BALANCING QUALITY AND COVERAGE IN DIFFUSION LANGUAGE MODELS WITH ADAPTIVE RISK PORTFOLIOS

Anonymous authors
Paper under double-blind review

ABSTRACT

Generating multiple responses can improve reasoning performance, but its effectiveness depends on balancing individual response quality and solution coverage. In diffusion language models, conservative token commitment can improve individual responses while limiting the coverage gained from additional samples. We introduce Confidence-Adaptive Risk Portfolios (CARP), a training-free decoding method that allocates complementary commitment policies across a population of responses. A budget-dependent reserve of trajectories uses an adaptive rule that constrains the cumulative confidence deficit of eligible token proposals at each denoising step, while retaining stochastic selection within the eligible set. The remaining trajectories follow two fixed tempered confidence-threshold policies to support broader exploration. A progress-based computation guard limits prolonged constrained decoding. Together, these components balance conservative commitment and stochastic exploration without additional training or correctness feedback. Experiments on MATH-500, HumanEval, MBPP, and GSM8K with LLaDA and DREAM compare CARP with TCT and ODD. CARP achieves the highest mean pass@k at the reported budgets on MATH-500 and HumanEval, improves early-budget performance on MBPP, and remains competitive as GSM8K approaches saturation. These results support population-level commitment allocation as a practical approach to improving response quality and solution coverage in diffusion language models.

1 INTRODUCTION

Large language models are conventionally decoded autoregressively, committing one token at a time in a fixed left-to-right order (Brown et al., 2020). Diffusion language models have recently emerged as an alternative that generates sequences through repeated prediction and commitment of masked tokens (Austin et al., 2021a; Sahoo et al., 2024; Nie et al., 2025b; Ye et al., 2025). Their parallel decoding makes the choice of commitment policy critical: aggressive unmasking can introduce errors, while conservative policies consume more computation (Wu et al., 2026a; Jazbec et al., 2026). Rather than resolving this tradeoff for a single response, one can instead spend inference compute on a population of responses: coverage, the fraction of problems solved by at least one response, grows with the response budget (Brown et al., 2024; Snell et al., 2025), though how much it grows depends heavily on how different the sampled responses are from each other (Levi, 2025; Kamigaito et al., 2025). Self-consistency-style aggregation over samples improves reasoning performance in a related way (Wang et al., 2023), and in code generation specifically, success is often defined by at least one correct response within a population (Kulal et al., 2019; Chen et al., 2021), so individual-response quality alone does not characterize the objective. This raises a parallel question for diffusion decoding. Because coverage gains come from diversity rather than population size alone, what matters is not only *how many* responses are generated but also *how* each trajectory's commitment policy affects its own quality and, in turn, the population's coverage at a given budget.

Existing diffusion decoding methods pursue diversity by applying one decoding rule uniformly across the population (Lamont et al., 2026; Olausson et al., 2026). This leaves an open question: whether every trajectory in a finite response population must share the same commitment policy, or whether

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.

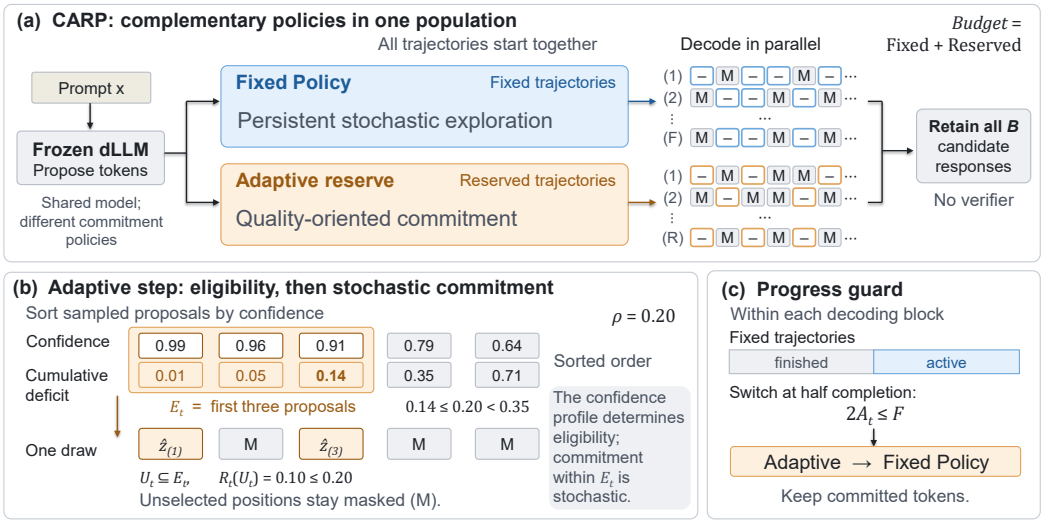


Figure 1: Overview of Confidence-Adaptive Risk Portfolios (CARP).

complementary policies — without an additional model or a correctness verifier — can better allocate a fixed inference budget between quality and coverage.

A finite response budget creates two competing requirements: each trajectory must produce a high-quality solution, while the population must also explore enough alternatives to solve problems that individual trajectories miss. Applying the same conservative rule to every trajectory risks spending extra computation on similar outcomes (Olausson et al., 2026), which motivates assigning different commitment roles to trajectories within a single population.

Therefore, we propose Confidence-Adaptive Risk Portfolios (CARP), a training-free decoding method that jointly specifies population allocation and commitment constraints. A budget-dependent minority of trajectories uses a more cost-heavy, detailed decoding strategy with adaptive confidence thresholds. The remaining trajectories retain stochastic commitment under two persistent confidence thresholds. The method requires no correctness verifier.

Our contributions are as follows. First, we formulate heterogeneous token commitment as a population-level decoding policy. Second, we introduce a commitment rule that caps the summed confidence deficit an adaptive trajectory commits at each step while retaining stochastic selection within the eligible set. Third, we evaluate CARP against baselines on four benchmarks with two models and demonstrate overall better performance. We also use component ablations to demonstrate the effectiveness of our design. Algorithm portfolios and confidence-profile decoding provide relevant precedents (Gomes & Selman, 2001; Kasa et al., 2026); our contribution is their specific integration into a stochastic diffusion decoding population.

2 Related Work

Scaling and diversity in inference-time compute. Repeated sampling scales inference-time compute by generating more opportunities to solve a problem (Brown et al., 2024), with pass@k measuring whether at least one candidate is correct (Chen et al., 2021). Adaptive allocation offers a complementary approach by matching compute to problem difficulty (Snell et al., 2025; Yu et al., 2025; Zuo & Zhu, 2026). Levi (2025) provides a statistical model of repeated-attempt coverage, while Kamigaito et al. (2025) analyze inference scaling through a bias–diversity decomposition of utility-based quality estimates in minimum Bayes risk decoding. Prompt-diversification studies further show that how candidates are generated can matter beyond their number (Wang et al., 2026b). These findings motivate evaluating individual-response quality and complementary successes across samples, and can instead treat sample count or textual variation alone as sufficient indicators of coverage. CARP addresses this balance through population-level commitment allocation.

