BETTER SELECTOR, BETTER DECODER? EVALUATING TOKEN SELECTION IN DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Parallel diffusion decoding can accelerate text generation by filling several masked positions per forward pass. Its accuracy depends on both which positions are selected and how many are filled at once. Adaptive decoders couple these decisions, because earlier selections shape the context used to determine later counts. Changing a selector can therefore also change its schedule of per-step counts, while fixing the schedule leaves unclear whether the measured advantage generalizes. We record the per-step counts that adaptive decoders choose and let different selectors fill exactly these counts. As a control, we spread the same positions evenly over the same number of forward passes. On GSM8K in both LLaDA and Dream, DAPD's selector is more accurate than confidence when both use DAPD's counts. By contrast, confidence is more accurate when both use counts from a confidence-threshold decoder. We also let a different selector fill the first half of the output positions. The original decoder then completes the answer more accurately by choosing new counts than by reusing its recorded counts. These findings motivate the Decision-Value Protocol. It recommends evaluating a selector together with the rule that sets its counts, choosing counts from the current text, and choosing complete decoders by accuracy and measured latency. In a later study on ASDiv, following the protocol selects a decoder that is 9.38 points more accurate than the one a selector comparison favors.

1 INTRODUCTION

Masked diffusion language models generate text by repeatedly predicting tokens at masked positions (Nie et al., 2025; Ye et al., 2025). Unlike standard left-to-right generation, this process can fill several positions in one model evaluation, reducing the number of evaluations needed to produce a sequence. Tokens filled in the same step cannot condition on one another, so filling dependent positions together can produce inconsistent choices (Kang et al., 2026; Zhang et al., 2026). A decoder therefore needs to choose both how many positions to fill at once and which positions to fill.

The sequence of per-step counts is the *schedule*, and the rule that chooses positions is the *selector*. A common way to compare selectors is to fix the schedule and change only the selection rule (Ben-Hamu et al., 2025; Asano et al., 2026; Kang et al., 2026; Kim et al., 2025a). Selector rankings can change even between hand-designed schedules (Luxembourg et al., 2026; Sahin et al., 2026). With adaptive decoders, the selector also helps determine the schedule, because earlier selections change the text used to choose later counts (Wu et al., 2025; Ben-Hamu et al., 2025; Kim et al., 2026; 2025b). This raises three questions. Does a selector's advantage persist when another decoder supplies the counts? After a different selector changes the generated text, do counts recorded on the original text still work as well as counts chosen anew? And if one selector beats another under the same counts, does that advantage carry over to the accuracy and speed of a complete decoder, which chooses both its positions and its counts?

The same selector can win under one decoder's counts and lose under another's. On the full GSM8K (Cobbe et al., 2021) test set in LLaDA and Dream, we compare confidence with the selector of DAPD, a recent dependency-aware parallel decoder (Kim et al., 2026). Both selectors run under

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: **Counts change selector rankings and decoder choices.** (a) DAPD minus confidence accuracy, in percentage points, on 1,319 GSM8K problems. Budgets differ across count sources. (b) Mean accuracy of the sources' own selectors (solid) and the other selectors (dashed), averaged over four count sources on 256 problems per task. Balancing preserves every problem's forward passes in every block. Colors identify models in (a,b). (c) On Dream ASDiv, the confidence and entropy selectors use the entropy rule's counts on the left. The complete C95 decoder and the entropy rule use their own counts on the right. Both comparisons share the entropy rule's run. Whiskers are descriptive 95% bootstrap intervals for gaps in (a) and accuracies in (b,c). Paired contrasts are in Appendices B and C. Dream uses one 256-token block in (b,c).

counts recorded from DAPD and from C95, a decoder with confidence threshold 0.95. The decoder that supplies the counts is the *count source*. Both selectors receive the same counts within each comparison, and we adapt the DAPD selector to fill exactly these counts. In both models, the DAPD selector is more accurate under DAPD's counts, whereas confidence is more accurate under C95's counts (Finding 1, Figure 1a). This reversal also holds among problems on which the two count sources use similar forward-pass budgets (Section 4.1).

With two more decoders, KLASS and an entropy rule, the own selectors outperform the other selectors when averaged over all four count sources. We compare all four selectors under all four decoders' counts on 256 problems from each of GSM8K and ASDiv (Miao et al., 2020). To change how the counts are allocated without changing the budget, we also spread each block's positions evenly over its recorded number of forward passes. These *balanced* counts shrink the average lead, to near zero on ASDiv in both models (Finding 1, Figure 1b and Section 4.2).

Recorded counts can also become less useful when the generated text changes. A simpler selector replaces a decoder's own selector for the first half of the positions. The decoder then continues from two identical copies of the resulting text and decoding history, once with its recorded counts and once choosing new counts. The new counts give higher accuracy in all six settings (Finding 2, Section 4.3).

A selector's advantage under the same counts need not carry over to its complete decoder (Finding 3). On LLaDA GSM8K, confidence trails the DAPD selector under DAPD's counts, yet the complete C95 decoder matches the observed accuracy of DAPD-Staged with 31.3% lower latency (Section 5.1).

The three findings motivate the Decision-Value Protocol (DVP), with one practice for each: evaluate a selector together with its count rule, choose counts from the current text, and choose complete decoders by their accuracy and measured latency (Protocol 1). We wrote the protocol down before running the four-selector study and then applied it to that study. On Dream ASDiv, the entropy selector beats confidence under the entropy rule's counts, which would favor the entropy rule if used to choose a complete decoder. Comparing the complete decoders, as the protocol asks, selects C95 instead, which is 9.38 points more accurate (Figure 1c).

Contributions.

- We design a controlled evaluation that varies one decision while holding the other fixed. Selectors fill identical counts, either recorded from an adaptive decoder or balanced over the same forward passes, and we compare continuations with recorded or new counts starting from identical changed text (Sections 3 and 4.3).
- We find that selector rankings depend on the count source (Finding 1), that counts chosen anew beat recorded counts after the text changes (Finding 2), and that a selector's advantage under the same counts need not carry over to complete decoders (Finding 3).
- We turn these findings into DVP and apply it to the four-selector study. The accuracy gaps under shared counts often differ from those between complete decoders. On Dream ASDiv, using the selector ranking to choose a complete decoder would select the less accurate candidate, which DVP avoids (Section 5.2).

2 Related work

Selectors compared under given counts. Confidence, entropy and margin are compared with one position per step (Ben-Hamu et al., 2025, Fig. 2), and so are the confidence and KL orderings of KLASS (Kim et al., 2025b, Table 5). Where-to-Unmask learns a ranking of positions from an oracle that uses the margin of the ground-truth token (Asano et al., 2026). Other comparisons fill a fixed number of positions per step (Kim et al., 2025a; Kang et al., 2026). DPRM replaces the ordering module while retaining the host decoder's budget mechanism (Bu et al., 2026). Our comparisons instead reuse the realized counts recorded from another decoder's trajectory.

Unmasking order in masked image generation. Masked image generators face the same two decisions. MaskGIT reveals a scheduled number of tokens per step and chooses them by confidence (Chang et al., 2022). Hayakawa et al. (2026) show that this confidence-based choice acts as implicit temperature sampling, and study adaptive rules for choosing which positions to unmask. The Halton scheduler replaces the confidence-based choice with a quasi-random, low-discrepancy spatial order and reports lower FID than the confidence scheduler (Besnier et al., 2025). These studies compare position rules under a shared schedule of how many tokens each step reveals, whereas in the adaptive decoders we study the selector also shapes the schedule.

Selector rankings across schedules. In Plan for Speed, confidence and random selection reverse their ranking between fixed-count and dilated schedules (Luxembourg et al., 2026, Table 3). The ranking of ADAS against confidence also changes with the number of positions filled per step on Dream GSM8K (Sahin et al., 2026). The counts in our comparisons instead come from adaptive decoders and depend on earlier selections. Their balanced counterparts keep each block's number of forward passes.

Dependencies, complete decoders and cost. Analyses of parallel decoding link generation errors to dependencies among positions filled together (Zhang et al., 2026; Kang et al., 2026). Adaptive decoders use confidence, entropy, estimated dependencies or distribution stability to decide what to fill (Wu et al., 2025; Ben-Hamu et al., 2025; Kim et al., 2026; 2025b). Fast-dLLM++ picks the set of positions to fill from the sorted confidence profile, so it chooses the positions and their number together (Kasa et al., 2026). Caching and pruning change the cost of each forward pass (Wu et al., 2025; Zuo et al., 2026).

Controlled comparisons. Rankings of machine learning methods can depend on evaluation choices. Recurrent language-model architectures ranked differently after each received the same large-scale hyperparameter search (Melis et al., 2018), and many GAN variants reached similar scores with enough tuning and restarts (Lucic et al., 2018). For adaptive decoding, the count source is one such choice. We compare selectors under recorded counts from different decoders and balance those counts at the same budget within each source.

3 COMPARING SELECTORS UNDER RECORDED COUNTS

Selectors and schedules. In the main two-selector comparisons, the model fills 256 masked positions in four blocks of $L = 64$, from left to right. Let h_t denote the partly filled sequence and the decoding history before step t. At each step, the frozen model f_θ predicts a token $\hat{y}_{t,i}$ and a score $s_{t,i}$, such as confidence, for every masked position i. Among the masked positions of the current block, a *count rule* $\mathcal{K}$ then chooses how many to fill, and a *selector* σ chooses which ones:

$$(\hat{y}_t, s_t) = f_\theta(h_t), \quad k_t = \mathcal{K}(h_t, s_t), \quad W_t = \sigma(h_t, s_t, k_t), \quad h_{t+1} = \mathrm{Update}(h_t, W_t, \hat{y}_t). \tag{1}$$

The decoders we study never change a filled position afterwards (Nie et al., 2025; Wu et al., 2025). The counts $\mathbf{k} = (k_1, \ldots, k_T)$ of a block form its schedule, and T is the number of forward passes that the block uses.

Under a given schedule, write $A(\sigma, \mathbf{k})$ for the accuracy of selector σ when it fills exactly the counts $\mathbf{k}$, which now give a count for every step of every problem. The accuracy gap between two selectors under $\mathbf{k}$ is

$$G(\mathbf{k}) = A(\sigma_1, \mathbf{k}) - A(\sigma_2, \mathbf{k}). \tag{2}$$

The ranking carries over from $\mathbf{k}_0$ to $\mathbf{k}_1$ when $G(\mathbf{k}_0)$ and $G(\mathbf{k}_1)$ have the same sign. An adaptive count rule sets k_t from the sequence h_t produced by its own selector (Equation (1)).

Recorded and balanced comparisons. A separate adaptive-decoder run records counts $\mathbf{k}^{\mathrm{rec}}$ for each block b of each problem x, using T_{xb} forward passes per block (Appendix B). A selector following this schedule fills k^{rec}_{xbt} positions at step t. The balanced schedule spreads the block's L positions evenly over the same steps:

$$k^{\mathrm{bal}}_{xbt} = \lfloor L/T_{xb} \rfloor + \mathbf{1}\{t \le L \bmod T_{xb}\}, \qquad t = 1, \ldots, T_{xb}. \tag{3}$$

Both schedules use exactly the same number of forward passes in every block of every problem. Budgets can differ between count sources.

Under either schedule, *confidence* (C) fills the k_t most confident positions (Nie et al., 2025). The *DAPD selector* (D) (Kim et al., 2026) starts from the set of positions that DAPD proposes to fill. Each position has a node score that combines its normalized dependency score and normalized confidence by multiplication. If the proposal exceeds k_t, we keep its highest-scoring positions. Otherwise, we keep the proposal and add the highest-scoring remaining positions until k_t are filled. Ties go to the earlier position, and token predictions do not change. Reusing DAPD's own counts requires no additions or removals and reproduces its positions and final tokens exactly. The two selectors under the two schedules give four runs. All four share DAPD's numerical implementation, including graph computation and precision.

For the confidence–DAPD comparison, we write $G = A(D) - A(C)$, so a positive gap means that the DAPD selector is more accurate. The *gain* of selector σ from recorded rather than balanced counts is

$$V_\sigma = A(\sigma_{\mathrm{rec}}) - A(\sigma_{\mathrm{bal}}), \tag{4}$$

and the difference between these gains is the change in the accuracy gap:

$$I = V_D - V_C = G(\mathbf{k}^{\mathrm{rec}}) - G(\mathbf{k}^{\mathrm{bal}}) = \big[A(D_{\mathrm{rec}}) - A(C_{\mathrm{rec}})\big] - \big[A(D_{\mathrm{bal}}) - A(C_{\mathrm{bal}})\big]. \tag{5}$$

A positive I means that the recorded counts help D more than C, and a negative I that they help C more. A ranking reversal requires G itself to change sign between two decoders' recorded counts. For the KLASS selector (Kim et al., 2025b), the same contrast is $I = V_K - V_C$.

