CONTEXTUAL COMMITMENT FOR PARALLEL MASKED DIFFUSION DECODING

Anonymous authors
Paper under double-blind review

ABSTRACT

Masked diffusion decoders can use unresolved predictions as context before committing them to discrete tokens. Choosing which predictions to commit and how strongly to update this provisional context requires accounting for dependence between the two groups. Yet the model provides only per-position distributions, which do not determine their joint dependence. To address this problem, we propose Contextual Commitment (CoCom). Under one dependence budget, the selected commitments constrain how strongly pending predictions can update the next input. We couple these decisions by bounding total correlation in a reference observation of committed variables and randomly retained pending variables. The bound uses only current marginals and holds for every compatible joint distribution. Both decisions therefore reuse the current model predictions, with zero additional forward passes, no additional training, and no separately tuned update strength. Across four backbones and four benchmarks, CoCom improves the accuracy–computation tradeoff over the compared accelerated decoders. Budgets selected on LLaDA-1.5 transfer directly to LLaDA-Ins and Dream, where CoCom achieves the highest accuracy among accelerated decoders in seven of eight settings. Ablations support the benefit of selecting commitments and pending updates jointly.

1 INTRODUCTION

Masked diffusion language models (MDLMs) predict tokens at all unresolved positions in a single forward pass (Sahoo et al., 2024; Nie et al., 2025). In standard hard-token decoding, a prediction enters the next input only when it replaces a mask with a discrete token. Other predictions remain unavailable as context, even when they could help resolve uncertainty elsewhere (Hersche et al., 2026b; Zhu et al., 2025). Making them available requires choosing discrete tokens, but individually plausible predictions can be incompatible when chosen together (Gu et al., 2018). The decoder must therefore balance the value of providing context against the risk of committing incompatible predictions.

Confidence- and dependency-based selection guides which predictions become visible as discrete tokens (Wu et al., 2026; Kim et al., 2026). Verification and remasking allow these choices to be revised as context changes (Hong et al., 2026; Xiang et al., 2026). Continuous-input methods allow predictions to provide context before a discrete choice is made, preserving uncertainty in revisable inputs (Hersche et al., 2026b; Zhu et al., 2025; Ye et al., 2026). However, the dependence in the resulting context extends across committed tokens and pending predictions. The dependence associated with a pending update can change with the selected commitments, even when the pending marginals remain fixed. Separate controls on commitment dependence and pending uncertainty leave this interaction unaccounted for.

Consider an unresolved operator and two operands in a generated program. Writing the operator provides context for predicting the operands, whose current predictions may also be supplied as revisable inputs. Figure 1 uses two constructed joint distributions to illustrate the dependence between these predictions. In the left panel, the operator determines either operand, so observing an operand alongside it adds redundant information. In the right panel, the operands share information with each other but not with the operator. The operand marginals are identical across panels, and the commitment contains just one variable, giving it zero internal dependence in both cases. Nevertheless, the dependence of the combined observation differs. Separate measures of commitment cost and

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.

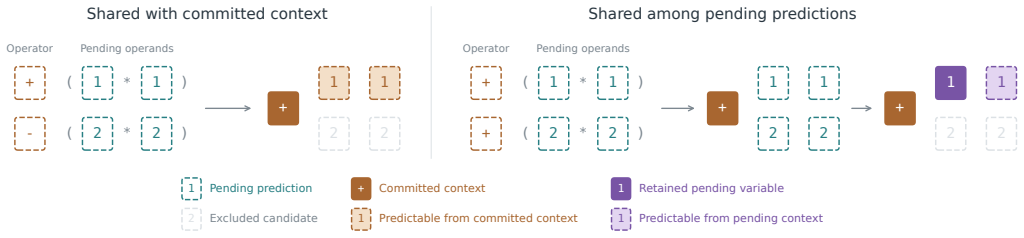


Figure 1: Shared information in committed and pending predictions. The two configurations in each panel are equally likely. Arrows show conditioning on solid-box values in a reference observation. In the left panel, either operand is predictable from the committed operator, yet retaining it adds dependence to the reference observation because they share information. In the right panel, the operator reveals neither operand, but retaining one determines the other.

pending uncertainty therefore miss an interaction that matters when both groups become context. Accounting for this interaction is difficult because the decoder receives only marginal predictions, not their joint distribution.

The marginals nevertheless contain enough information to bound this dependence across all compatible joint distributions. We propose Contextual Commitment (CoCom), which uses such a bound to select permanent writes and revisable context updates under one budget. The two decisions are represented by a reference observation that retains all committed variables and independently retains each pending variable with a common probability. Its total correlation accounts for dependence within and between the groups, while the retention probability controls how much pending information it includes. A computable upper bound on this quantity determines the feasible commitments and retention probabilities from the current marginals. For a given candidate order, CoCom first selects the largest feasible commitment prefix, then the largest feasible retention probability. Using that probability as the pending embedding-update strength makes the update depend on the selected commitments. Both decisions reuse the current model predictions with zero additional forward passes and no additional training. A single budget governs both decisions, so the pending-update strength requires no separate tuning.

Our main contributions are as follows.

- We formulate permanent commitment and revisable context updates as a joint decision under one dependence budget. The formulation captures dependence between committed and pending predictions that separate controls miss, making the permitted update strength depend on which predictions are committed even at equal hard cost and fixed pending marginals.
- We derive a computable bound on this joint dependence using only the model's current marginals, valid for every compatible joint distribution and exact for two illustrative classes. The bound gives a prefix scan and scalar search for selecting commitments and update strength. CoCom implements this solution with zero additional forward passes and no additional training, using one budget for both decisions without separately tuning the pending-update strength.
- Across four backbones and four benchmarks, CoCom combines leading accuracy among accelerated decoders with fewer model evaluations than Native, including when budgets transfer across backbones. Controlled ablations support the contributions of joint pending updates and commitment selection at comparable computation.

2 Related Work

Dependence in parallel decoding. Independent marginal predictions need not form a coherent joint output, a central difficulty in non-autoregressive generation (Gu et al., 2018). Under exact conditional marginals, conditional total correlation characterizes the error of adaptive parallel sampling (Wen et al., 2026). Confidence thresholds select predictions by marginal certainty (Wu et al., 2026), while

EB-Sampler separates model error from joint dependence error and bounds the latter to select a reveal batch (Ben-Hamu et al., 2025). DAPD and DeMask use attention-based dependencies and learned conditional influence to guide selection (Kim et al., 2026; Ringel et al., 2026). Local autoregressive decoding adds sequential conditioning (Hersche et al., 2026a), and CoCommit coordinates selected tokens through a learned marker and an additional pass through the backbone's final layers (Yao, 2026). These methods address parallel dependence by selecting which predictions to reveal or coordinating them before commitment. When pending predictions also enter the next context, dependence within the committed batch captures only part of the cost. CoCom accounts for both groups and their interaction, so different commitments can permit different pending updates even at equal hard cost.

Revision and provisional context. Iterative refinement revises provisional outputs (Lee et al., 2018; Ghazvininejad et al., 2019), while verification and remasking revisit visible tokens in diffusion decoding (Dong et al., 2026; Hong et al., 2026). COVER preserves context during verification to reduce repeated revocation of the same prediction (Xiang et al., 2026). Mixed embeddings and belief states retain information at unresolved positions (Hersche et al., 2026b; Zhu et al., 2025). ReMix combines continuous states with confidence-based commitment and rejection (Ye et al., 2026). These approaches use provisional predictions as context while allowing that context to be revised. The question here is how strongly to update this revisable context alongside permanent writes. Our bound makes that strength depend on which predictions are committed, linking persistent inputs to hard admission.

Context and decoding dynamics. Information-gain sampling evaluates successor uncertainty (Yang et al., 2026), and persistent-context analysis studies coordination through visible evidence in uniform discrete diffusion (Hayakawa, 2026). Other approaches derive continuous updates and discrete decisions from model dynamics. x-Prediction Flow learns token-wise step sizes after model alignment (Wang et al., 2026), while Sticky Jump Diffusions derives drift and commitment by reversing a common forward process (Jutras-Dubé et al., 2026). CoCom addresses the decoding decision for an existing masked model, without additional training or changes to model parameters. A shared dependence budget links which predictions become permanent to how strongly the remaining predictions enter as revisable inputs. The next section derives this joint decision from the model's current predictions.

3 Contextual Commitment

We formulate the joint choice of permanent commitments and revisable inputs at a decoding step. The derivation first separates dependence within and between these groups, then bounds the total cost using the current marginals. For a given candidate order, this bound yields a commitment prefix and its associated pending-update strength.

3.1 Decoding state and contextual commitment

At each decoding step, some predictions become permanent tokens, while the others can enter the next input as revisable context. Let M_t be the unresolved positions and s_t the current input embeddings, including the fixed prompt. One model evaluation gives a marginal distribution $p_{t,i}$ and a candidate token $w_{t,i}$ at each unresolved position,

$$p_{t,i} = p_\theta(X_i = \cdot \mid s_t), \qquad w_{t,i} = \arg\max_{v \in \mathcal{V}} p_{t,i}(v), \quad i \in M_t. \tag{1}$$

Here $\mathcal{V}$ is the vocabulary, and ties use a fixed deterministic rule. A proposed write is the labeled pair $(i, w_{t,i})$, specifying both the position and the token that becomes context.

Fix the current state and suppress t. Within the active block, let $M \subseteq M_t$ be the unresolved positions. The decision selects a nonempty commitment set $C \subseteq M$ and a common update strength $\alpha \in [0, 1]$ for the pending positions $P = M \setminus C$. Together, these choices determine which predictions become fixed context and how strongly the rest enter alongside them. We constrain both choices through the dependence of the resulting reference observation.

3.2 Dependence in the next context

To assess commitment and pending updates together, we consider an observation that retains all committed predictions and a random subset of pending predictions. Conditional on the current state, let $X \sim Q$ have the fixed marginals $p_M = (p_i)_{i \in M}$. Each pending position is retained with probability α, with retention events mutually independent and independent of X. Writing $R \subseteq P$ for the retained subset, the observation contains $X_{C \cup R}$ and erases the remaining positions (Cover & Thomas, 2006). Its total correlation (Watanabe, 1960) is the expected dependence of the retained variables (Appendix A.2),

$$F_Q(C, \alpha) := \mathbb{E}_R \, \mathrm{TC}_Q(X_{C \cup R}). \tag{2}$$

Here $\mathrm{TC}_Q(X_S) = \sum_{i \in S} H(p_i) - H_Q(X_S)$ is the sum of marginal entropies minus joint entropy, measured in nats. At $\alpha = 0$, this cost measures dependence within the commitments alone. Increasing α includes more pending predictions, reaching the full set at $\alpha = 1$.

Evaluating this cost requires a joint law, but the model supplies only marginals. We allow arbitrary dependence among the token variables and impose the budget for every joint law with these marginals. Writing $\Gamma(p_M)$ for this family, the shared constraint with budget $\beta \geq 0$ is

$$\sup_{Q \in \Gamma(p_M)} F_Q(C, \alpha) \leq \beta. \tag{3}$$

To expose the interaction between the two decisions, we separate dependence within each group from dependence between them using the entropy chain rule.

Proposition 3.1 (Dependence shared by the two decisions)**.** *For every $Q \in \Gamma(p_M)$,*

$$F_Q(C, \alpha) = \underbrace{\mathrm{TC}_Q(X_C)}_{\text{within commitments}} + \underbrace{\alpha \sum_{i \in P} I_Q(X_i; X_C)}_{\text{commitment–pending dependence}} + \underbrace{\mathbb{E}_R \, \mathrm{TC}_Q(X_R \mid X_C)}_{\text{conditional pending dependence}}. \tag{4}$$

Here $\mathrm{TC}_Q(X_R \mid X_C) = \sum_{i \in R} H_Q(X_i \mid X_C) - H_Q(X_R \mid X_C)$, where conditional entropies average over X_C. Empty sets and singletons have zero total correlation. Moreover, $F_Q(C, \alpha)$ is nondecreasing and convex in α.

The cross term in Proposition 3.1 is absent from a commitment-only constraint and determines the initial slope in α (Appendix A.2). Even a singleton commitment, whose own total correlation is zero, can therefore restrict pending retention.

Consider $m = |P|$ pending marginals all equal to p, with entropy $h > 0$. Changing only the committed singleton gives the following exact costs.

