PROGRESSIVE CONFORMAL PREDICTION FOR REMAINING-LENGTH ESTIMATION IN LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Autoregressive decoding creates an "unknown horizon": although serving engines allocate KV pages on demand, admission and preemption decisions are made before a request's future demand is known. We introduce **Progressive Mondrian Conformal Prediction**, a training-free method that converts stopping probability, predictive entropy, and decoding progress into online upper bounds on remaining length. We distinguish an empirical pooled estimator from prompt-exchangeable fixed-step and two-fold request-level constructions; the latter provide finite-sample step or whole-request coverage without assuming independence among states of one request. Across models from 3.8B to 70B, the pooled method reduces mean width by 41–49% relative to global calibration at the same nominal level. The request construction records 4.8% empirical request violations at a 10% budget. A vLLM prototype confirms that the bound can drive admission metadata, while the statistical theorem is explicitly separated from empirical multi-request OOM and preemption outcomes. The bound adds 0.22% per-state overhead.

1 INTRODUCTION

Large Language Models (LLMs) generate one token at a time until an engine-defined stopping condition is met. Consequently, a serving system does not know a request's remaining work at admission or during decoding. This "unknown horizon" is operationally important because every additional token consumes a decoding iteration and extends the Key–Value (KV) cache.

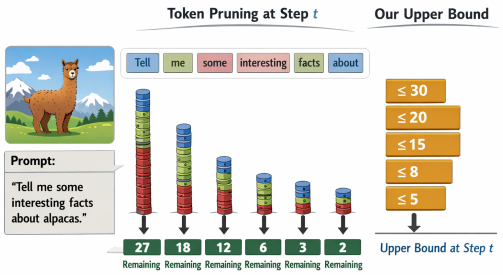


Figure 1: Overview of the Progressive Remaining-Length Estimation workflow.

Modern systems use paging, chunked prefill, live migration, phase disaggregation, and optimized attention kernels (Kwon et al., 2023; Agrawal et al., 2024b; Sun et al., 2024; Zhong et al., 2024; Ye et al., 2025; Yu et al., 2026). In particular, vLLM allocates physical KV blocks incrementally rather than reserving every request's maximum length. The unresolved decision is how much future demand to admit or retain before memory pressure forces preemption or swapping. A *per-request, state-dependent ceiling* supplies this information with a controllable failure probability. The loss is asymmetric: underestimation can trigger preemption, whereas overestimation makes an admission rule unnecessarily conservative.

We introduce **Progressive Remaining-Length Estimation**, which updates this ceiling after every decoded token without an auxiliary model. Figure 1 shows the interface: an admission controller converts $\widehat{U}_t$ to $\lceil \widehat{U}_t / B \rceil$ pages of logical headroom while the engine continues allocating physical pages on demand. The statistical target must match deployment: we report both step violation, $\Pr(R_t > \widehat{U}_t)$, and request violation, $\Pr(\exists t < T : R_t > \widehat{U}_t)$.

We combine a training-free stopping-probability anchor with **Mondrian Conformal Prediction** (Vovk et al., 2005; Angelopoulos & Bates, 2023), using entropy and progress to model heteroscedastic residuals. Our contributions are: **(1)** an online remaining-length target and lightweight logit-based

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.

anchor; **(2)** a pooled estimator, an exact fixed-step construction, and an exact two-fold request construction whose exchangeable unit is a complete prompt trajectory; **(3)** explicit treatment of sparse cells, general stopping predicates, within-request dependence, anchor saturation, and guarantee scope; and **(4)** evaluation across three backbones and six datasets, with predictor/calibrator effects reported separately and systems outcomes treated as empirical rather than conformal guarantees.

2 Background and Related Work

2.1 LLM Serving and Length-Aware Scheduling

Reactive serving systems manage uncertainty after it appears: vLLM allocates paged KV memory on demand (Kwon et al., 2023), Sarathi-Serve chunks prefills (Agrawal et al., 2024b), Llumnix migrates live requests (Sun et al., 2024), and DistServe separates prefill from decode (Zhong et al., 2024). Recent engines further optimize dynamic attention and memory pressure (Ye et al., 2025; Yu et al., 2026), while Vidur provides a validated simulator (Agrawal et al., 2024a). Prediction-based schedulers estimate a length (Qiu et al., 2024), rank requests (Fu et al., 2024), or repeatedly estimate remaining work (Choi et al., 2025). Most directly, TIE fits a heavy-tailed output-length distribution and schedules using a tail-aware risk score (Zheng et al., 2026). Thus uncertainty-aware length scheduling is not new. Our narrower claim is a lightweight *progressive conformal upper bound* with fixed-step or request-level coverage semantics; it can inform admission control but is not a replacement for on-demand paging or a claim of scheduling novelty.

