0.9265

comparison without tests. Stability AUC measures agreement with the sequential reference, not task correctness.

A.7 Calibration

Predicted stability should not be read as a calibrated probability on deployed states. On LLaDA-1.5, expected calibration error (six probability bands) is 0.002–0.005 on held-out problems labelled with the prefix-selected protocol used for training, and 0.008–0.013 when the replay offset is set to its deployment value of zero. On an uncensored sample of all masked positions at oracle-guided states, stability prevalence falls from 88.5% to 57.1% and calibration error rises to 0.197 (0.220 when the position index used there in place of the replay offset is set to zero). The selected-label and uncensored populations therefore differ, and neither measures calibration on the states visited by a parallel decoder.

Table 7: **Sensitivity to the NFE matching window.** The nearest-cost pairing procedure is repeated with different maximum absolute NFE differences. Lower, equal, and higher loss refer to our decoder relative to its paired comparator. Percentages use all eligible pairs, including ties. The final column reports lower/higher-loss counts within the Learn2PD subset. Pairs reuse configurations and evaluation problems, so counts are descriptive rather than independent statistical tests.

Maximum $\lvert\Delta\mathrm{NFE}\rvert$	Pairs	Win	Tie	Loss	Learn2PD: win / loss
2	117	90 (77%)	1	26	15 / 0
3.5	164	117 (71%)	4	43	17 / 1
5	183	129 (70%)	5	49	18 / 1
10	234	159 (68%)	8	67	23 / 2

B Design choices

B.1 Sensitivity to the cost-matching window.

Table 7 repeats the comparison using maximum NFE differences of 2, 3.5, 5, and 10. Our decoder has lower relative accuracy loss in 68–77% of eligible pairs across these windows. Tightening the main-text window from 3.5 to 2 NFE retains 117 pairs, of which 90 (77%) favour our decoder. Thus, the descriptive majority advantage persists under tighter cost matching. Changing the window changes the eligible comparison set; the 10-NFE window is a broad sensitivity check.

Table 8: **Effect of the product cutoff on decoding.** Complete decoding with the same deployed head H_L and unchanged weights. Speedup is $256/\overline{\mathrm{NFE}}$. HumanEval evaluations include the head's 32 training problems. Single-token steps are reported as rounded percentages.

	HumanEval			MBPP	
ρ	Accuracy (%)	Speedup	Single-token steps (%)	Accuracy (%)	Speedup
0.95	33.54	5.22×	38	33.20	11.21×
0.99	40.24	4.16×	50	37.00	9.15×
0.999	42.68	2.75×	63	38.60	6.64×

B.2 Cutoff settings and decoding behavior.

We retain the cutoff grids used in the original experiments. The training-subset study uses $\rho \in \{0.95, 0.99\}$ to examine faster decoding, while the training-set-size study uses $\rho \in \{0.99, 0.999\}$, including a more conservative setting. Within each study, every head is evaluated at the same cutoffs. Both pairs cover distinct measured speed ranges, with $\rho = 0.99$ shared between the studies.

Higher ρ imposes a stricter cumulative-product criterion, allowing shorter commitment prefixes at a given decoding state. Table 8 illustrates this effect using the same deployed head at all three cutoffs. Increasing ρ from 0.95 to 0.999 raises observed HumanEval accuracy from 33.54% to 42.68%, while reducing NFE-based speedup from 5.22× to 2.75×. MBPP shows the same direction: accuracy rises from 33.20% to 38.60%, while speedup decreases from 11.21× to 6.64×. Thus, the evaluated cutoffs span faster and more conservative decoding while retaining substantial parallelism. Fixed cutoffs can produce different costs across trained heads.

Table 9: **Matched-cost pairs by setting.** Counts of pairs in which our decoder has lower, equal, or higher relative accuracy loss (lower is better), together with the absolute difference in mean NFE.

Setting	Pairs	Ours lower	Tied	Ours higher	Median $\lvert\Delta\text{NFE}\rvert$	Max $\lvert\Delta\text{NFE}\rvert$
LLaDA-1.5 GSM8K	26	18	1	7	0.89	2.35
LLaDA-1.5 MBPP	34	26	1	7	0.73	2.64
LLaDA-1.5 HumanEval	20	12	1	7	1.13	3.48
LLaDA-8B GSM8K	8	4	0	4	2.23	2.69
LLaDA-8B MBPP	7	6	0	1	1.43	2.89
LLaDA-8B HumanEval	7	4	0	3	0.38	3.40
Dream-7B GSM8K	25	19	0	6	1.22	3.36
Dream-7B MBPP	18	9	1	8	1.53	3.24
Dream-7B HumanEval	19	19	0	0	1.60	3.05
All settings	164	117	4	43	1.14	3.48

C Complete-decoder comparisons and transfer

C.1 Matching and statistical protocol

Pairwise comparisons pair measured operating points whose mean NFE differs by at most 3.5, without interpolation, and report both achieved costs. The window is an operational matching tolerance, not an equivalence guarantee; near 20 NFE a 3.5-NFE gap is a substantial fraction of the cost. Accuracy differences are evaluated on identical problems with two-sided exact McNemar tests, which compare correctness vectors and do not adjust for unequal computation; w/l denotes the number of problems solved only by the first or only by the second policy. Paired 95% intervals use $20{,}000$ problem-level bootstrap resamples. Reported p-values are nominal unless a correction family is stated, and a nonsignificant difference is not interpreted as equivalence. Exploratory operating points were selected after inspecting sweeps, so corrections within a reported family do not account for that selection.

C.2 Matched-cost pairs

The matched-cost analysis includes all full-split operating points for the deployed stability heads with product stopping (H_L on LLaDA and H_D on Dream) and for confidence thresholding, DAWN, ADAS, DAPD, Learn2PD, and LocalLeap. The evaluated splits contain $1{,}319$ GSM8K, 500 MBPP (test ids 11–510), and 164 HumanEval problems. Runs on calibration sets, subsets, and the separate 374-problem MBPP evaluation set are excluded.

Each comparator operating point is matched to the nearest head operating point on the same task and backbone, provided their mean NFE differs by at most 3.5. Distance ties favour the lower-NFE point. A head operating point may match multiple comparator points, while comparator points without an eligible match are excluded. Identical operating points are counted once. Relative accuracy loss is measured against the sequential baseline for the same task and backbone,

$$\ell = (A_{\text{seq}} - A)/A_{\text{seq}},$$

with negative values retained when accuracy exceeds the sequential baseline. All comparisons use unrounded values.

This procedure produces 164 pairs from 275 comparator configurations. Our decoder has lower relative accuracy loss in 117 pairs (71%), equal loss in 4, and higher loss in 43. By comparator, it has lower loss in 30 of 40 confidence comparisons, 29 of 41 DAWN comparisons, 25 of 36 ADAS comparisons, 6 of 16 DAPD comparisons, 17 of 18 Learn2PD comparisons, and 10 of 13 LocalLeap comparisons.

The advantage is strongest at aggressive operating points. Our decoder has lower loss in 28 of 61 pairs where the comparator loses less than 5% of sequential accuracy, 10 of 13 at 5–10%, 18 of 21 at 10–20%, 9 of 13 at 20–30%, and 52 of 56 at 30% or more. Overall, it has lower loss in 79 of 90 pairs where the comparator loses at least 10%. Table 9 reports the breakdown by task and backbone.

Table 10: **Training-problem exclusion audit.** Head accuracy and paired differences with tuned confidence (Conf.) and DAWN before and after excluding head-training problems. Retained sets contain 132 HumanEval or 1,269 GSM8K problems. Bold differences have nominal $p < 0.05$.