Committed marginal	Hard cost	Pending marginals	$\sup_{Q \in \Gamma(p_M)} F_Q(C, \alpha)$
Deterministic	0	$p, \ldots, p$	$[m\alpha - 1 + (1-\alpha)^m]h$
p	0	$p, \ldots, p$	$m\alpha h$

Both costs are attained by coupling the uncertain variables to a shared draw from p (Appendix A.4). Figure 2(a,b) contrasts their quadratic and linear growth near $\alpha = 0$, despite equal hard costs and pending marginals. Panel (c) shows the largest feasible strength at the bound's intersection with the budget in a HumanEval state.

3.3 A computable bound

The preceding contrast uses explicit couplings. A decoder must capture this dependence on commitment using only its marginal predictions, without enumerating joint laws or retained subsets. Let $B(C)$ bound dependence within the commitments, and let $\Delta h_{(1)}(C) \geq \cdots \geq \Delta h_{(m)}(C)$ be the sorted increments in the joint-entropy lower bound when individual pending positions join C, where $m = |P|$. These quantities give the extension bound

$$\underbrace{B_{\mathrm{ext}}(C, \alpha)}_{\text{extension bound}} = \underbrace{B(C)}_{\text{commitment cost}} + \underbrace{\alpha \sum_{i \in P} H(p_i)}_{\text{pending marginal cost}} \underbrace{- \sum_{r=1}^{m} \Delta h_{(r)}(C) \alpha (1-\alpha)^{r-1}}_{\text{joint-entropy correction}}. \tag{5}$$

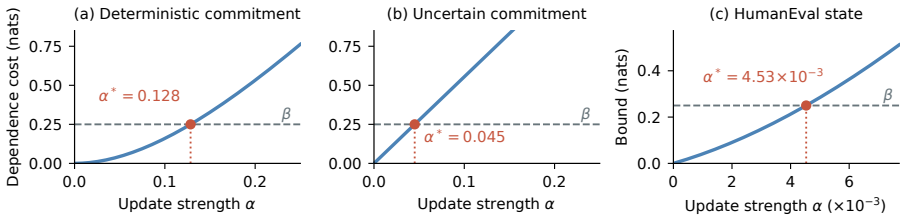


Figure 2: Commitment-dependent feasible strengths. Exact costs for eight identical binary pending marginals at $\beta = 0.25$ (a,b), and $\mathcal{B}(C, \alpha)$ for HumanEval task 0, step 1 (c). Dots mark the budget intersections.

The first two terms account for commitments and pending marginals separately. The correction subtracts the additional joint entropy guaranteed when pending variables join the commitments. It depends on which predictions are committed, even at equal hard cost and pending uncertainty. We next construct the commitment cost and entropy increments from the marginals.

Bounding dependence requires a lower bound on joint entropy, since the marginal entropies are already known. For a nonempty $S \subseteq M$, let m_S be the majorization meet of its marginals after descending rearrangement, namely their greatest lower bound under majorization. Its entropy $h_S = H(m_S)$ satisfies $H_Q(X_S) \geq h_S$ for every compatible law (Li, 2021). Subtracting this lower bound gives

$$B(S) = \sum_{i \in S} H(p_i) - h_S. \tag{6}$$

Appendix A.3 gives the finite meet construction. To extend this bound to pending updates, we examine how it changes when a position joins the commitments. The added position contributes its marginal entropy and can raise the joint-entropy lower bound. The following lemma establishes the set bound and bounds the entropy increment.

Lemma 3.2 (Set cost and entropy increments)**.** *Every compatible joint law satisfies* $\mathrm{TC}_Q(X_S) \leq B(S)$*. The bound is zero for a singleton and nondecreasing under set inclusion. For* $i \in P$*, define*

$$\Delta h_i(C) = h_{C \cup \{i\}} - h_C, \qquad 0 \leq \Delta h_i(C) \leq H(p_i). \tag{7}$$

Then $B(C \cup \{i\}) - B(C) = H(p_i) - \Delta h_i(C)$.

Lemma 3.2 lets us bound dependence in the full reference observation. The largest increment in a retained subset gives a joint-entropy correction, whose average yields B_{ext}. Convexity from Proposition 3.1 gives a second bound by interpolating between retaining no pending variables and retaining them all. The next theorem combines these bounds and establishes the monotonicity needed to select an update strength.

Theorem 3.3 (Upper bound on the total correlation of the reference observation)**.** *For all* $\alpha \in [0, 1]$ *and* $Q \in \Gamma(p_M)$*,*

$$F_Q(C, \alpha) \leq \mathcal{B}(C, \alpha) := \min\{B_{\mathrm{end}}(C, \alpha), B_{\mathrm{ext}}(C, \alpha)\}, \tag{8}$$

$$B_{\mathrm{end}}(C, \alpha) = (1 - \alpha)B(C) + \alpha B(M). \tag{9}$$

Here $B_{\mathrm{ext}}(C, \alpha)$ *is given in Equation 5. The function* $\mathcal{B}(C, \cdot)$ *is continuous and nondecreasing, with* $\mathcal{B}(C, 0) = B(C)$ *and* $\mathcal{B}(C, 1) = B(M)$.

Proof sketch. For the endpoint bound, convexity places $F_Q(C, \cdot)$ below the chord joining its endpoint costs, which are bounded by $B(C)$ and $B(M)$. For the extension bound, every retained subset satisfies $H_Q(X_{C \cup R}) \geq h_C + \max_{i \in R} \Delta h_i(C)$, with maximum zero when R is empty. The probability that rank r is the first retained increment is $\alpha(1 - \alpha)^{r-1}$. Taking the expected maximum gives B_{ext}. Appendix A proves the endpoint and monotonicity statements. □

The condition $\mathcal{B}(C, \alpha) \leq \beta$ is sufficient for Equation 3. For the two singleton examples in Figure 2(a,b), $\Delta h_i(C) = h$ and $\Delta h_i(C) = 0$, respectively, so the bound equals their exact worst-case costs. For marginals ordered by majorization, B_{ext} computes the retained-subset average of B exactly (Corollary A.1 in Appendix A.4). Evaluation requires neither subset enumeration nor auxiliary model forwards. Arithmetic costs appear in Appendix A.3.

3.4 Decision and execution

The bound jointly determines commitment and pending-update strength. Let $\pi = (i_1, \ldots, i_n)$ order the $n = |M|$ candidates from the current forward pass, with prefixes $C_k = \{i_1, \ldots, i_k\}$. The decision rule takes this order as given. Our experiments use increasing position order. We prioritize permanent progress, then maximize the expected number $(n-k)\alpha$ of retained pending predictions under the same budget,

$$\operatorname*{lex\,max}_{k \in \{1,\ldots,n\},\, \alpha \in [0,1]} \big(k, (n-k)\alpha\big) \quad \text{subject to} \quad \mathcal{B}(C_k, \alpha) \le \beta. \tag{10}$$

Although the constraint couples k and α, the endpoint and monotonicity properties in Theorem 3.3 allow us to check prefix feasibility before selecting the update strength, yielding the following solution.

Corollary 3.4 (Solution of the prefix-constrained optimization)**.** *For any fixed candidate order π, with $\alpha^* = 0$ when $k^* = n$, Equation 10 is solved by*

$$\begin{aligned} k^* &= \max\{k \in \{1, \ldots, n\} : B(C_k) \le \beta\}, \\ \alpha^* &= \max\{\alpha \in [0,1] : \mathcal{B}(C_{k^*}, \alpha) \le \beta\} \quad (k^* < n). \end{aligned} \tag{11}$$

The first stage needs only $B(C_k)$ because setting $\alpha = 0$ minimizes the bound for each prefix. Since B is nondecreasing under set inclusion, a prefix scan finds k^*, and scalar bisection then finds α^*. A singleton has zero cost, ensuring progress. Appendix A.5 gives the proof.

The selected prefix and strength determine the next input. Let e be the token embedding map and z_t record the discrete tokens on the L generation positions, with unresolved positions marked MASK. Their stored inputs $s_{t,i}$ start at the mask embedding. We replace committed inputs by their candidate embeddings and move pending inputs a fraction α^* toward their current candidates,

$$s_{t+1,i} = \begin{cases} e(w_{t,i}), & i \in C_{k^*}, \\ (1-\alpha^*)s_{t,i} + \alpha^* e(w_{t,i}), & i \in M \setminus C_{k^*}. \end{cases} \tag{12}$$

For $i \in C_{k^*}$, we also set $z_{t+1,i} = w_{t,i}$ permanently. Pending positions remain MASK in z_{t+1}, so their stored inputs can continue to change. Inputs outside M are unchanged. The retention probability thus sets the weight of pending predictions in the next input. For fixed candidates, replacing $s_{t,i}$ by $e(w_{t,i})$ with probability α^* and otherwise preserving it has exactly the expectation in Equation 12 (Appendix A.6). This connects the commitment-dependent retention probability to the interpolation used in the next model input.

Each new forward pass recomputes the marginals, the admitted prefix, and the update strength. Algorithm 1 summarizes the decoding rule.

Scope and limitations. CoCom uses the current marginals without auxiliary forward passes. Directly evaluating how alternative context updates change subsequent predictions would require additional model evaluations. The guarantee therefore concerns the current reference observation, not the accumulated embedding state or subsequent model predictions. Candidate order and argmax writers are taken as given.

4 Experiments

Our experiments assess whether joint commitment and pending updates improve the accuracy–computation tradeoff. We then examine how commitment changes feasible update strengths and use controlled ablations to evaluate the bound and the treatment of pending inputs.

We evaluate LLaDA-Ins-8B (Nie et al., 2025), LLaDA-1.5-8B (Zhu et al., 2026), Dream-Ins-7B (Ye et al., 2025), and DiffuCoder-7B-Base (Gong et al., 2026) on HumanEval (Chen et al., 2021), MBPP (Austin et al., 2021), GSM8K (Cobbe et al., 2021), and MATH500 (Lightman et al., 2024). For each model and benchmark, all methods share prompts, evaluation, a 256-token generation canvas, and one deterministic completion per task. Appendix C specifies the evaluation sets, prompts, and scoring procedures.

We compare CoCom with DAPD (Kim et al., 2026), ReMix (Ye et al., 2026), Saber (Dong et al., 2026), WINO (Hong et al., 2026), and COVER (Xiang et al., 2026), using Native, each model's

fixed 256-step decoder without early stopping, as a reference. Decoder settings and stopping rules are listed in Table 6 in Appendix C. CoCom's dataset-wise budgets are selected on LLaDA-1.5 and transferred unchanged to LLaDA-Ins and Dream. DiffuCoder uses the same budgets except on GSM8K. Appendix B.1 details selection and transfer. We assess each operating point by accuracy and mean model forward evaluations (NFE). Wall time and decoder overhead appear in Appendix D.1.

4.1 Main Results

Table 1: Accuracy and decoding work across four models and four benchmarks. Shading identifies Native. Bold marks the best result among accelerated decoders for each model and benchmark.