2.2 Uncertainty Quantification in Language Generation

LLM uncertainty is commonly summarized by token entropy, likelihood, sample agreement, or semantic dispersion (Kuhn et al., 2023; Lin et al., 2024); recent surveys emphasize that these scores are useful features but are not coverage guarantees by themselves (Campos et al., 2024). Black-box methods can avoid logits altogether (Su et al., 2024), whereas our white-box serving setting exposes logits at no extra model call. Most of this literature targets *semantic* events such as answer correctness or factuality. We instead quantify an *operational* random variable—future decoding work—whose error has an asymmetric systems cost.

2.3 Conformal Prediction for Language and Sequences

Conformal prediction (CP) provides finite-sample marginal coverage under exchangeability (Vovk et al., 2005; Angelopoulos & Bates, 2023). Language applications include sampled response sets (Quach et al., 2024), multiple-choice question answering (Kumar et al., 2023), factuality guarantees (Mohri & Hashimoto, 2024), optimized decision sets (Vishwakarma et al., 2025), non-exchangeable token generation (Ulmer et al., 2024), and multi-group long-form guarantees (Liu & Wu, 2025). These works quantify content or next-token uncertainty, not the remaining decoding horizon. Our setting also requires care about the exchangeable unit: steps within a request are dependent. We therefore separate an empirical pooled estimator from exact fixed-step calibration and a two-fold trajectory score. Mondrian grouping is used for efficiency and state adaptivity; it does not imply arbitrary conditional validity beyond the prespecified groups (Gibbs et al., 2025).

3 Method

We address *remaining-length estimation* in autoregressive LLMs. At state t, exactly t output tokens have been materialized and the model is about to draw token y_{t+1}; $t = 0$ denotes the prompt-only state. Let R_t be the number of additional nonterminal output tokens before stopping. For a prespecified state index t, we seek an upper bound $\widehat{U}_t$ satisfying

$$\Pr\left(R_t \le \widehat{U}_t\right) \;\ge\; 1-\alpha, \tag{1}$$

where $\alpha \in (0,1)$ is a user-specified miscoverage level.

We use a training-free anchor derived from the instantaneous stopping probability, Mondrian groups defined by uncertainty features, and sparse-cell backoff. Three modes must be distinguished: Full-Step pools dependent states and is evaluated empirically; Full-Fixed uses one state per prompt and has fixed-t coverage; Full-Request uses one maximum score per prompt and has simultaneous request coverage.

3.1 SETUP AND NOTATION

Model and Decoding. Let $\mathcal{M}$ be an autoregressive LLM over vocabulary $\mathcal{V}$, with $y_{1:0} = \emptyset$. Given prompt x, at state $t \in \{0, \dots, T_{\max} - 1\}$,

$$y_{t+1} \sim \mathcal{M}(\cdot \mid x, y_{1:t}), \qquad y_{t+1} \in \mathcal{V}.$$

The decoding policy and maximum content-token budget $T_{\max} \in \mathbb{N}$ are fixed.

Stopping Time. Let $\tau(y_{1:s}, x)$ be a measurable predicate that fires after candidate token y_s (EOS, stop-string completion, grammar acceptance, or a tool boundary). To match the geometric anchor, T counts nonterminal, KV-bearing output tokens and excludes the final control token:

$$T \triangleq \min(\{s - 1 : 1 \le s \le T_{\max},\ \tau(y_{1:s}, x) = 1\} \cup \{T_{\max}\}). \tag{2}$$

The primary experiments use $\tau(y_{1:s}, x) = \mathbf{1}\{y_s = \texttt{eos}\}$. A scheduler that reserves storage for the terminal control token can add one deterministic slot to every bound.

Target Variable. At state t, the *true remaining length* is

$$R_t \triangleq \max\{0,\ T - t\}. \tag{3}$$

For fixed-step calibration, use the absorbing values $(R_t, Z_t, \widehat{R}_t^{\mathrm{pt}}) = (0, z_{\mathrm{abs}}, 0)$ for $t \ge T$, with fixed z_{abs} assigned to its own bin.