		Head accuracy (%)		Full set		Retained set	
Head NFE[a]	Comparator @ NFE[b]	Full	Retained	Δ	p	Δ	p
HumanEval / LLaDA-1.5							
39.7 (39.4)	Conf. @ 41.6	23.78	25.00	+1.22	0.83	+3.03	0.48
42.0 (43.1)	Conf. @ 41.6	27.44	29.55	+4.88	0.18	+7.58	0.06
HumanEval / LLaDA-8B							
40.5 (40.4)	Conf. @ 37.8	20.12	18.94	+2.44	0.61	+0.00	1.00
44.5 (45.1)	Conf. @ 44.5	28.05	27.27	+3.66	0.26	+2.27	0.63
GSM8K / Dream-7B							
30.8[c]	Conf. @ 29.8	64.67	64.62	**+19.33**	4.7×10^{-34}	**+19.39**	8.5×10^{-33}
30.8[c]	DAWN @ 29.9	64.67	64.62	**+19.03**	1.4×10^{-34}	**+18.91**	1.9×10^{-32}
34.0[c]	Conf. @ 34.9	68.23	68.24	**+7.73**	8.0×10^{-9}	**+7.64**	1.7×10^{-8}
34.0[c]	DAWN @ 35.0	68.23	68.24	**+7.28**	7.3×10^{-8}	**+6.93**	4.2×10^{-7}

[a] Full-set mean NFE; retained-set mean in parentheses where per-problem costs were logged. [b] Comparator full-set mean NFE. [c] Head full-set mean NFE.

Because configurations and evaluation problems recur across pairs, these counts summarize the measured operating points rather than independent statistical tests.

C.3 Training-overlap audit

H_L was trained on 32 HumanEval problems and H_D on 50 GSM8K test problems, so the HumanEval evaluations on both LLaDA backbones and the GSM8K evaluation on Dream include head-training problems. Table 10 repeats the matched-cost comparisons with tuned confidence and DAWN at two exploratory operating points in each affected setting after excluding those problems. The LLaDA HumanEval differences remain statistically unresolved after exclusion, and the Dream GSM8K gains remain significant on the other 1,269 problems. Head NFE on the retained problems is given where per-problem costs were logged; comparator NFE refers to the full split.

C.4 Native versus ported Dream heads

Table 11 compares H_D (native) with H_L ported unchanged to Dream, with its own feature standardization, on identical problems. Each measured point of one head is paired with the nearest measured point of the other within 3.5 mean NFE, without interpolation, so the two cutoffs can differ; all 11 eligible pairs are shown. H_D is ahead in all six GSM8K pairs and in three of five MBPP pairs, and the difference is unresolved near 60 NFE. The two heads differ in training backbone, training task, and number of training problems, so these comparisons do not isolate the effect of model family. Δ is native minus ported with a paired-bootstrap 95% interval over problems; bold marks Holm-adjusted McNemar $p < 0.05$ in the family of the source analysis; the operating points come from sweeps and are exploratory.

C.5 Training-draw and sample-size protocols

Training draws. Five additional heads were trained with the deployed recipe on independent random draws of 32 of the 164 HumanEval problems (seeds 1–5; seed 0 gives H_L; the largest pairwise overlap between draws is 9 problems). Each head was evaluated at $\rho = 0.95$ and 0.99 on the full GSM8K (1,319), MBPP (500), and HumanEval (164) splits; the HumanEval evaluation includes each head's own training problems.

Table 11: **Dream-7B: Dream-trained versus LLaDA-trained head.** Accuracy (%)@mean NFE for each head's nearest measured points within 3.5 mean NFE; Δ is native minus ported with paired-bootstrap 95% intervals; bold: Holm-adjusted McNemar $p < 0.05$. Exploratory.

Task	Cutoff ρ (native / ported)	Native	Ported	Δ [95% CI]
GSM8K	0.85 / 0.9	64.67@30.8	59.21@32.7	**+5.5 [+2.9, +8.0]**
	0.9 / 0.9	68.23@34.0	59.21@32.7	**+9.0 [+6.7, +11.4]**
	0.95 / 0.95	69.75@39.1	65.58@37.0	**+4.2 [+1.9, +6.4]**
	0.99 / 0.997	73.69@51.3	72.48@52.4	+1.2 [+0.1, +2.4]
	0.99 / 0.995	73.69@51.3	70.81@49.8	**+2.9 [+1.5, +4.2]**
	0.997 / 0.999	74.22@60.1	73.31@60.6	+0.9 [+0.0, +1.8]
MBPP	0.9 / 0.9	37.60@20.7	32.80@20.0	+4.8 [+1.2, +8.4]
	0.9 / 0.95	37.60@20.7	39.80@21.9	−2.2 [−5.0, +0.6]
	0.95 / 0.95	46.40@25.1	39.80@21.9	**+6.6 [+3.0, +10.2]**
	0.99 / 0.997	53.60@42.9	50.60@42.8	**+3.0 [+1.4, +4.8]**
	0.997 / 0.999	54.80@59.2	55.00@57.3	−0.2 [−1.4, +1.0]

Training-set size. Five heads were trained with LLaDA-1.5 on 15, 37, 75, 112, and 142 problems (10–95% of a 150-problem GSM8K label set) and evaluated on all 164 HumanEval problems at $\rho = 0.99$ and 0.999, a task different from their training task.

Table 12: **Score × stopping-rule interaction, all 13 four-arm blocks (exploratory).** I is in accuracy points with pointwise 95% bootstrap intervals not adjusted for multiplicity; bold I: Holm $p < 0.05$. [§] Bootstrap resolution floor. [†] An arm is more than 3.5 NFE from the block target, or the arms differ by more than 3.5 NFE. NFE is the achieved mean NFE of head/product (HP), head/threshold (HT), margin/product (MP), and margin/threshold (MT). Definitions and procedures are given in the text.

Target NFE	I (pp)	Pointwise 95% percentile-bootstrap CI	Raw p (finite-resample)	Holm p (13 blocks)	Sign-flip exact Holm p (sens.)	NFE HP/HT/MP/MT	Cost
Block: LLaDA-1.5 GSM8K							
33.2	−4.62	[−8.34, −0.83]	0.017	0.17	0.18	33.2/33.1/31.3/33.3	
37.7	−0.91	[−4.17, +2.35]	0.61	1	1	37.7/37.6/36.9/38.3	
41.8	−4.62	[−7.96, −1.29]	0.0066	0.073	0.090	41.8/37.6/36.9/41.0	†
Block: LLaDA-1.5 MBPP							
22.8	**+7.20**	[+2.60, +11.80]	0.0020	0.024	0.034	22.8/25.6/24.4/22.4	
28.0	−0.60	[−4.00, +2.60]	0.76	1	1	28.0/28.0/26.8/28.4	
Block: LLaDA-1.5 HumanEval							
39.7	+4.27	[−3.66, +12.20]	0.34	1	1	39.7/37.1/39.8/40.9	†
42.0	+1.83	[−6.10, +10.37]	0.71	1	1	42.0/43.7/39.8/40.9	†
Block: LLaDA-8B-Instruct GSM8K							
37.8	−2.50	[−6.14, +1.14]	0.19	1	1	37.8/36.0/39.2/36.0	
42.7	−3.26	[−6.52, +0.00]	0.055	0.44	0.44	42.7/41.6/39.2/41.7	
Block: LLaDA-8B-Instruct MBPP							
33.6	**+10.80**	[+6.20, +15.40]	1.0×10^{-4} §	0.0013§	5.5×10^{-5}	33.6/33.0/34.0/32.5	
39.0	+5.00	[+0.80, +9.20]	0.023	0.20	0.24	39.0/36.9/36.5/35.6	
Block: LLaDA-8B-Instruct HumanEval							
40.5	+0.61	[−7.93, +9.15]	0.97	1	1	40.5/40.5/40.1/42.5	
44.5	−1.22	[−9.15, +6.10]	0.81	1	1	44.5/44.4/44.1/45.2	