		HumanEval		MBPP		GSM8K		MATH500	
Model	Method	Acc. (%) ↑	Mean NFE ↓	Acc. (%) ↑	Mean NFE ↓	Acc. (%) ↑	Mean NFE ↓	Acc. (%) ↑	Mean NFE ↓
LLaDA-1.5-8B	Native	40.24	256.00	36.80	256.00	57.24	256.00	24.00	256.00
	DAPD	26.83	**31.75**	32.80	**50.64**	54.51	**36.52**	32.60	**65.50**
	ReMix	39.02	45.80	32.20	67.12	80.14	64.56	33.20	66.64
	Saber	40.85	132.93	38.40	99.36	56.86	103.18	36.00	100.10
	WINO	37.20	48.93	37.80	72.77	78.17	44.38	34.20	77.50
	COVER	36.59	69.80	38.00	97.61	**82.49**	79.01	35.80	103.61
	CoCom	**43.90**	64.76	**40.40**	51.58	80.97	61.39	**38.60**	96.37
LLaDA-Ins-8B	Native	34.76	256.00	37.00	256.00	60.50	256.00	24.40	256.00
	DAPD	25.61	**30.62**	33.80	52.63	54.28	**38.39**	32.60	**69.18**
	ReMix	32.32	43.65	35.20	70.60	76.72	66.46	33.40	69.84
	Saber	36.59	129.17	37.00	101.12	60.73	105.48	36.40	102.17
	WINO	31.71	48.00	35.40	75.38	76.95	44.42	36.40	80.04
	COVER	37.20	68.53	37.80	101.47	**80.67**	80.58	33.40	107.14
	CoCom	**41.46**	64.02	**40.20**	**52.08**	80.52	62.39	**38.00**	99.45
Dream-Ins-7B	Native	60.98	256.00	59.40	256.00	80.52	256.00	44.60	256.00
	DAPD	17.68	91.64	53.40	31.63	60.80	47.39	37.00	92.01
	ReMix	35.37	**23.41**	44.20	**27.75**	37.91	**28.66**	29.60	**49.74**
	Saber	20.73	155.96	51.20	46.84	49.73	115.00	36.20	106.28
	WINO	39.02	38.48	49.80	46.29	42.46	51.19	37.00	96.77
	COVER	43.29	40.62	47.80	55.94	71.04	63.19	34.40	119.77
	CoCom	**60.37**	49.87	**58.20**	30.86	**76.12**	49.45	**41.80**	101.92
DiffuCoder-7B-Base	Native	63.41	256.00	62.60	256.00	49.05	256.00	8.20	256.00
	DAPD	50.61	69.38	54.80	74.87	23.88	83.34	5.20	84.27
	ReMix	17.07	57.82	49.40	54.25	30.63	74.27	6.00	**68.62**
	Saber	60.37	115.51	**62.80**	121.27	42.84	120.06	7.80	109.77
	WINO	62.80	99.38	62.60	96.55	43.21	118.70	8.60	113.55
	COVER	65.85	130.96	61.60	130.70	44.73	142.78	7.40	138.29
	CoCom	**68.90**	**55.14**	60.20	**41.84**	**48.45**	**60.36**	**9.40**	107.30

Accuracy and transfer across backbones. Compared with representative accelerated baselines, CoCom achieves the highest accuracy in thirteen of sixteen combinations of models and benchmarks. Relative to the strongest baseline in each of these settings, its accuracy advantage ranges from 0.80 to 17.08 percentage points. This performance extends to backbones that reuse the budgets selected on LLaDA-1.5. With these budgets transferred unchanged, CoCom leads the accelerated decoders in seven of eight settings on LLaDA-Ins and Dream, covering both code generation and mathematical reasoning.

Accuracy–computation tradeoff. Several settings combine higher accuracy with lower computation. On LLaDA-Ins MBPP and DiffuCoder HumanEval and GSM8K, CoCom achieves both the highest accuracy and the lowest NFE among accelerated decoders. For example, on DiffuCoder HumanEval, it reaches 68.90% accuracy at 55.14 NFE, compared with 65.85% at 130.96 NFE for COVER, the most accurate baseline. Where another accelerated decoder is more accurate, CoCom offers a lower-computation operating point. On DiffuCoder MBPP, its accuracy is 2.60 percentage points below Saber, while NFE falls from 121.27 to 41.84, a 65.5% reduction. On the two LLaDA GSM8K comparisons, CoCom is 1.52 and 0.15 points below COVER while using about 22% fewer forwards.

Acceleration and wall time. Relative to Native, CoCom reduces model evaluations across all sixteen settings. On Dream, this acceleration trades some accuracy relative to Native while retaining the highest accuracy among accelerated methods. The reduction in forward evaluations also translates into shorter decoding time on LLaDA-1.5, with wall-clock speedups of 2.20–$4.81\times$ over Native and the shortest runtime among the compared decoders on MBPP. Appendix D.1 reports the accuracy–NFE operating points in Figure 6 and wall-time measurements in Table 7.

4.2 Mechanism Validation

The central prediction is that changing the commitment can change the feasible pending-update strength even when hard cost and pending marginals are fixed. We first test this property of the reference bound on LLaDA-1.5 decoding states, then examine the additional update strength it permits throughout generation. Section 4.3 evaluates the effect of the update rule on final accuracy.

Commitment-dependent update strength. We hold pending marginals and the budget fixed and compare singleton commitments with equal hard cost $B(C) = 0$. Each pair consists of the minimum- and maximum-entropy nonterminal candidates, with both positions excluded from the shared pending set. Changing the commitment changes the feasible strength in 96.6% of HumanEval pairs and 97.7% of MATH500 pairs, with mean absolute changes of 0.035 and 0.028 (Figure 3, with full summaries in Table 9). We denote the feasible strength for the maximum-entropy commitment minus that for the minimum-entropy commitment by $\Delta\alpha^*$. Every task contains at least one pair with different feasible strengths. The comparison shows that equal hard costs and identical pending marginals can admit different updates, as predicted by the joint constraint.

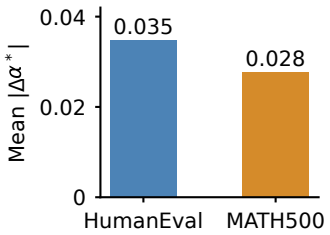


Figure 3: Commitment effect on update strength.

This sensitivity persists when the extension bound B_{ext} is used alone, with unchanged counts of affected pairs and nearly identical mean differences (Table 8 in the appendix). At fixed hard cost and pending marginals, its dependence on commitment comes through the entropy increments $\Delta h_i(C)$.

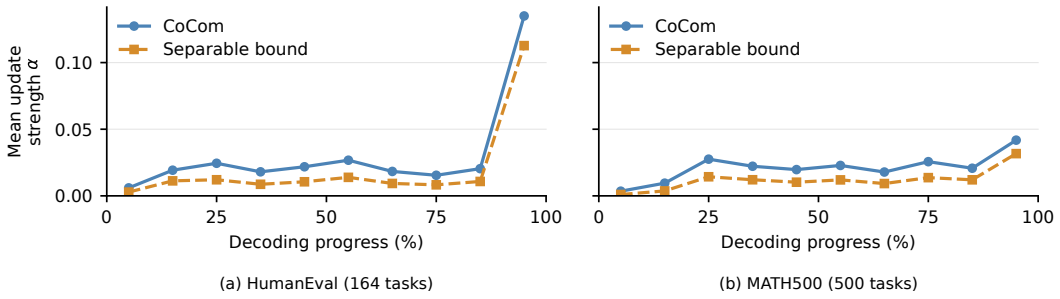


Figure 4: Update strength during decoding, averaged within task and progress bin, then across tasks with pending positions. Both bounds use the same CoCom states.

Additional update strength throughout decoding. The joint bound permits at least as much update as the separable bound at a given state. Figure 4 examines when this difference occurs along actual decoding trajectories. Mean feasible strength is larger in every progress bin on HumanEval and MATH500. A strictly larger update is permitted at 96.5% and 98.6% of pending steps, respectively, and occurs in every task (Appendix D.3). Thus, the additional update capacity is available across the decoding process, including the early and intermediate steps when many predictions remain unresolved.

4.3 Ablation Studies

We test the contributions of the joint update rule, persistent pending context, and commitment selection on LLaDA-1.5. Three variants retain CoCom's commitment rule and change pending inputs. *Separable update bound* chooses α using $B(C) + \alpha \sum_{i \in P} H(p_i) \leq \beta$, omitting the joint-entropy correction. *Mask-only pending inputs* keeps unresolved inputs at the MASK embedding. *Reset pending inputs each step* applies the CoCom update from the MASK embedding each time, discarding accumulated context.

Three further variants retain the pending-update function and replace commitment selection. *Marginal entropy threshold* screens each candidate individually using $H(p_i) \leq \beta$. *Fixed quota (4)* commits up to the first four nonterminal candidates in position order, regardless of their joint cost. *No bound*

admits all nonterminal candidates and the first terminal candidate without a dependence constraint. All variants retain the benchmark stopping rules. Table 2 reports HumanEval and MATH500 results. Appendix D.4 includes MBPP and GSM8K.

Table 2: Pending-update and commitment-selection ablations. Conditional accuracy is computed among parseable programs or extracted answers. W/T/L counts CoCom's task-level wins, ties, and losses.

(a) HumanEval

Variant	ACC (%)	Parse rate (%)	Conditional accuracy (%)	NFE	W/T/L
CoCom	**43.90**	99.39	44.17	64.76	–
Separable update bound	40.24	99.39	40.49	64.48	9/152/3
Mask-only pending inputs	40.85	98.78	41.36	65.15	7/155/2
Reset pending inputs each step	39.02	98.78	39.51	65.21	11/150/3
Marginal entropy threshold	40.85	97.56	41.88	74.49	11/147/6
Fixed quota (4)	15.24	29.27	52.08	62.95	47/117/0
No bound	0.61	5.49	11.11	3.51	71/93/0

(b) MATH500

Variant	ACC (%)	Answer extraction rate (%)	Conditional accuracy (%)	NFE	W/T/L
CoCom	**38.60**	100.00	38.60	96.37	–
Separable update bound	38.20	100.00	38.20	96.25	22/458/20
Mask-only pending inputs	37.00	100.00	37.00	95.80	29/450/21
Reset pending inputs each step	37.80	100.00	37.80	96.35	24/456/20
Marginal entropy threshold	37.80	100.00	37.80	119.33	36/432/32
Fixed quota (4)	27.60	97.20	28.40	63.39	85/385/30
No bound	2.20	80.40	2.74	2.84	184/314/2

Joint pending updates improve accuracy at similar NFE. The joint update rule improves accuracy with little change in decoding work. The separable-bound variant reduces HumanEval accuracy from 43.90% to 40.24%, while NFE changes from 64.76 to 64.48. The parse rate is unchanged, and CoCom gains nine tasks while losing three. MBPP (Table 10(c)) shows a further 2.20-point gain at similar NFE (51.58 versus 51.14), while the MATH500 gain is smaller at 0.40 points. The mask-only and reset variants also lower accuracy on all three benchmarks. Together, these comparisons support using the joint bound to update persistent pending context, with the largest gains on the code benchmarks.

Commitment selection matters at comparable NFE. Commitment selection improves accuracy beyond what can be explained by the number of forward evaluations alone. Committing a fixed quota of four nonterminal candidates gives nearly the same NFE as CoCom on HumanEval (62.95 versus 64.76), but accuracy falls from 43.90% to 15.24%. On GSM8K (Table 10(d)), NFE remains close (61.13 versus 61.39), while accuracy falls from 80.97% to 60.42% despite a 99.77% answer-extraction rate. The marginal entropy threshold also reduces accuracy while requiring more forwards on every benchmark. Unrestricted admission provides an extreme case, completing in about three to four forwards but sharply reducing accuracy.

The gains from pending-context updates are larger on the code benchmarks, while the GSM8K comparisons particularly emphasize commitment selection. Its update variants remain within 0.23 accuracy points of CoCom, whereas fixed-quota admission loses 20.55 points at comparable NFE.

5 Conclusion and Future Work

CoCom couples permanent commitment and revisable context under a shared dependence budget through a computable bound valid for all joint laws compatible with current marginals. Across four backbones and four benchmarks, it achieves strong accuracy among accelerated decoders without auxiliary model evaluations. Ablations support the joint update bound and persistent pending inputs. Future work includes extending the reference guarantee to subsequent model predictions and jointly selecting candidate order, writer tokens, and update strength.

AI USE STATEMENT

We used generative AI tools to assist with writing, generating synthetic data for illustrative examples, and writing some scripts. The authors are responsible for the research and the final content.

REPRODUCIBILITY STATEMENT

Appendix A provides theoretical assumptions and proofs. Appendix B specifies decoder implementation, budget settings, and numerical conventions. Appendix C describes benchmark prompts, baseline settings, and evaluation procedures. Appendix D provides supplementary experimental evidence, including runtime measurements, mechanism analyses, case studies, and additional ablation results. We will make the implementation available with the final paper.

REFERENCES

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. *arXiv preprint arXiv:2108.07732*, 2021. URL https://arxiv.org/abs/2108.07732.

Heli Ben-Hamu, Itai Gat, Daniel Severo, Niklas S Nolte, and Brian Karrer. Accelerated Sampling from Masked Diffusion Models via Entropy Bounded Unmasking. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 55981–56007. Curran Associates, Inc., 2025. doi: 10.52202/085713-1874.

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. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

Ferdinando Cicalese and Ugo Vaccaro. Supermodularity and Subadditivity Properties of the Entropy on the Majorization Lattice. *IEEE Transactions on Information Theory*, 48(4):933–938, 2002. doi: 10.1109/18.992785.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/abs/2110.14168.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2 edition, 2006. ISBN 9780471241959.