Decoders and experimental setup. The count sources are DAPD, KLASS, C95 and an entropy-threshold decoder (Table 1). DAPD (Kim et al., 2026) fills positions that a dependency graph built from attention marks as independent. We use its released code, in the variant DAPD-Staged on LLaDA, which adds a staged confidence threshold, and DAPD-Direct on Dream, which also fills every position of confidence 1.0 directly. KLASS (Kim et al., 2025b) supplies selector K, adapted by the same rule, ranking positions by confidence. C95 is a confidence threshold as in Fast-dLLM (Wu et al., 2025), run in the DAPD implementation rather than as the full Fast-dLLM system. The entropy rule is a simple threshold added for the four-selector study. Under its own decoder's counts, each selector reproduces that decoder exactly.

Table 1: **Adaptive count sources.** Each decoder sets its counts while it decodes, and its own selector chooses the positions. Implementation is the code in which each selector runs.

Decoder	Each step fills	Own selector	Implementation
DAPD	positions its dependency graph marks as independent	DAPD (D)	DAPD's
KLASS	positions whose predictions are confident and stable	KLASS (K)	KLASS's
C95	positions with confidence ≥ 0.95, else the top one	Confidence (C)	DAPD's
Entropy rule	positions with entropy ≤ 0.1 nats, else the lowest	Entropy (E)	DAPD's

LLaDA-8B-Instruct (Nie et al., 2025) and Dream-v0-Instruct-7B (Ye et al., 2025) run with BF16 weights. Every run decodes deterministically at batch size one and generates 256 positions, in four blocks of 64 unless stated otherwise. The main task is GSM8K (Cobbe et al., 2021), with all 1,319 test problems, DAPD's five-shot prompts and accuracy of the final answer. Other studies use MATH500, ARC-Challenge, ASDiv and held-out GSM8K and MATH problems (Appendix A). Confidence intervals use paired resampling of problems, except for the exact intervals in the continuation study. Each study fixed the level of its main intervals in advance from its number of main tests, between 95% and 99.5%. All other intervals are descriptive 95% intervals, and a Bonferroni correction across studies serves as a sensitivity check (Appendix A).

4 Selector comparisons under different counts

4.1 Two selectors swap ranks between two decoders' counts

The DAPD selector is more accurate than confidence under DAPD's counts, but confidence is more accurate under C95's counts. On LLaDA, the DAPD selector goes from leading by 25.25 points to trailing by 12.81 points. Dream shows the same reversal (Figure 1a and Table 2). In both models, the winner is the selector whose own decoder produced the counts.

Balancing each block's counts across the same forward passes makes the gaps much smaller (Table 2). Compared with these balanced counts, each decoder's recorded counts raise its own selector's observed accuracy and lower the other's (Table 6 in Appendix B). For a hand-designed schedule that ends with a few large steps, balancing instead raises both selectors' observed accuracy (Table 22 in Appendix D).

The two count sources use different forward-pass budgets, but the rankings still reverse among problems whose budgets differ by at most 10%, with both gaps' intervals excluding zero (Appendix B.2). Under C95's counts, DAPD must fill positions outside its original proposal, 31% of all filled positions on LLaDA and 15% on Dream, so this comparison measures the adapted DAPD selector rather than the complete DAPD decoder. Removing development problems or scoring only up to the first end-of-sequence token changes the interaction I by less than one point in either model (Appendix B.1).

On MATH500 and ARC and under KLASS's counts, whenever the interval for I excludes zero, recorded counts also help their own selector more than the other (Table 5 in Appendix B). Both intervals include zero when counts come from neither compared selector. Unlike GSM8K, ARC

Table 2: **Same GSM8K problems, opposite rankings.** Gaps are in points ($n = 1{,}319$ per row), and positive $D - C$ favors DAPD. Within each row, all four runs use the same number of forward passes per problem and block. Budgets differ between count sources. NFE is the mean number of forward passes per problem. Recorded gaps have descriptive 95% paired intervals. Intervals for I are 97.5% (DAPD) or 98.75% (C95), and all four also exclude zero under the Bonferroni correction across studies (*). Full accuracies and balanced-gap intervals are in Tables 6 and 8 in Appendix B.

Model	Counts	NFE	Recorded $D - C$ [95% CI]	Balanced $D - C$	I [CI]
LLaDA	DAPD	110.97	$+25.25$ $[22.37, 28.28]$	$+6.37$	$+18.88^*$ $[14.40, 23.28]$
LLaDA	C95	89.97	-12.81 $[-15.54, -10.08]$	$+2.35$	-15.16^* $[-19.94, -10.61]$
Dream	DAPD	65.28	$+23.12$ $[20.39, 25.93]$	$+3.34$	$+19.79^*$ $[15.47, 24.11]$
Dream	C95	72.39	-11.98 $[-14.56, -9.40]$	$+0.91$	-12.89^* $[-17.44, -8.34]$

Table 3: **Balancing reduces the average lead of each decoder's own selector.** Own-selector accuracy minus the mean of the other three, averaged over the four count sources (percentage points, $n = 256$ per row). Balanced counts keep each block's forward passes. Paired 98.75% intervals correct for four main tests per task. † marks the only reported interval that includes zero under the correction across studies. Individual-selector results and intervals for the balanced leads are in Tables 15 and 19 in Appendices C and C.1.

Task	Model	Recorded lead A_R [CI]	Balanced lead A_U	Reduction $A_R - A_U$ [CI]
GSM8K	LLaDA	$+18.95$ $[15.62, 22.27]$	$+3.29$	$+15.66$ $[11.56, 19.95]$
	Dream	$+6.32$ $[3.17, 9.51]$	$+1.46$	$+4.85^{\dagger}$ $[0.78, 8.90]$
ASDiv	LLaDA	$+9.24$ $[6.02, 12.43]$	$+0.68$	$+8.56$ $[4.56, 12.57]$
	Dream	$+14.71$ $[11.20, 18.33]$	-0.16	$+14.88$ $[10.74, 19.14]$

loses accuracy for both selectors when counts are balanced, and C95's interaction has an interval including zero among problems with similar budgets (Appendix B.6).

4.2 Four selectors: an average lead that balancing shrinks

Beyond the confidence–DAPD pair, the four-selector comparison includes KLASS and entropy under all four decoders' recorded and balanced counts. All combinations use the same 256 GSM8K problems in both models (Appendix C). Dream uses one 256-token block here, the setting in which the released KLASS decoder runs. Under each count source, the own-selector lead is its accuracy minus the mean accuracy of the other three selectors. Averaging over the four sources gives A_R for recorded counts and A_U for balanced counts. Their difference, $A_R - A_U$, measures how much balancing reduces the average lead. Constant additive differences in selector or count-source quality cancel in these averages.

On GSM8K, the average own-selector lead is about 19 points on LLaDA and 6 on Dream. Balancing reduces it in both models (Figure 1b and Table 3). With the same decoders and block structures on 256 ASDiv problems, both models again show a positive average lead under recorded counts (Appendix C.1). Balancing at the same budget reduces this average lead to near zero.

All four selectors follow this pattern on ASDiv in both models and on GSM8K in LLaDA: each has a positive observed lead under its own recorded counts, and balancing shrinks it. On GSM8K in Dream, however, entropy falls below the other selectors' mean, DAPD ranks first under every source, and only two of the four leads shrink. Without entropy, Dream's reduction is small, and with all four selectors its interval includes zero under the correction across studies. On GSM8K, the average recorded lead stays positive when only the three selectors sharing DAPD's implementation are compared, when development problems are removed, and when scoring stops at the first end-of-sequence token (Tables 15, 17 and 18 in Appendix C).

Finding 1. The count source can change which selector wins. Across four selectors, the average lead of a decoder's own selector appears on both GSM8K and ASDiv, and balancing at the same forward-pass budget reduces it. On ASDiv, the remaining average lead is near zero in both models.

4.3 New counts beat recorded counts after the text changes

Recorded counts can favor the selector that produced them (Finding 1), but changing the selector also changes the text from which later counts are chosen. Fast-dLLM's comparison of threshold and fixed-count rules changes both the rule and the text (Wu et al., 2025). We instead compare recorded and new counts from identical copies of the changed text and its decoding history (Figure 2a).

A simpler selector creates the changed prefix by filling the first 128 positions using counts recorded from a complete run of the original decoder. It ranks positions by confidence for KLASS and by DAPD's node scores for DAPD, without DAPD's check that positions filled together are independent. From identical copies of this prefix and its history, the original decoder either reuses the original run's remaining counts (N_{ref}), fitting its proposals to them, or chooses counts from its current text (N_{new}), allowing the remaining budget to change. Their accuracy difference is $Q = \mathbb{E}_x[Y_x(N_{\mathrm{new}}) - Y_x(N_{\mathrm{ref}})]$, where $Y_x(N)$ is 1 if continuation N answers problem x correctly

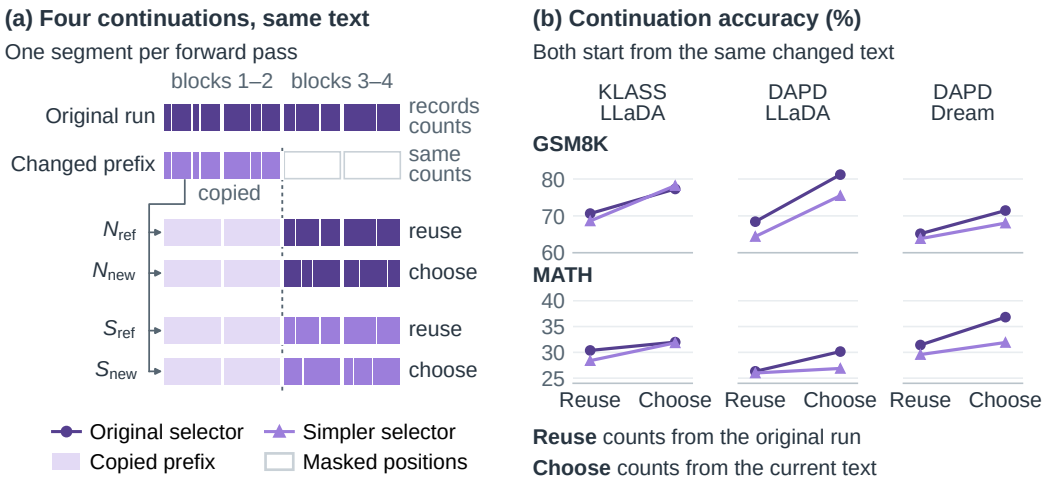

Figure 2: **New counts improve continuation from the same changed text.** (a) A simpler selector refills the first half using the original run's counts. Four continuations share copies of this changed prefix and its history. Each segment represents one forward pass. Its width shows how many tokens are filled. The dashed divider marks the start of continuation. (b) Final accuracy of separate continuations that reuse counts from the original run or choose counts from the current text. Circles denote the original selector and triangles the simpler selector. GSM8K uses 2,084 held-out training problems and MATH 4,096 test problems outside MATH500. The original decoder's paired gains Q and their joint 95% intervals are in Table 24 in Appendix E.

and 0 otherwise. The simpler selector also continues from the same copies, using recorded counts (S_{ref}) or counts recomputed by the original decoder's rule from the simpler selector's current text (S_{new}).

Choosing new counts improves accuracy for both selectors in all six settings (Figure 2b). The original decoder gains up to 12.76 points, and all six paired gain intervals exclude zero (Table 24 in Appendix E). Its mean number of forward passes changes little. The difference between the original decoder's gain and the simpler selector's gain has an interval including zero in every setting (Tables 25 and 26 in Appendix E). These results concern counts recorded from another text, rather than schedules fixed before decoding, such as the dilated schedule of Luxembourg et al. (2026).

Finding 2. After a shared change to the text, choosing new counts improves continuation accuracy in all six settings tested, for the original decoder and for the simpler selector alike.

5 From selector comparisons to decoder choice

5.1 Comparing complete decoders

The studies behind Finding 1 and Finding 2 vary one decision at a time and measure only accuracy. A complete decoder makes both decisions, and its deployment value also depends on latency, block structure and end-of-sequence handling. We compare DAPD-Staged and DAPD-Direct, run with their released code, with C95 and with EB-Sampler (Ben-Hamu et al., 2025) at error tolerance 0.1 (EB0.1). Appendix F gives their settings.

Under Staged's counts, confidence trails the DAPD selector by 25.25 points. As complete decoders on LLaDA with four blocks, however, C95 and Staged reach the same observed accuracy of 77.10%, and C95 has 31.3% lower latency. Direct is faster still, with similar accuracy (Table 4). EB0.1 has accuracy close to that of C95 in all three settings. Direct's accuracy difference from C95 has an interval including zero even without the 64 tuning problems (Appendix F).