D SCORE–STOPPING EXPERIMENTS

D.1 INTERACTION BLOCKS

Table 12 reports the interaction $I = (\text{margin}_{\text{prod}} - \text{margin}_{\text{thr}}) - (\text{head}_{\text{prod}} - \text{head}_{\text{thr}})$, which equals the I of Section 6, for all 13 exploratory four-arm blocks, computed per problem on identical problems and averaged, in accuracy points. Intervals are pointwise 95% percentile-bootstrap intervals from 20,000 joint resamples of problems across the four arms (seed 20260921), not adjusted for multiplicity. The primary raw p is the two-sided bootstrap value $p = \min\{1,\ 2(1 + \min(c_{\le}, c_{\ge}))/(B+1)\}$ with $B = 20{,}000$, where $c_{\le}$ and $c_{\ge}$ count resamples with $I^* \le 0$ and $I^* \ge 0$; its smallest reportable value, $2/(B+1) \approx 1.0\times10^{-4}$, is the resolution of a 20,000-resample bootstrap, not a bound on the population p (marked [§], together with the Holm value derived from it). Holm adjustment is over the family of these 13 blocks. As a sensitivity analysis, an exact paired sign-flip randomization test on the per-problem contrasts, whose null is that each contrast is symmetric about zero, gives the same Holm-significant blocks. A block is flagged [†] when an arm is more than 3.5 NFE from the block target or the arms differ by more than 3.5 NFE. Dream-7B is excluded because it has no margin or learned-score threshold arm. The analysis is exploratory: correction does not account for the earlier operating-point search, blocks share problems and some arms, and HumanEval ($n = 164$) is low powered.

Within-rule differences. The within-rule head–margin differences of the four MBPP blocks form a family of eight contrasts (exact McNemar, Holm over the eight). We call a block a reversal when both of its differences are Holm-significant in this family with opposite signs. Only LLaDA-8B MBPP near 33.6 NFE qualifies (+5.40 under thresholding, Holm $p = 0.0051$; −5.40 under product stopping, Holm $p = 0.015$). On LLaDA-1.5 MBPP near 22.8 NFE, the threshold difference is +8.60 (Holm $p = 8.2\times10^{-7}$) and the product difference +1.40 is unresolved (Holm $p = 1$), an attenuation. At the second MBPP target of each backbone the interaction is unresolved after correction.

Table 13: **Factorial arm registry.** Accuracy (%)@mean NFE of every measured score–rule arm in the 18 exploratory target blocks (two per setting). Dream-7B has no margin or learned-score threshold arm.

Target NFE	Raw, thr.	Raw, prod.	Margin, thr.	Margin, prod.	Head, thr.	Head, prod.
LLaDA-1.5 GSM8K						
37.7	70.58@36.8	75.59@38.5	75.66@38.3	77.26@36.9	73.01@37.6	75.51@37.7
41.8	77.79@39.8	75.59@38.5	78.17@41.0	77.26@36.9	73.01@37.6	76.72@41.8
LLaDA-1.5 MBPP						
22.8	20.00@22.8	30.00@25.6	27.20@22.4	31.80@24.4	35.80@25.6	33.20@22.8
28.0	35.80@27.6	34.40@27.5	36.20@28.4	36.00@26.8	36.60@28.0	37.00@28.0
LLaDA-1.5 HumanEval						
39.7	22.56@41.6	30.49@40.5	30.49@40.9	29.88@39.8	28.66@37.1	23.78@39.7
42.0	22.56@41.6	30.49@40.5	30.49@40.9	29.88@39.8	29.88@43.7	27.44@42.0
LLaDA-8B GSM8K						
37.8	72.10@40.0	70.51@35.0	70.28@36.0	73.16@39.2	66.94@36.0	72.33@37.8
42.7	72.10@40.0	74.91@41.3	74.45@41.7	73.16@39.2	73.09@41.6	75.06@42.7
LLaDA-8B MBPP						
33.6	11.80@32.1	14.40@30.6	16.20@32.5	21.80@34.0	21.60@33.0	16.40@33.6
39.0	19.60@39.9	21.00@38.6	20.40@35.6	23.00@36.5	24.40@36.9	22.00@39.0
LLaDA-8B HumanEval						
40.5	17.68@37.8	29.88@42.1	31.71@42.5	29.27@40.1	23.17@40.5	20.12@40.5
44.5	24.39@44.5	31.10@44.7	31.71@45.2	31.71@44.1	26.83@44.4	28.05@44.5
Dream-7B GSM8K						
30.8	45.34@29.8	60.80@29.0	–	65.13@30.8	–	64.67@30.8
34.0	60.50@34.9	67.63@33.8	–	67.17@33.1	–	68.23@34.0
Dream-7B MBPP						
20.7	37.80@21.7	35.00@21.2	–	38.60@20.9	–	37.60@20.7
25.1	45.00@26.9	46.00@27.6	–	43.40@24.0	–	46.40@25.1
Dream-7B HumanEval						
29.1	10.37@27.8	21.34@27.2	–	25.00@28.8	–	28.05@29.1
48.1	35.98@49.1	43.90@46.0	–	51.83@50.5	–	46.34@48.1

D.2 Factorial arm registry and contrasts

Equal-cost stopping-rule comparison. On LLaDA-1.5 GSM8K, raw-confidence thresholding achieves 70.6% accuracy and raw-confidence product stopping achieves 75.3%, both at 36.8 mean NFE. The gain is 4.7 percentage points (nominal exact McNemar $p = 0.0006$). The stability-head product achieves 75.5% at 37.7 mean NFE, differing from the raw-confidence product by $+0.2$ points (nominal $p = 0.91$).

LLaDA-8B MBPP follow-up protocol. We select a cutoff separately for each combination of head or margin scoring and threshold or product stopping. Selection uses 90 calibration problems (ids 511–600), choosing the configuration closest to 33.0 mean NFE without using accuracy. The selected cutoffs are $\tau = 0.965$ for head threshold, $\rho = 0.92$ for head product, $\tau = 0.37$ for margin threshold, and $\rho = 0.12$ for margin product. These configurations are evaluated unchanged on 374 problems (ids 601–974). The primary interaction uses an exact sign-flip test and a 95% interval from $20{,}000$ joint problem-level bootstrap resamples. The two within-rule head–margin comparisons use exact McNemar tests with Holm correction across those two tests.

D.3 Additional block-size checks

We evaluate three settings at block size 64; LLaDA-1.5 GSM8K, LLaDA-1.5 MBPP, and Dream-7B MBPP. All runs use 256-token continuations and full test splits. Heads are retrained on block-64 labels using the procedure in Appendix A.2, which produces 63 labels per block and 252 per problem. These experiments evaluate the method at another block size, rather than transfer of an unchanged head between block sizes.

Table 14: **Score contrasts in the factorial blocks.** Accuracy difference in points (Holm-adjusted p within the block's family of m contrasts); bold: adjusted $p < 0.05$; † the two arms differ by more than 3.5 mean NFE.