Yihong Dong, Zhaoyu Ma, Xue Jiang, Zhiyuan Fan, Jiaru Qian, Yongmin Li, Jianha Xiao, Zhi Jin, and Ge Li. Saber: Efficient Sampling with Adaptive Acceleration and Backtracking Enhanced Remasking for Diffusion Language Model in Code Generation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3623–3642. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.165.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. Mask-Predict: Parallel Decoding of Conditional Masked Language Models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 6112–6121. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1633.

Shansan Gong, Ruixiang Zhang, Huangjie Zheng, Jiatao Gu, Navdeep Jaitly, Lingpeng Kong, and Yizhe Zhang. DiffuCoder: Understanding and Improving Masked Diffusion Models for Code Generation. In *International Conference on Learning Representations*, 2026.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O. K. Li, and Richard Socher. Non-Autoregressive Neural Machine Translation. In *International Conference on Learning Representations*, 2018.

Satoshi Hayakawa. From Truncation to Commitment: Persistent Context in Uniform Discrete Diffusion. *arXiv preprint arXiv:2609.01043*, 2026. URL https://arxiv.org/abs/2609.01043.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring Mathematical Problem Solving With the MATH Dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021.

Michael Hersche, Nicolas Menet, Ronan Tanios, and Abbas Rahimi. Locally Coherent Parallel Decoding in Diffusion Language Models. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026a.

Michael Hersche, Samuel Moor-Smith, Thomas Hofmann, and Abbas Rahimi. Soft-Masked Diffusion Language Models. In *International Conference on Learning Representations*, 2026b.

Feng Hong, Geng Yu, Yushi Ye, Haicheng Huang, Huangjie Zheng, Ya Zhang, Yanfeng Wang, and Jiangchao Yao. Wide-In, Narrow-Out: Revokable Decoding for Efficient and Effective DLLMs. In *International Conference on Learning Representations*, 2026.

Pascal Jutras-Dubé, Patrick Pynadath, Jeremy Lu, Yuan Gao, and Ruqi Zhang. Sticky Jump Diffusions: A Unifying View of Masked, Continuous, and Hybrid Diffusion. *arXiv preprint arXiv:2607.10951*, 2026. URL https://arxiv.org/abs/2607.10951.

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*, 2026.

Jason Lee, Elman Mansimov, and Kyunghyun Cho. Deterministic Non-Autoregressive Neural Sequence Modeling by Iterative Refinement. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 1173–1182. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1149.

Cheuk Ting Li. Efficient Approximate Minimum Entropy Coupling of Multiple Probability Distributions. *IEEE Transactions on Information Theory*, 67(8):5259–5268, 2021. doi: 10.1109/tit.2021.3076986.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's Verify Step by Step. In *International Conference on Learning Representations*, 2024.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large Language Diffusion Models. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 50608–50646. Curran Associates, Inc., 2025. doi: 10.52202/085713-1689.

Liran Ringel, Ameen Ali, and Yaniv Romano. Dependency-Guided Parallel Decoding in Discrete Diffusion Language Models. *arXiv preprint arXiv:2604.02560*, 2026. URL https://arxiv.org/abs/2604.02560.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and Effective Masked Diffusion Language Models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 130136–130184. Curran Associates, Inc., 2024. doi: 10.52202/079017-4135.

Weitian Wang, Lianlei Shan, Shubham Rai, Cecilia De La Parra, and Akash Kumar. x-Prediction Flow: Efficient Continuous Decoding for Masked Diffusion Language Models. *arXiv preprint arXiv:2606.29066*, 2026. URL https://arxiv.org/abs/2606.29066.

Satosi Watanabe. Information Theoretical Analysis of Multivariate Correlation. *IBM Journal of Research and Development*, 4(1):66–82, 1960. doi: 10.1147/rd.41.0066.

Chuling Wen, Weijie Liang, and Jian Lu. Conditional Total Correlation and the Serial Depth of Adaptive Parallel Sampling. *arXiv preprint arXiv:2608.25505*, 2026. URL https://arxiv.org/abs/2608.25505.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dLLM: Training-free Acceleration of Diffusion LLM by Enabling KV Cache and Parallel Decoding. In *International Conference on Learning Representations*, 2026.

Yanzheng Xiang, Lan Wei, Yizhen Yao, Qinglin Zhu, Hanqi Yan, Chen Jin, Philip Teare, Dandan Zhang, Lin Gui, Amrutha Saseendran, and Yulan He. Stop the Flip-Flop: Context-Preserving Verification for Fast Revocable Diffusion Decoding. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Kaisen Yang, Jayden Teoh, Kaicheng Yang, Yitong Zhang, and Alex Lamb. Improving Sampling for Masked Diffusion Models via Information Gain. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Lin Yao. Don't commit alone: Joint token commitment in diffusion language models, 2026. URL https://arxiv.org/abs/2607.04469.

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. URL https://arxiv.org/abs/2508.15487.

Yushi Ye, Feng Hong, Huangjie Zheng, Xu Chen, Zhiyong Chen, Yanfeng Wang, and Jiangchao Yao. Rejection Mixing: Fast Semantic Propagation of Mask Tokens for Efficient DLLM Inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17619–17629, June 2026.

Fengqi Zhu, Rongzhen Wang, Shen Nie, Xiaolu Zhang, Chunwei Wu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. LLaDA 1.5: Variance-Reduced Preference Optimization for Large Language Diffusion Models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11425–11460. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.524.

Qinglin Zhu, Yizhen Yao, Runcong Zhao, Yanzheng Xiang, Amrutha Saseendran, Chen Jin, Philip Teare, Bin Liang, Yulan He, and Lin Gui. Latent Refinement Decoding: Enhancing Diffusion-Based Language Models by Refining Belief States. *arXiv preprint arXiv:2510.11052*, 2025. URL https://arxiv.org/abs/2510.11052.

A Theoretical Details and Proofs

All results are conditional on the current decoding state and the marginals returned by the model. Reference variables have finite support, retention gates are mutually independent and independent of these variables, and logarithms are natural. We derive the dependence bound, characterize its tightness and slack, and establish the prefix decision rule. We then explain how the selected retention probability is used in the continuous input update.

A.1 Setting and notation

At a fixed decoding step, the prompt, committed tokens, and stored pending inputs are part of the conditioning state. The active unresolved positions form M. Each marginal p_i is a distribution on the

finite vocabulary $\mathcal{V}$. For $S \subseteq M$, write $p_S = (p_i)_{i \in S}$ and $X_S = (X_i)_{i \in S}$. The compatible joint laws form

$$\Gamma(p_S) = \left\{ Q \text{ on } \mathcal{V}^S : \sum_{x_S : x_i = v} Q(x_S) = p_i(v) \text{ for all } i \in S,\ v \in \mathcal{V} \right\}.$$

This set is nonempty because it contains the product coupling. We use $H(p_i) = -\sum_v p_i(v) \log p_i(v)$ with natural logarithms and $0 \log 0 = 0$, so information quantities are in nats. For any such law, total correlation also has the representation

$$\mathrm{TC}_Q(X_S) = D_{\mathrm{KL}}\left(Q \,\middle\|\, \bigotimes_{i \in S} p_i \right) = \sum_{i \in S} H(p_i) - H_Q(X_S).$$

We take the entropy of the empty vector to be zero. Empty sets and singletons therefore have zero total correlation. Meets are used only for nonempty sets, including the nonempty commitment set C.

The candidate token w_i is the deterministic marginal argmax, with a fixed tie rule. Selecting C selects the labeled writes $\{(i, w_i) : i \in C\}$. The reference vector $X \sim Q$, in contrast, retains the full marginal uncertainty. Its joint law ranges over $\Gamma(p_M)$ when certifying the reference requirement. The same current marginals thus supply both the candidate writes and the bound.

A.2 Erasure observation and total correlation

Let $P = M \setminus C$, and let $G_i \sim \mathrm{Bernoulli}(\alpha)$ be independent for $i \in P$ and independent of X. With $\perp \notin \mathcal{V}$, define

$$Y_i = \begin{cases} X_i, & i \in C, \\ X_i, & i \in P \text{ and } G_i = 1, \\ \perp, & i \in P \text{ and } G_i = 0. \end{cases}$$

The observation identifies $R = \{i \in P : G_i = 1\}$. Consequently, $F_Q(C, 0) = \mathrm{TC}_Q(X_C)$ and $F_Q(C, 1) = \mathrm{TC}_Q(X_M)$. At intermediate strengths, independent retention controls how much pending information is observed. The following proof first expresses this cost as an average over retained sets, then derives the decomposition and convexity used to bound it.

Proof of Proposition 3.1. Write $h_{\mathrm{b}}(\alpha) = -\alpha \log \alpha - (1 - \alpha) \log(1 - \alpha)$. Each pending observation has entropy $h_{\mathrm{b}}(\alpha) + \alpha H(p_i)$. The full observation identifies the gate vector, so its joint entropy is

$$H(Y) = |P| h_{\mathrm{b}}(\alpha) + \mathbb{E}_R H_Q(X_{C \cup R}).$$

Subtracting this joint entropy from the marginal entropy sum proves Equation 2.

For a fixed R, the entropy chain rule gives

$$\mathrm{TC}_Q(X_{C \cup R}) = \mathrm{TC}_Q(X_C) + \sum_{i \in R} I_Q(X_i; X_C) + \mathrm{TC}_Q(X_R \mid X_C).$$

Averaging and using $\Pr(i \in R) = \alpha$ proves Equation 4. These steps use standard entropy and mutual-information identities (Cover & Thomas, 2006).

To establish convexity, differentiate the finite expectation with respect to each retention probability and then set them all to α. If R_{-i} retains positions in $P \setminus \{i\}$, the result is

$$F'_Q(C, \alpha) = \sum_{i \in P} \mathbb{E} I_Q(X_i; X_{C \cup R_{-i}}), \tag{13}$$

$$F''_Q(C, \alpha) = 2 \sum_{i < j,\ i, j \in P} \mathbb{E} I_Q(X_i; X_j \mid X_{C \cup R_{-ij}}) \geq 0. \tag{14}$$

Here the derivatives are with respect to α, and R_{-ij} excludes both indices. The first derivative is also nonnegative. The identities extend to endpoints by continuity. □

At zero retention, the derivative reduces to

$$\left. \frac{\partial F_Q(C, \alpha)}{\partial \alpha} \right|_{\alpha = 0} = \sum_{i \in P} I_Q(X_i; X_C).$$

Retaining one pending position has probability of order α, whereas retaining two has probability of order α^2. Dependence between committed and pending variables therefore sets the initial slope, while dependence among pending variables contributes at higher order. A constraint based only on pending uncertainty misses the first contribution.

A.3 Computable dependence bound

The preceding identities hold for each compatible joint law Q. To obtain a constraint from the marginals alone, we now lower-bound joint entropy. Recall that m_S is the majorization meet, $h_S = H(m_S)$, $B(S) = \sum_{i\in S} H(p_i) - h_S$, and $\Delta h_i(C) = h_{C\cup\{i\}} - h_C$ for $i \in P$.

Proof of Lemma 3.2. We use the entropy properties of the majorization lattice (Cicalese & Vaccaro, 2002) and give the finite construction below.

For probability vectors, $u \preceq v$ means that every descending partial sum of u is at most the corresponding sum of v, after zero padding to a common length. Write $u \wedge v$ for their majorization meet and $u \otimes v$ for the product probability vector. For nonempty S, the finite meet construction is

$$A_S(r) = \min_{i\in S} \sum_{j=1}^{r} p_i^{\downarrow}(j), \qquad m_S(r) = A_S(r) - A_S(r-1), \quad A_S(0) = 0. \tag{15}$$

Here $p_i^{\downarrow}(j)$ is the jth largest marginal probability and $r \in \{1, \dots, |\mathcal{V}|\}$. Each marginal cumulative curve $A_i(r) = \sum_{j=1}^{r} p_i^{\downarrow}(j)$ is nondecreasing and discretely concave. Their pointwise minimum $A_S(r) = \min_{i\in S} A_i(r)$ is nondecreasing. For every $i \in S$ and interior rank r, discrete concavity gives

$$2A_i(r) \geq A_i(r-1) + A_i(r+1) \geq A_S(r-1) + A_S(r+1).$$

Taking the minimum over i proves discrete concavity of A_S. Since $A_S(0) = 0$ and $A_S(|\mathcal{V}|) = 1$, its increments form a descending probability vector m_S. Let q_S be the joint mass vector, sorted in descending order and zero-padded to the common vocabulary-product length. For every marginal p_i, grouping the joint atoms according to coordinate i gives $q_S \preceq p_i$ (after zero padding), since the sum of the r largest joint atoms cannot exceed the sum of the r largest grouped masses. Thus q_S is a lower bound of the marginals in the majorization lattice, and hence $q_S \preceq m_S$, their meet. Schur concavity gives $H_Q(X_S) = H(q_S) \geq H(m_S) = h_S$ and hence Lemma 3.2.