Masked diffusion and efficient token commitment. Masked diffusion models generate text through repeated prediction of missing tokens (Austin et al., 2021a; Ou et al., 2025). Advances in training objectives and modeling frameworks have improved discrete and masked diffusion language modeling (Sahoo et al., 2024; Lou et al., 2024; Nie et al., 2025a; Arriola et al., 2025), with LLaDA and Dream extending these approaches to large instruction-following models (Nie et al., 2025b; Ye et al., 2025). Unmasking order and parallelism affect both generation quality and computation (Kim et al., 2025; Jazbec et al., 2026). Fast-dLLM combines confidence-aware parallel decoding with caching (Wu et al., 2026a). EB-Sampler adapts the number of parallel commitments using an entropy-based criterion motivated by sampling-error analysis (Ben-Hamu et al., 2025), whereas Fast-dLLM++ uses sorted confidence profiles and a Fréchet-based criterion (Kasa et al., 2026). Learned unmasking policies optimize commitment decisions through reinforcement learning (Jazbec et al., 2026), while Order-Token Search jointly explores generation orders and token choices using likelihood-based scoring (Shen et al., 2026). These methods provide close precedents for adaptive commitment. Our rule instead constrains the cumulative confidence deficit of sampled proposals and retains stochastic selection within the feasible set; it is used alongside persistent stochastic policies rather than applied uniformly to the population.

Exploration, diversity, and population allocation. Order flexibility does not automatically yield diverse successful responses (Ni et al., 2026). Fang et al. (2026) analyze how confidence-based decoding can improve a local quality proxy while restricting sequence-level exploration. Tempered confidence-threshold (TCT) decoding softens commitment through stochastic position selection (Olausson et al., 2026); orthogonal diverse diffusion (ODD) modifies intermediate logits to repel each candidate from the feature subspace of preceding candidates (Lamont et al., 2026); and Time-Annealed Perturbation Sampling gradually reduces perturbations to the conditioning embeddings during decoding (Wu et al., 2026b). Diversity-oriented decoding also has autoregressive precedents, including Diverse Beam Search (Vijayakumar et al., 2018). Other approaches revise generated tokens through remasking (Wang et al., 2026a) or resample interacting diffusion trajectories using reward-based potentials (Singhal et al., 2025). In contrast, CARP retains all trajectories and uses irreversible commitment without reward-guided resampling or logit steering. Its allocation of complementary commitment policies follows the algorithm-portfolio principle (Gomes & Selman, 2001). TCT supplies the stochastic commitment primitive, while the contribution is its integration with a budget-dependent adaptive reserve and a progress-based guard.

3 Problem Setting

We study training-free generation of candidate populations with masked diffusion language models. Given a prompt x and a positive integer response budget B, the goal is to generate B candidate responses that balance individual quality, solution coverage, and computational cost. We propose Confidence-Adaptive Risk Portfolios (CARP), which addresses this problem by allocating different token-commitment policies across the population and adapting their behavior to model confidence.

A masked diffusion language model generates a response through iterative token prediction and commitment. Let z_t denote the partially masked response at decoding step t, and let M_t contain the active masked positions in the current generation block. For each position $i \in M_t$, the frozen model produces logits $\ell_{\theta,i}(v \mid x, z_t)$ over vocabulary $\mathcal{V}$. Given token temperature $T_{\text{tok}} > 0$, a proposal is sampled according to

$$\hat{z}_{t,i} \sim p_{\theta,T_{\text{tok}},i}(\cdot \mid x, z_t), \qquad p_{\theta,T,i}(v \mid x, z_t) = \frac{\exp\left(\ell_{\theta,i}(v \mid x, z_t)/T\right)}{\sum_{v' \in \mathcal{V}} \exp\left(\ell_{\theta,i}(v' \mid x, z_t)/T\right)}, \quad T > 0.$$

We define the confidence of a sampled proposal using its untempered model probability:

$$c_{t,i} = p_{\theta,1,i}(\hat{z}_{t,i} \mid x, z_t).$$

This confidence refers to the sampled token rather than necessarily the highest-probability token. We treat it as a model uncertainty signal, without assuming calibration to token correctness or whole-response validity.

A commitment policy selects a subset $U_t \subseteq M_t$ of positions whose proposals are revealed. Under the irreversible commitment setting considered here, the response state and active positions are updated

within the current block as

$$z_{t+1,i} = \begin{cases} \hat{z}_{t,i}, & i \in U_t, \\ z_{t,i}, & i \notin U_t, \end{cases} \qquad M_{t+1} = M_t \setminus U_t.$$

Unselected positions remain masked and are predicted again at subsequent steps. Once the current block is complete, generation proceeds to the next block and initializes its active masked positions. Consequently, commitment determines both parallel decoding progress and the context available for future predictions.

To formalize population-level commitment, let $z_t^{(b)}$ and $M_t^{(b)}$ denote the state and active positions of trajectory $b \in \{1, \dots, B\}$. Let $\mathcal{H}_t$ denote the information available before commitment at step t, including the prompt, current population state, sampled token proposals and their confidences, policy assignments, and preceding generation history. Write

$$U_t^{(b)} \sim \pi_t^{(b)}\left(\cdot \mid \mathcal{H}_t\right), \qquad U_t^{(b)} \subseteq M_t^{(b)},$$

where $\pi_t^{(b)}$ is the commitment policy of trajectory b. This formulation permits policies to differ across trajectories, change during decoding, and depend on information shared across the population. All trajectories share the same frozen model and token proposal mechanism, while their commitment decisions can produce different subsequent contexts and completed responses. Our method specifies these policies through a budget-dependent allocation and a local confidence-deficit constraint. No correctness feedback is available during generation.

For evaluation, let $c_u \in \{0, \dots, B\}$ be the number of correct responses in the size-B population generated for problem u. Over N problems, we use the pass@k estimator of Chen et al. (2021):

$$\widehat{\text{pass@}k} = \frac{1}{N} \sum_{u=1}^{N} \left[1 - \frac{\binom{B-c_u}{k}}{\binom{B}{k}} \right], \qquad k \in \{1, \dots, B\}, \tag{1}$$

with a zero numerator when $B - c_u < k$. At $k = 1$, this measures average individual-response correctness; at $k = B$, it measures the fraction of problems solved by the full population. Intermediate values measure the success probability of a uniformly drawn size-k subset of the realized population. For heterogeneous or coupled populations, this need not equal the success probability of an independent run configured with response budget k, particularly when policy allocation depends on B. It also differs from evaluating the first k responses, whose coverage can depend on policy ordering.