Target NFE	m	Head − raw, prod.	Head − margin, prod.	Head − margin, thr.	Margin − raw, prod.
LLaDA-1.5 GSM8K					
37.7	10	−0.08 (1)	−1.74 (0.62)	−2.65 (0.22)	+1.67 (0.62)
41.8	10	+1.14 (1)	−0.53 (1)†	**−5.16** (2×10^{-4})	+1.67 (0.78)
LLaDA-1.5 MBPP					
22.8	10	+3.20 (0.22)	+1.40 (0.51)	**+8.60** (8×10^{-7})	+1.80 (0.51)
28.0	10	+2.60 (0.77)	+1.00 (1)	+0.40 (1)	+1.60 (1)
LLaDA-1.5 HumanEval					
39.7	9	−6.71 (0.52)	−6.10 (0.52)	−1.83 (1)†	−0.61 (1)
42.0	9	−3.05 (1)	−2.44 (1)	−0.61 (1)	−0.61 (1)
LLaDA-8B GSM8K					
37.8	9	+1.82 (0.66)	−0.83 (0.66)	−3.34 (0.10)	+2.65 (0.21)†
42.7	9	+0.15 (1)	+1.90 (0.71)	−1.36 (1)	−1.74 (0.72)
LLaDA-8B MBPP					
33.6	9	+2.00 (0.28)	**−5.40** (0.01)	**+5.40** (0.005)	**+7.40** (2×10^{-5})
39.0	9	+1.00 (1)	−1.00 (1)	**+4.00** (0.04)	+2.00 (1)
LLaDA-8B HumanEval					
40.5	9	**−9.76** (0.03)	−9.15 (0.06)	−8.54 (0.06)	−0.61 (1)
44.5	9	−3.05 (1)	−3.66 (1)	−4.88 (0.94)	+0.61 (1)
Dream-7B GSM8K					
30.8	4	**+3.87** (0.01)	−0.45 (0.77)	–	**+4.32** (0.004)
34.0	4	+0.61 (1)	+1.06 (1)	–	−0.45 (1)
Dream-7B MBPP					
20.7	4	+2.60 (0.65)	−1.00 (0.69)	–	+3.60 (0.25)
25.1	4	+0.40 (1)	+3.00 (0.42)	–	−2.60 (0.55)†
Dream-7B HumanEval					
29.1	4	+6.71 (0.30)	+3.05 (0.53)	–	+3.66 (0.53)
48.1	4	+2.44 (0.62)	−5.49 (0.24)	–	+7.93 (0.06)†

Training. The LLaDA-1.5 heads use HumanEval labels. Three full heads are trained on three 32-problem samples: draw 0 uses the training problems of H_L, while draws 1 and 2 use those of the first two training-draw heads. The confidence-only head uses draw 0. The Dream-7B full and confidence-only heads use 80 GSM8K problems sampled from the first 400 test problems. Each head is therefore evaluated on tasks different from its training task. All heads follow the training recipe in Appendix A.3.

Additional equal-cost stopping-rule comparison. For the factorial comparisons, cutoffs were chosen by cost alone on the first 50 test problems, after which the full split was evaluated at the setting nearest each target; if the first setting missed the 3.5-NFE window, one additional cutoff was estimated from measured cost points and evaluated. Table 13 lists every measured arm of the 18 target blocks, two per setting. Tables 14 and 15 give the contrasts used to interpret scores and rules, with exact McNemar tests Holm-adjusted within each block's original family of m contrasts. The families of the two LLaDA-1.5 GSM8K and MBPP blocks also contain a contrast with an entropy-budget stopping control (our implementation of the entropy-accumulation principle of EB-Sampler (Ben-Hamu et al., 2025), not the authors' release) that is not discussed here, and the adjusted values are those of the original analysis. Adverse results are retained, for example the head's 9.76-point deficit to raw-confidence product on LLaDA-8B HumanEval near 40.5 NFE (Holm $p = 0.03$). The source package lists all 136 contrasts of these blocks with intervals and win/loss counts.

Cutoffs. We evaluate both reused and retuned cutoffs. On LLaDA-1.5 GSM8K, the reused settings are $\tau = 0.55$ for raw-confidence thresholding, $\rho = 0.85$ for the confidence-only product, and $\rho = 0.90$ for the full-head product. On LLaDA-1.5 MBPP, the confidence-only and full heads reuse $\rho = 0.95$ and $\rho = 0.99$, respectively. On Dream-7B MBPP, raw-confidence thresholding reuses $\tau = 0.65$ and the full head reuses $\rho = 0.95$. Because there is no corresponding block-32 Dream confidence-only run, its initial cutoff $\rho = 0.85$ is borrowed from LLaDA.

Table 15: **Stopping-rule contrasts in the factorial blocks.** Product minus threshold for each score, in points (Holm-adjusted p within the block's family of m contrasts). † The compared configurations differ by more than 3.5 mean NFE.

Target NFE	m	Raw: prod. − thr.	Margin: prod. − thr.	Head: prod. − thr.
LLaDA-1.5 GSM8K				
37.7	10	$+\mathbf{5.00}$ (0.002)	$+1.59$ (0.62)	$+2.50$ (0.29)
41.8	10	-2.20 (0.43)	-0.91 (1)†	$+\mathbf{3.71}$ (0.02)†
LLaDA-1.5 MBPP				
22.8	10	$+\mathbf{10.00}$ (5×10^{-6})	$+4.60$ (0.06)	-2.60 (0.31)
28.0	10	-1.40 (1)	-0.20 (1)	$+0.40$ (1)
LLaDA-1.5 HumanEval				
39.7	9	$+7.93$ (0.37)	-0.61 (1)	-4.88 (0.68)
42.0	9	$+7.93$ (0.37)	-0.61 (1)	-2.44 (1)
LLaDA-8B GSM8K				
37.8	9	-1.59 (0.66)†	$+2.88$ (0.16)	$+\mathbf{5.38}$ (9×10^{-4})
42.7	9	$+2.81$ (0.25)	-1.29 (1)	$+1.97$ (0.62)
LLaDA-8B MBPP				
33.6	9	$+2.60$ (0.27)	$+\mathbf{5.60}$ (0.01)	$-\mathbf{5.20}$ (0.01)
39.0	9	$+1.40$ (1)	$+2.60$ (0.87)	-2.40 (0.87)
LLaDA-8B HumanEval				
40.5	9	$+\mathbf{12.20}$ (0.001)†	-2.44 (1)	-3.05 (1)
44.5	9	$+6.71$ (0.49)	$+0.00$ (1)	$+1.22$ (1)
Dream-7B GSM8K				
30.8	4	$+\mathbf{15.47}$ (7×10^{-21})	–	–
34.0	4	$+\mathbf{7.13}$ (9×10^{-7})	–	–
Dream-7B MBPP				
20.7	4	-2.80 (0.65)	–	–
25.1	4	$+1.00$ (1)	–	–
Dream-7B HumanEval				
29.1	4	$+\mathbf{10.98}$ (0.006)	–	–
48.1	4	$+7.93$ (0.12)	–	–

Retuning selects cutoffs using mean NFE on the first 20 evaluation problems, targeting the cost of the corresponding block-32 comparison without using accuracy. These problems remain in the full-split evaluation. Retuning retains the GSM8K threshold and head cutoffs and the Dream-7B threshold cutoff; the other selected settings appear in Table 16. The GSM8K raw-confidence product sweep uses $\rho \in \{0.20, 0.30, 0.40, 0.50\}$.

Comparisons. On GSM8K, raw-confidence thresholding is compared with both the nearest-cost product configuration and the other product configuration within 3.5 mean NFE. Each learned head is compared with the nearest raw-confidence product configuration. Comparisons outside the 3.5-NFE window are flagged in the table and reported as cutoff comparisons. Holm adjustment covers all 14 contrasts in Table 16, separately from the 13 interaction tests in Section 6.

Results. On GSM8K, raw-confidence product stopping improves accuracy over thresholding by 5.61 percentage points under the nearest-cost pairing (Holm $p = 0.0011$). The other eligible pairing gives a 0.91-point difference (nominal $p = 0.537$), showing sensitivity to the cost pairing. Under product stopping, learned-head differences from raw confidence range from -1.29 to -0.08 points (nominal $p \geq 0.327$).