For an extension by p_i, associativity gives $m_{C\cup\{i\}} = m_C \wedge p_i$. The product vector $m_C \otimes p_i$ is a common majorization lower bound, so

$$h_C \leq h_{C\cup\{i\}} \leq h_C + H(p_i).$$

This proves $0 \leq \Delta h_i(C) \leq H(p_i)$ and monotonicity of B. A product coupling also establishes $B(S) \geq 0$, and a singleton has its own marginal as meet, giving $B(\{i\}) = 0$. □

Proof of Theorem 3.3. We bound the same reference cost in two ways. The endpoint interpolation uses convexity, while the extension bound uses a joint-entropy restriction for each retained set.

Convexity established above yields

$$F_Q(C, \alpha) \leq (1-\alpha)F_Q(C, 0) + \alpha F_Q(C, 1) \leq (1-\alpha)B(C) + \alpha B(M).$$

For a nonempty retained R, joint entropy is at least that of every subset $C \cup \{i\}$, giving

$$H_Q(X_{C\cup R}) \geq \max_{i\in R} h_{C\cup\{i\}} = h_C + \max_{i\in R} \Delta h_i(C).$$

For empty R use the lower bound h_C. Order the $m = |P|$ increments as $\Delta h_{(1)}(C) \geq \cdots \geq \Delta h_{(m)}(C)$. Rank r is the first retained rank with probability $\alpha(1-\alpha)^{r-1}$, so the expected maximum is $\sum_{r=1}^{m} \Delta h_{(r)}(C)\alpha(1-\alpha)^{r-1}$. Subtracting the expected joint-entropy lower bound from the expected marginal entropy sum gives B_{ext}. Together with the endpoint bound B_{end}, this proves the inequality in Theorem 3.3. Ties can be ordered arbitrarily.

To prove monotonicity of B_{ext}, write

$$B_{\text{ext}}(C, \alpha) = B(C) + \mathbb{E}_R g(R), \qquad g(R) = \sum_{i\in R} H(p_i) - \max_{i\in R} \Delta h_i(C).$$

The maximum over an empty set is taken to be zero here. Adding a coordinate increases the maximum by at most $\Delta h_i(C) \le H(p_i)$, so g is nondecreasing. Coupling the retained sets with common uniform random variables proves that its expectation is nondecreasing in α. The same holds for B_{end} because $B(C) \le B(M)$, and therefore for their minimum. Both are continuous. At zero they equal $B(C)$. At one, $B_{\mathrm{end}} = B(M)$ and

$$B_{\mathrm{ext}}(C, 1) - B(M) = h_M - h_C - \max_{i \in P} \Delta h_i(C) \ge 0,$$

which proves the other endpoint. For $P = \varnothing$, use empty sums, giving $\mathcal{B}(C, \alpha) = B(C)$ for all α. □

Evaluation cost. After sorting the vocabulary probabilities, the extension terms for a fixed commitment set require $O(|M||\mathcal{V}|)$ arithmetic. Sorting the increments costs $O(|P| \log |P|)$, and each subsequent scalar evaluation costs $O(|P|)$. These operations use the current model marginals without additional model evaluations or retained-subset enumeration.

Complementarity of the two bounds. For nonempty P, write $H_{\mathrm{sum}}(P) = \sum_{i \in P} H(p_i)$, $\Delta h_M(C) = h_M - h_C$, and $J_C(\alpha) = \sum_{r=1}^{m} \Delta h_{(r)}(C) \alpha (1-\alpha)^{r-1}$, where $m = |P|$. Since $B(M) = B(C) + H_{\mathrm{sum}}(P) - \Delta h_M(C)$, the bound has the equivalent form

$$\mathcal{B}(C, \alpha) = B(C) + \alpha H_{\mathrm{sum}}(P) - \max\{\alpha \Delta h_M(C), J_C(\alpha)\}. \tag{16}$$

It subtracts the larger of two nonnegative entropy corrections from the separable cost $B(C) + \alpha H_{\mathrm{sum}}(P)$. Every strength admitted by the separable bound is therefore also admitted by $\mathcal{B}$ at the same commitment and budget. For $\alpha > 0$, the extension bound is no larger than the chord exactly when

$$\frac{J_C(\alpha)}{\alpha} = \sum_{r=1}^{m} \Delta h_{(r)}(C)(1-\alpha)^{r-1} \ge \Delta h_M(C).$$

The left side is nonincreasing. Thus the active bound can switch at most once, from B_{ext} at smaller strengths to B_{end} at larger strengths, allowing ties. If $\sum_i \Delta h_i(C) \le \Delta h_M(C)$, the chord is no larger throughout $[0, 1]$. If $\sum_i \Delta h_i(C) > \Delta h_M(C)$, the extension bound is strictly smaller for sufficiently small positive strengths. At $\alpha = 1$, $\Delta h_{(1)}(C) \le \Delta h_M(C)$ ensures that the chord is no larger. These relations characterize when each correction improves the common constraint.

A.4 Tightness and approximation slack

The derivation replaces joint entropy by the majorization lower bound and bounds the resulting average without enumerating retained subsets. We first identify when B_{ext} computes that average exactly. This does not by itself establish attainment by a compatible joint law. We then give two classes with a common coupling that attains every retained-subset bound, and finally examine cases where the joint-entropy bound has slack.

Corollary A.1 (Exact subset averaging for ordered marginals)**.** *If the current marginal vectors are totally ordered by majorization, then for every $\alpha \in [0, 1]$,*

$$\mathcal{B}(C, \alpha) = B_{\mathrm{ext}}(C, \alpha) = \mathbb{E}_R B(C \cup R).$$

This includes every family of binary marginals.

Proof. Suppose the marginal vectors are totally ordered by majorization. The meet of each nonempty set is then its least concentrated member, so $h_S = \max_{i \in S} H(p_i)$. For every retained subset R,

$$h_{C \cup R} = h_C + \max_{i \in R} \Delta h_i(C),$$

with maximum zero for empty R. The extension bound consequently equals $\mathbb{E}_R B(C \cup R)$. To compare it with the chord, set $\Delta h_{(m+1)}(C) = 0$ and write

$$J_C(\alpha) = \sum_{r=1}^{m} \big(\Delta h_{(r)}(C) - \Delta h_{(r+1)}(C)\big)\big[1 - (1-\alpha)^r\big].$$

Each bracket is concave in α, so J_C is concave and B_{ext} is convex. For ordered marginals, $\Delta h_M(C) = \Delta h_{(1)}(C)$, giving $B_{\text{ext}}(C, 1) = B(M)$. Together with $B_{\text{ext}}(C, 0) = B(C)$, convexity implies $B_{\text{ext}} \le B_{\text{end}}$ and hence

$$\mathcal{B}(C, \alpha) = B_{\text{ext}}(C, \alpha) = \mathbb{E}_R B(C \cup R). \tag{17}$$

The empty-pending case follows directly. This class includes all binary marginal families and does not require identical marginals. Thus B_{ext} computes the retained-subset average exactly for this class. □

To establish tightness of the reference cost itself, the subset bounds must be attained by the same joint law on X_M. The two constructions below satisfy this stronger requirement.

Attainment of the two cases in Section 3. Suppose X_C is deterministic and all m pending marginals are p. For a nonempty retained subset of size r, joint entropy is at least $h = H(p)$, so its total correlation is at most $(r-1)h$. Making all pending variables identical attains every subset bound simultaneously. Consequently the largest expected cost is

$$h\,\mathbb{E}(|R| - 1)_+ = h\{m\alpha - [1 - (1-\alpha)^m]\}.$$

Here $(x)_+ = \max\{x, 0\}$, $B(C) = 0$ and every $\Delta h_i(C) = h$, so B_{ext} attains the same value. If instead C contains one variable with marginal p, every retained set contains at least one such variable. Its cost is at most $|R|h$, attained by making all variables identical. Thus the worst cost is $m\alpha h$, and now every $\Delta h_i(C) = 0$, making both B_{end} and B_{ext} exact. Both constructions attain the supremum over compatible couplings. At the other extreme, the product coupling has zero cost for the same marginals. □

Small budgets at a fixed commitment. The two attained cases also give different scaling laws for feasible strength. This comparison holds the commitment set fixed to isolate its effect on the pending update. The prefix optimization considered below also changes the size of that set. Fix $m \ge 2$ identical pending marginals of entropy $h > 0$. For a deterministic singleton commitment,

$$\mathcal{B}(C, \alpha) = h\binom{m}{2}\alpha^2 + O(\alpha^3), \qquad \alpha_C^*(\beta) \sim \sqrt{\frac{2\beta}{m(m-1)h}} \quad (\beta \downarrow 0),$$

where $\alpha_C^*(\beta)$ is the largest feasible strength with C held fixed. The separable rule instead gives $\alpha_{\text{sep}} = \beta/(mh)$ for small budgets. Replacing the committed marginal by p, with the same pending marginals and singleton hard cost, gives $\alpha_C^*(\beta) = \beta/(mh)$ as well. Thus the commitment can change the asymptotic order of feasible strength, and a fixed rescaling of the separable budget cannot reproduce both cases. These comparisons fix C before optimizing strength. In the deterministic case, when this singleton is the first candidate, the prefix rule also admits the next variable at zero hard cost. Its selected prefix therefore differs from the singleton held fixed here.

Slack in the joint-entropy bound. The attained cases above do not make the majorization bound exact for arbitrary marginals. To examine its slack without also approximating a retained-subset average, consider the commitment-only endpoint, where $\mathcal{B}(C, 0) = B(C)$. Finite coupling examples quantify the gap between this bound and the largest compatible total correlation.

Fix the marginal vectors at a single state and define

$$\mathrm{TC}^*(S) = \max_{Q \in \Gamma(p_S)} \mathrm{TC}_Q(X_S), \tag{18}$$

$$\Delta(S) = B(S) - \mathrm{TC}^*(S) = \min_{Q \in \Gamma(p_S)} H_Q(X_S) - H(m_S) \ge 0. \tag{19}$$

The maximum and minimum exist because the finite coupling polytope is compact and entropy is continuous. The quantity Δ measures the slack of the majorization bound relative to the largest dependence consistent with the available marginals. It is distinct from the slack relative to a particular joint distribution. For any $Q \in \Gamma(p_S)$,

$$B(S) - \mathrm{TC}_Q(X_S) = \Delta(S) + \big[\mathrm{TC}^*(S) - \mathrm{TC}_Q(X_S)\big]. \tag{20}$$

The second term depends on the particular coupling. For example, a product coupling has zero total correlation even when $\Delta = 0$ and the worst compatible coupling attains a positive bound. The decomposition separates the bound's approximation slack from variation across couplings.

When a feasible coupling has joint masses equal to the meet entries up to permutation and zero padding, its joint entropy is $H(m_S)$ and the bound is attained. Identical marginals admit such a coupling by making the coordinates identical. Another example is a deterministic coarsening of one coordinate, when its finer marginal is the meet.

We compute both attained and unattained cases using two coordinates with binary and ternary supports. For a binary marginal $p = (b, 1-b)$ and a K-category marginal q, every coupling is determined by a row x with

$$0 \le x_j \le q_j, \qquad \sum_{j=1}^{K} x_j = b,$$

and second row $q - x$. A vertex has at most one coordinate strictly between its bounds. If two coordinates were interior, a small mass transfer between them would express the point as the midpoint of two feasible points. Conversely, fixing all but at most one coordinate at their bounds determines a vertex whenever the remaining coordinate is feasible. Concavity of entropy ensures that its minimum is attained at a vertex. We therefore inspect these vertices using exact rational arithmetic for the marginal constraints and floating-point evaluation only for entropy. This gives the global minimum for each finite example.