Exchangeability Assumption. We assume the n calibration prompt–trajectory pairs and the test pair are exchangeable (i.i.d. prompts decoded with independent randomness are sufficient). The model, tokenizer, stopping predicate, decoding policy, and $T_{\max}$ are fixed between calibration and test. No proof assumes independence among states of one trajectory.

3.2 ONLINE SIGNAL EXTRACTION

At each realized state $t < T$, the model outputs a logit vector $\ell_t \in \mathbb{R}^{|\mathcal{V}|}$, which is normalized via softmax into a probability distribution π_t:

$$\pi_t(v) \triangleq \frac{\exp(\ell_t(v))}{\sum_{u \in \mathcal{V}} \exp(\ell_t(u))}, \quad v \in \mathcal{V}.$$

We extract three scalar features that empirically correlate with termination and uncertainty:

1. **Termination Probability (p_t^{stop}):** The probability mass of tokens that complete the stopping predicate:
$$p_t^{\mathrm{stop}} \triangleq \sum_{v \in \mathcal{V}} \pi_t(v)\, \mathbf{1}\{\tau(y_{1:t} \circ v, x) = 1\}. \tag{4}$$
For EOS stopping, $p_t^{\mathrm{stop}} = p_t^{\mathrm{eos}} = \pi_t(\texttt{eos})$.
2. **Predictive Entropy (H_t):** A measure of the model's uncertainty regarding the next token:
$$H_t \triangleq -\sum_{v \in \mathcal{V}} \pi_t(v) \log \pi_t(v), \tag{5}$$
with $0 \log 0 \triangleq 0$.
3. **Decoding Progress (ρ_t):** The normalized position within the maximum budget:
$$\rho_t \triangleq \frac{t}{T_{\max}} \in [0, 1]. \tag{6}$$

These variables form the random feature vector $Z_t \triangleq (p_t^{\mathrm{stop}}, H_t, \rho_t)$; z_t denotes its realized value. Both are determined online by the current prefix, logits, and decoder stopping state.

3.3 Training-Free Point Prediction

Before applying conformal prediction, we require a point prediction of the remaining length. To avoid auxiliary training and a second model on the decoding path, we derive a training-free anchor.

Geometric Approximation. Set $p = p_t^{\text{stop}}$. If future candidates independently stopped with constant probability p, the number of nonterminal tokens before stopping would be geometric on $\{0, 1, \ldots\}$ with mean $(1-p)/p$. This motivates the anchor only; conformal validity does not assume a geometric stopping law.

Feasible Point Predictor. Using this intuition, and enforcing the hard constraint that the sequence cannot exceed the remaining budget ($T_{\max} - t$), we define the point predictor $\widehat{R}_t^{\text{pt}}$:

$$\widehat{R}_t^{\text{pt}} \triangleq \min\left\{\frac{1-p_t^{\text{stop}}}{\max\{p_t^{\text{stop}}, \varepsilon\}},\ T_{\max} - t\right\}, \tag{7}$$

where $\varepsilon > 0$ is a small constant (e.g., 10^{-6}) to prevent division by zero. This predictor serves as a dynamic anchor that reacts instantly to the model's logits.

All tables use Eq. (7). With the numerical floor inactive and $L_t = T_{\max} - t$, the anchor saturates iff $p_t^{\text{stop}} \le 1/(L_t + 1)$; a nonnegative correction cannot tighten it. Appendix C gives the audit and signed-score alternative.

3.4 Progressive Conformal Upper Bound

The point predictor $\widehat{R}_t^{\text{pt}}$ is merely a heuristic and provides no safety guarantees. We transform it into a rigorous upper bound using conformal prediction.

Nonconformity score. We measure underestimation using the one-sided positive residual:

$$S_t \triangleq \left(R_t - \widehat{R}_t^{\text{pt}}\right)_+ = \max\{0,\ R_t - \widehat{R}_t^{\text{pt}}\}. \tag{8}$$

A score of $S_t = 0$ implies the point predictor was safe (it overestimated or exactly predicted R_t). A large positive S_t implies the predictor underestimated the remaining length.

Signed-residual calibration remains valid but is not evaluated here (Appendix C).