On LLaDA-1.5 MBPP, the full head improves over the confidence-only head by 6.00 points with reused cutoffs (Holm $p = 0.0029$). After retuning, the difference is 2.20 points, with a 95% interval of $[-0.40, +5.00]$ and nominal $p = 0.144$.

On Dream-7B MBPP, none of the six comparisons is significant after Holm correction ($p \geq 0.605$). The smallest nominal p is 0.0504 for the confidence-only head versus thresholding with the initial cutoffs; this comparison lies outside the matching window. Its bootstrap interval excludes zero, while its exact McNemar test does not reject at 0.05: the interval and test use different procedures. Significance conclusions use the Holm-adjusted exact tests.

Table 16: **Additional checks at block size** 64**.** Accuracy (%) and mean NFE on full test splits with 256-token continuations. Raw denotes raw confidence; Conf.-only and Full denote learned heads. Full d identifies training draw d on LLaDA-1.5. Thr. uses threshold τ; Prod. uses product cutoff ρ. Each contrast is the row's policy minus the policy in "vs."; Δ is in percentage points, with an unadjusted paired 95% problem-bootstrap interval. w/l counts problems solved only by the row's policy / only by its comparator. Tests are two-sided exact McNemar tests; Holm adjustment covers all 14 contrasts. G3 is G1's nearest-cost product partner; G2 is the other eligible partner. †The pair exceeds the 3.5-NFE matching window and is reported as a cutoff comparison.

ID	Score	Rule	Cutoff	Acc.	NFE	vs.	Δ [95% CI]	w/l	p	Holm p
LLaDA-1.5 / GSM8K: stopping rules and learned scores										
G1	Raw	Thr.	0.55	69.67	32.3					
G2	Raw	Prod.	0.20	70.58	30.5	G1	+0.91 [−1.75, +3.49]	165/153	0.537	1.000
G3	Raw	Prod.	0.30	75.28	33.1	G1	+5.61 [+2.88, +8.34]	209/135	7.8×10^{-5}	0.0011
G4	Raw	Prod.	0.40	78.17	36.9					
G5	Raw	Prod.	0.50	79.53	41.3					
G6	Conf.-only	Prod.	0.85	74.75	32.7	G3	−0.53 [−2.96, +1.90]	132/139	0.716	1.000
G7	Full 0	Prod.	0.90	74.00	34.7	G3	−1.29 [−3.64, +1.21]	125/142	0.327	1.000
G8	Full 1	Prod.	0.90	75.21	33.0	G3	−0.08 [−2.50, +2.35]	134/135	1.000	1.000
G9	Full 2	Prod.	0.90	74.00	33.3	G3	−1.29 [−3.79, +1.14]	126/143	0.329	1.000
LLaDA-1.5 / MBPP: reused block-32 cutoffs										
M1	Conf.-only	Prod.	0.95	23.40	22.1					
M2	Full 0	Prod.	0.99	29.40	23.8	M1	+6.00 [+3.00, +9.20]	47/17	2.3×10^{-4}	0.0029
LLaDA-1.5 / MBPP: retuned cutoffs										
M3	Conf.-only	Prod.	0.97	28.40	26.6					
M4	Full 0	Prod.	0.995	30.60	25.9	M3	+2.20 [−0.40, +5.00]	29/18	0.144	1.000
Dream-7B / MBPP: common threshold comparator										
D1	Raw	Thr.	0.65	41.20	22.6					
Dream-7B / MBPP: reused or borrowed cutoffs										
D2	Conf.-only	Prod.	0.85	37.00	17.8	D1	−4.20† [−8.20, −0.20]	42/63	0.0504	0.605
D3	Full	Prod.	0.95	39.60	20.9	D1	−1.60 [−5.40, +2.20]	43/51	0.470	1.000
						D2	+2.60 [−1.40, +6.40]	56/43	0.228	1.000
Dream-7B / MBPP: retuned cutoffs										
D4	Conf.-only	Prod.	0.90	42.60	20.2	D1	+1.40 [−1.80, +4.60]	35/28	0.450	1.000
D5	Full	Prod.	0.97	43.00	26.1	D1	+1.80 [−1.60, +5.40]	45/36	0.374	1.000
						D4	+0.40† [−3.20, +3.80]	41/39	0.911	1.000

D.4 Product transformations

This section formalizes the score dependence of cumulative-product stopping. Unlike thresholding, which depends only on whether individual scores exceed a cutoff, the product rule depends on score magnitudes. Consequently, the numerical cutoff ρ is tied to the scale of the score being accumulated. Throughout, B denotes the block length and C_ρ the corresponding additive budget.

Setup. Fix the confidence ordering and commit the first proposal unconditionally. Let $q_1, \ldots, q_{m-1} \in [0, 1]$ denote the scores of the remaining candidates, where m is the number of masked positions. Product stopping accepts the longest additional prefix satisfying

$$\prod_{j=1}^{k} q_j \geq \rho.$$

Here k counts additional commitments, the empty product for $k = 0$ equals one, and the total commitment count is $1 + k$, consistent with Equation 1. For positive scores, this is equivalent to the additive condition

$$\sum_{j=1}^{k} c_j \leq C_\rho, \qquad c_j = -\log q_j, \qquad C_\rho = -\log \rho, \tag{2}$$

so each accepted position contributes a non-negative cost c_j, while ρ determines the total budget.

Power transforms preserve the policy. For any common $a > 0$, replacing each score by q_j^a and the cutoff by ρ^a leaves every prefix decision unchanged:

$$\prod_{j=1}^{k} q_j^a \geq \rho^a \iff \left(\prod_{j=1}^{k} q_j\right)^a \geq \rho^a \iff \prod_{j=1}^{k} q_j \geq \rho.$$

This remains true for prefixes containing a zero. With identical proposals, numerical conventions, and tie rules, the resulting decoding trajectory is therefore unchanged in exact arithmetic. In the additive view, a common power simply rescales every cost c_j and the budget C_ρ by the same factor.

At a fixed cutoff, only powers preserve all decisions above the cutoff. Let $f : [0,1] \to [0,1]$ be continuous and strictly increasing, and define the induced cost transformation

$$g(c) = -\log f(e^{-c}),$$

which is also continuous and strictly increasing. Fix $\rho \in (0,1)$ and a transformed cutoff ρ', with budgets $C = -\log \rho$ and $C' = -\log \rho'$. Suppose the original and transformed product rules make the same decision for every candidate vector of length at least three.

Because prefix sums are nondecreasing, reaching a prefix of length $k \geq 2$ is equivalent to the k-th prefix satisfying the corresponding budget. Thus, for $k = 2$ and $k = 3$, the acceptance sets

$$\left\{\sum_{j \leq k} c_j \leq C\right\} \quad \text{and} \quad \left\{\sum_{j \leq k} g(c_j) \leq C'\right\}$$

coincide. Continuity implies that their boundaries coincide as well:

$$\sum_j c_j = C \quad \iff \quad \sum_j g(c_j) = C'.$$

For two coordinates satisfying $c_1 + c_2 = C$,

$$g(c_1) + g(C - c_1) = C'.$$

Comparing this identity with the corresponding three-coordinate boundary yields

$$g(u) + g(v) = g(u+v), \qquad u, v \geq 0, \quad u + v \leq C.$$

This is Cauchy's equation on a triangle. Monotonicity of g therefore implies

$$g(c) = ac \qquad \text{for } c \in [0, C],$$

with $a = C'/C > 0$. Hence, at a single cutoff,

$$f(q) = q^a \qquad \text{for } q \in [\rho, 1],$$

and $\rho' = \rho^a$. Below ρ, the transformation is unconstrained by policy equivalence because such a score is rejected by both rules.