Table 3: Exact coupling optima and majorization-bound slack, in nats.

p	q	$B(S)$	$\mathrm{TC}^*(S)$	$\Delta(S)$
$(.5, .5)$	$(.5, .5)$	.693147	.693147	.000000
$(.5, .5)$	$(.5, .3, .2)$	.693147	.693147	.000000
$(.5, .5)$	$(.6, .4)$	.673012	.422810	.250201
$(.5, .5)$	$(.6, .2, .2)$	.613765	.422810	.190954

The first two cases attain the bound. The last two establish that the meet need not be realizable as a joint mass vector with the required marginals. For example, in the last case $m_S = (.5, .3, .2)$, whereas a minimum-entropy coupling has nonzero masses $(.5, .2, .2, .1)$. The resulting gap is approximately 0.191 nats, illustrating slack caused by the marginal constraints on a realizable joint mass vector.

For a large-vocabulary set, $B(S)$ is directly computable, but Δ additionally requires a minimum-entropy coupling. A feasible coupling $\widetilde{Q}$ can provide the certified interval

$$\mathrm{TC}_{\widetilde{Q}}(X_S) \le \mathrm{TC}^*(S) \le B(S), \qquad 0 \le \Delta(S) \le H_{\widetilde{Q}}(X_S) - H(m_S).$$

A.5 Prefix decision and budget boundaries

The preceding analysis establishes a sufficient dependence constraint, including cases where it is conservative. We now optimize under this computable constraint, rather than the unknown exact worst-case cost. For the candidate order supplied at the current step, the feasible decisions belong to the prefix family $C_1, \ldots, C_n$. The objective in Equation 10 first maximizes the number of permanent writes and then the expected number of retained pending predictions, $\mathbb{E}|R| = (n-k)\alpha$. The order and the candidate tokens are fixed within this optimization and are supplied again at the next step.

Proof of Corollary 3.4. For a given prefix, monotonicity and the value at $\alpha = 0$ imply that some strength is feasible if and only if $B(C_k) \le \beta$. Thus the largest such prefix maximizes the first coordinate of the lexicographic objective. For $k < n$, the second coordinate is strictly increasing in α. Continuity and compactness give a largest feasible strength. If $k = n$, that coordinate is zero and we choose $\alpha = 0$ by convention. □

Since $B(C_1) = 0$ and $\beta \ge 0$, at least one prefix is feasible. Monotonicity of $B(C_k)$ makes the feasible prefixes an initial segment. After choosing $C = C_{k^*}$, continuity and monotonicity of $\mathcal{B}(C, \cdot)$

make the feasible strengths a closed interval $[0, \alpha^*]$. If $B(M) \leq \beta$, full commitment is feasible and the pending set is empty. Otherwise, scalar bisection finds the endpoint of the feasible interval.

If $B(C) < \beta$ and pending positions remain, continuity ensures that a positive strength is feasible, even when the next permanent write exceeds the budget. For a general fixed commitment set with $B(C) = \beta$, positive strengths are feasible exactly where $\mathcal{B}(C, \alpha) = B(C)$. This can occur in degenerate cases. For example, a deterministic committed set and a single pending variable have zero reference cost at every strength. In contrast, if both $B_{\text{end}}(C, \alpha)$ and $B_{\text{ext}}(C, \alpha)$ exceed $B(C)$ for every $\alpha > 0$, the only feasible strength is zero.

A.6 Continuous input updates across steps

The prefix decision selects a retention probability under a bound on the token-valued reference observation. To construct the next model input, CoCom uses this probability as the fraction of each pending input replaced by its current candidate embedding. At zero strength the stored input is retained, and at unit strength it is replaced by the candidate embedding.

For a fixed candidate token and fixed current input, let $\widetilde{s}_i = (1 - G_i)s_i + G_i e(w_i)$ with $G_i \sim \text{Bernoulli}(\alpha)$. Then

$$\mathbb{E}[\widetilde{s}_i \mid s_i, w_i] = (1 - \alpha)s_i + \alpha e(w_i),$$

which gives the pending-input update in Equation 12. The reference bound governs retention of the uncertain token variable X_i, while execution uses the resulting probability to interpolate fixed argmax embeddings. The identity establishes this input-level correspondence. Dependence guarantees for subsequent nonlinear model predictions require further analysis.

Iterating the deterministic update gives the exact stored-state expansion before commitment,

$$s_{t,i} = \left[\prod_{u=0}^{t-1}(1 - \alpha_{u,i})\right] s_{0,i} + \sum_{r=0}^{t-1} \alpha_{r,i} \left[\prod_{u=r+1}^{t-1}(1 - \alpha_{u,i})\right] e(w_{r,i}),$$

where $\alpha_{u,i} = \alpha_u^*$ when position i receives a pending update at step u and $\alpha_{u,i} = 0$ otherwise. Empty products equal one and empty sums equal zero. The coefficients are nonnegative and sum to one, so the stored input is a convex combination of its initial mask embedding and previous candidate embeddings. They describe the contributions of previous predictions even when the argmax token changes across steps.

B Decoder Implementation

B.1 Budget selection and transfer across backbones

CoCom has one scalar control parameter, the dependence budget β, which jointly controls permanent commitment and pending-input updates. We select dataset-wise budgets on LLaDA-1.5 and transfer them to the other backbones, with one exception described below. Table 4 lists the resulting configuration.

Table 4: Dependence budgets β in nats.

Model	HumanEval	MBPP	GSM8K	MATH500
LLaDA-1.5-8B	0.25	0.25	0.50	0.25
LLaDA-Ins-8B	0.25	0.25	0.50	0.25
Dream-Ins-7B	0.25	0.25	0.50	0.25
DiffuCoder-7B-Base	0.25	0.25	0.15	0.25

We use the same selection procedure across benchmarks, varying β on LLaDA-1.5 while holding the other decoder settings fixed and comparing accuracy with mean NFE. Table 5(a) illustrates this procedure on the first 100 MBPP evaluation tasks. Budgets of 0.20 and 0.25 tie for the highest accuracy in this scan, while 0.25 requires fewer forward evaluations. The selected MBPP budget is then used unchanged across all four backbones.

The sole transfer exception is DiffuCoder GSM8K, where a separate 200-question scan selects $\beta = 0.15$, the highest-accuracy setting in Table 5(b). DiffuCoder is trained specifically for code generation (Gong et al., 2026). This specialization may lead to different marginal uncertainty on mathematical reasoning tasks and could partly explain the preference for a smaller dependence budget on GSM8K. All main results use the complete benchmark sets described in Appendix C.

Table 5: Budget selection examples. Panel (a) illustrates selection on LLaDA-1.5 using 100 MBPP tasks. Panel (b) shows the sole backbone-specific exception, using 200 GSM8K questions with DiffuCoder.

β	ACC (%) ↑	Mean NFE ↓
(a) LLaDA-1.5-8B, MBPP		
0.05	44.00	68.88
0.10	45.00	60.59
0.15	45.00	55.32
0.20	**46.00**	51.84
0.25	**46.00**	49.53
0.30	44.00	47.67
(b) DiffuCoder-7B-Base, GSM8K		
0.15	**52.00**	58.63
0.25	49.50	53.28
0.35	47.50	49.86
0.50	47.50	45.77

B.2 Decoding loop and special tokens

The benchmark protocol instantiates the candidate order π as increasing position order, which the context ablations retain while varying the pending-update rule.

Algorithm 1 states the budget-feasible rule of Corollary 3.4 for the supplied candidate order. Both commitment and pending updates use the same marginal distributions and set cost.

The hard gate evaluates the meet lazily. With exact, unmodified marginals, $h_C \geq \max_{i \in C} H(p_i)$ implies $B(C) \leq \sum_{i \in C} H(p_i) - \max_{i \in C} H(p_i)$. It can therefore accept a prefix when this cheaper

Algorithm 1 Contextual Commitment (CoCom), prefix and update selection

Require: Frozen model p_θ, prompt, canvas length L, block length b, fixed budget $\beta \geq 0$, candidate-order rule

Initialize $z_i \leftarrow [\text{MASK}]$ and $s_i \leftarrow e([\text{MASK}])$ for $i \in [L]$
Prepend fixed prompt embeddings to s and partition generation positions into blocks of length b
while unresolved positions remain and the stopping rule is not satisfied **do**
    $M \leftarrow$ unresolved positions in the block containing the leftmost mask
    $(p_i)_{i \in M} \leftarrow p_\theta(\cdot \mid s)$ ▷ One model forward pass
    $w_i \leftarrow \arg\max_v p_i(v)$ for $i \in M$
    Order M as $\pi = (i_1, \ldots, i_n)$ and define $C_k = \{i_1, \ldots, i_k\}$
    $k^* \leftarrow \max\{k \in \{1, \ldots, n\} : B(C_k) \leq \beta\}$ ▷ A singleton is feasible
    $C \leftarrow C_{k^*}$ ▷ The selected commitment satisfies $B(C) \leq \beta$
    if some $w_i = [\text{MASK}]$ for $i \in C$ **then**
        stop because a selected candidate is MASK
    end if
    $P \leftarrow M \setminus C$ and $\alpha \leftarrow 0$
    if $P \neq \varnothing$ **then**
        Compute $B(M)$ and $(\Delta h_i(C))_{i \in P}$ using the same marginals as admission
        $\alpha \leftarrow \max\{a \in [0, 1] : \mathcal{B}(C, a) \leq \beta\}$ ▷ Scalar search with feasible endpoint 0
    end if
    $s_i \leftarrow (1 - \alpha)s_i + \alpha\, e(w_i)$ for $i \in P$ ▷ z_i remains MASK
    $z_i \leftarrow w_i$ and $s_i \leftarrow e(w_i)$ for $i \in C$ ▷ Permanent commitment
    Apply the benchmark stopping rule and retain all other stored inputs
end while
return the completion and its stopping status

upper bound is at most β and otherwise evaluate the meet. In exact arithmetic, this shortcut gives the same acceptance decisions as directly testing $B(C) \leq \beta$.

The hard gate uses FP32 entropies for prefix screening. When a meet calculation is needed, it subtracts the meet entropy gain on the retained coordinates from this entropy bound. The hard and pending calculations reuse the sorted marginal probabilities. The pending update recomputes its costs from normalized FP64 marginals and uses 52 bisection iterations, returning the feasible lower endpoint.

EOS, PAD and MASK are special tokens handled separately during admission. The scan skips leading special-token candidates until an ordinary token is admitted, falling back to the leftmost candidate if all are terminal. A MASK argmax invokes the failure guard. Decoding stops at mask exhaustion or a generated EOS with no preceding unresolved position. Benchmark-specific stopping conditions are listed in Table 6.

C Evaluation Protocols

CoCom inference used one NVIDIA GeForce RTX 4090 or NVIDIA vGPU-48GB per worker with BF16 precision, PyTorch 2.8.0 and CUDA 12.8, and Transformers 4.46.2 for LLaDA-1.5 and Dream or 4.57.0 for LLaDA-Ins and DiffuCoder.

We evaluate 164 HumanEval tasks (Chen et al., 2021), the official 500-task MBPP slice (rows 10–509), all 1,319 GSM8K test questions (Cobbe et al., 2021), and the 500 test problems in `HuggingFaceH4/MATH-500`, a subset of MATH (Hendrycks et al., 2021).

Table 6 gives sampling settings and stopping rules. All reported decoder runs use deterministic sampling under fixed model weights, prompts, and tie-breaking. Block-length tuples follow the order HumanEval, MBPP, GSM8K, and MATH500. Extraction is evaluator-side unless a generation stop is explicitly listed. CoCom's EOS handling is specified in Appendix B.2. Each task is evaluated with one deterministic completion.

Table 6: Decoder-specific settings and stopping rules.