Constructing the bound. If we can find a quantile value q such that $\Pr(S_t \le q) \ge 1 - \alpha$, then by algebraic rearrangement:

$$\begin{aligned} R_t - \widehat{R}_t^{\text{pt}} &\le q \\ R_t &\le \widehat{R}_t^{\text{pt}} + q. \end{aligned}$$

Since $R_t \le L_t \triangleq T_{\max} - t$ deterministically, we may clip without weakening coverage:

$$\widehat{U}_t(q) \triangleq \min\{L_t, \widehat{R}_t^{\text{pt}} + q\}.$$

3.5 Conditioning via Mondrian Binning

A single global quantile q is suboptimal because the distribution of residuals S_t is highly non-stationary. For example, the model is often highly uncertain early in generation (large residuals) but very confident near the end (small residuals).

To adapt to this, we employ **Mondrian Conformal Prediction**. We partition the feature space into bins to group steps with similar characteristics.

Binning Strategy. We define sets of discretization edges for our features:

$$\mathcal{E}_{\text{stop}} = \{e_j^{\text{stop}}\}, \quad \mathcal{E}_H = \{e_j^H\}, \quad \mathcal{E}_\rho = \{e_j^\rho\}.$$

Let $\text{bin}(u; \mathcal{E})$ return the unique half-open interval induced by ordered edges $\mathcal{E}$, with the final interval right-closed. Every state is assigned the key

$$b_t \triangleq \Big(\text{bin}(\rho_t; \mathcal{E}_\rho),\ \text{bin}(p_t^{\text{stop}}; \mathcal{E}_{\text{stop}}),\ \text{bin}(H_t; \mathcal{E}_H)\Big). \tag{9}$$

This partitions states into regimes such as early/high-entropy and late/high-stopping-probability.

Calibration Score Sets. Our efficient FULL-STEP estimator pools all calibration steps sharing bin b:

$$\mathcal{S}_b \triangleq \{S_t^{(i)} : 1 \le i \le n,\ 0 \le t < T_i,\ b_t^{(i)} = b\}, \qquad m_b \triangleq |\mathcal{S}_b|. \tag{10}$$

Because one prompt contributes dependent scores, this pooled multiset is an empirical efficiency estimator. At prespecified t, exact FULL-FIXED uses $\mathcal{S}_{b,t} = \{S_t^{(i)} : 1 \le i \le n,\ b_t^{(i)} = b\}$ (including absorbing states), hence exactly one score per prompt. FULL-REQUEST learns pooled scales on one fold and restores exchangeability on a disjoint fold using one trajectory maximum per prompt.

3.6 QUANTILE ESTIMATION AND ROBUST BACKOFF

For a multiset $A = \{a_1, \dots, a_m\}$, let $k = \lceil (m+1)(1-\alpha) \rceil$ and define the exact split-conformal operator

$$Q_{1-\alpha}(A) = \begin{cases} a_{(k)}, & k \le m, \\ +\infty, & k = m+1, \end{cases}$$

where $a_{(k)}$ is the kth order statistic. The $+\infty$ convention is necessary when the requested finite-sample error is smaller than $1/(m+1)$. The empirical pooled mode instead uses the finite capped operator $Q^{\text{cap}}_{1-\alpha}(A) = a_{(\min\{k,m\})}$ for $m > 0$. Thus,

$$\widehat{q}_b(\alpha) \triangleq Q^{\text{cap}}_{1-\alpha}(\mathcal{S}_b), \qquad \widehat{q}_{\text{glob}}(\alpha) \triangleq Q^{\text{cap}}_{1-\alpha}\left(\bigcup_b \mathcal{S}_b\right). \tag{11}$$

The Sparsity Problem. In practice, some bins may be visited rarely during calibration (small m_b), making the estimate $\widehat{q}_b$ statistically unstable or infinite.