If instead the two rules agree for every cutoff $\rho \in (0,1)$, with a corresponding mapping $\rho \mapsto \rho'$, the linear regions overlap for all C_ρ. They must therefore share a common slope a, and letting $\rho \to 0$ gives

$$g(c) = ac \qquad \text{for all } c \geq 0,$$

or equivalently

$$f(q) = q^a \qquad \text{for all } q \in [0,1].$$

Thus, a common power is the only continuous strictly increasing transformation to which product stopping is blind at every cutoff; at a fixed cutoff, it is the only such transformation on the score range $[\rho, 1]$. The statement concerns policy equivalence at every possible state, not equality of outcomes on a particular run. The argument requires three candidate scores after the mandatory first commitment, which is compatible with our block length $B = 32$.

Consequences. Two scores can induce the same ranking while producing different prefix lengths at the same numerical cutoff. A shared ρ is therefore not a shared computational budget. This motivates three choices made throughout the paper: comparisons are made at matched achieved

cost, raw-confidence product is used as the control for the learned product, and changing the score while keeping ρ fixed is treated only as a diagnostic of cutoff transfer.

Whether selecting a new cutoff restores a desired operating point is an empirical question and is not implied by the argument above. The asymmetry with thresholding is also important: under thresholding, any strictly increasing score transformation can be absorbed into a corresponding cutoff wherever an inverse exists, whereas under product stopping this is generally not possible. Finite-precision rounding can still change boundary decisions, so no bitwise implementation equivalence is claimed. Finally, this derivation does not establish that the learned score contains additional information, that it is calibrated, or that changing prefix allocation improves task accuracy.

Table 17: **Local reference disagreement at approximately matched mean commitment counts.** Oracle-guided states; head disagreement in percent; Δ is head minus the named score in percentage points with paired 95% bootstrap intervals over problems (negative favors the head).

Target k	Head (%)	Δ raw confidence	Δ margin	Δ confidence-only control
LLaDA-1.5 HumanEval				
7.1	6.61	-2.25 [-2.66, -1.85]	-1.61 [-1.97, -1.27]	-0.01 [-0.27, $+0.27$]
5.8	4.59	-0.64 [-0.87, -0.41]	-0.51 [-0.71, -0.30]	-0.14 [-0.29, $+0.01$]
LLaDA-8B HumanEval				
7.3	6.57	-2.63 [-3.04, -2.22]	-1.89 [-2.25, -1.54]	-0.11 [-0.41, $+0.20$]
5.9	4.51	-0.64 [-0.90, -0.41]	-0.51 [-0.70, -0.32]	-0.04 [-0.22, $+0.13$]
LLaDA-1.5 GSM8K				
9.5	4.13	-1.79 [-2.12, -1.47]	-1.39 [-1.69, -1.09]	-0.06 [-0.27, $+0.16$]
7.9	2.45	-0.55 [-0.75, -0.37]	-0.37 [-0.53, -0.22]	-0.17 [-0.31, -0.04]
LLaDA-1.5 MBPP				
9.4	7.95	-2.69 [-3.09, -2.29]	-1.96 [-2.30, -1.63]	$+0.15$ [-0.16, $+0.47$]
7.9	5.26	-0.80 [-1.06, -0.54]	-0.52 [-0.74, -0.32]	$+0.10$ [-0.08, $+0.29$]

E Local stability and task accuracy

E.1 Common-state local-agreement diagnostic

Table 17 reports the complete common-state comparison. At approximately matched mean commitment counts, learned stability yields lower reference disagreement than raw confidence and margin in all eight evaluated settings. Its differences from the learned confidence-only control are small.

E.2 Oracle-guided states and count matching

Oracle-guided states. For each problem we first compute the sequential one-token-per-step reference continuation $y^\star$ from the initial state, then generate a guided trajectory block by block. At each state the backbone is evaluated on the current masked set; every masked position of the active block is recorded with its proposal, features, and stability label (proposal equal to $y^\star$ at that position). The trajectory then advances by writing the reference token at every position whose proposal already agrees with $y^\star$, or at the highest-confidence position if none agrees. A state is one such backbone evaluation, identified by problem and step, and all masked positions of the active block are stored (no subsampling). These are therefore oracle-guided diagnostic states with labels defined against the sequential reference, not states visited by the sequential decode or by a deployed decoder; committing about 10–13 tokens per state, a problem contributes about 19–25 states rather than 256. The two HumanEval cells use 132 problems after excluding the 32 head-training problems; GSM8K and MBPP use 150 problems each. This yields 3,282, 3,209, 2,884, and 3,174 states for LLaDA-1.5 HumanEval, LLaDA-8B HumanEval, LLaDA-1.5 GSM8K, and LLaDA-1.5 MBPP, respectively. Unless noted otherwise, diagnostics use raw task prompts (zero-shot, no chat template). The same HumanEval-trained LLaDA predictor is used unchanged in all four cells.

Matching commitment volume. All scores use confidence ordering and product stopping with the first candidate always committed. Each score receives its own cutoff so that comparisons approximately match mean commitment count. Cutoffs are selected by five-fold problem-level cross-fitting: each fold is evaluated using a cutoff fitted by bisection on the other four folds. Target volumes are the head's mean counts at $\rho = 0.95$ and 0.99, weighted by problem and fixed before disagreement is computed; achieved counts are averaged over states and are 5–9% above the targets, closely matched across scores, as shown in Table 18. For the three settings beyond LLaDA-1.5 HumanEval, the targets and reporting protocol were fixed before collection. Equal per-state prefix lengths would select identical candidates, because every score shares the confidence order.

Metric, control, and uncertainty. At each state, disagreement is the fraction of committed proposals that differ from the sequential-reference tokens at those positions, averaged first within each problem and then across problems. The confidence-only control has confidence and log-confidence as inputs (4,417 parameters) and was trained on the same 32 problems, labels, and recipe as H_L;

Table 18: **Cross-fitted cutoffs and achieved counts.** Range of the five fold-specific cutoffs ρ for each score, and achieved mean commitment counts (head / raw confidence / margin / confidence-only control).

Target k	Head ρ	Raw ρ	Margin ρ	Control ρ	Achieved k
LLaDA-1.5 HumanEval					
7.1	0.911–0.934	0.270–0.318	0.114–0.150	0.884–0.911	7.47 / 7.43 / 7.44 / 7.46
5.8	0.984–0.988	0.561–0.614	0.400–0.467	0.973–0.977	6.13 / 6.11 / 6.11 / 6.13
LLaDA-8B HumanEval					
7.3	0.905–0.923	0.238–0.279	0.094–0.124	0.868–0.893	7.80 / 7.74 / 7.75 / 7.78
5.9	0.982–0.986	0.535–0.589	0.372–0.441	0.970–0.976	6.34 / 6.30 / 6.31 / 6.32
LLaDA-1.5 GSM8K					
9.5	0.908–0.927	0.255–0.292	0.098–0.124	0.894–0.913	10.01 / 9.96 / 9.97 / 9.99
7.9	0.985–0.988	0.592–0.634	0.427–0.482	0.976–0.978	8.36 / 8.34 / 8.36 / 8.35
LLaDA-1.5 MBPP					
9.4	0.861–0.907	0.174–0.232	0.055–0.091	0.822–0.870	10.13 / 10.07 / 10.08 / 10.09
7.9	0.978–0.985	0.501–0.586	0.333–0.433	0.965–0.974	8.60 / 8.57 / 8.58 / 8.59

Table 19: **The local advantage on states reached by parallel decoding.** Count-matched LLaDA-1.5 GSM8K diagnostic on oracle-guided states (five-shot prompts) and on states visited by two parallel decoders; target volumes are each population's head counts at $\rho = 0.95$ and 0.99 and therefore differ. Head disagreement and paired differences are in percentage points; negative differences favour the head. Brackets give paired 95% bootstrap intervals over problems.