Decoder	Sampling and decoder parameters	Generation stopping
Native	256 steps. Deterministic sampling for all backbones.	All 256 scheduled forwards with no early stop.
DAPD	Direct selection, layer ratio 0.3. Blocks $(256, 128, 256, 128)$. LLaDA $\tau \in [0.01, 0.05]$, with GSM8K $[0.005, 0.05]$. Dream and DiffuCoder $[0.005, 0.01]$.	Active-mask exhaustion.
ReMix	Block length 128. Confidence threshold 0.8, or 0.9 on GSM8K. Mixing ratio 0.5 and JS threshold 0.3. Official LLaDA rules with backbone-interface adaptation for Dream and DiffuCoder.	Active-mask exhaustion. Evaluation uses text before the first EOS. NFE and time cover the full canvas.
Saber	$n = 2$. $\mu = 2$ on HumanEval and 8 otherwise. Blocks $(256, 128, 256, 128)$.	Standard mask-scheduling termination.
WINO	Block length 128. Threshold 0.7, or 0.6 on GSM8K. Backward threshold 0.9.	Verification and mask-scheduling termination.
COVER	Draft threshold 0.9. At most 15 unmasked positions per step. Attention guidance and KV-cached reverification. Maximum reverification 2, or 5 on GSM8K. Blocks $(64, 128, 64, 128)$.	Draft, verification, and mask-exhaustion schedule.
CoCom	Block length 64. Fixed budgets in Table 4. Prefix admission and persistent pending inputs.	EOS/mask rule. MBPP also uses a code-fence stop. DiffuCoder uses HumanEval and GSM8K answer boundaries.

C.1 Prompt construction

HumanEval and MATH500 are zero-shot. MBPP uses rows 1–3 as fixed demonstrations, each containing a problem, test cases, and reference code. GSM8K uses the first eight training examples in dataset order, retaining their full reference solutions and terminal `#### answer`. Each message is rendered with `apply_chat_template` and `add_generation_prompt=True`. The masked canvas follows the assistant marker immediately, without assistant-side prefill.

C.2 Evaluation

HumanEval. Task success requires passing all assertions in the task's test suite within a 10-second execution timeout. The evaluator assembles the generated function body or extracted target definition with the canonical prompt and imports. For DiffuCoder, extraction accommodates continuation bodies and complete function outputs while retaining generated dependencies. Its program selection uses text and syntax alone, before assertion execution.

MBPP. The evaluator extracts the first code block, adapts the function name, and requires all assertions to pass, with a 10-second timeout per assertion. DiffuCoder extraction preserves the

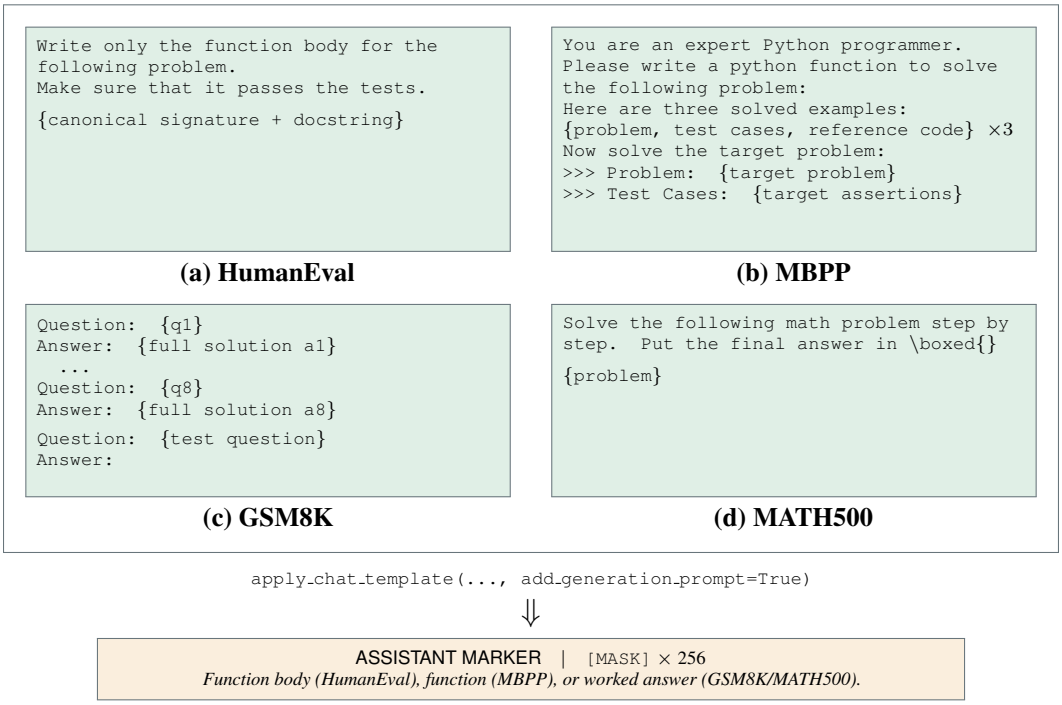


Figure 5: User-message templates for all four benchmarks and their shared conversion to a masked generation canvas. Braces denote inserted content.

complete program. Its shared evaluation protocol uses relative tolerance 10^{-9} and absolute tolerance 10^{-12} for floating-point equality assertions, including nested containers, while preserving exact comparisons for discrete values and container structure. For both code benchmarks, parse rate measures syntactic validity, assertion fraction averages the proportion of assertions passed with zero for invalid programs, and conditional accuracy is $P(\text{pass} \mid \text{parse})$.

GSM8K. Extraction prioritizes the final boxed answer, then a #### answer, followed by labelled or unambiguous concluding values, with the last numeric token as fallback. Predictions are compared exactly with the reference answer after numeric-format normalization.

MATH500. Extraction prioritizes boxed answers, with fallbacks for answer tags, final-answer lines, and the last numeric value. Predictions and references undergo the same LaTeX normalization and are compared by normalized equality or symbolic equivalence, with numerical tolerance 10^{-6}. Structured answers are compared componentwise, preserving dimensions and ordering except for unordered solution sets. Conditional accuracy for both mathematical benchmarks is measured among successfully extracted answers.

D Additional Experiments and Analysis

D.1 Accuracy and computational cost

Figure 6 compares accuracy and NFE across all sixteen combinations of models and benchmarks.

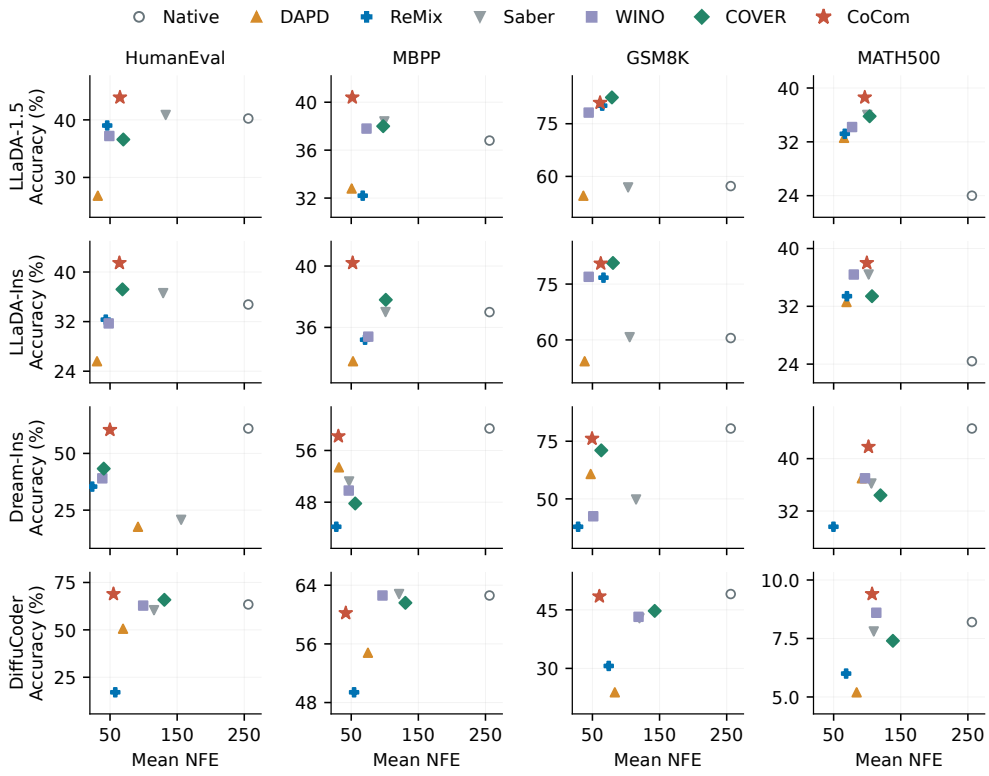


Figure 6: Operating points for accuracy and NFE from Table 1 across all sixteen panels, one for each model and benchmark. Native is shown with an open circle.

Table 7 reports decoder wall time on LLaDA-1.5, including all decoder operations and work not captured by the forward count alone. For example, WINO's shadow positions increase the number of positions processed within forward evaluations by 7.29%–25.92%.

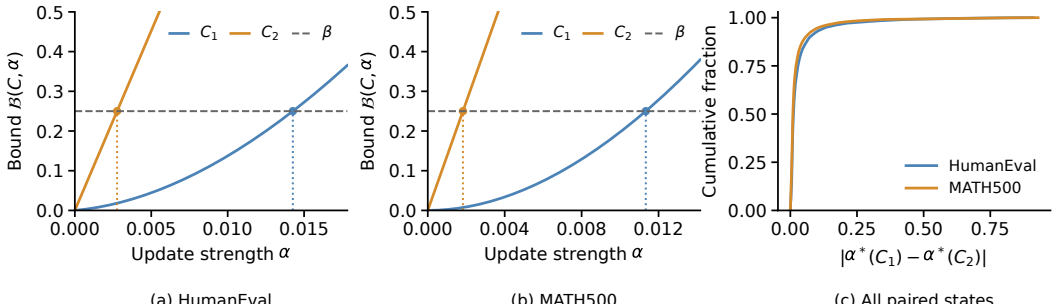
Table 7: Decoder time per task on LLaDA-1.5 in seconds, excluding model loading and evaluation.

Model	Method	HumanEval	MBPP	GSM8K	MATH500
LLaDA-1.5	Native	14.86	31.54	52.31	12.53
	DAPD	1.91	6.69	8.28	3.33
	ReMix	2.94	9.20	14.49	3.67
	Saber	7.52	12.02	20.61	4.84
	WINO	4.50	16.34	17.43	6.02
	COVER	4.47	13.72	18.85	5.81
	CoCom	4.38	6.55	12.51	5.70

D.2 Commitment-dependent feasible updates

We evaluate commitment-dependent feasibility at 10,620 HumanEval and 48,186 MATH500 decoding states from LLaDA-1.5, covering 164 and 500 tasks, respectively. At each eligible state, we select the minimum- and maximum-entropy nonterminal argmax labeled pairs, breaking ties by position. Both candidate positions are excluded from a common pending set P. Each paired reference observation

then contains P and one candidate as its singleton commitment. The unselected candidate is outside that reference observation. Thus the comparison holds pending marginals, the budget, and $B(C) = 0$ fixed while changing the committed prediction. Candidate selection depends only on the current marginal entropies. Figure 7 shows the paired costs and the distribution of feasible-strength differences.



(a) HumanEval (b) MATH500 (c) All paired states

Figure 7: Commitment-dependent feasible updates. Left and middle show pairs nearest the median strength difference. The right panel shows the distribution over all pairs. Dashed lines mark the budget and dotted lines mark feasible strengths.

Table 8: Commitment sensitivity with and without endpoint interpolation. Both bounds give the same changed-pair counts.

	Changed pairs	Mean $\lvert\Delta\alpha^*\rvert$	
Dataset		$\mathcal{B}$	B_{ext} alone
HumanEval	9,955/10,309	0.03469	0.03464
MATH500	46,003/47,067	0.02766	0.02762

For fixed $B(C)$, α, and pending entropies, B_{ext} depends on commitment through the extension increments alone,

$$B_{\text{ext}}(C, \alpha) = B(C) + \alpha \sum_{i \in P} H(p_i) - \mathbb{E}_R \max_{i \in R} \Delta h_i(C),$$

where the maximum over the empty set is zero. Table 9 summarizes the changes under $\mathcal{B} = \min\{B_{\text{end}}, B_{\text{ext}}\}$. It complements Table 8 with medians and means that weight tasks equally.

Table 9: Feasible-strength changes with fixed pending marginals. Changed pairs have $|\Delta\alpha^*| > 10^{-10}$. Task mean weights tasks equally. The other summaries weight pairs equally.