The relative latency of these decoders also changes with the block structure. Moving Dream from one 256-token block to four 64-token blocks changes C95 from taking 26.5% more time than Direct to taking 31.9% less (Table 4). The cost per forward pass differs as well: both DAPD variants spend

Table 4: **Complete decoders: accuracy and latency depend on the deployment setting.** GSM8K ($n = 1{,}319$), 256 positions and batch size one. Accuracy in percent and mean request time in seconds, with all four decoders paired within each problem on one H200 GPU. C95 and EB0.1 were tuned on 64 LLaDA GSM8K problems that remain in this set, and they were used on Dream unchanged (Appendix F removes these problems). Forward-pass counts are in Table 27 in Appendix F.

	LLaDA, 4×64		Dream, 1×256		Dream, 4×64	
Decoder	Acc.	Time	Acc.	Time	Acc.	Time
DAPD-Staged	77.10	5.797	54.66	3.713	68.54	6.408
DAPD-Direct	77.26	3.289	56.63	3.559	74.22	4.126
C95	77.10	3.982	41.62	4.501	71.80	2.810
EB0.1	76.88	3.962	41.70	5.169	72.02	2.984

about 8 and 24 ms more per step than C95 on LLaDA and Dream across the full decoding loop (Table 28 in Appendix F).

Whereas block structure changes the latency ranking, end-of-sequence (EOS) handling can change the accuracy ranking. One rule postpones filling positions predicted as EOS while any other candidate remains. In Dream's published single-block setting, it raises C95 from 41.62% to 62.55% and lowers DAPD-Direct from 56.63% to 42.00%, which reverses their ranking. This large gain for C95 does not recur with four blocks (Appendix F).

Finding 3. A selector's advantage under the same counts need not carry over to complete decoders. Block structure can change their relative latency, and end-of-sequence handling can reverse their accuracy ranking.

5.2 Decision-Value Protocol

Findings 1–3 motivate the three practices of Protocol 1. We wrote the protocol down from the two-selector comparisons, the continuation study and the complete-decoder comparisons, before running the four-selector study. That study extends Finding 1 (Section 4.2), and the rest of this section applies the protocol to it.

1	**Evaluate a selector together with the rule that sets its counts.** State whose counts a comparison uses, how selectors are fitted to them and the budget. Before claiming a ranking, compare the selectors under several decoders' counts and balanced counts with the same number of forward passes (Finding 1, Sections 3 and 4).
2	**Choose counts from the current text.** Before reusing counts recorded on another text, such as another decoder's schedule, compare them with counts chosen anew from the current text (Finding 2, Section 4.3).
3	**Choose complete decoders by accuracy and measured latency.** Compare complete decoders on paired accuracy and measured latency under the target block structure, end-of-sequence rule and hardware, and report forward passes separately (Finding 3, Section 5).

Protocol 1: **Decision-Value Protocol.** Three practices for comparing selectors and choosing decoders. The four-selector study on GSM8K and ASDiv reveals differences between selector comparisons and complete-decoder comparisons that the first and third practices address (Section 5.2).

The four-selector study also evaluates each complete decoder with its own counts, as the third practice requires for accuracy. If a shared-count selector ranking is used to choose a complete decoder, it can favor a different candidate. We call this a *reversed choice* when the selector and complete-decoder gaps have opposite signs and both intervals exclude zero. A *gap discrepancy* occurs when the difference between these two gaps has an interval excluding zero. For each task and model, eight recorded or balanced schedules and two fixed counts give ten comparisons of six selector pairs, 60 in all.

On GSM8K, no selector comparison reverses a choice between complete decoders, because every complete-decoder difference that a selector comparison contradicts has an interval including zero. On LLaDA, each decoder's own selector ranks first under its counts, although the entropy rule ranks last as a complete decoder (Table 17 in Appendix C). On Dream, DAPD ranks first under every count source and as a complete decoder.

On ASDiv in LLaDA, the accuracy gaps between selectors are much larger than those between complete decoders. The complete decoders differ by at most 3.52 points, and all six pairwise intervals include zero (Table 20 in Appendix C.1). Yet every selector comparison except the one under the entropy rule's own recorded counts ranks the entropy selector last, 10.94 to 23.83 points below the best selector, with intervals that exclude zero, and this includes balanced and fixed counts. As a complete decoder, the entropy rule is only 0.39 points below the best one.

Each of these selector comparisons has a gap discrepancy involving the entropy rule and at least one other decoder. None reverses a choice, because all six gaps between the complete decoders have intervals including zero. The four count sources also name four different winners, so the choice would depend on which decoder supplied the counts.

On Dream ASDiv, using the selector ranking for decoder choice changes the recommendation. Consider choosing between C95 and the entropy rule. Under the entropy rule's counts, the entropy selector leads confidence by 7.42 points, so the selector comparison favors the entropy rule. Comparing the complete decoders, as the protocol asks, selects C95, which is 9.38 points more accurate. Both 95% intervals exclude zero, so using this selector ranking for decoder choice yields a reversed choice. The entropy selector's lead over KLASS under the same counts gives a second one (Appendix C.1). DAPD remains first among all four complete decoders, as it is under every count source.

Across both tasks, gap discrepancies are more common than reversed choices (Table 21 in Appendix C.1). Of the 240 pairwise gaps, 89 differ between the selector and complete-decoder comparisons, most of them under a decoder's recorded counts. Among confidence, DAPD and KLASS, 28 of 120 gaps differ, with no reversed choice. Selector comparisons under recorded counts thus often give gaps of a different size than the complete decoders show.

6 Conclusion

Because the winning selector can change with the count source (Finding 1), a comparison under one source ranks the selectors for those counts only. After the text changes, counts chosen anew improve the continuation in every setting we tested (Finding 2). Choosing a complete decoder then requires its own accuracy and measured latency (Finding 3). In the four-selector study, this comparison avoids both reversed choices from using selector rankings to choose decoders (Section 5.2).

Limitations. The evidence covers two model families, with the four-selector study confined to two arithmetic tasks and the implementations and block structures tested. Budgets differ across count sources, so comparisons on similar-budget subsets do not isolate the effect of the source. Balancing and continuation also change later text and selected positions, which prevents attributing their effects to a single mechanism. The GSM8K cohort contains previously studied problems. The ASDiv problems had been used only in our study of hand-designed schedules, so no adaptive decoder had run on them before the replication, but earlier exposure to the benchmark remains (Appendices C and C.1).

Both reversed choices in Section 5.2 involve the entropy rule, a threshold decoder that we added for the four-selector study rather than a released decoder. Among the other three decoders, no selector comparison makes a reversed choice on either task, although some gaps differ between the two comparisons. Beyond using selector rankings for decoder choice, DVP has not been compared with another decision rule. DVP has been applied to these decoders at batch size one, with ASDiv measuring accuracy and forward passes rather than latency.

AI use statement. Generative AI tools assisted with language editing, searching for relevant literature, debugging and executing analysis code, and checking mathematical arguments. The authors independently verified the cited literature, source code, experimental outputs, numerical results, and mathematical arguments. The authors take full responsibility for the final content of the paper.

Reproducibility statement. Appendix A describes the datasets, decoding settings, budget checks and statistical methods. The study-specific designs and additional results appear in Appendices B to F. Upon acceptance, we will publicly release the experimental code, configurations, evaluation item IDs and analysis scripts.

REFERENCES

Hikaru Asano, Tadashi Kozuno, Kuniaki Saito, and Yukino Baba. Where-to-Unmask: Ground-truth-guided unmasking order learning for masked diffusion language models. *arXiv preprint arXiv:2602.09501*, 2026. URL https://arxiv.org/abs/2602.09501.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Heli Ben-Hamu, Itai Gat, Daniel Severo, Niklas Nolte, and Brian Karrer. Accelerated sampling from masked diffusion models via entropy bounded unmasking. *arXiv preprint arXiv:2505.24857*, 2025. URL https://arxiv.org/abs/2505.24857.

Victor Besnier, Mickael Chen, David Hurych, Eduardo Valle, and Matthieu Cord. Halton scheduler for masked generative image transformer. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2503.17076.

Dake Bu, Wei Huang, Andi Han, Si Wu, Hau-San Wong, Qingfu Zhang, Taiji Suzuki, and Atsushi Nitanda. DPRM: A plug-in Doob h transform-induced token-ordering module for discrete diffusion models. *arXiv preprint arXiv:2604.24357*, 2026. URL https://arxiv.org/abs/2604.24357. Version 3.

Changxiao Cai and Gen Li. Confidence-based decoding is provably efficient for diffusion language models. *arXiv preprint arXiv:2603.22248*, 2026. URL https://arxiv.org/abs/2603.22248.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. URL https://arxiv.org/abs/2202.04200.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021. URL https://arxiv.org/abs/2107.03374.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

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

Satoshi Hayakawa, Yuhta Takida, Masaaki Imaizumi, Hiromi Wakaki, and Yuki Mitsufuji. Demystifying MaskGIT sampler and beyond: Adaptive order selection in masked diffusion. *Transactions on Machine Learning Research*, 2026. URL https://arxiv.org/abs/2510.04525.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset, 2021. URL https://arxiv.org/abs/2103.03874.

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. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2512.09106.

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

Siva Rajesh Kasa, Yasong Dai, Sumit Negi, and Hongdong Li. Fast-dLLM++: Fréchet profile decoding for faster diffusion LLM inference. *arXiv preprint arXiv:2606.02955*, 2026. URL https://arxiv.org/abs/2606.02955.

Bumjun Kim, Dongjae Jeon, Moongyu Jeon, and Albert No. DAPD: Dependency-aware parallel decoding via attention for diffusion LLMs. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://icml.cc/virtual/2026/poster/64624.

Jaeyeon Kim, Kulin Shah, Vasilis Kontonis, Sham M. Kakade, and Sitan Chen. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 30749–30768. PMLR, 2025a. URL https://proceedings.mlr.press/v267/kim25ah.html.

Seo Hyun Kim, Sunwoo Hong, Hojung Jung, Youngrok Park, and Se-Young Yun. KLASS: KL-guided fast inference in masked diffusion models. In *Advances in Neural Information Processing Systems*, volume 38, 2025b. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/8514c1f12dcdd061ac8521feb78f28ca-Paper-Conference.pdf.

Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs created equal? a large-scale study. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://arxiv.org/abs/1711.10337.

Omer Luxembourg, Haim Permuter, and Eliya Nachmani. Plan for speed: Dilated scheduling for masked diffusion language models. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2506.19037.

Gábor Melis, Chris Dyer, and Phil Blunsom. On the state of the art of evaluation in neural language models. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=ByJHuTgA-.

Shen-yun Miao, Chao-Chun Liang, and Keh-Yih Su. A diverse corpus for evaluating and developing English math word problem solvers. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 975–984, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.92. URL https://aclanthology.org/2020.acl-main.92/.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025. doi: 10.48550/arXiv.2502.09992. URL https://arxiv.org/abs/2502.09992.

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. URL https://arxiv.org/abs/2606.10829.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dllm: Training-free acceleration of diffusion LLM by enabling KV cache and parallel decoding. *arXiv preprint arXiv:2505.22618*, 2025. URL https://arxiv.org/abs/2505.22618.

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. doi: 10.48550/arXiv.2508.15487. URL https://arxiv.org/abs/2508.15487.

Shaorong Zhang, Longxuan Yu, Rob Brekelmans, Luhan Tang, Salman Asif, and Greg Ver Steeg. Generation order and parallel decoding in masked diffusion models: An information-theoretic perspective. *arXiv preprint arXiv:2602.00286*, 2026. doi: 10.48550/arXiv.2602.00286. URL https://arxiv.org/abs/2602.00286.

Fengrui Zuo, Zhiwei Ke, Yiming Liu, Wenqi Lou, Chao Wang, and Xuehai Zhou. Window-Diffusion: Accelerating diffusion language model inference with windowed token pruning and caching. *arXiv preprint arXiv:2601.20332*, 2026. URL https://arxiv.org/abs/2601.20332.

A PROBLEMS, DECODERS AND STATISTICS

Problems. The main two-selector comparisons use all 1,319 GSM8K (Cobbe et al., 2021) test problems with DAPD's five-shot prompts and flexible answer extraction. The additional two-selector comparisons use all 500 MATH500 problems from MATH (Hendrycks et al., 2021), scored by exact match of the final answer, and all 1,172 ARC-Challenge (Clark et al., 2018) test problems with a fixed zero-shot prompt and a strict parser that takes the last explicit answer letter (Section 4.1). The hand-designed schedules also use HumanEval (Chen et al., 2021) and MBPP-sanitized (Austin et al., 2021) (421 problems, scored by execution) and 1,429 ASDiv (Miao et al., 2020) problems. The continuation study uses 2,084 held-out GSM8K training problems and 4,096 MATH test problems outside MATH500. The four-selector study uses 256 GSM8K test problems and 256 ASDiv problems, each set chosen by a hash of its identifiers (Appendices C and C.1).

Decoding. All runs use BF16 weights, batch size one and 256 generated positions. Unless stated otherwise, the positions form four blocks of 64. Dream uses one block of 256 in the four-selector study and in DAPD's published setting (Section 5). Answers are scored on the whole output, and scoring only up to the first end-of-sequence token serves as a sensitivity check.