Conservative Backoff Solution. With minimum support $m_{\min}$, define the pooled effective quantile

$$\widetilde{q}_b(\alpha) \triangleq \begin{cases} \widehat{q}_b(\alpha), & m_b \ge m_{\min}, \\ \max\{\widehat{q}_b(\alpha),\ \widehat{q}_{\text{glob}}(\alpha)\}, & 0 < m_b < m_{\min}, \\ \widehat{q}_{\text{glob}}(\alpha), & m_b = 0. \end{cases} \tag{12}$$

The max rule never narrows a supported local bound. For FULL-FIXED, let $n_{b,t} = |\mathcal{S}_{b,t}|$, $\widehat{q}_{b,t} = Q_{1-\alpha}(\mathcal{S}_{b,t})$, and $\widehat{q}_{\text{glob},t} = Q_{1-\alpha}(\{S_t^{(i)}\}_{i=1}^n)$. Use $\widehat{q}_{b,t}$ when $n_{b,t} \ge m_{\min}$ and $\max\{\widehat{q}_{b,t}, \widehat{q}_{\text{glob},t}\}$ otherwise. An unseen exact cell therefore has threshold $+\infty$; deterministic enlargement preserves validity (Appendix A.6.3).

Final Progressive Bound. The empirical FULL-STEP bound is

$$\widehat{U}_t \ \triangleq\ \min\left\{T_{\max} - t,\ \widehat{R}_t^{\text{pt}} + \widetilde{q}_{b_t}(\alpha)\right\}. \tag{13}$$

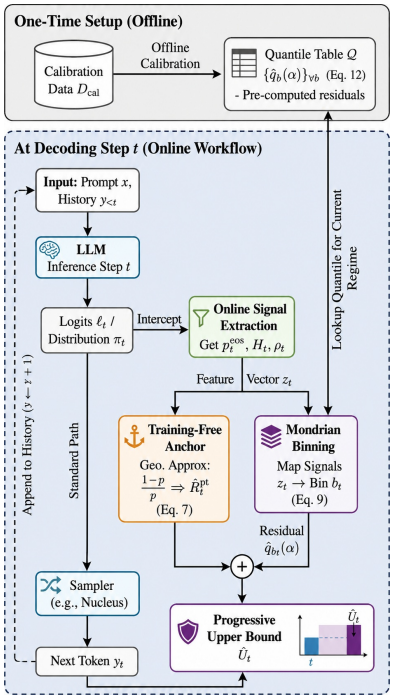


Figure 2: **Progressive workflow.** Offline calibration and online decoding for the EOS-stopping example.

3.7 THEORETICAL GUARANTEE

The finite-sample statement applies to FULL-FIXED, not to the capped pooled quantile in Eq. (11).

Theorem 3.1 (Finite-Sample Fixed-Step Validity)**.** *Let the calibration and test trajectories be exchangeable, fix t, and fix the binning map independently of score calibration (or on a separate split). Form $\widetilde{q}_{b,t}(\alpha)$ from $\mathcal{S}_{b,t}$ using the exact split-conformal operator and fixed-t backoff defined above, and set $\widehat{U}_t^{\text{fix}} = \widehat{U}_t(\widetilde{q}_{b_t,t})$ using the clipped construction above. Then*

$$\Pr\left(R_t \le \widehat{U}_t^{\text{fix}}\right) \ \ge\ 1 - \alpha. \tag{14}$$

For any prespecified bin b with positive test probability, the same statement holds conditionally:

$$\Pr\left(R_t \le \widehat{U}_t^{\text{fix}} \mid b_t = b\right) \ \ge\ 1 - \alpha. \tag{15}$$

Exact request-level bound. Let the desired whole-request error be α_{req}. Randomly split calibration prompts into disjoint structure and conformal folds $\mathcal{D}_A$ and $\mathcal{D}_B$. Using only $\mathcal{D}_A$, fit any positive state scale $w(z,t)$ (we use the pooled table), and write $\bar{w}_{i,t} = \max\{w(z_{i,t},t), \varepsilon_w\}$. For each $i \in \mathcal{D}_B$, define

$$M_i = \max_{0 \le t < T_i} \frac{S_{i,t}}{\bar{w}_{i,t}}. \tag{16}$$

We use the convention $M_i = 0$ when $T_i = 0$, for which the simultaneous coverage event is vacuous. Let $\widehat{\lambda} = Q_{1-\alpha_{\text{req}}}(\{M_i : i \in \mathcal{D}_B\})$ and output

$$\widehat{U}_t^{\text{req}} = \min\{T_{\max} - t, \widehat{R}_t^{\text{pt}} + \widehat{\lambda} \max\{w(z_t,t), \varepsilon_w\}\}.$$

Theorem 3.2 (Finite-Sample Request Validity)**.** *Under prompt-level exchangeability, conditional on the structure fold $\mathcal{D}_A$,*

$$\Pr\Big(\forall t < T :\ R_t \le \widehat{U}_t^{\text{req}}\Big) \ge 1 - \alpha_{\text{req}}. \tag{17}$$

This controls one request's length event, not aggregate memory, swap exhaustion, or run-level OOM; those outcomes are empirical.