State source	Target k	Head	Head − raw confidence	Head − margin
Oracle-guided states (five-shot prompts)	9.8	5.86%	−2.10 [−2.49, −1.71]	−1.57 [−1.91, −1.25]
	8.3	3.42%	−0.66 [−0.92, −0.41]	−0.54 [−0.75, −0.35]
Raw-confidence product ($\rho = 0.30$)	7.2	3.22%	−3.34 [−3.69, −2.98]	−2.47 [−2.80, −2.15]
	6.0	0.91%	−1.07 [−1.27, −0.87]	−0.71 [−0.88, −0.55]
Deployed head product ($\rho = 0.90$)	6.4	3.13%	−3.84 [−4.28, −3.40]	−2.67 [−3.00, −2.33]
	5.3	0.79%	−1.26 [−1.46, −1.06]	−0.70 [−0.86, −0.55]

it differs from raw confidence only in the value fed to the product rule, does not match the full head's capacity, and was not run on the parallel-decoder or fork populations. Paired 95% intervals use $20{,}000$ problem-level bootstrap resamples. The 16 head-versus-raw and head-versus-margin contrasts form one Holm family, all of which survive; the eight full-head-versus-control contrasts form a separate family, fixed before the control was run, none of which survives (smallest adjusted $p = 0.070$).

E.3 Prompt-matched and parallel-decoder states

Table 19 repeats the diagnostic on LLaDA-1.5 GSM8K (150 problems, five-shot evaluation prompts) on oracle-guided states and on the states visited by two parallel decoders, raw-confidence product at $\rho = 0.30$ and the deployed head at $\rho = 0.90$. Each recorded parallel-decoder state is one step of the evaluated decoder: for all 150 problems, the number of recorded states equals the logged NFE and the per-step commitment counts equal the logged ones. Labels come from restarting one-token-per-step decoding from the recorded state for the current block, whereas oracle-guided labels use the original reference, and each population's targets are its own head counts at $\rho = 0.95$ and 0.99, so the populations are compared descriptively. The raw-product trajectories yield $5{,}523$ states and $103{,}538$ labelled positions, and the head trajectories $5{,}689$ states and $98{,}322$ positions; achieved counts differ between scores by 0.03–0.06 tokens per state. These results show that the local advantage is not an artifact of oracle-guided states; they do not compare the future states that different commitments would induce.

Table 20: **Validation-selected evaluations: outcomes.** Accuracy (%)@achieved mean NFE on the evaluation problems; the first target of each backbone is primary. Dashes mark policies not run on that backbone.

Backbone	Target NFE	Head, prod.	Raw conf., prod.	Margin, prod.	Raw conf., thr.	ADAS
GSM8K (450 problems)						
LLaDA-1.5	30	66.00@31.1	70.22@30.7	74.44@33.2	63.56@32.4	–
	37	75.78@37.3	76.89@36.4	77.11@36.1	72.89@36.7	–
LLaDA-8B	38	71.78@37.7	73.33@37.8	74.22@36.6	70.89@36.8	–
	46	76.00@42.6	75.33@45.6	77.78@45.8	74.44@44.3	–
Dream-7B	29	60.22@30.6	60.00@29.6	63.78@31.4	44.67@30.3	–
	36	63.33@34.5	66.00@36.1	67.78@35.4	60.22@34.9	–
MBPP (374 problems)						
LLaDA-1.5	23	39.84@22.69	38.77@24.89	39.84@23.33	–	35.83@20.62
	28	46.26@26.52	41.44@26.72	43.32@26.94	–	44.39@27.30
Dream-7B	20	37.17@19.37	36.10@19.29	37.97@21.27	31.82@19.85	–
	30	54.28@28.53	51.07@28.46	52.41@27.02	56.42@30.77	–

E.4 VALIDATION-SELECTED EVALUATIONS

GSM8K design. Calibration uses GSM8K `train`[0:100] and evaluation `train`[100:550] (450 problems). Neither overlaps the canonical test split, and no head was trained on GSM8K `train`. Product cutoffs are selected from $\{0.10, 0.20, 0.30, 0.40, 0.50, 0.70, 0.85, 0.90, 0.95, 0.99\}$ and confidence thresholds from $\{0.45, 0.50, 0.55, 0.60, 0.65, 0.70, 0.80, 0.90\}$; before evaluation, margin cutoff $\rho = 0.05$ was added for LLaDA-1.5 because the grid did not reach the aggressive target. For each backbone, policy, and target, the setting with calibration NFE closest to the target is selected without inspecting accuracy, and all settings were fixed before decoding the evaluation problems. The harness draws five-shot exemplars from GSM8K `train` and removes only the current evaluation problem from its prompt, so an evaluation problem can appear as an exemplar for another. Backbone pretraining exposure is unknown, and prior use of the GSM8K `train` split in a companion project could not be verified from accessible records.

MBPP design. Evaluation uses MBPP task ids 601–974 (374 problems), disjoint from the historical test split (11–510), the 90 validation problems used for selection (511–600), and the ten-problem prompt split (1–10), of which ids 2–4 supply the three fixed few-shot examples. LLaDA-1.5 and Dream-7B use the same evaluation set, so their results are not independent, and later evaluations, including the LLaDA-8B follow-up, reuse it; prior use of ids 511–974 in a companion project could not be verified from accessible records. Settings are selected by validation NFE alone, targeting 23 and 28 NFE on LLaDA-1.5 and 20 and 30 on Dream-7B. Calibration grids were, for LLaDA-1.5, head product $\rho \in \{0.80, 0.85, 0.88, 0.90, 0.92, 0.95, 0.97, 0.99\}$, raw-confidence product $\rho \in \{0.30, 0.35, 0.40, 0.45, 0.50, 0.55, 0.75, 0.80, 0.85\}$, and margin product $\rho \in \{0.15, 0.20, 0.25, 0.30, 0.35, 0.40, 0.60, 0.70, 0.80\}$, plus the two ADAS values $\gamma \in \{3, 1.5\}$, which come from earlier exploratory runs on the MBPP test split and were assigned to the targets by nearest calibration NFE; for Dream-7B, head product $\rho \in \{0.85, 0.88, 0.90, 0.92, 0.96, 0.97, 0.98\}$, raw-confidence product $\rho \in \{0.1, 0.2, \ldots, 0.6\}$, margin product $\rho \in \{0.02, 0.05, 0.10, 0.15, 0.20, 0.30\}$, and raw-confidence threshold $\tau \in \{0.40, 0.50, 0.55, 0.60, 0.65, 0.70\}$. Three selected settings lie at a grid maximum: LLaDA-1.5 head $\rho = 0.99$ and Dream-7B margin $\rho = 0.30$ and threshold $\tau = 0.70$. Tables 20 and 21 give the outcomes and the selected settings for both tasks.

Tests. Accuracy uses exact paired McNemar tests, and paired 95% intervals for accuracy and NFE use 20,000 problem-level bootstrap resamples. On GSM8K, the prespecified primary family compares head product with raw-confidence product at the aggressive target on each backbone, Holm-corrected over the three backbones; the other contrasts are secondary and nominal. On MBPP, Holm correction is applied separately to the six head-versus-comparator tests of each backbone. Table 22 lists the head contrasts. Among the secondary rule contrasts, raw-confidence product minus threshold on GSM8K is $+6.67$ and $+4.00$ points on LLaDA-1.5 ($p = 0.012$ and 0.085), $+2.44$ and $+0.89$ on LLaDA-8B ($p = 0.31$ and 0.73), and $+15.33$ and $+5.78$ on Dream-7B ($p = 9.5\times10^{-8}$ and 0.015) at the aggressive and second budgets; on Dream-7B MBPP it is $+4.28$ points at the primary target ($p = 0.125$) and -5.35 at the secondary target ($p = 0.008$), where product stopping uses 2.31

Table 21: **Validation-selected evaluations: selected settings.** Cutoff ρ, threshold τ, or ADAS parameter γ, with calibration mean NFE in parentheses; accuracy is not used for selection.