Budgets and reproduction. Within each count source, recorded and balanced runs use the same number of forward passes in every block of every problem, verified for every run. The GSM8K replay checks find exact reproduction of each decoder's proposals, filled positions and final tokens under its own recorded counts, for every problem. The tables of this appendix write recorded and balanced counts as R (or rec) and U (or bal), with prefixes C, D and K for confidence, the DAPD selector and the KLASS selector. Some captions call the decoder whose counts are reused the *donor*.

Intervals. Unless stated otherwise, intervals are percentile intervals from 10,000 paired bootstrap draws over problems, which keep all runs of a problem together. The cross-study intervals for the four-selector studies use 400,000 paired draws with seed 20260925. The fixed-count ranking intervals use 100,000 draws (Appendix C). Each study set its interval level when its design was fixed, from the number of main tests in its family: 97.5% for DAPD's counts and for ARC, 98.75% for C95's counts on GSM8K, the four-selector study and the complete decoders, 99.17% for the hand-designed schedules, 99.5% for the end-of-sequence rule and 95% for the KLASS studies. Other contrasts are descriptive 95% intervals. An interval that includes zero shows neither equivalence nor noninferiority.

Paired exact intervals. For w wins and l losses among n pairs, let $d = w+l$, $q = \Pr(\text{discordance})$ and $p = \Pr(\text{win} \mid \text{discordance})$. Then $\Delta = q(2p-1)$, estimated by $(w-l)/n$. Clopper–Pearson bounds for d/n and w/d, each with per-tail error $\alpha/(4m)$, define a rectangle, and the extrema of $q(2p-1)$ over it bound the effect. A union bound gives coverage of at least $1-\alpha$ for m contrasts. When $d = 0$, we use $p \in [0, 1]$. For Q, $m = 6$ and $\alpha = .05$.

Correction over all main tests. The cross-study Bonferroni correction covers all 161 main tests preceding the four-selector study, including tests from studies omitted from the paper. At the resulting two-sided level of 99.9689%, 85 of the 100 main tests whose intervals exclude zero in their own family still exclude zero. The four-selector study adds four tests, and Table 15 gives its results under the correction over 165 tests. The ASDiv replication adds four more, and Table 19 gives its results under the correction over 169 tests. Tests remain in this correction even if their study is removed from the paper.

Development problems. Earlier studies used some benchmark problems during development, 182 GSM8K and 79 MATH500 problems in total. We also chose C95 and EB0.1 on 64 LLaDA GSM8K problems. Results without these problems are sensitivity checks (Table 11 and Appendix F). Removing them does not remove earlier exposure to the benchmarks.

B TWO-SELECTOR COMPARISONS

The four-run design covers thirteen settings (Table 5), including the four GSM8K settings in Table 2, whose individual-run accuracies are in Table 6. DAPD's counts come from the released

Staged decoder on LLaDA and Direct decoder on Dream, using all GSM8K and MATH500 problems without selecting problems. We fixed its design before any of its runs, after earlier studies, including the hand-designed schedules, had been read. On GSM8K, balancing changes confidence by $+9.63$ pp $[6.52, 12.66]$ on LLaDA and $+10.46$ pp $[7.58, 13.42]$ on Dream, and the DAPD selector by -9.25 pp $[-11.75, -6.75]$ and -9.33 pp $[-11.83, -6.82]$ (descriptive 95%). On MATH500 the corresponding changes are $+1.80$, $+7.00$, -4.80 and -4.80 pp.

Table 5: **All thirteen selector comparisons under recorded counts.** Accuracy (%) of confidence (C) and the other selector named in each row group: DAPD (D) or KLASS (K). Each uses recorded or balanced counts with the same number of forward passes. I is the recorded gap over confidence minus the balanced gap, in points. Within a row, all four runs use the same number of forward passes in every problem and block, while budgets and implementations differ between rows. Rows are grouped by the selector whose decoder produced the counts, a grouping we noticed after the earlier settings had been read. Intervals use the level of each study (97.5% for DAPD's counts and ARC, 99.17% for C95's counts on ARC, 98.75% for C95's counts on GSM8K and 95% for the KLASS studies). Bold intervals exclude zero at that level, and * marks those that still exclude zero after the correction over 161 tests (Appendix A). The last column says when the design was fixed: Dev., after development runs and before the full results; Pro., before any output on the task; Ext., before the ARC effects were read; Pre., with the sign predicted before the output; Out., after related results were known.

Model / task	n	C_{rec}	C_{bal}	Other rec	Other bal	I [CI]	Design
DAPD produced the counts; D is the adapted DAPD selector							
LLaDA / GSM8K	1,319	51.86	61.49	77.10	67.85	$+\mathbf{18.88}^{*}$ [14.40, 23.28]	Dev.
Dream / GSM8K	1,319	51.10	61.56	74.22	64.90	$+\mathbf{19.79}^{*}$ [15.47, 24.11]	Dev.
LLaDA / MATH500	500	21.20	23.00	28.60	23.80	$+\mathbf{6.60}$ [0.60, 12.60]	Dev.
Dream / MATH500	500	22.20	29.20	37.40	32.60	$+\mathbf{11.80}^{*}$ [5.20, 18.20]	Dev.
LLaDA / ARC	1,172	71.08	64.59	78.41	70.65	$+1.28$ [-2.05, 4.52]	Pro.
Dream / ARC	1,172	83.45	75.34	90.02	74.83	$+\mathbf{7.08}^{*}$ [4.27, 9.98]	Pro.
The C95 confidence threshold produced the counts; D is the adapted DAPD selector							
LLaDA / GSM8K	1,319	78.01	65.81	65.20	68.16	$-\mathbf{15.16}^{*}$ [$-19.94, -10.61$]	Pre.
Dream / GSM8K	1,319	72.33	61.26	60.35	62.17	$-\mathbf{12.89}^{*}$ [$-17.44, -8.34$]	Pre.
LLaDA / ARC	1,172	69.20	57.59	69.03	66.98	$-\mathbf{9.56}^{*}$ [$-14.45, -4.75$]	Ext.
Dream / ARC	1,172	89.51	73.81	77.47	75.17	$-\mathbf{13.40}^{*}$ [$-17.32, -9.47$]	Ext.
KLASS produced the counts; K is the adapted KLASS selector							
LLaDA / GSM8K	1,319	63.91	69.75	74.30	70.20	$+\mathbf{9.93}^{*}$ [7.28, 12.66]	Out.
Neither compared selector produced the counts: DAPD counts, K is the adapted KLASS selector							
LLaDA / GSM8K	1,319	51.18	61.41	51.55	60.65	$+1.14$ [-1.14, 3.49]	Out.
Neither compared selector produced the counts: KLASS counts, D is the adapted DAPD selector							
LLaDA / GSM8K	1,319	63.31	69.45	64.44	70.74	-0.15 [-4.17, 4.02]	Out.

Table 6: **Complete four-run results for the main GSM8K comparison.** Accuracy (%) and gains (points), with $n = 1{,}319$ per row. All four runs in a row use the same number of forward passes in each problem and block; mean NFE differs between count sources. V_C and V_D are the gains from recorded counts, and $I = V_D - V_C$. The intervals for I are 97.5% for DAPD's counts and 98.75% for C95's counts. Asterisks mark intervals that exclude zero after correction over the 161 earlier main tests. Table 2 presents the ranking gaps with their intervals.

Model	Counts	NFE	C_{rec}	C_{bal}	D_{rec}	D_{bal}	V_C	V_D	$D - C$	I [CI]
LLaDA	DAPD	110.97	51.86	61.49	77.10	67.85	-9.63	$+9.25$	$+25.25$	$+\mathbf{18.88}^{*}$ [14.40, 23.28]
LLaDA	C95	89.97	78.01	65.81	65.20	68.16	$+12.21$	-2.96	-12.81	$-\mathbf{15.16}^{*}$ [$-19.94, -10.61$]
Dream	DAPD	65.28	51.10	61.56	74.22	64.90	-10.46	$+9.33$	$+23.12$	$+\mathbf{19.79}^{*}$ [15.47, 24.11]
Dream	C95	72.39	72.33	61.26	60.35	62.17	$+11.07$	-1.82	-11.98	$-\mathbf{12.89}^{*}$ [$-17.44, -8.34$]

B.1 C95's counts on GSM8K

We fixed this study after the earlier settings, including C95's counts on ARC, had been read, and before any C95 run on GSM8K. It reuses the GSM8K test problems of the earlier studies, so it is not a fresh confirmation. Its family has four main tests with 98.75% intervals, $I < 0$ and $H > 0$ in each model. H is confidence's loss from balancing on the fastest third of problems minus its loss on the slowest third, with the thirds formed from C95's forward passes before any other run (Table 7).

Table 7: **C95's counts on GSM8K** ($n = 1{,}319$ per model). Accuracy in percent and effects in points. $\Delta_C = C_{\text{rec}} - C_{\text{bal}}$ and $\Delta_D = D_{\text{rec}} - D_{\text{bal}}$, so $I = \Delta_D - \Delta_C$. Rows marked 98.75% are the main tests. Other intervals are descriptive 95% intervals, and the last two rows use the correction over all main tests.

Quantity	LLaDA	Dream
C_{rec} / C_{bal} / D_{rec} / D_{bal}	78.01 / 65.81 / 65.20 / 68.16	72.33 / 61.26 / 60.35 / 62.17
Mean NFE (all runs)	89.97	72.39
Δ_C	$+12.21$ $[9.63, 14.78]$	$+11.07$ $[8.49, 13.57]$
Δ_D	-2.96 $[-5.76, -0.15]$	-1.82 $[-4.62, 0.99]$
I (98.75%)	$\mathbf{-15.16}$ $[-19.94, -10.61]$	$\mathbf{-12.89}$ $[-17.44, -8.34]$
H (98.75%)	$\mathbf{+12.98}$ $[5.01, 20.96]$	$+1.59$ $[-6.38, 9.34]$
I without 182 development problems	-15.48 $[-19.44, -11.52]$	-12.84 $[-16.71, -9.06]$
I scored to the first end-of-sequence token	-14.25 $[-17.89, -10.69]$	-12.74 $[-16.22, -9.25]$
I, correction over all main tests	$[-21.99, -8.42]$	$[-19.48, -6.29]$
H, correction over all main tests	$[1.59, 24.60]$	$[-10.02, 13.21]$

Table 8: **The $D - C$ gap on GSM8K under two decoders' counts** ($n = 1{,}319$ per setting), in points, with descriptive 95% paired bootstrap intervals, for the gaps in Table 2. Budgets match within a decoder's counts but not between decoders.

Counts	Model	Mean NFE	Recorded gap	Balanced gap
DAPD	LLaDA	110.97	$+25.25$ $[22.37, 28.28]$	$+6.37$ $[3.49, 9.25]$
DAPD	Dream	65.28	$+23.12$ $[20.39, 25.93]$	$+3.34$ $[0.61, 5.99]$
C95	LLaDA	89.97	-12.81 $[-15.54, -10.08]$	$+2.35$ $[-0.38, 5.08]$
C95	Dream	72.39	-11.98 $[-14.56, -9.40]$	$+0.91$ $[-1.74, 3.64]$

Both values of I are negative, so the prediction holds in both models. H is positive on LLaDA, but its Dream interval includes zero. From the fastest to the slowest third, balancing costs confidence 18.00, 13.61 and 5.01 points on LLaDA but 10.93, 12.93 and 9.34 points on Dream, so the relation between forward-pass count and balancing loss differs between models. Under C95's counts, the DAPD selector fills positions outside DAPD's own proposal in 31.17% (recorded) and 23.57% (balanced) of its filled positions on LLaDA and in 14.95% and 13.81% on Dream, and every problem needs this adaptation at least once.

B.2 GSM8K problems with similar budgets

The two count sources differ in total forward passes as well as in their allocation. On GSM8K, DAPD uses 110.97 and 65.28 mean forward passes and C95 uses 89.97 and 72.39 (LLaDA, Dream). The budget-matched subsets restrict this comparison to problems with equal totals, equal counts in every block, or totals within 10% or 20% of each other (Table 9), using the subset definitions and intervals of the ARC analysis (Appendix B.6). Within 10%, both decoders use almost the same mean budget, and the reversal holds in both models with intervals that exclude zero, as does the difference between the two interactions. Equal totals are rare on LLaDA, and equal budgets in every block are too rare to interpret. Matching changes which problems are compared. On LLaDA, 251 of the 321 matched problems are among the third on which C95 is slowest, while on Dream the matched problems come from all thirds (201, 170 and 122 from the fastest to the slowest).