4 Confidence-Adaptive Risk Portfolios

4.1 Motivation: separate trajectory quality from population coverage

Generating a useful candidate population requires both high-quality individual responses and sufficient exploration to solve problems that additional samples would otherwise continue to miss. In diffusion language models, token commitment influences this tradeoff by determining which predictions become context for subsequent decoding steps. Conservative commitment can defer uncertain decisions until more context becomes available, but applying the same conservative policy across all trajectories may limit exploration and prolong the decoding process.

Existing approaches address population diversity through different mechanisms. ODD (Lamont et al., 2026) discourages redundancy through interactions across samples, while TCT introduces stochasticity into token commitment (Olausson et al., 2026). Together, these approaches motivate a complementary allocation question: should all B trajectories use the same commitment rule? A population may benefit from combining trajectories with different commitment behaviors, allowing some to favor conservative local decisions while others retain broader opportunities for exploration.

A fixed confidence threshold does not control the *aggregate uncertainty* committed in one step. For example, committing ten proposals with confidence 0.9 has total confidence deficit 1, despite every proposal being individually confident. Constraining this aggregate deficit provides a way to regulate local commitment. However, applying such a constraint to every trajectory may suppress useful branching and increase computation. This tradeoff motivates reserving a subset of trajectories for

Algorithm 1 CARP: commitments within one diffusion block

Require: Fixed/adaptive roles from Eqs. (2) and (3)

while some trajectory has masked positions in the block **do**
  Evaluate the frozen model; sample token proposals and compute raw confidences.
  Count active fixed-policy trajectories A_t.
  For fixed trajectories, draw stochastic threshold commitments at their assigned τ_a.
  if $2A_t > F$ **then**
    For each active adaptive trajectory, compute $m_t^\star$ and E_t.
    If $m_t^\star > 0$, draw U_t by Eq. (7); otherwise commit the highest-confidence active proposal.
  else
    For adaptive trajectories, draw stochastic threshold commitments at their assigned τ_a.
  Apply empty-draw fallbacks, commit proposals, and update masks.

quality-oriented adaptive commitment while retaining stochastic coverage trajectories. These are intended roles, not assumptions that either policy is uniformly superior.

CARP implements this separation through two components: (i) a budget-dependent adaptive reserve; and (ii) stochastic commitment inside a confidence-risk constraint. All responses are retained. There is no verifier, response repair, logit steering, or post-generation selection.

4.2 Budget-dependent allocation

Let $B \geq 4$ be an even integer and let $0 < \tau_\ell < \tau_h < 1$ be two fixed thresholds. Allocate

$$R(B) = \min\left\{B - 2,\ 2\left\lceil \frac{\sqrt{B}}{2} \right\rceil\right\}, \qquad F(B) = B - R(B) \tag{2}$$

adaptive (R) and fixed-policy trajectories (F), respectively. For each $a \in \{\ell, h\}$, assign $F(B)/2$ fixed trajectories to the stochastic confidence-threshold commitment rule at τ_a, and assign $R(B)/2$ adaptive trajectories to allowance

$$\rho_a = 1 - \tau_a. \tag{3}$$

This correspondence has a simple interpretation: a single proposal of confidence τ_a exhausts allowance ρ_a. It does *not* make the cumulative constraint equivalent to thresholding.

Thus, at larger response budgets, the adaptive reserve grows in number while occupying a vanishing fraction of the population. Two simple requirements motivate this choice. The reserve should grow with B, so that larger budgets include more quality-oriented trajectories, while its share should shrink, so that most additional responses go to coverage. A constant reserve fails the first requirement, and a fixed fraction fails the second. The square root is the simplest rule between these extremes: a uniformly drawn size-k subset contains $kR(B)/B \approx k/\sqrt{B}$ adaptive trajectories in expectation, so a subset of about $\sqrt{B}$ responses is expected to include one of them. The square-root exponent is a design choice, not a proven optimal allocation.

For $B = 64$ and $(\tau_\ell, \tau_h) = (0.6, 0.8)$, the allocation is 28 fixed trajectories per threshold and four adaptive trajectories per allowance, with $(\rho_\ell, \rho_h) = (0.4, 0.2)$. The method still requires thresholds and temperatures; it ties the reserve size and allowances to existing quantities rather than eliminating hyperparameters.

4.3 Stochastic commitment under a confidence-risk constraint

Consider an active trajectory while its adaptive commitment policy is in use. Condition on its current masked state and sampled token proposals, as in Section 3. Let $M_t \neq \varnothing$ denote the set of active masked positions at step t, and let $c_{t,i}$ denote the frozen model's confidence in the sampled token proposal at position i. This confidence is defined as the proposal's untempered probability rather than the maximum probability over the vocabulary.

For each position $i \in M_t$, define its confidence deficit $r_{t,i} = 1 - c_{t,i}$. For any subset $S \subseteq M_t$ of masked positions, define the aggregate confidence deficit of their proposals as

$$r_{t,i} = 1 - c_{t,i}, \qquad \mathcal{R}_t(S) = \sum_{i \in S} r_{t,i}. \tag{4}$$

A higher-confidence proposal contributes a smaller deficit. We call $\mathcal{R}_t$ a confidence-risk *proxy*: model confidence need not be calibrated to token correctness or program validity.

Given the trajectory's allowance ρ_a, the eligibility problem selects as many positions as possible without exceeding this aggregate deficit:

$$\max_{S \subseteq M_t} |S| \quad \text{subject to} \quad \mathcal{R}_t(S) \le \rho_a. \tag{5}$$

Here $|S|$ is the number of selected positions, and ρ_a is the per-step allowance assigned in Eq. (3). The same allowance is used at every adaptive step; it is not depleted across the trajectory.

Let $m = |M_t|$ be the number of active masked positions. Order their confidences as $c_{t,(1)} \ge \cdots \ge c_{t,(m)}$, breaking ties consistently. The index (j) denotes the position with the jth-highest confidence, rather than the jth position in the sequence. Define

$$m_t^\star = \max \left\{ j \in \{0, \dots, m\} : \sum_{h=1}^{j} (1 - c_{t,(h)}) \le \rho_a \right\}. \tag{6}$$

Thus, $m_t^\star$ is the largest number of highest-confidence proposals whose combined deficit fits the allowance. The empty prefix has deficit zero, so $m_t^\star = 0$ when no proposal is affordable.

Let $E_t = \{(1), \dots, (m_t^\star)\}$ be the corresponding set of *eligible positions*, with $E_t = \varnothing$ when $m_t^\star = 0$. This prefix is computed separately for each adaptive trajectory. When $m_t^\star > 0$, set $\tau_t^\star = c_{t,(m_t^\star)}$, the lowest confidence among eligible proposals. The adaptive threshold is therefore a function of the whole confidence profile rather than a fixed schedule. For instance, an allowance 0.10 admits two proposals from confidences $(0.99, 0.98, 0.80)$ but only one from $(0.94, 0.93, 0.80)$. The eligible set and its threshold are recomputed from new predictions at each adaptive step.