Backbone	Target	Head, product	Raw conf., product	Margin, product	Raw conf., threshold	ADAS
GSM8K (calibration on `train[0:100]`*)*						
LLaDA-1.5	30	$\rho = 0.70$ (30.4)	$\rho = 0.10$ (30.6)	$\rho = 0.05$ (32.1)	$\tau = 0.45$ (32.3)	–
LLaDA-1.5	37	$\rho = 0.90$ (36.5)	$\rho = 0.30$ (35.5)	$\rho = 0.10$ (34.8)	$\tau = 0.55$ (37.2)	–
LLaDA-8B	38	$\rho = 0.90$ (37.8)	$\rho = 0.30$ (37.4)	$\rho = 0.10$ (36.6)	$\tau = 0.55$ (37.8)	–
LLaDA-8B	46	$\rho = 0.95$ (42.3)	$\rho = 0.50$ (46.7)	$\rho = 0.30$ (47.1)	$\tau = 0.65$ (44.0)	–
Dream-7B	29	$\rho = 0.85$ (30.1)	$\rho = 0.20$ (29.2)	$\rho = 0.10$ (30.1)	$\tau = 0.45$ (29.5)	–
Dream-7B	36	$\rho = 0.90$ (33.7)	$\rho = 0.40$ (35.0)	$\rho = 0.20$ (35.6)	$\tau = 0.55$ (35.0)	–
MBPP (calibration on ids 511–600*)*						
LLaDA-1.5	23	$\rho = 0.95$ (22.2)	$\rho = 0.40$ (25.0)	$\rho = 0.20$ (24.1)	–	$\gamma = 3$ (21.0)
LLaDA-1.5	28	$\rho = 0.99$ (28.7)	$\rho = 0.50$ (27.5)	$\rho = 0.35$ (28.1)	–	$\gamma = 1.5$ (29.0)
Dream-7B	20	$\rho = 0.88$ (20.2)	$\rho = 0.20$ (20.0)	$\rho = 0.10$ (19.9)	$\tau = 0.50$ (20.4)	–
Dream-7B	30	$\rho = 0.97$ (28.9)	$\rho = 0.50$ (29.1)	$\rho = 0.30$ (30.6)	$\tau = 0.70$ (30.6)	–

Table 22: **Validation-selected evaluations: head contrasts.** Head minus the named policy at each target; accuracy and NFE differences with paired 95% bootstrap intervals; p from exact McNemar tests. Holm-adjusted values are given only within each family.

Contrast	Target	Δ accuracy, 95% CI	Δ NFE, 95% CI	p	Holm p
GSM8K; Holm over the three primary tests					
LLaDA-1.5					
Head − raw prod.	30 (prim.)	−4.22 [−9.33, +0.67]	+0.40 [−0.34, +1.13]	0.118	0.355
Head − raw prod.	37	−1.11 [−4.89, +2.67]	+0.87 [+0.06, +1.68]	0.657	–
Head − margin prod.	30	−8.44 [−12.89, −4.00]	−2.10 [−2.94, −1.26]	3.3×10^{-4}	–
Head − margin prod.	37	−1.33 [−5.11, +2.44]	+1.18 [+0.38, +1.98]	0.572	–
LLaDA-8B					
Head − raw prod.	38 (prim.)	−1.56 [−5.78, +2.67]	−0.04 [−0.77, +0.68]	0.538	1
Head − raw prod.	46	+0.67 [−2.89, +4.22]	−2.97 [−3.75, −2.17]	0.81	–
Head − margin prod.	38	−2.44 [−6.67, +1.78]	+1.18 [+0.46, +1.92]	0.31	–
Head − margin prod.	46	−1.78 [−5.33, +1.78]	−3.14 [−3.95, −2.33]	0.382	–
Dream-7B					
Head − raw prod.	29 (prim.)	+0.22 [−4.67, +5.11]	+1.03 [+0.40, +1.68]	1	1
Head − raw prod.	36	−2.67 [−7.11, +1.78]	−1.52 [−2.19, −0.87]	0.281	–
Head − margin prod.	29	−3.56 [−8.00, +0.89]	−0.81 [−1.42, −0.20]	0.141	–
Head − margin prod.	36	−4.44 [−8.67, +0.00]	−0.86 [−1.52, −0.21]	0.0569	–
MBPP; Holm over the six tests of each backbone					
LLaDA-1.5					
Head − raw confidence	23	+1.07 [−2.94, +5.08]	−2.20 [−3.47, −0.98]	0.689	1.000
Head − margin	23	+0.00 [−3.74, +3.48]	−0.65 [−1.63, +0.35]	1.000	1.000
Head − ADAS	23	+4.01 [−0.27, +8.56]	+2.07 [+0.98, +3.18]	0.096	0.401
Head − raw confidence	28	+4.81 [+1.34, +8.29]	−0.20 [−1.42, +1.00]	0.0096	0.057
Head − margin	28	+2.94 [0.00, +5.88]	−0.42 [−1.55, +0.67]	0.080	0.401
Head − ADAS	28	+1.87 [−0.80, +4.55]	−0.78 [−1.78, +0.20]	0.230	0.689
Dream-7B					
Head − raw confidence	20	+1.07 [−4.01, +6.15]	+0.07 [−0.81, +0.94]	0.755	1.000
Head − margin	20	−0.80 [−5.88, +4.01]	−1.90 [−2.94, −0.88]	0.832	1.000
Head − raw confidence threshold	20	+5.35 [+0.27, +10.43]	−0.48 [−1.39, +0.43]	0.047	0.282
Head − raw confidence	30	+3.21 [−1.07, +7.49]	+0.07 [−1.62, +1.76]	0.175	0.876
Head − margin	30	+1.87 [−1.87, +5.61]	+1.51 [+0.13, +2.89]	0.401	1.000
Head − raw confidence threshold	30	−2.14 [−6.15, +1.87]	−2.24 [−3.86, −0.61]	0.358	1.000

fewer mean NFE. The change in the head's MBPP advantage over raw-confidence product between budgets is also unresolved: +3.74 points on LLaDA-1.5 (95% interval $[-1.07, +8.56]$, $p = 0.142$) and +2.14 on Dream-7B ($[-4.55, +8.82]$, $p = 0.556$).

E.5 Single-decision rollouts

We replay the raw-confidence ($\rho = 0.30$) and deployed-head ($\rho = 0.90$) product decoders on the 150 GSM8K problems of Appendix E.3; replayed NFE and correctness match the original evaluation for all 150 problems. At each recorded state, the head and raw-confidence prefix lengths are computed

with the cross-fitted cutoffs that match their mean commitment counts on these states. States where the two lengths differ are candidate forks, and up to twelve are sampled per problem, evenly along the trajectory. One branch commits the head's prefix and the other raw confidence's; both then continue with the base policy that produced the state. This yields $1{,}733$ forks from raw-product trajectories and $1{,}736$ from head trajectories. At the sampled forks, the head commits means of 6.73 and 6.43 tokens, against 6.66 and 6.21 for raw confidence. Immediate disagreement counts committed tokens at the fork that differ from the reference obtained by restarting one-token-per-step decoding from the fork for the current block; final correctness is the task outcome of each completed branch. Intervals use $20{,}000$ bootstrap resamples clustered by problem, because forks from the same problem are not independent.