Table 9: **GSM8K problems on which DAPD and C95 use similar budgets.** Problems are kept when the two decoders' total forward passes are equal, equal in every block, or within 10% or 20% of the larger total. $D - C$ is the gap between the DAPD selector and confidence under each decoder's recorded counts, and $\Delta I = I_{\mathrm{C95}} - I_{\mathrm{DAPD}}$. Entries are in points with descriptive 95% paired bootstrap intervals (10,000 draws). Subsets with fewer than 30 problems are too small to interpret.

Problems	n	NFE (DAPD/C95)	$D - C$, DAPD's counts	$D - C$, C95's counts	ΔI
LLaDA					
All	1,319	111.0/90.0	$+25.25$ $[22.37, 28.28]$	-12.81 $[-15.54, -10.08]$	-34.04 $[-39.35, -28.96]$
Same total	14	114.4/114.4	$+14.29$ $[-14.29, 50.00]$	-14.29 $[-42.86, 14.29]$	-14.29 $[-71.43, 42.86]$
Same per block	0	—	—	—	—
Within 10%	321	112.9/112.1	$+32.40$ $[25.86, 38.94]$	-11.53 $[-17.76, -5.60]$	-33.02 $[-44.24, -21.81]$
Within 20%	639	112.9/108.6	$+31.61$ $[26.92, 36.15]$	-10.64 $[-15.02, -6.57]$	-35.68 $[-43.67, -27.70]$
Dream					
All	1,319	65.3/72.4	$+23.12$ $[20.39, 25.93]$	-11.98 $[-14.56, -9.40]$	-32.68 $[-37.53, -27.82]$
Same total	59	56.0/56.0	$+28.81$ $[18.64, 40.68]$	-15.25 $[-25.42, -6.78]$	-45.76 $[-64.41, -28.81]$
Same per block	9	42.4/42.4	$+22.22$ $[0.00, 55.56]$	$+0.00$ $[0.00, 0.00]$	-22.22 $[-55.56, 0.00]$
Within 10%	493	65.8/66.1	$+22.72$ $[18.46, 26.98]$	-14.60 $[-18.46, -10.75]$	-35.70 $[-42.60, -28.60]$
Within 20%	885	65.4/67.6	$+21.58$ $[18.31, 24.86]$	-13.79 $[-16.72, -10.96]$	-32.99 $[-38.53, -27.57]$

B.3 KLASS'S COUNTS

Native KLASS runs on LLaDA GSM8K with its own implementation and settings (confidence threshold 0.6, KL threshold 0.015, history length 2). Under its counts, balancing raises confidence by $+5.84$ pp $[2.88, 8.87]$ and lowers the KLASS selector by -4.09 pp $[-6.75, -1.44]$, so $I = +9.93$ points (Table 5). Under DAPD's counts, the same implementation gives $I = +1.14$ pp $[-1.14, 3.49]$ for confidence and the KLASS selector. After these results were known, we also reused KLASS's counts with confidence and the DAPD selector, in DAPD's implementation. These four runs reach 63.31, 69.45, 64.44 and 70.74% (C_{rec}, C_{bal}, D_{rec}, D_{bal}) at 96.33 mean NFE, and $I = -0.15$ pp $[-4.17, 4.02]$ includes zero. Balancing raises the DAPD selector by 6.29 pp $[3.49, 9.10]$. This agrees with a separate advance prediction that its loss from balancing would be smaller than the 9.25 points it loses under its own decoder's counts. All intervals in this subsection are 95% intervals.

B.4 THE FAILED SIZE PREDICTION

Before the results under DAPD's counts were read, we predicted their size by scaling the effects of the hand-designed schedules. With $\rho = (D_U - D_{SL})/(C_U - C_{SL})$ from those schedules (0.028 on LLaDA and 0.572 on Dream), the prediction was $I \approx B_C(1 - \rho)$, with a tolerance of ± 5 points. With $B_X = -V_X$ denoting the gain from balancing, Table 10 splits $I = B_C - B_D$ into the two selectors' gains from balancing. The confidence term matches, but the DAPD selector's term has the opposite sign in both models, and the prediction misses by 9.15 and 15.32 points. In the hand-designed schedules, the DAPD selector always fills counts that DAPD did not produce, while under DAPD's recorded counts it follows its own decoder's counts. The prediction had no term for this difference. We formed this account after the results were known.

Table 10: **Decomposing the failed size prediction.** B_X is selector X's GSM8K accuracy gain from balancing DAPD's counts, in points. Thus $B_X = -V_X$ and $I = B_C - B_D$. Observed gains carry descriptive 95% intervals, and observed interactions the intervals of Table 5.

Model	Term	Predicted	Observed	Observed − predicted
LLaDA	B_C (confidence)	$+10.01$	$+9.63$ $[6.52, 12.66]$	-0.38
LLaDA	B_D (DAPD selector)	$+0.28$	-9.25 $[-11.75, -6.75]$	-9.53
LLaDA	I	$+9.73$	$+18.88$ $[14.40, 23.28]$	$+9.15$
Dream	B_C (confidence)	$+10.46$	$+10.46$ $[7.58, 13.42]$	0.00
Dream	B_D (DAPD selector)	$+5.99$	-9.33 $[-11.83, -6.82]$	-15.32
Dream	I	$+4.47$	$+19.79$ $[15.47, 24.11]$	$+15.32$

B.5 DEVELOPMENT PROBLEMS

Removing the 261 development problems (182 GSM8K and 79 MATH500) leaves every interaction under DAPD's counts positive, with intervals that exclude zero (Table 11).

Table 11: **Results under DAPD's counts without the** 261 **development problems.** Accuracy in percent and interactions in points with 97.5% intervals.

Model / task	n	C-R	C-U	D-R	D-U	I [97.5% CI]
LLaDA / GSM8K	1,137	51.36	61.30	77.04	68.07	$+18.91$ $[14.16, 23.66]$
Dream / GSM8K	1,137	50.48	61.30	74.41	65.79	$+19.44$ $[14.86, 24.10]$
LLaDA / MATH500	421	19.95	23.04	28.74	23.52	$+8.31$ $[1.90, 14.96]$
Dream / MATH500	421	20.67	28.50	36.82	31.83	$+12.83$ $[5.94, 19.71]$

B.6 ARC-CHALLENGE

Unparsed ARC outputs count as wrong. The study's two main tests, one per model, use 97.5% intervals. Under DAPD's counts, I is positive on Dream but its LLaDA interval includes zero. A second family of six tests, fixed before the ARC effects were read, adds a confidence-ranked adaptation rule and C95's counts (Table 12). The adaptation rule changes the filled positions but not the conclusions, and under C95's counts I is negative in both models. A negative I need not reverse the ranking: on LLaDA, the gap under C95's counts is only -0.17 points.

Table 12: **ARC under two adaptation rules and two decoders' counts** ($n = 1{,}172$ per model). Interactions in points. The first row has the 97.5% intervals of the main tests, and the other rows 99.17% intervals from the six-test family. The lower block gives the recorded and balanced $D - C$ gaps as point estimates.

Quantity	LLaDA	Dream
DAPD's counts, node-score rule, I	$+1.28$ $[-2.05, 4.52]$	$+7.08$ $[4.27, 9.98]$
DAPD's counts, confidence rule, I	-0.94 $[-4.44, 2.56]$	$+5.20$ $[2.30, 8.28]$
Change between the two rules	-2.22 $[-4.69, 0.28]$	-1.88 $[-4.27, 0.51]$
C95's counts, node-score rule, I	-9.56 $[-14.45, -4.75]$	-13.40 $[-17.32, -9.47]$
DAPD's counts: recorded / balanced gap	$+7.34$ / $+6.06$	$+6.57$ / -0.51
C95's counts: recorded / balanced gap	-0.17 / $+9.39$	-12.03 / $+1.37$

Problems with similar budgets. On ARC, DAPD uses 116.37 and 22.34 mean forward passes and C95 uses 72.38 and 25.52 (LLaDA, Dream). On problems where the two decoders' forward passes differ by at most 10%, the interaction under C95's counts is -4.63 points on 108 LLaDA problems and 0.00 points on 247 Dream problems, and the difference between decoders includes zero in both models (Table 13). These problems are not a random subset, so matching changes the problems as well as the budget and cannot isolate the decoder.

Table 13: **ARC interactions on problems with similar budgets.** I under DAPD and C95's counts and their difference $\Delta I = I_{\mathrm{C95}} - I_{\mathrm{DAPD}}$, in points, with descriptive 95% paired bootstrap intervals for ΔI (10,000 draws). The matched subset keeps problems whose two forward-pass totals differ by at most 10% of the larger. The four LLaDA problems with exactly equal budgets give a degenerate interval.

Model	Subset	n	I_{DAPD}	I_{C95}	ΔI [95% CI]
LLaDA	All items	1,172	$+1.28$	-9.56	-10.84 $[-14.76, -7.00]$
LLaDA	Exact total	4	$+25.00$	$+25.00$	0.00 (degenerate)
LLaDA	Exact per block	0	–	–	–
LLaDA	Within 10%	108	-1.85	-4.63	-2.78 $[-13.91, 9.26]$
Dream	All items	1,172	$+7.08$	-13.40	-20.48 $[-23.98, -17.06]$
Dream	Exact total	80	$+3.75$	$+1.25$	-2.50 $[-7.50, 2.50]$
Dream	Exact per block	56	$+3.57$	$+1.79$	-1.79 $[-7.14, 3.57]$
Dream	Within 10%	247	$+3.24$	0.00	-3.24 $[-7.69, 0.81]$

C FOUR-SELECTOR STUDY

Design. The study crosses four selectors with the counts of their four decoders (Table 1). It uses the first 256 GSM8K test problems after sorting the SHA256 hashes of `routeA-20260924:` followed by `gsm8k:test:`i, where i is the zero-based test index. The problems were not chosen by correctness or speed. The same problems are used in LLaDA and Dream. For every problem, each decoder first records its counts. Each selector then runs under these recorded counts and under balanced counts with the same forward passes in every block, and it also runs with two and with four positions per step. This gives 44 generations per problem and model, or 22,528 in total, which took about 40 GPU hours. LLaDA uses four 64-token blocks and Dream one 256-token block, the settings under which the released KLASS decoder runs. The Dream results therefore use a different block structure from the Dream pair comparison in Section 4.1 and are not pooled with them.

Checks. Within each decoder's counts, all recorded and balanced runs use exactly the same number of forward passes for every problem and block. Under its own decoder's recorded counts, each selector reproduces the decoder exactly. An independent check recomputed these budgets, these reproductions and the main intervals from the raw traces. Confidence, the DAPD selector and the entropy selector share DAPD's numerical implementation, while the KLASS selector keeps the implementation of KLASS. The rows of the matrix therefore differ in implementation as well as in ranking rule. Under another decoder's counts, a selector must often fill positions that its own decoder would not have chosen at that step. This happens for 20–62% of the filled positions on LLaDA and for 1–69% on Dream, depending on the selector and the decoder.

What was fixed in advance. Before the runs, we fixed the design, the four main tests and their predicted signs, and the 98.75% interval level that corrects for these four tests. The main tests are A_R and $A_R - A_U$ in each model, and we predicted both to be positive. We also fixed a breadth criterion, which asked that at least three of the four own selectors show each effect. We chose 256 problems per model from the run time of a small test run, before reading any result. We completed the rules for interpreting the results after the generation had started but before any accuracy was read, and no decision used partial results.

Table 14: **Four-selector study: accuracy (%) of each selector under each decoder's counts.** Rows are selectors and columns are the decoders whose counts are reused, on the same 256 GSM8K test problems in both models. Recorded counts reproduce each decoder exactly on the diagonal. Balanced counts spread the same forward passes evenly within each block. LLaDA uses four 64-token blocks and Dream one 256-token block.

Selector	C95's counts	DAPD's counts	KLASS's counts	Entropy rule's counts
LLaDA, recorded counts				
Confidence	78.91	51.95	65.23	58.59
DAPD	69.53	78.52	64.45	60.94
KLASS	72.66	55.47	74.61	61.72
Entropy rule	42.19	44.14	43.36	73.83
LLaDA, balanced counts				
Confidence	68.75	64.45	71.09	69.14
DAPD	65.23	67.97	67.97	70.70
KLASS	67.97	60.55	71.88	69.53
Entropy rule	60.55	51.56	64.45	65.62
Dream, recorded counts				
Confidence	41.80	35.55	42.19	37.11
DAPD	47.66	51.95	48.44	44.53
KLASS	41.80	35.94	42.19	35.55
Entropy rule	19.92	18.75	24.22	33.20
Dream, balanced counts				
Confidence	49.22	42.97	46.48	46.88
DAPD	47.66	50.00	46.88	48.44
KLASS	49.61	43.75	48.05	47.66
Entropy rule	32.81	30.08	37.11	32.03

Table 15: **Four-selector study: lead of each decoder's own selector.** Under each decoder's counts, the accuracy of the decoder's own selector minus the mean accuracy of the other three selectors, in points. Individual-selector intervals are descriptive 95% paired bootstrap intervals. The means over the four decoders are A_R (recorded, 98.75% intervals) and A_U (balanced, descriptive 95% intervals). The lower part gives the study's four main tests, A_R and $A_R - A_U$, with 98.75% intervals that correct for these four tests. It repeats them without the 30 development problems, with answers scored only up to their first end-of-sequence token, and under a Bonferroni correction over all 165 main tests of the paper.