Eligibility permits commitment but does not require it. Instead of committing the entire affordable prefix deterministically, retain TCT's stochastic position decision. When $m_t^\star > 0$, independently draw, for each $i \in M_t$,

$$b_{t,i} \overset{\text{ind}}{\sim} \text{Bernoulli}\left(\mathbf{1}\{i \in E_t\}\, \sigma\left(\frac{c_{t,i} - \tau_t^\star}{T_{\text{pos}}} \right) \right), \qquad U_t = \{i \in M_t : b_{t,i} = 1\}. \tag{7}$$

Here $b_{t,i}$ is a binary commitment indicator: $b_{t,i} = 1$ selects the proposal at position i, while $b_{t,i} = 0$ leaves that position unselected before fallback. The indicator $\mathbf{1}\{i \in E_t\}$ assigns zero selection probability to ineligible positions. The logistic sigmoid $\sigma(u) = (1 + \exp(-u))^{-1}$ and position temperature $T_{\text{pos}} > 0$ determine selection probabilities within the eligible set. Since $c_{t,i} \ge \tau_t^\star$ for every $i \in E_t$, each eligible position has selection probability at least $1/2$.

The set U_t contains the positions actually selected for commitment, whereas E_t contains all positions eligible for selection. If $m_t^\star > 0$ but $U_t = \varnothing$, replace U_t by the singleton containing the highest-confidence member of E_t. If $m_t^\star = 0$, directly commit the highest-confidence active proposal to ensure progress. Use the same deterministic tie-breaking rule as in the confidence ordering. This latter *unaffordable fallback* is explicitly outside the proxy constraint.

Committed proposals become part of the context for subsequent predictions; positions outside the final set U_t remain masked. The per-state token proposal distribution is unchanged: only the set of proposals committed to the sequence is modified.

This rule is well-defined and respects its allowance. The sorted prefix E_t is the largest set of proposals whose summed deficit fits ρ_a; every committed set satisfies $\mathcal{R}_t(U_t) \le \rho_a$; and each adaptive step with $m_t^\star > 0$ commits at least $m_t^\star/2 + 2^{-m_t^\star}$ proposals in expectation (Proposition 1, Appendix A). These guarantees concern the confidence-deficit proxy, not response correctness. Restricting eligibility can exclude uncertain positions that stochastic thresholding may preserve, which is why the constraint is confined to the adaptive reserve. Sorting costs $O(m \log m)$ per active adaptive trajectory and adds no model forward.

Table 1: LLaDA pass@k across all benchmarks (%, mean $\pm$ one standard deviation).

Dataset	Method	pass@4	pass@8	pass@16	pass@64
MATH-500	TCT	47.80±0.20	56.18±0.29	63.75±0.34	75.26±0.10
	ODD	49.31±0.17	56.67±0.23	63.60±0.45	75.60±1.06
	CARP	**49.87±0.06**	**57.53±0.32**	**64.55±0.53**	**75.87±0.84**
HumanEval	TCT	38.75±0.46	47.29±0.64	54.84±0.94	67.89±0.35
	ODD	41.11±0.35	47.92±0.44	53.83±0.34	62.60±0.35
	CARP	**46.01±0.12**	**52.78±0.07**	**58.67±0.15**	**68.09±0.70**
GSM8K	TCT	89.81±0.65	94.60±0.81	97.33±0.88	–
	ODD	91.45±0.30	95.19±0.23	97.00±0.33	–
	CARP	**92.08±0.39**	**95.81±0.29**	**98.11±0.19**	–
MBPP	TCT	57.72±0.13	64.94±0.15	70.57±0.47	**77.56±0.45**
	ODD	57.96±0.13	63.87±0.18	68.30±0.24	75.36±0.81
	CARP	**60.66±0.27**	**66.43±0.39**	**70.96±0.78**	**77.56±1.38**

Algorithm 1 summarizes commitment within a decoding block. Fixed-policy and adaptive trajectories begin together; the progress switch changes the policy of unfinished adaptive trajectories while preserving their committed tokens. Appendix D provides further details on this switch and on measuring the adaptive reserve's contribution to coverage.

5 Experiments

This section presents the experimental setup in Section 5.1, compares results in Section 5.2, and examines the method's components in Section 5.3. Additional experimental details and results appear in Appendix C.

5.1 Setup

Experimental setup. Our evaluation uses two diffusion language models, `LLaDA-8B-Instruct` (LLaDA) and `Dream-v0-Instruct-7B` (Dream), and reports pass@k (Chen et al., 2021), defined in Section 3. Unless otherwise noted, we use a generation length of 256, a maximum of 256 diffusion steps, a block length of 32, and a token temperature of 0.8. All decoding builds on Fast-dLLM's confidence-aware parallel decoding (Wu et al., 2026a). The main result tables report means with ± 1 standard deviation across generation seeds.

Datasets. The evaluation covers four benchmarks: MATH-500 (Lightman et al., 2024), HumanEval (Chen et al., 2021), MBPP (Austin et al., 2021b), and GSM8K (Cobbe et al., 2021). MATH-500 consists of 500 competition-level mathematics problems spanning diverse topics and difficulty levels, and is commonly used to evaluate multi-step reasoning. GSM8K consists of grade-school math word problems that require two to eight steps of basic arithmetic reasoning; we randomly pick 300 questions from its 1,319-problem test split for evaluation. HumanEval consists of 164 programming problems and evaluates code generation through functional correctness. MBPP consists of entry-level Python programming problems, each paired with a natural-language task description and three test cases; we evaluate on its 257-problem sanitized test split.

Baselines. We compare with two training-free approaches to diverse diffusion sampling. **TCT** (Olausson et al., 2026) samples token proposals from the model and commits each masked position with a probability determined by its proposal confidence, a confidence threshold, and a position temperature. This stochastic commitment rule provides the decoding primitive used by CARP. **ODD** (Lamont et al., 2026) steers intermediate logits to encourage each candidate's features to depart from the subspace spanned by preceding candidates in the batch.

5.2 Experiment Results

HumanEval results. Tables 1 and 2 show that CARP achieves maximum absolute gains of 7.26 and 10.28 percentage points over TCT on LLaDA and Dream, respectively, both at pass@4. The

Table 2: DREAM pass@k across all benchmarks (%, mean $\pm$ one standard deviation).