Dataset	Pairs	Changed	Mean $\lvert\Delta\alpha^*\rvert$	Median	Task mean
HumanEval	10,309	9,955	0.03469	0.01150	0.03530
MATH500	47,067	46,003	0.02766	0.00951	0.02844

D.3 Decoding progress and budget use

Figure 8 separates permanent progress from pending updates on HumanEval and GSM8K. The selected hard-set cost and combined cost show how the same budget accommodates both decisions. Progress is divided into ten equal bins relative to each task's total NFE. Update strength and budget use are averaged over steps with pending positions whose commitment cost $B(C)$, recomputed from normalized FP64 marginals, does not exceed β.

In Figure 4, the joint bound permits a strictly larger update than the separable bound at 9,699/10,047 HumanEval pending steps and 45,654/46,289 MATH500 pending steps, using a difference threshold of 10^{-10}. Every task contains such a step (164/164 and 500/500, respectively). Ten equal progress bins are defined by the forward index divided by the task's total NFE. Strengths are averaged within

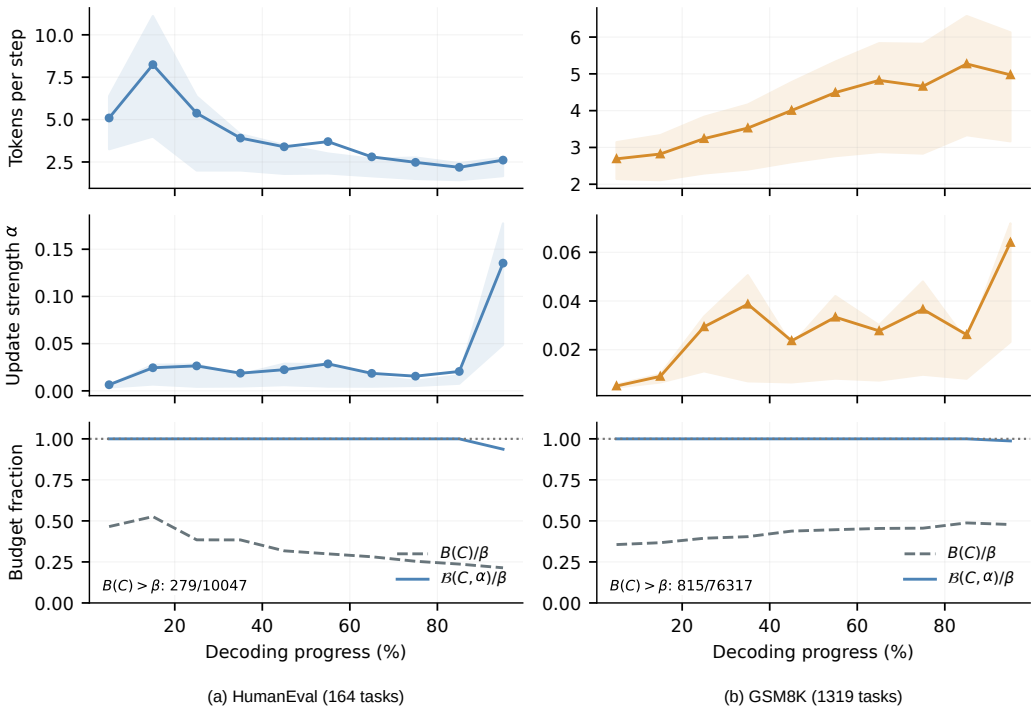


Figure 8: Decoding progress, update strength, and budget use on LLaDA-1.5. Curves show means and shading the interquartile range across task-bin averages.

each task and bin, then across contributing tasks. All pending steps are included, with both strengths set to zero when $B(C) > \beta$.

D.4 Ablation studies

Table 10 extends the main-text ablation to all four benchmarks on LLaDA-1.5. The six variants test pending-input updates and permanent-commitment selection.

Pending updates. Mask-only inputs remove the revisable context, while resetting pending inputs removes its persistence across steps. The separable-bound variant retains this state but replaces the contextual update constraint with $B(C) + \alpha \sum_{i \in P} H(p_i) \le \beta$, giving

$$\alpha_{\text{sep}} = \min \left\{ 1, \frac{\beta - B(C)}{\sum_{i \in P} H(p_i)} \right\}.$$

This expression applies when $B(C) \le \beta$, with $\alpha_{\text{sep}} = 1$ for a zero denominator and $\alpha_{\text{sep}} = 0$ when $B(C) > \beta$. Since $\Delta h_i(C) \ge 0$, $\mathcal{B}(C, \alpha) \le B_{\text{ext}}(C, \alpha) \le B(C) + \alpha \sum_{i \in P} H(p_i)$, so this rule permits no larger update under the same budget at a given state.

All three variants have lower accuracy than CoCom on HumanEval, MBPP, and MATH500 at similar NFE in these comparisons. On MBPP, the separable-bound variant loses 2.20 percentage points with little change in parse rate. The effect is smaller on GSM8K, where all three variants remain within 0.23 points of CoCom. The separable rule yields six fewer correct HumanEval solutions overall.

Commitment selection. These variants retain the pending-update and stopping rules while changing the submitted set. Marginal threshold admits positions with entropy at most β, fixed quota admits the first four non-terminal positions, and no bound admits all ordinary positions plus the first terminal candidate.

Marginal threshold requires more forward passes on every benchmark without matching CoCom's accuracy. Fixed quota also loses accuracy at nearly matched NFE on HumanEval and GSM8K,

showing that commitment selection matters beyond the number of model evaluations. Removing the bound produces the largest losses. The failures include invalid code and incorrect reasoning, with fixed-quota GSM8K accuracy falling to 60.42% despite a 99.77% answer-extraction rate. Thus, the strong sensitivity to admission on GSM8K persists even where the pending-update variants yield similar accuracy.

Table 10: Complete ablation results. Conditional accuracy is measured among parseable programs or extracted answers. W/T/L counts CoCom's task-level wins, ties, and losses.

(a) HumanEval

Variant	ACC (%)	Parse rate (%)	Conditional accuracy (%)	Assertion fraction	NFE	W/T/L
CoCom	**43.90**	99.39	44.17	0.65	64.76	–
Separable update bound	40.24	99.39	40.49	0.63	64.48	9/152/3
Mask-only pending inputs	40.85	98.78	41.36	0.64	65.15	7/155/2
Reset pending inputs each step	39.02	98.78	39.51	0.63	65.21	11/150/3
Marginal entropy threshold	40.85	97.56	41.88	0.62	74.49	11/147/6
Fixed quota (4)	15.24	29.27	52.08	0.18	62.95	47/117/0
No bound	0.61	5.49	11.11	0.01	3.51	71/93/0

(b) MATH500

Variant	ACC (%)	Answer extraction rate (%)	Conditional accuracy (%)	Assertion fraction	NFE	W/T/L
CoCom	**38.60**	100.00	38.60	–	96.37	–
Separable update bound	38.20	100.00	38.20	–	96.25	22/458/20
Mask-only pending inputs	37.00	100.00	37.00	–	95.80	29/450/21
Reset pending inputs each step	37.80	100.00	37.80	–	96.35	24/456/20
Marginal entropy threshold	37.80	100.00	37.80	–	119.33	36/432/32
Fixed quota (4)	27.60	97.20	28.40	–	63.39	85/385/30
No bound	2.20	80.40	2.74	–	2.84	184/314/2

(c) MBPP

Variant	ACC (%)	Parse rate (%)	Conditional accuracy (%)	Assertion fraction	NFE	W/T/L
CoCom	**40.40**	98.40	41.06	0.466	51.58	–
Separable update bound	38.20	98.20	38.90	0.439	51.14	24/463/13
Mask-only pending inputs	38.20	98.00	38.98	0.442	51.16	26/459/15
Reset pending inputs each step	39.00	98.20	39.71	0.449	51.01	22/463/15
Marginal entropy threshold	38.80	98.60	39.35	0.448	63.41	24/460/16
Fixed quota (4)	18.20	42.80	42.52	0.197	40.28	120/371/9
No bound	0.00	2.00	0.00	0.000	3.78	202/298/0

(d) GSM8K

Variant	ACC (%)	Answer extraction rate (%)	Conditional accuracy (%)	Assertion fraction	NFE	W/T/L
CoCom	**80.97**	100.00	80.97	–	61.39	–
Separable update bound	80.74	100.00	80.74	–	61.52	53/1216/50
Mask-only pending inputs	80.89	100.00	80.89	–	61.31	56/1208/55
Reset pending inputs each step	80.74	100.00	80.74	–	61.59	59/1204/56
Marginal entropy threshold	80.59	100.00	80.59	–	74.39	83/1158/78
Fixed quota (4)	60.42	99.77	60.56	–	61.13	340/910/69
No bound	13.42	77.10	17.40	–	3.75	910/390/19

D.5 Case Studies

Figures 9 and 10 show LLaDA-1.5 trajectories for HumanEval task 1, MBPP and GSM8K task 0, and MATH500 task 3, comprising two passing and two failing completions. The windows follow candidate changes through commitment, aligning argmax tokens, pending-update strength, and new commitments across the active block on a common forward-step axis. Pending inputs are embedding mixtures, and the dashed curve gives the separable-bound strength at the same CoCom states. Orange borders mark candidate changes. Position indices start at zero and forward labels at one. `3 sp` denotes three spaces.

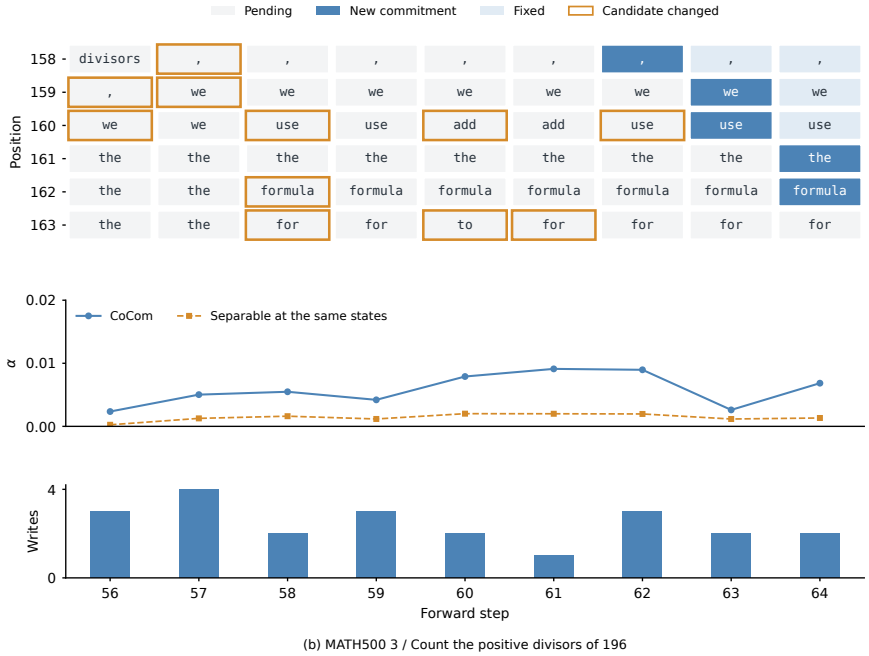


Figure 9: Code-generation trajectories on HumanEval (top) and MBPP (bottom). Candidates change across forward passes before being committed.

In Figure 9, HumanEval position 142 changes from `paren` through `in` and `_string` to `char`, which is committed at forward 24. The final program is parseable but passes 0/4 test assertions. MBPP position 47 changes from whitespace to `s` and then `result`, which is committed at forward 20. Its final code contains `result = s[:-char]` and fails all three tests.



(a) GSM8K 0 / Revenue from the remaining eggs



(b) MATH500 3 / Count the positive divisors of 196

Figure 10: Reasoning trajectories on GSM8K (top) and MATH500 (bottom). Both completions pass evaluation. The examples show candidate revision and delayed commitment of a stable candidate.

In Figure 10, GSM8K position 92 changes between `1` and `Let` before committing `1` at forward 36. The completion calculates $(16 - 3 - 4) \times 2 = 18$, and its extracted answer is correct. In MATH500, position 160 changes from `we` to `use`, then to `add`, before returning to `use` at forward

62 and committing it at forward 63. The completion correctly factors $196 = 2^2 \times 7^2$ and obtains $(2+1)(2+1) = 9$ positive divisors.