Own selector	Recorded counts	Balanced counts
LLaDA		
Confidence	$+17.45$ $[13.28, 21.61]$	$+4.17$ $[0.52, 7.68]$
DAPD	$+27.99$ $[22.40, 33.59]$	$+9.11$ $[3.39, 14.71]$
KLASS	$+16.93$ $[12.50, 21.35]$	$+4.04$ $[0.39, 7.68]$
Entropy rule	$+13.41$ $[7.81, 19.01]$	-4.17 $[-9.51, 1.17]$
Mean (A_R, A_U)	$+18.95$ $[15.62, 22.27]$	$+3.29$ $[1.14, 5.44]$
Dream		
Confidence	$+5.34$ $[1.82, 8.85]$	$+5.86$ $[1.95, 9.77]$
DAPD	$+21.88$ $[15.76, 27.86]$	$+11.07$ $[4.82, 17.32]$
KLASS	$+3.91$ $[0.52, 7.29]$	$+4.56$ $[1.04, 8.07]$
Entropy rule	-5.86 $[-11.59, -0.13]$	-15.62 $[-21.09, -10.29]$
Mean (A_R, A_U)	$+6.32$ $[3.17, 9.51]$	$+1.46$ $[-0.68, 3.55]$
Main test	A_R	$A_R - A_U$
LLaDA		
All 256 problems (98.75%)	$+18.95$ $[15.62, 22.27]$	$+15.66$ $[11.56, 19.95]$
Without development problems	$+18.81$ $[15.15, 22.46]$	$+15.15$ $[10.62, 19.64]$
Scored to the first end-of-sequence token	$+18.98$ $[15.64, 22.30]$	$+16.21$ $[12.11, 20.49]$
Correction over 165 main tests	$[14.13, 23.70]$	$[9.64, 21.71]$
Dream		
All 256 problems (98.75%)	$+6.32$ $[3.17, 9.51]$	$+4.85$ $[0.78, 8.90]$
Without development problems	$+6.34$ $[3.08, 9.85]$	$+4.90$ $[0.70, 9.40]$
Scored to the first end-of-sequence token	$+6.64$ $[3.52, 9.80]$	$+5.18$ $[1.17, 9.11]$
Correction over 165 main tests	$[1.86, 11.00]$	$[-0.98, 10.81]$

Results. On LLaDA, both averages are positive, all four own selectors show both effects, and the results hold under every sensitivity check (Tables 14 and 15). On Dream, both averages are positive with 98.75% intervals that exclude zero. The lead appears for three of the four own selectors, and the shrinkage for only two, so Dream does not meet the breadth criterion. Under the correction over all 165 main tests of the paper, Dream's lead still excludes zero, but its shrinkage does not. Removing any one selector together with its decoder leaves both averages positive in both models, although removing the entropy rule leaves Dream's shrinkage at only 0.52 points.

A change to the output-length check. Our plan also marked the main tests as limited when any run gave a median answer shorter than half the median answer of confidence at four positions per step. On Dream, this threshold is 21 words. The entropy selector under the counts of C95 and of KLASS gives medians of 18 and 19.5 words, with accuracies of 19.92% and 24.22%, so the plan marked both Dream tests as limited. After seeing the complete results, we decided to treat this check as a diagnostic rather than as a veto on the average effects. The threshold had not been calibrated, short answers can be correct, and the entropy selector's answers are short even under its own counts, with a median of 21.5 words. We report the original flag, and the estimates and intervals are unchanged. Because we made this change after the results were known, we do not describe the Dream results as a full confirmation of the plan.

Table 16: **Four-selector study: rankings of complete decoders and of selectors at fixed counts.** Accuracy (%) of the four complete adaptive decoders and of the four selectors when every step fills a fixed number k of positions. τ is Kendall's rank correlation between the fixed-k ranking and the ranking of complete decoders. These are point-estimate rankings, and many pairwise differences have intervals that include zero.

Model	Setting	Confidence	DAPD	KLASS	Entropy rule	τ
LLaDA	Complete decoders	78.91	78.52	74.61	73.83	—
	Fixed $k = 2$	76.56	75.78	73.83	71.48	1.00
	Fixed $k = 4$	61.72	66.41	62.50	53.52	0.33
Dream	Complete decoders	41.80	51.95	42.19	33.20	—
	Fixed $k = 2$	49.22	47.27	50.78	36.72	0.33
	Fixed $k = 4$	45.31	46.88	45.70	31.64	1.00

Rankings at fixed counts. We predicted that at least one pair of selectors would rank in opposite orders at a fixed count and as complete decoders, with both differences excluding zero under a correction over the 36 pairwise differences. These secondary intervals use 100,000 paired bootstrap draws with seed 20260924. No pair met this criterion at either fixed count in either model (Table 16). The ranking at two positions per step matches the ranking of the complete decoders on LLaDA, and the ranking at four positions per step matches it on Dream. On GSM8K, no fixed-count comparison establishes a reversal of the complete-decoder ranking.

Applying the protocol to the later four-selector study. On LLaDA, every count source ranks its own selector first, with gaps of 6.25–23.05 points over the strongest other selector, whereas the complete decoders differ by at most 5.08 points (Table 17). The entropy selector ranks first under its own counts but last as a complete decoder. On Dream, DAPD ranks first under every count source and as a complete decoder. These are point-estimate rankings, and several gaps between complete decoders have intervals containing zero. We stated the protocol before running the four-selector study and compared these rankings after the matrix results were available. Beyond the choices that selector comparisons imply, DVP has not been compared with another decision rule.

Table 17: **Rankings under one decoder's counts and as complete decoders.** Winners and comparators are chosen by observed accuracy on the 256 problems. Each gap compares the decoder's own selector with the most accurate other selector under its counts, or with the most accurate other complete decoder. Parentheses name that comparator. Intervals are descriptive 95% paired bootstrap intervals for these named pairs, with the comparator held fixed across bootstrap draws. They are not simultaneous tests of the winners.

Counts of	Best selector	Own minus best other Recorded counts	Own minus best other Complete decoders
LLaDA			
Confidence	Confidence	$+6.25$ $[2.73, 10.16]$ (KLASS)	$+0.39$ $[-4.69, 5.08]$ (DAPD)
DAPD	DAPD	$+23.05$ $[16.79, 29.30]$ (KLASS)	-0.39 $[-5.08, 4.69]$ (Confidence)
KLASS	KLASS	$+9.38$ $[4.30, 14.45]$ (Confidence)	-4.30 $[-8.98, 0.39]$ (Confidence)
Entropy rule	Entropy rule	$+12.11$ $[5.47, 18.75]$ (KLASS)	-5.08 $[-10.16, 0.00]$ (Confidence)
Dream			
Confidence	DAPD	-5.86 $[-12.11, 0.39]$ (DAPD)	-10.16 $[-16.80, -3.91]$ (DAPD)
DAPD	DAPD	$+16.02$ $[9.38, 22.66]$ (KLASS)	$+9.77$ $[3.52, 16.41]$ (KLASS)
KLASS	DAPD	-6.25 $[-12.50, 0.00]$ (DAPD)	-9.77 $[-16.41, -3.52]$ (DAPD)
Entropy rule	DAPD	-11.33 $[-18.36, -4.69]$ (DAPD)	-18.75 $[-25.39, -12.11]$ (DAPD)

Subsets of selectors. Without KLASS, the three remaining selectors share DAPD's implementation. In this subset, the mean recorded-count lead is 22.53 points on LLaDA and 8.40 on Dream, and both it and its shrinkage have descriptive 95% intervals excluding zero. Without entropy, Dream's lead remains positive, but its shrinkage is only 0.52 points with interval $[-3.13, 4.30]$. Each subset keeps the count sources of its own three selectors (Table 18).

Table 18: **The four-selector summary on subsets of selectors.** A_R, A_U and $A_R - A_U$ computed as in Section 4.2 over the listed selectors and their decoders. Confidence, DAPD and the entropy rule share DAPD's implementation, and KLASS uses its own. Points with descriptive 95% paired bootstrap intervals.

Model	Selectors	A_R	A_U	$A_R - A_U$
LLaDA	C, D, E (one impl.)	$+22.53$ $[19.34, 25.78]$	$+3.84$ $[1.04, 6.71]$	$+18.68$ $[14.45, 23.05]$
LLaDA	C, D, K	$+14.13$ $[11.26, 17.06]$	$+3.32$ $[0.72, 5.99]$	$+10.81$ $[6.90, 14.65]$
LLaDA	C, K, E	$+18.49$ $[15.30, 21.74]$	$+1.63$ $[-0.46, 3.65]$	$+16.86$ $[13.15, 20.57]$
LLaDA	D, K, E	$+20.64$ $[17.51, 23.83]$	$+4.36$ $[1.50, 7.23]$	$+16.28$ $[12.24, 20.31]$
Dream	C, D, E (one impl.)	$+8.40$ $[5.21, 11.65]$	$+2.28$ $[-0.65, 5.21]$	$+6.12$ $[1.82, 10.48]$
Dream	C, D, K	$+3.39$ $[0.52, 6.25]$	$+2.86$ $[0.20, 5.40]$	$+0.52$ $[-3.13, 4.30]$
Dream	C, K, E	$+5.60$ $[3.32, 7.94]$	-0.33 $[-1.95, 1.30]$	$+5.92$ $[2.99, 8.92]$
Dream	D, K, E	$+7.88$ $[4.75, 11.00]$	$+1.04$ $[-1.76, 3.84]$	$+6.84$ $[2.67, 11.07]$

C.1 Replication on ASDiv

Design. The ASDiv cohort contains the first 256 of the 1,429 problems from the hand-designed schedule study (Appendix D) after sorting the SHA256 hashes of `dvp-asdiv-20260925:asdiv:` followed by the original ASDiv identifier (for example, `nluds-0001`). That study ran only confidence and the DAPD selector under hand-designed counts, so no adaptive decoder had been run on these problems. We ran this replication after writing down the protocol (Protocol 1). The models, decoders, block structures, balanced and fixed counts, scoring and the 44 generations per problem and model are those of the GSM8K study. This gives 22,528 generations, which took about 40 H200 GPU hours. The diagonal of each matrix is the decoder's own run. In the GSM8K study, each selector reproduced its decoder exactly under the decoder's recorded counts.

Checks. Within each decoder's counts, every recorded and balanced run used exactly the decoder's number of forward passes in every block. Every balanced and fixed-count run also passed the step-by-step audit of the GSM8K study. A smoke run on two further ASDiv problems tested the code before the formal runs, and its outputs are not part of the results.

What was fixed in advance. We wrote down the following expectations before reading any output of the formal runs. A_R would be positive in both models. On LLaDA, the own selector would rank first under at least three of the four decoders' counts, and its lead would exceed the corresponding gap between the complete decoders. On Dream, the DAPD selector would rank first under most decoders' counts. The first runs covered the complete decoders and the recorded-count cells. We added the balanced and fixed-count runs after those runs had started but before any result was read. For them, we expected $A_R - A_U$ to be positive on LLaDA and stated no expectation for Dream, whose GSM8K interval had included zero under the correction over all main tests. The four main tests are A_R and $A_R - A_U$ in each model, with 98.75% intervals.

Results. On LLaDA, the own-selector ranking prediction holds under all four decoders' counts. Its lead exceeds the corresponding gap between complete decoders in 11 of 12 pairs. The exception is confidence against KLASS under C95's counts, with a lead of 0.78 points against a gap of 2.34 points between the complete decoders. Both main averages are positive in both models (Table 19). In both models, every own selector also has a positive observed lead that shrinks under balancing, so both effects meet the breadth criterion of the GSM8K study. Under balanced counts, the average lead is 0.68 points $[-1.20, 2.54]$ on LLaDA and -0.16 points $[-2.25, 1.86]$ on Dream. The written expectations did not include the reversed choice in Section 5 under the entropy rule's counts on Dream, which is the only count source on either task with a reversed choice.

Comparisons and complete decoders. On LLaDA, these own-selector leads contrast with small differences between complete decoders: the largest gap is 3.52 points, and all six pairwise 95% intervals include zero (Table 20). The entropy selector ranks last in every comparison except the one under its own recorded counts, 10.94 to 23.83 points below the best selector. As a complete decoder, the entropy rule is 0.39 points below DAPD. On Dream, the comparison under the entropy

Table 19: **ASDiv replication: lead of each decoder's own selector.** Under each decoder's counts, the accuracy of the decoder's own selector minus the mean accuracy of the other three selectors, in points, with descriptive 95% paired bootstrap intervals. The means over the four decoders are A_R (recorded, 98.75% interval) and A_U (balanced, 95% interval). The lower part gives the four main tests of the replication, A_R and $A_R - A_U$, with 98.75% intervals that correct for these four tests. It repeats them with answers scored only up to their first end-of-sequence token, and under a Bonferroni correction over all 169 main tests of the paper, including these four.