Dataset	Method	pass@4	pass@8	pass@16	pass@64
MATH-500	TCT	54.93±0.26	64.32±0.43	71.93±0.60	82.53±1.15
	ODD	58.63±0.24	66.80±0.32	73.80±0.42	83.53±1.36
	CARP	**59.10±0.31**	**67.48±0.53**	**74.45±0.76**	**84.33±1.56**
HumanEval	TCT	55.00±0.85	66.39±1.12	74.90±1.31	85.16±1.41
	ODD	61.31±0.49	70.01±0.64	76.71±0.66	84.96±0.93
	CARP	**65.28±0.63**	**73.74±0.74**	**79.90±0.79**	**87.60±1.53**
GSM8K	TCT	89.41±0.33	94.76±0.61	97.56±0.84	–
	ODD	91.23±0.19	**95.52±0.34**	**98.00±0.00**	–
	CARP	**91.80±0.04**	95.51±0.21	97.56±0.69	–
MBPP	TCT	66.32±0.26	73.51±0.38	78.49±0.44	85.47±0.22
	ODD	68.45±0.14	73.89±0.16	78.02±0.40	84.05±1.03
	CARP	**71.30±0.11**	**76.64±0.23**	**80.98±0.14**	**87.75±0.83**

diversity analysis in Figs. 2a and 2b measures the breadth of the successful population. Each bar reports the mean number of distinct program strings per task that pass the official HumanEval tests among the first k responses. A taller bar therefore indicates more distinct passing implementations. CARP exceeds TCT at every reported budget on both models and exceeds ODD at most budgets, with the exception of Dream at $k = 16$. These results support improved diversity among correct programs.

MATH-500 results. Tables 1 and 2 show that CARP attains the highest mean pass@k at every reported budget for both backbones. Its maximum absolute gains over TCT are 4.17 percentage points on DREAM and 2.07 on LLADA, both at pass@4. All methods use the same population budget of 64 responses per problem. A complementary analysis of successful-problem coverage by difficulty level appears in Appendix C.

MBPP results. On the 257-task MBPP-sanitized test split, CARP achieves maximum absolute gains over TCT of 2.94 percentage points on LLADA and 4.98 on DREAM, both at pass@4. On LLADA, TCT and CARP have the same mean pass@64 to the reported precision, although TCT has a smaller standard deviation (Tables 1 and 2).

GSM8K results. On the 300 randomly selected questions, CARP achieves the highest mean pass@k in four of the six model–budget comparisons in Tables 1 and 2. All methods approach saturation at $k = 16$, leaving relatively little room for improvement; we therefore report results through this budget. The maximum absolute gains over TCT are 2.27 percentage points on LLADA and 2.39 on DREAM, both at pass@4.

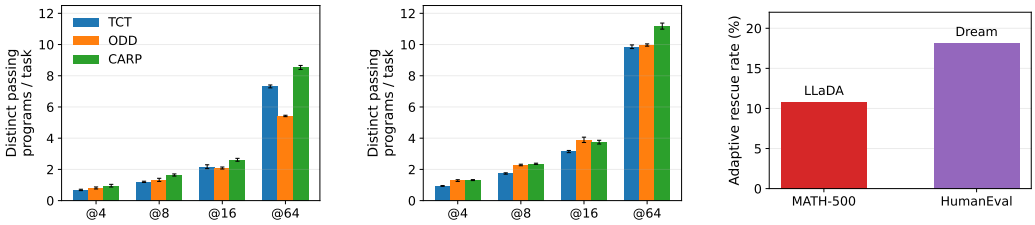
5.3 ABLATION STUDIES

The following analyses examine the contribution of the adaptive reserve and the sensitivity to the portfolio's confidence thresholds.

Table 3: Dream MATH-500 results (CARP vs Fixed).

Population	pass@4	pass@8	pass@16	pass@32
Fixed 0.6/0.8 only	58.56 ± 0.27	67.14 ± 0.43	74.28 ± 0.58	79.92 ± 0.81
Complete CARP	$\mathbf{59.10 \pm 0.31}$	$\mathbf{67.48 \pm 0.53}$	$\mathbf{74.45 \pm 0.76}$	$\mathbf{80.05 \pm 1.05}$

Adaptive rule. Fig. 2c isolates the complementary contribution of the adaptive reserve within a single CARP population. For each benchmark, we first condition on tasks that none of the fixed-threshold trajectories solve. The bar reports the fraction of this subset for which at least one adaptive-reserve trajectory is correct. Thus, the analysis credits the adaptive rule only when it recovers a solution absent from every fixed portfolio trajectory, rather than comparing independently sampled populations. We



(a) LLaDA (b) Dream (c) Adaptive reserve

Figure 2: Useful diversity and adaptive-reserve complementarity. Panels (a) and (b) show the mean number of distinct HumanEval programs per task that pass the official tests among the first k responses. Panel (c) shows the fraction of fixed-portfolio failures recovered by the adaptive reserve.

additionally include a comparison of an equal split of fixed rules (same number of responses as CARP) compared to CARP in Table 3. The only difference is the presence of adaptive trajectories or not.

Portfolio thresholds. Table 4 compares confidence-threshold pairs on Dream HumanEval. The 0.5/0.7 pair outperforms the 0.7/0.9 pair at larger k. The 0.7/0.9 pair gives the best pass@4, but its high thresholds reduce performance at large k. Overall, our selected 0.6/0.8 pair offers the best balance and the strongest pass@k across most budgets. Generally speaking, a low threshold (like 0.3 $\sim$0.5) is likely to hurt early performance, while a high threshold (like 0.7$\sim$1.0) is not beneficial for scaling performance.

Table 4: Dream HumanEval results (%) across confidence-threshold pairs.

Method	pass@4	pass@8	pass@16	pass@32	pass@64
CARP (0.5/0.7)	62.20	67.07	73.17	80.49	84.76
CARP (0.6/0.8)	65.28	73.74	**79.90**	**84.45**	**87.60**
CARP (0.7/0.9)	**67.93**	**75.03**	75.88	79.90	83.54

6 Discussion and Limitation

CARP treats conservative commitment as a specialized role within a response population. The results across two models and four benchmarks support combining this role with persistent stochastic commitment policies. Gains are most visible at smaller response budgets, while differences narrow as coverage approaches saturation. The HumanEval program analysis and adaptive-reserve analysis provide complementary evidence: the former measures distinct passing implementations, and the latter identifies tasks solved by the adaptive reserve but missed by all fixed-policy trajectories in the same population.

More broadly, CARP shows that trajectories in one response population need not share a single commitment policy, and that dividing commitment roles lets a population balance quality and coverage rather than trading one for the other. A budget-dependent adaptive reserve caps the summed confidence deficit committed at each step to protect individual response quality; fixed-threshold trajectories preserve stochastic exploration to extend coverage, and a progress switch limits prolonged conservative decoding. Across MATH-500, HumanEval, MBPP, and GSM8K with LLaDA and Dream, this allocation improves pass@k relative to TCT and ODD over a range of response budgets, without an additional model or correctness verifier. These findings suggest that assigning complementary commitment roles across trajectories is a practical way to balance quality and coverage in diffusion language models.