Own selector	Recorded counts	Balanced counts
LLaDA		
Confidence	$+7.16$ $[3.13, 11.20]$	$+2.08$ $[-2.21, 6.25]$
DAPD	$+11.20$ $[6.38, 16.02]$	$+7.16$ $[1.95, 12.24]$
KLASS	$+12.37$ $[7.94, 16.67]$	$+3.39$ $[-0.52, 7.29]$
Entropy rule	$+6.25$ $[1.56, 11.20]$	-9.90 $[-14.84, -5.08]$
Mean (A_R, A_U)	$+9.24$ $[6.02, 12.43]$	$+0.68$ $[-1.20, 2.54]$
Dream		
Confidence	$+10.68$ $[7.55, 13.67]$	$+3.26$ $[-0.39, 6.90]$
DAPD	$+38.28$ $[32.55, 44.14]$	$+13.28$ $[7.68, 18.75]$
KLASS	$+6.25$ $[3.26, 9.11]$	-0.78 $[-4.82, 3.13]$
Entropy rule	$+3.65$ $[-2.34, 9.64]$	-16.41 $[-22.14, -10.68]$
Mean (A_R, A_U)	$+14.71$ $[11.20, 18.33]$	-0.16 $[-2.25, 1.86]$
Main test	A_R	$A_R - A_U$
LLaDA		
All 256 problems (98.75%)	$+9.24$ $[6.02, 12.43]$	$+8.56$ $[4.56, 12.57]$
Scored to the first end-of-sequence token	$+9.18$ $[5.92, 12.37]$	$+8.82$ $[4.85, 12.84]$
Correction over 169 main tests	$[4.62, 13.80]$	$[2.80, 14.36]$
Dream		
All 256 problems (98.75%)	$+14.71$ $[11.20, 18.33]$	$+14.88$ $[10.74, 19.14]$
Scored to the first end-of-sequence token	$+14.68$ $[11.26, 18.23]$	$+15.14$ $[11.04, 19.37]$
Correction over 169 main tests	$[9.70, 19.86]$	$[8.76, 21.09]$

rule's recorded counts puts the entropy selector ahead of confidence by 7.42 points $[0.78, 14.06]$ and ahead of KLASS by 6.64 points $[0.39, 12.90]$. As complete decoders, C95 and KLASS both lead the entropy rule by 9.38 points $[5.08, 13.67]$. The complete C95 and KLASS decoders are scored identically on every Dream problem and write the same text on 253 of the 256, although their counts differ on every problem. The fixed and balanced comparisons on Dream have Kendall correlations of 0.91 with the complete-decoder ranking.

Short answers on Dream. In the single-block Dream setting, C95, KLASS and the entropy rule give empty answers on 12.9%, 12.9% and 22.7% of problems under their own counts, compared with at most 5.5% under balanced counts. Empty answers are more frequent on the diagonal of the recorded-count matrix (12.3%) than off it (7.5%). By the output-length check of the GSM8K study, whose Dream threshold is 11 words here, four Dream runs are flagged. These are the entropy rule itself, the entropy selector under the counts of C95 and of KLASS, and confidence under the entropy rule's counts. We report this check as a diagnostic, as for GSM8K. On LLaDA, at most 3.1% of answers in any run are empty.

Reversed choices and gap discrepancies on both tasks. Across all ten selector comparisons on each task and model, the only two reversed choices involve the entropy selector against confidence and against KLASS under the entropy rule's counts on Dream ASDiv (Table 21). Of the 89 gap discrepancies, 56 come from recorded counts, 21 from balanced counts and 12 from fixed counts. Among confidence, DAPD and KLASS, 17 of the 28 gap discrepancies come from recorded counts. The largest of these 28 occurs in the 256-problem GSM8K matrix. Under DAPD's counts on LLaDA, the DAPD selector leads confidence by 26.56 points, whereas the complete C95 decoder is 0.39 points more accurate than the complete DAPD decoder.

Table 20: **ASDiv replication: accuracy (%) of each selector in each comparison.** Rows are ways of comparing the four selectors on the same 256 ASDiv problems. The first row runs each decoder with its own counts, and its last line gives each decoder's mean number of forward passes. The other rows run every selector under one decoder's recorded or balanced counts, or with a fixed number k of positions per step. The most accurate selector in each row is in bold. τ is Kendall's rank correlation with the ranking of the complete decoders. LLaDA uses four 64-token blocks and Dream one 256-token block.

Comparison	Confidence	DAPD	KLASS	Entropy rule	τ
LLaDA					
Complete decoders	77.34	**78.52**	75.00	78.12	—
forward passes	77.2	111.8	78.2	101.6	
Recorded counts of C95	**77.34**	71.09	76.56	62.89	−0.33
Recorded counts of DAPD	66.02	**78.52**	71.09	64.84	0.00
Recorded counts of KLASS	65.23	71.48	**75.00**	51.17	−0.33
Recorded counts of the entropy rule	68.36	75.00	72.27	**78.12**	0.33
Balanced counts of C95	73.05	**75.00**	74.22	63.67	0.00
Balanced counts of DAPD	69.14	**74.61**	72.66	60.55	0.00
Balanced counts of KLASS	72.66	**79.30**	76.56	67.58	0.00
Balanced counts of the entropy rule	75.78	78.12	**78.52**	67.58	−0.33
Fixed $k = 2$	80.08	**84.38**	81.64	71.48	0.00
Fixed $k = 4$	75.00	**80.86**	73.44	62.11	0.33
Dream					
Complete decoders	72.27	**78.52**	72.27	62.89	—
forward passes	80.1	35.2	119.5	98.4	
Recorded counts of C95	72.27	**75.00**	72.27	37.50	1.00
Recorded counts of DAPD	48.83	**78.52**	48.83	23.05	1.00
Recorded counts of KLASS	72.27	**81.25**	72.27	44.53	1.00
Recorded counts of the entropy rule	55.47	**66.02**	56.25	62.89	0.18
Balanced counts of C95	66.80	**72.27**	67.58	50.78	0.91
Balanced counts of DAPD	63.28	**71.88**	62.11	50.39	0.91
Balanced counts of KLASS	65.62	**72.66**	64.06	56.25	0.91
Balanced counts of the entropy rule	65.62	**73.44**	66.02	51.95	0.91
Fixed $k = 2$	67.58	**75.78**	67.19	60.94	0.91
Fixed $k = 4$	69.53	**74.22**	72.27	55.08	0.91

Table 21: **Differences between selector and complete-decoder comparisons.** For each task and model, the ten comparisons (recorded and balanced counts of each decoder, and a fixed $k = 2$ or $k = 4$) each give six selector pairs, 60 in total. A *reversed choice* favors one decoder while the complete decoders favor the other, with both 95% intervals excluding zero. A *gap discrepancy* is a pair for which the selector gap minus the complete-decoder gap has a 95% paired interval excluding zero. The last column counts only the 30 pairs among confidence, DAPD and KLASS.

Task	Model	Reversed choices	Gap discrepancies (of 60)	Without entropy (of 30)
GSM8K	LLaDA	0	20	7
GSM8K	Dream	0	18	8
ASDiv	LLaDA	0	35	10
ASDiv	Dream	2	16	3

D HAND-DESIGNED SCHEDULES

Design. Every hand-designed schedule uses 32 forward passes per 64-position block, or 128 per problem, in both models. At the balanced level U, every step fills two positions. At the moderate and strong levels, the first steps fill one position each and 16 or 6 large steps fill the rest, with at most 3 or 7 positions per step. The large steps come last (ML, SL) or first (ME, SE), and each early schedule reverses its late counterpart. Confidence runs at all five levels, and the DAPD selector at U and SL.

Predictions. Before any output, we predicted three effects. More concentrated counts lower accuracy at the same number of forward passes. For the same counts, large steps placed late beat large steps placed early. The DAPD selector's lead over confidence grows with concentration. Each

model has six main tests with 99.17% intervals, and a prediction holds in a model only if all its tests do.

Results. On GSM8K, increasing concentration from moderate to strong costs confidence 10.46 and 10.16 points, and at strong concentration late steps beat early ones by 8.11 and 14.63 points (LLaDA, Dream, Table 22). The step from balanced to moderate concentration includes zero in all six settings, so the first prediction does not hold as a whole in either model. The second prediction holds on Dream, and the third on LLaDA, where the DAPD selector's lead grows by 10.61 points. On Dream GSM8K, this growth is +4.70 points, and its interval includes zero under the six-test correction. On code, strong concentration costs confidence 9.03 and 5.94 points of execution accuracy, and the DAPD selector's lead grows by 14.73 points on LLaDA.

Table 22: **Designed schedule controls.** All arms use 32 forward passes per 64-position block. Columns report GSM8K ($n = 1{,}319$) and pooled code execution (HumanEval with MBPP-sanitized, $n = 421$). The concentration rows give confidence's accuracy loss from balanced to moderate-late counts and from moderate-late to strong-late counts, and the DAPD selector's loss from balanced to strong-late counts; the placement rows give late minus early at each level; the last row is the change in the DAPD–confidence gap from balanced to strong-late counts. Entries are paired contrasts in pp with 99.17% intervals fixed for six comparisons per model–cohort family. Bold estimates have intervals excluding zero. Appendix D summarizes the MATH500 and ASDiv results.

	GSM8K		Pooled code	
Contrast	LLaDA	Dream	LLaDA	Dream
Moderate concentration	+0.45 [−2.81, 3.71]	+0.83 [−2.88, 4.50]	+2.38 [−2.85, 7.60]	+0.48 [−5.23, 6.26]
Strong concentration	**+10.46** [6.82, 14.05]	**+10.16** [6.37, 14.03]	**+9.03** [3.33, 14.49]	**+5.94** [0.71, 11.40]
DAPD concentration	+0.30 [−2.73, 3.26]	**+6.29** [2.65, 9.86]	−3.33 [−7.84, 1.19]	+1.66 [−3.56, 7.13]
Moderate placement	+0.83 [−2.65, 4.40]	**+4.47** [0.66, 8.34]	+3.33 [−1.90, 8.31]	+5.23 [−0.71, 11.16]
Strong placement	**+8.11** [4.02, 12.28]	**+14.63** [10.39, 18.88]	+4.04 [−1.90, 10.21]	**+9.03** [2.38, 15.68]
Selector-gap interaction	**+10.61** [5.99, 15.47]	+4.70 [−0.45, 9.78]	**+14.73** [7.60, 22.57]	+4.75 [−3.09, 12.59]

Other tasks. On MATH500, Dream shows a strong-concentration cost of 16.60 pp [11.40, 21.80] and a growth of the DAPD selector's lead of 16.60 pp [9.40, 23.60]. The LLaDA effects are smaller, and no prediction holds as a whole in either model. On 1,429 ASDiv problems, the lead grows by +6.09 pp [2.38, 9.80] on LLaDA and +4.20 pp [1.19, 7.28] on Dream, with 97.5% intervals. Between balanced and strong late counts, confidence falls from 79.29 to 72.92% and from 80.90 to 75.86%, while the DAPD selector barely moves, from 83.07 to 82.79% and from 83.90 to 83.07%. Under the correction over all main tests, the Dream ASDiv interval includes zero.

Balanced counts as a comparator. Balanced counts are not a proven optimum. An entropy-based rule for setting counts differs from balanced counts by −4.25 pp [−8.19, −0.53] on LLaDA and +0.83 pp [−2.50, 4.13] on Dream, and budgets that move forward passes between blocks do not clearly improve on balanced counts either.

Code under recorded counts. An earlier run balanced DAPD's recorded counts for confidence alone on the code tasks, and all four of its main tests include zero. We therefore compare the two selectors on code only under hand-designed schedules.

Table 23: **The same comparison at a common fixed count.** Gap $D-C$ (points) between the DAPD selector and confidence on the same 1,319 GSM8K problems. The first column fills two positions per step in every block, from the hand-designed schedules of Appendix D. The other columns use the recorded counts of the two adaptive decoders (Table 2). All runs share problems, prompts, checkpoints, four 64-position blocks, scorer, implementation and adaptation rule. The columns come from separate runs with different budgets, so comparisons across columns are descriptive. Brackets give 95% bootstrap intervals within each run.

Model	Common $k = 2$	DAPD's counts	C95's counts
LLaDA	+2.43 [0.15, 4.78]	+25.25 [22.37, 28.28]	−12.81 [−15.54, −10.08]
mean NFE	128.00	110.97	89.97
Dream	+3.71 [1.14, 6.29]	+23.12 [20.39, 25.93]	−11.98 [−14.56, −9.40]
mean NFE	128.00	65.28	72.39

Table 24: **New counts beat recorded ones after the same change to the text.** Accuracy in percent and Q in points, with six simultaneous paired Clopper–Pearson intervals providing joint 95% coverage. Wins:losses count the problems that only N_{new} or only N_{ref} answers correctly. GSM8K uses $2{,}084$ held-out training problems, MATH uses $4{,}096$ test problems outside MATH500, and KLASS runs on LLaDA.