As for our limitations, the square-root reserve size and the half-completion progress switch are heuristic design choices. They are motivated by simple scaling and progress arguments rather than derived from an optimality criterion, and other reserve sizes or switching rules may perform better for other response budgets, models, or tasks.

AI USE STATEMENT

We have not used generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, provide ingredients for or write proofs, propose or refine hypotheses, design or give feedback on research methodology or experiments, implement the proposed method, translate text, clean or reformat datasets, or interpret results. Qualitative and thematic data analysis is not applicable to this work. In this work, we used generative AI tools only for the tasks listed below: (1) edit the manuscript to improve readability, grammar, and clarity of the text, and (2) create and edit supporting software code, such as experiment scripts and evaluation utilities.

Ethics statement. This work adheres to the ICLR Code of Ethics. Our study involves no human subjects, user studies, or collection of personal data. All experiments use publicly available models and established public benchmarks (e.g., MATH-500, HumanEval, GSM8k, and MBPP), used in accordance with their licenses.

Reproducibility statement. We will publicly release our code upon publication to facilitate reproducibility.

REFERENCES

Marianne Arriola, Subham Sekhar Sahoo, Aaron Gokaslan, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Justin T Chiu, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=tyEyYT267x.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021a. URL https://openreview.net/forum?id=h7-XixPCAL.

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*, 2021b.

Heli Ben-Hamu, Itai Gat, Daniel Severo, Niklas Nolte, and Brian Karrer. Accelerated sampling from masked diffusion models via entropy bounded unmasking. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=WBcBhT1NKO.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling, 2024. URL https://arxiv.org/abs/2407.21787.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

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.

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.

Liancheng Fang, Aiwei Liu, Henry Peng Zou, Yankai Chen, Enze Ma, Leyi Pan, Chunyu Miao, Wei-Chieh Huang, Xue Liu, and Philip S. Yu. Locally confident, globally stuck: The quality-exploration dilemma in diffusion language models, 2026. URL https://arxiv.org/abs/2604.00375.

Carla P. Gomes and Bart Selman. Algorithm portfolios. *Artif. Intell.*, 126(1–2):43–62, February 2001. ISSN 0004-3702. doi: 10.1016/S0004-3702(00)00081-3. URL https://doi.org/10.1016/S0004-3702(00)00081-3.

Metod Jazbec, Theo X. Olausson, Louis Béthune, Pierre Ablin, Michael Kirchhof, Joao Monteiro, Victor Guilherme Turrisi da Costa, Jason Ramapuram, and marco cuturi. Learning unmasking

policies for diffusion language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=F9NDKf5oPy.

Hidetaka Kamigaito, Hiroyuki Deguchi, Yusuke Sakai, Katsuhiko Hayashi, and Taro Watanabe. Diversity explains inference scaling laws: Through a case study of minimum Bayes risk decoding. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29060–29094, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1410. URL https://aclanthology.org/2025.acl-long.1410/.

Siva Rajesh Kasa, Yasong Dai, Sumit Negi, and Hongdong Li. Fast-dLLM++: Fréchet profile decoding for faster diffusion LLM inference. In *ICML 2026 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2026. URL https://openreview.net/forum?id=pdxEKvTsRq.

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 Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 30749–30768. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/kim25ah.html.

Sumith Kulal, Panupong Pasupat, Kartik Chandra, Mina Lee, Oded Padon, Alex Aiken, and Percy S Liang. Spoc: Search-based pseudocode to code. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/7298332f04ac004a0ca44cc69ecf6f6b-Paper.pdf.

Sean Lamont, Christian Walder, Paul Montague, Amir Dezfouli, and Michael Norrish. Free lunch for pass@k? low cost diverse sampling for diffusion language models. *arXiv preprint arXiv:2603.04893*, 2026. URL https://arxiv.org/abs/2603.04893.

Noam Itzhak Levi. A simple model of inference scaling laws. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=5ulCAfxJzc.

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 *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32819–32848. PMLR, 2024.

Zanlin Ni, Shenzhi Wang, Yang Yue, Tianyu Yu, Weilin Zhao, Yeguo Hua, Tianyi Chen, Jun Song, Cheng Yu, Bo Zheng, and Gao Huang. The flexibility trap: Rethinking the value of arbitrary order in diffusion language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=kpgURPRMGf.

Shen Nie, Fengqi Zhu, Chao Du, Tianyu Pang, Qian Liu, Guangtao Zeng, Min Lin, and Chongxuan Li. Scaling up masked diffusion models on text. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=WNvvwK0tut.

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 *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b. URL https://openreview.net/forum?id=KnqiC0znVF.

Theo X. Olausson, Metod Jazbec, Xi Wang, Armando Solar-Lezama, Christian A. Naesseth, Stephan Mandt, and Eric Nalisnick. A tale of two temperatures: Simple, efficient, and diverse sampling from diffusion language models. In *Third Conference on Language Modeling*, 2026. URL https://openreview.net/forum?id=mUyvTuvWOZ.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=sMyXP8Tanm.

Subham Sekhar Sahoo, Marianne Arriola, Aaron Gokaslan, Edgar Mariano Marroquin, Alexander M Rush, Yair Schiff, Justin T Chiu, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=L4uaAR4ArM.

Yangyi Shen, Tianjian Feng, Jiaqi Han, Wen Wang, Tianlang Chen, Chunhua Shen, Jure Leskovec, and Stefano Ermon. Improving diffusion language model decoding through joint search in generation order and token space. *arXiv preprint arXiv:2601.20339*, 2026. URL https://arxiv.org/abs/2601.20339.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 55810–55827. PMLR, 2025. URL https://proceedings.mlr.press/v267/singhal25b.html.

Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4FWAwZtd2n.

Ashwin K. Vijayakumar, Michael Cogswell, Ramprasath R. Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. Diverse beam search: Decoding diverse solutions from neural sequence models. In *AAAI Conference on Artificial Intelligence*, 2018. URL https://arxiv.org/abs/1610.02424.

Guanghan Wang, Yair Schiff, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Remasking discrete diffusion models with inference-time scaling. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026a. URL https://openreview.net/forum?id=IJryQAOy0p.

Tianchun Wang, Zichuan Liu, Yuanzhou Chen, Jonathan Light, Weiyang Liu, Haifeng Chen, Xiang Zhang, and Wei Cheng. On the effect of sampling diversity in scaling llm inference, 2026b. URL https://arxiv.org/abs/2502.11027.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

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 *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=3Z3Is6hnOT.

Jingxuan Wu, Zhenglin Wan, Yuzhe YANG, Yiqiao Huang, Chubin Zhang, Xingrui Yu, Ivor Tsang, and Yang You. Time-annealed perturbation sampling: Diverse generation for diffusion language models. In *ICML 2026 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2026b. URL https://openreview.net/forum?id=97dB4GrUzx.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models, 2025. URL https://arxiv.org/abs/2508.15487.

Yue Yu, Qiwei Di, Quanquan Gu, and Dongruo Zhou. On the limits of test-time compute: Sequential reward filtering for better inference. *arXiv preprint arXiv:2512.04558*, 2025.

Bowen Zuo and Yinglun Zhu. Strategic scaling of test-time compute: A bandit learning approach. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=0mNnINd2z5.

CONTENTS OF THE APPENDIX

A Local guarantees and their scope **15**
A.1 Local feasibility and parallel progress 15

B Prompts **17**

C Additional Experiment Results **18**

D Component Analysis Details **18**
D.1 Progress-based switching and batched decoding 18

A LOCAL GUARANTEES AND THEIR SCOPE

The following result describes the eligible-set construction and the commitment decisions made at a single adaptive step. The deficit bound concerns the model's confidence-risk proxy, rather than the probability of an incorrect prediction, and does not require a calibration assumption.

A.1 LOCAL FEASIBILITY AND PARALLEL PROGRESS

For a nonempty eligible set E_t, let

$$q_{t,i} = \sigma\left(\frac{c_{t,i} - \tau_t^\star}{T_{\mathrm{pos}}}\right), \qquad i \in E_t,$$

denote the probability of selecting position i before applying the empty-draw fallback. Conditional on the pre-commitment information H_t, these selections are independent.

Proposition 1 (Local feasibility and parallel progress)**.** *Condition on the pre-commitment information H_t, including the sampled proposals, the allowance $\rho_a \geq 0$, and the position temperature $T_{\mathrm{pos}} > 0$. Suppose $c_{t,i} \in [0,1]$ for every active position. Then the sorted feasible prefix E_t has maximum cardinality among all subsets $S \subseteq M_t$ satisfying $\mathcal{R}_t(S) \leq \rho_a$.*

If $m_t^\star > 0$, the final committed set U_t, including the empty-draw fallback, satisfies

$$\mathcal{R}_t(U_t) \leq \rho_a \qquad \textit{for every selection outcome}, \tag{8}$$

and

$$\mathbb{E}[|U_t| \mid H_t] = \sum_{i \in E_t} q_{t,i} + \prod_{i \in E_t} (1 - q_{t,i}) \geq \frac{m_t^\star}{2} + 2^{-m_t^\star}. \tag{9}$$

The two bounds concern a step using the adaptive rule with a nonempty eligible set. All statements are in exact arithmetic.

Proof. Fix H_t. The sampled proposals, their confidences, and the eligible set are now fixed. The only remaining randomness comes from the Bernoulli decisions that determine which eligible positions are selected.

We first show that the prefix E_t contains as many positions as the allowance permits. Write $r_{t,i} = 1 - c_{t,i}$ and order the deficits as

$$r_{t,(1)} \leq r_{t,(2)} \leq \cdots \leq r_{t,(m)}, \qquad m = |M_t|.$$

This is the same ordering as sorting confidences from highest to lowest. Consider any subset S of $j \geq 1$ positions, and let $s_1 < \cdots < s_j$ be the ranks of its elements in this ordering. Since $s_h \geq h$, the hth selected deficit is at least the hth smallest deficit overall. Therefore,

$$\mathcal{R}_t(S) = \sum_{h=1}^{j} r_{t,(s_h)} \geq \sum_{h=1}^{j} r_{t,(h)}.$$

Thus, among all subsets of size j, the first j positions have the smallest total deficit. By the definition of $m_t^\star$, the prefix E_t is feasible, whereas every longer prefix exceeds ρ_a. The inequality above then shows that every subset larger than E_t also exceeds the allowance. This proves the maximum-cardinality claim, including the case $m_t^\star = 0$.

Now suppose $m_t^\star > 0$, and let U_t^0 denote the set selected before the fallback. Only eligible positions can be selected, so $U_t^0 \subseteq E_t$. If U_t^0 is nonempty, the final set is $U_t = U_t^0$. Otherwise, the fallback selects the highest-confidence member of E_t. In either case, $U_t \subseteq E_t$. Since $r_{t,i} \geq 0$, removing positions from a set cannot increase its total deficit. Hence,

$$\mathcal{R}_t(U_t) = \sum_{i \in U_t} r_{t,i} \leq \sum_{i \in E_t} r_{t,i} = \mathcal{R}_t(E_t) \leq \rho_a.$$

This holds for every realization of the Bernoulli decisions and proves (8).

It remains to determine how many positions are committed on average. Let $X_t = |U_t^0|$ be the number selected before the fallback. The fallback does not change the set when $X_t \geq 1$ and adds exactly one position when $X_t = 0$. Consequently,

$$|U_t| = X_t + \mathbf{1}\{X_t = 0\}.$$

By linearity of expectation,

$$\mathbb{E}[X_t \mid H_t] = \sum_{i \in E_t} q_{t,i}.$$

Moreover, conditional independence of the selections gives

$$\Pr(X_t = 0 \mid H_t) = \prod_{i \in E_t} (1 - q_{t,i}),$$

because an empty draw requires every eligible position to be rejected. Taking expectations in the expression for $|U_t|$ therefore yields

$$\mathbb{E}[|U_t| \mid H_t] = \sum_{i \in E_t} q_{t,i} + \prod_{i \in E_t} (1 - q_{t,i}).$$

To obtain the lower bound, recall that $\tau_t^\star$ is the lowest confidence in E_t. Thus $c_{t,i} \geq \tau_t^\star$ for every eligible position. Since $T_{\text{pos}} > 0$ and the sigmoid is increasing,

$$q_{t,i} = \sigma\left(\frac{c_{t,i} - \tau_t^\star}{T_{\text{pos}}}\right) \geq \sigma(0) = \frac{1}{2}.$$

The expectation contains both the expected number of initial selections and the probability of an empty draw. Increasing a selection probability raises the first term but lowers the second, so we must consider their combined effect.