Task / decoder	n	N_{ref}	N_{new}	Wins:losses	Q [interval] (pp)
GSM8K / KLASS	2,084	70.63	77.30	178:39	$+6.67$ $[4.04, 9.67]$
GSM8K / DAPD, LLaDA	2,084	68.43	81.19	319:53	$+12.76$ $[9.27, 16.49]$
GSM8K / DAPD, Dream	2,084	65.16	71.45	156:25	$+6.29$ $[3.84, 9.02]$
MATH / KLASS	4,096	30.37	31.98	123:57	$+1.61$ $[0.54, 3.00]$
MATH / DAPD, LLaDA	4,096	26.32	30.15	278:121	$+3.83$ $[2.15, 5.81]$
MATH / DAPD, Dream	4,096	31.42	36.82	303:82	$+5.40$ $[3.61, 7.40]$

A common fixed count for the pair of selectors. At the common fixed count, both selectors fill two positions per step in every block, using the same problems, prompts, checkpoints, block structure, scorer, implementation and adaptation rule as Section 4. At this common count, the DAPD selector leads confidence by 2.43 and 3.71 points (Table 23). The fixed-count ranking thus agrees with the ranking under DAPD's counts, but its lead is six to ten times smaller, and it disagrees with the ranking under C95's counts. We assembled this comparison after all three runs had been read, and its columns come from separate runs with different budgets. It is therefore a descriptive reference for one fixed count.

E Continuation after the text changes

Design. A complete run of the original decoder records its counts for four 64-position blocks. A simpler selector fills the first two blocks with these counts, and four continuations then start from identical copies of the partly filled sequence and its history. N_{ref} fits the original decoder's current proposal to the remaining recorded counts, and N_{new} lets the original decoder choose freely. S_{ref} lets the simpler selector fill the remaining recorded counts, and S_{new} lets it fill as many positions as the original decoder currently proposes. The simpler selector ranks positions by DAPD's node scores for DAPD and by confidence for KLASS. Both S continuations still compute the original proposals, so S_{new} is not a cheap standalone selector.

Problems and settings. The study uses $2{,}084$ held-out GSM8K training problems, after removing 27 near-duplicate templates, and $4{,}096$ MATH test problems outside MATH500 and the development examples. Both sets were fixed before these runs, and no other comparison in the paper uses them. The two sets are evaluated with two model families, so the six settings are not six independent populations. KLASS uses a confidence threshold of 0.6, history length 2 and KL thresholds of 0.015 (GSM8K) and 0.01 (MATH), with chat prompts and its own parser. DAPD uses its staged thresholds, $0.01/0.05$ on LLaDA and $0.005/0.05$ on Dream. GSM8K uses five-shot prompts and flexible extraction, and MATH uses the Minerva four-shot prompt on LLaDA and the official Dream prompt and scorer.

Results. Choosing new counts improves continuation accuracy in all six settings, with simultaneous paired Clopper–Pearson intervals for Q excluding zero (Table 24 and Appendix A). All intervals of the secondary interaction include zero (Table 25), while mean forward passes change little (Table 26). With the simpler selector on DAPD/LLaDA GSM8K, choosing counts from its current text raises accuracy by 11.08 points, but it stays below the original decoder.

Table 25: **Continuation accuracy and the simpler selector's gain from new counts.** Accuracies are in percent; the original decoder's continuation levels are in Table 24. "Original" is its complete run without the changed prefix. The four continuations share the same changed prefix. $Q_S = S_{\text{new}} - S_{\text{ref}}$ is the simpler selector's gain from new counts. The last column subtracts Q_S from the original selector's gain Q, with joint 95% coverage across six comparisons. Every difference interval includes zero.

Task / implementation	Original	S_{ref}	S_{new}	$Q - Q_S$ [interval] (pp)
GSM8K / KLASS	78.74	68.62	78.21	$-2.93\ [-7.05, 1.21]$
GSM8K / DAPD–L	80.37	64.40	75.48	$+1.68\ [-5.01, 8.35]$
GSM8K / DAPD–D	69.72	63.82	68.04	$+2.06\ [-2.00, 6.11]$
MATH / KLASS	32.28	28.39	31.84	$-1.83\ [-4.25, 0.60]$
MATH / DAPD–L	30.08	26.00	26.88	$+2.95\ [-0.86, 6.76]$
MATH / DAPD–D	33.64	29.57	31.91	$+3.05\ [-0.23, 6.32]$

Table 26: **Forward passes of the continuations.** "Original" counts the complete run of the original decoder. The four continuation columns count only the remaining two blocks after the common 128-position prefix. They exclude donor generation, prefix execution and diagnostic checks. The two continuations with recorded counts use the same counts for every problem. The two that choose new counts may change both the number and the allocation of forward passes.

Task / implementation	Original (full)	S_{ref}	N_{ref}	S_{new}	N_{new}
GSM8K / KLASS	97.176	37.167	37.167	39.236	38.340
GSM8K / DAPD–L	110.210	13.401	13.401	10.355	13.411
GSM8K / DAPD–D	101.373	5.669	5.669	4.704	5.879
MATH / KLASS	125.117	54.508	54.508	55.849	54.903
MATH / DAPD–L	123.609	18.438	18.438	13.765	18.385
MATH / DAPD–D	116.250	14.522	14.522	11.681	14.730

Relation to scheduling rules. Fixed-step decoding fills the same number of positions at every step (Nie et al., 2025). The DUS planner fixes a dilated order and schedule before decoding starts (Luxembourg et al., 2026). Adaptive rules choose counts from the current text (Wu et al., 2025; Ben-Hamu et al., 2025; Jazbec et al., 2026), and an entropy-sum rule of this kind is provably efficient (Cai & Li, 2026). Fast-dLLM reports higher accuracy for its threshold rule than for a fixed number of positions per step at a similar number of forward passes (Wu et al., 2025). Those rules also produce different text before later counts are chosen. Our continuations instead begin with identical text and decoding history.

Scope. The two continuations can differ in their selected positions, later text and total forward passes. Nearly equal mean forward passes do not match every problem's budget. Recording the original run and rebuilding the changed text are extra costs outside the suffix columns. Removing the 63 GSM8K problems that an earlier development run had touched leaves 2,021 problems and gives $Q = +6.78\,\text{pp}\ [4.08, 9.86]$, $+12.77\,\text{pp}\ [9.22, 16.55]$ and $+6.09\,\text{pp}\ [3.63, 8.85]$ for KLASS, DAPD on LLaDA and DAPD on Dream.

F Complete decoders

Setup. The complete decoders run on all 1,319 GSM8K test problems with DAPD's five-shot prompts, BF16 weights, 256 generated positions and batch size one on one H200 GPU. Staged and Direct run the released code, and C95 and EB0.1 use FP32 probabilities. Request time includes DAPD's graph computation. The four main tests, C95 − Direct in accuracy and in time for each model, use 98.75% intervals, and other contrasts are descriptive 95% intervals. The C95 count source in Table 2 applies the same 0.95 threshold to DAPD's native confidence values, whereas C95 here computes FP32 probabilities. These are separate implementations and runs. On LLaDA, the controlled runs reach 78.01% accuracy at 89.97 mean forward passes, while the complete-decoder runs reach 77.10% at 89.79 (Table 27). The latency in Table 4 belongs to the latter runs.

Table 27: **Mean forward passes of the complete decoders in Table 4.**

Decoder	LLaDA 4×64	Dream 1×256	Dream 4×64
Staged	110.966	56.365	101.702
Direct	62.726	55.362	65.277
C95	89.788	115.334	72.145
EB0.1	89.215	132.369	76.538

Choosing C95 and EB0.1. On 64 LLaDA GSM8K development problems, we compared three settings of each decoder, including the confidence thresholds 0.85, 0.9 and 0.95 and the EB bounds 0.01, 0.1 and 1. C95 and EB0.1 were the most accurate settings at the target request times of the native decoder and were used on Dream unchanged. EB orders positions by confidence and fills the longest prefix A_j with

$$\sum_{i \in A_j} h_i - \max_{i \in A_j} h_i \leq \gamma, \qquad \gamma = 0.1, \tag{6}$$

where h_i is the entropy of position i, and it always fills at least one position (Ben-Hamu et al., 2025).

Without the development problems. All 64 development problems belong to the 1,319 test problems. Without them, the accuracy difference between C95 and Direct still includes zero, and Direct is still faster.

Problems	n	C95	Direct	C95−Direct (pp)	C95/Direct time
All GSM8K	1,319	1017 (77.10%)	1019 (77.26%)	−0.15 [−2.96, 2.65]	1.211 [1.193, 1.229]
Without the 64	1,255	964 (76.81%)	971 (77.37%)	−0.56 [−3.43, 2.39]	1.212 [1.194, 1.232]

Time per step. Request time divided by forward passes is about 8 ms higher for DAPD than for C95 on LLaDA and about 24 ms higher on Dream (Table 28). The residual includes execution and workload differences, so it does not isolate the cost of graph construction.

Table 28: **Time per forward pass of the complete decoders.** Four-block GSM8K, one H200, batch size one. Time per step is mean request time divided by mean NFE. The final column subtracts a separately measured model forward at the median prompt length. It is a calibration residual that includes execution and workload differences; it does not isolate selector or graph-construction cost. All timings are in milliseconds except request time in seconds.

Model	Decoder	Forwards	Seconds	ms/step	Median-prompt forward	Residual
LLaDA	DAPD-Direct	62.7	3.289	52.5	43.4	**9.1**
	DAPD-Staged	111.0	5.797	52.2	43.4	**8.8**
	Confidence C95	89.8	3.982	44.3	43.4	0.9
	EB0.1	89.2	3.962	44.4	43.4	1.0
Dream	DAPD-Direct	65.3	4.126	63.2	37.1	**26.1**
	DAPD-Staged	101.7	6.408	63.0	37.1	**25.9**
	Confidence C95	72.1	2.810	39.0	37.1	1.9
	EB0.1	76.5	2.984	39.0	37.1	1.9

End-of-sequence rule and block structure. The end-of-sequence rule in Section 5.1 removes positions predicted as end-of-sequence from the candidates of a step, unless no other candidate remains, and it keeps the token predictions. On single-block Dream, it raises C95 from 41.62 to 62.55% and lowers Direct from 56.63 to 42.00% (Table 29), which reverses their ranking, with an interaction of 35.56 pp [29.49, 41.55]. On LLaDA, the rule mainly repairs a known collapse of C95, which reaches only 6.44% without it. With four blocks, the Dream interaction shrinks to +5.38 pp [1.97, 9.10] (Table 30). We fixed these tests after a pilot run and before the full evaluation, and their directions hold under three scoring rules.

Table 29: **The end-of-sequence rule on GSM8K with one block** (1,319 problems per model). Accuracy (%) under the original score, first-EOS truncation and first numeric-answer-marker truncation. C95e and De denote C95 and DAPD-Direct with the rule that delays end-of-sequence positions (Appendix F). Each problem runs all four decoders in one execution stack. Request time is measured; NFE can change.

Model	Decoder	Original	First EOS	Marker	Time (s)	NFE
LLaDA	C95	6.44	6.44	6.44	6.281	138.72
LLaDA	C95e	69.29	69.29	69.45	4.573	100.27
LLaDA	Direct	70.20	70.20	70.20	1.960	35.34
LLaDA	De	54.89	54.89	54.97	2.525	45.90
Dream	C95	41.62	41.62	41.55	4.460	115.33
Dream	C95e	62.55	62.70	62.70	6.215	160.64
Dream	Direct	56.63	57.01	56.94	3.551	55.36
Dream	De	42.00	42.15	43.52	6.018	93.60

Table 30: **The end-of-sequence rule interacts with the block structure.** GSM8K, 1,319 paired problems per model. Entries are percentage-point effects with 99.5% intervals, which correct for the ten main tests of this study. I_{64} is the accuracy gain from the EOS rule for C95 minus that for DAPD-Direct with four 64-position blocks. I_{256} is the same contrast with one 256-position block, and $\Delta I = I_{64} - I_{256}$. G_A^{EOS} is the C95e-minus-De accuracy gap with four blocks minus that gap with one block. C95e and De are defined in Table 29.

Model	I_{64}	ΔI	G_A^{EOS}
LLaDA	$+1.74\ [-0.38, 3.94]$	$-76.42\ [-82.34, -70.66]$	$-12.81\ [-17.59, -8.04]$
Dream	$+5.38\ [1.97, 9.10]$	$-30.17\ [-37.00, -23.43]$	$-17.59\ [-22.82, -12.21]$