Let $n = |E_t| = m_t^\star$, and define

$$f((q_i)_{i \in E_t}) = \sum_{i \in E_t} q_i + \prod_{i \in E_t} (1 - q_i).$$

Holding all other probabilities fixed, we have

$$\frac{\partial f}{\partial q_i} = 1 - \prod_{j \in E_t \setminus \{i\}} (1 - q_j) \geq 0,$$

since every factor in the product lies in $[0, 1]$. When $n = 1$, the product is empty and equals one, so the same conclusion holds. Therefore, increasing any individual selection probability cannot decrease f. Because every $q_{t,i}$ is at least $1/2$, the expectation is bounded below by its value when all selection probabilities equal $1/2$:

$$\mathbb{E}[|U_t| \mid H_t] = f((q_{t,i})_{i \in E_t}) \geq f((1/2)_{i \in E_t}) = \frac{n}{2} + \left(\frac{1}{2}\right)^n.$$

Substituting $n = m_t^\star$ proves (9). □

What would be needed for a correctness interpretation? Let H_t include the prompt, current state, sampled proposals, assigned allowance, and history before position sampling. Suppose that well-defined proposal failure events $D_{t,i}$ satisfy the additional, selection-conditional assumption

$$\mathbb{P}(D_{t,i} \mid H_t, U_t) \leq 1 - c_{t,i} \quad \text{for every } i \in U_t.$$

On an adaptive step with a nonempty eligible set, a conditional union bound would then give

$$\mathbb{P}\left(\bigcup_{i \in U_t} D_{t,i} \,\middle|\, H_t, U_t\right) \leq \sum_{i \in U_t} (1 - c_{t,i}) = \mathcal{R}_t(U_t) \leq \rho_a.$$

Independence of the failure events is not required. However, raw model confidence alone does not establish this assumption; correctness of open-ended code is also not generally reducible to agreement with a single reference token. Even under this assumption, a trajectory-level guarantee would need to account for failures across multiple steps. Steps after switching and unaffordable forced commitments are not covered by this bound. We therefore make no whole-response correctness guarantee. Floating-point rounding can additionally affect eligibility at the budget boundary; the proposition is stated in exact arithmetic.

B PROMPTS

The boxes show the user-message templates supplied to each model's chat template with the assistant generation prompt enabled. Bracketed fields are replaced by dataset content. MATH-500, HumanEval, and GSM8K use zero-shot prompts; MBPP uses one demonstration from the sanitized prompt split.

MATH-500. Prompt:

```
Put the final answer in \boxed{ }.
Problem:
[problem]

Solution:
```

HumanEval. Prompt:

```
Complete the following Python function. Return only Python code.

[function signature and docstring]
```

GSM8K. Prompt:

```
You are a math expert. You will be given a question to solve. Solve it step by step.
    Wrap the final answer in a \boxed{}.
Respond in the following format:
<reasoning>
Your reasoning here
</reasoning>
<answer>
\boxed{...}
</answer>

[question]
```

The literal prefix `<reasoning>` is appended after the chat template's assistant generation prompt.

MBPP. The demonstration is the first example in the sanitized prompt split (not the test set).

```
You are an expert Python programmer, and here is your task:
[demonstration task]
Your code should pass these tests:
[demonstration test 1]
[demonstration test 2]
[demonstration test 3]
```python
[demonstration solution]
```

You are an expert Python programmer, and here is your task:
[target task]
Your code should pass these tests:
[target test 1]
[target test 2]
[target test 3]
```

After applying the chat template, generation begins with the appended prefix:

```python
```

C Additional Experiment Results

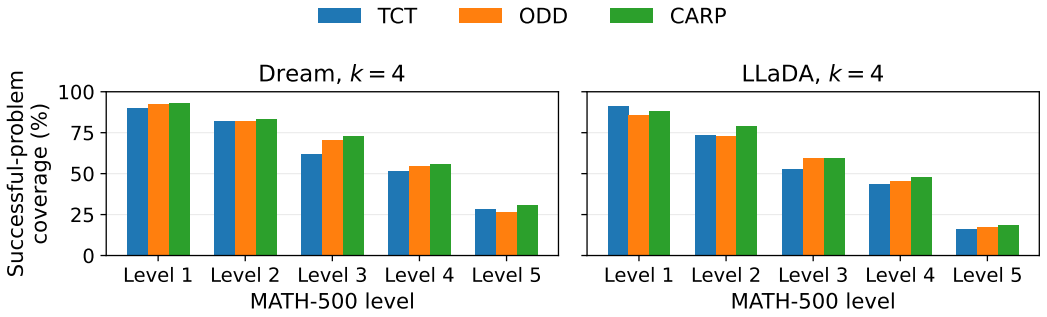


Figure 3: MATH-500 successful-problem coverage by difficulty level at $k = 4$. Each bar is the fraction of problems in a level with at least one correct response among the first four responses, averaged over three seeds.

MATH-500 successful coverage by difficulty. Figure 3 evaluates successful-problem coverage among the first four responses of each saved population. It reports the fraction of problems in each MATH difficulty stratum with at least one correct response, rather than counting surface forms of mathematically equivalent answers as distinct. On DREAM, CARP improves coverage over TCT at every level, with the largest gain at Level 3 (+10.80 percentage points). On LLADA, it gives up 3.10 percentage points at Level 1 while increasing coverage at Levels 2–5 by 5.19, 6.66, 4.43, and 2.24 percentage points, respectively. These results show improved prefix coverage in the harder strata. Because the prefixes retain generation order, this analysis also reflects the policies assigned to the leading trajectories; it does not isolate diversity across the entire population.

D Component Analysis Details

Adaptive-reserve complementarity. For each saved population, partition responses into adaptive and fixed-policy trajectories. A task contributes to the rescue numerator when at least one adaptive trajectory is correct and all fixed-policy trajectories are incorrect. The denominator includes every task unsolved by the fixed-policy trajectories. This conditional recovery rate measures the adaptive reserve's additional coverage within the same population. It does not measure the effect of replacing the adaptive reserve with an equal number of additional fixed-policy responses.

D.1 Progress-based switching and batched decoding

Fixed-policy and adaptive trajectories begin decoding together. Initially, fixed-policy trajectories use their assigned confidence thresholds, while adaptive trajectories apply the confidence-deficit constraint. The switch changes the commitment policy of an already active trajectory; it does not delay the start of adaptive generation.

The switching rule is motivated by computation in batched decoding. Restricting eligible proposals can reduce the number of tokens committed per step, causing adaptive trajectories to require more decoding steps. When multiple responses are generated together, these unfinished trajectories can prolong batch execution after other responses have completed. This concerns parallelism across responses, distinct from committing multiple tokens within a single response.

The fixed-policy population provides a reference for decoding progress. Let A_t count unfinished fixed-policy trajectories in the current block and let F be their total number. Once $2A_t \leq F$, unfinished adaptive trajectories switch to their corresponding fixed thresholds, removing the eligibility constraint.

Previously committed tokens remain unchanged and continue to condition subsequent predictions, preserving the influence of earlier adaptive decisions. The rule is re-evaluated in each new block.

The halfway threshold is a heuristic intended to balance conservative commitment with completion cost. Its runtime benefit depends on batching, microbatch scheduling, and the handling of inactive trajectories; the switching rule alone does not guarantee a runtime reduction.