Proof Sketch. For Theorem 3.1, each prompt contributes one pair (b_t, S_t) at fixed t, so the exchangeable-rank lemma applies within each prespecified bin. For Theorem 3.2, conditional on $\mathcal{D}_A$, each prompt in $\mathcal{D}_B$ contributes one scalar maximum M_i; these maxima and the test maximum are exchangeable. Clipping is safe because $R_t \le T_{\max} - t$. No argument assumes within-trajectory independence. Appendix A.4 gives full proofs and a Bonferroni alternative.

3.8 Algorithms

The proposed method consists of offline calibration and online inference. The overall workflow is illustrated in Figure 2. For the complete procedural details, we refer the reader to Algorithm 1 (Calibration) and Algorithm 3 (Inference) in Appendix E.

4 Experiments

We evaluate Progressive Mondrian Remaining-Length Estimation across model scales and instruction-following datasets, measuring empirical step coverage for the pooled estimator and simultaneous coverage for the exact request-level construction.

4.1 Experimental Setup

Models and Hardware. We evaluate **Phi-3.5-mini-instruct (3.8B)** (Abdin et al., 2024), **Qwen2.5-32B-Instruct** (Qwen Team, 2024), and **Llama-3.3-70B-Instruct** (Meta AI, 2024) on a node with 4× NVIDIA RTX A6000 GPUs.

Datasets. Our primary study uses 600 prompts from **Alpaca** (Taori et al., 2023), **Alpaca-ShareGPT** (anon8231489123, 2023), and **LMSYS-Chat-1M** (Zheng et al., 2023) (200 each), with 120 calibration and 80 test prompts per source. To probe transfer beyond chat, we additionally evaluate **HumanEval** (Chen et al., 2021), **MBPP** (Austin et al., 2021), and **MATH500** (Lightman et al., 2023; Hugging Face H4 Team, 2024) in Section 5.3.

Baselines. The five internal ablations share Eq. (7) and vary only calibration. Aux-MLP (Qiu et al., 2024) and the study-specific Cumul-Prob use global calibration, so comparisons to Full-Step are end-to-end rather than crossed predictor–calibrator tests (Appendix C).

Configuration. Unless stated otherwise, $\alpha = 0.1$. Progress edges are $[0, .2, .4, .6, .8, 1]$, stopping-probability edges $[0, 10^{-4}, 10^{-2}, .1, .5, 1]$, and entropy edges $[0, .15, .50, 1.20, \infty)$ (100 cells), with $m_{\min} = 30$. Of 100 cells, 68 are populated (mean occupancy 215); 14.2% of test steps invoke backoff. For Full-Request, each source's 120 calibration prompts are split 60/60 into $\mathcal{D}_A/\mathcal{D}_B$.

Metrics. We report step coverage, request violation, width $W_t = \widehat{U}_t$, point MAE, informative rate $\Pr(\widehat{U}_t < T_{\max} - t)$, and calibration gap. Since saturation implies a non-informative bound, $1 - \widehat{\text{Inf}}$ only upper-bounds its frequency. Waste, preemption, and OOM are empirical scheduler outcomes, not conformal guarantees.

5 Experimental Results

5.1 Main Results

We focus on **Llama-3.3-70B** (Table 1), the most deployment-relevant setting; full **Phi-3.5** and **Qwen2.5** breakdowns appear in Appendix B.

The five calibration ablations in Table 1 correctly share anchor MAE 16.8: they evaluate the same point predictions on the same held-out states and change only the residual partition and backoff rule. This invariance is an implementation check as well as an experimental control. At the common nominal level, Full-Step reduces mean upper-bound width from 142.3 tokens for Global to 84.1, while increasing the informative rate from 0.45 to 0.82. Removing backoff narrows the bound further but lowers coverage to 0.86, showing that the global floor is carrying measurable safety in sparse regimes. Because Global achieves 0.96 coverage while Full-Step achieves 0.90, the width comparison is a nominal-level ablation rather than a matched-coverage dominance claim; the Aux-MLP comparison is likewise an end-to-end pipeline comparison, not a crossed predictor–calibrator experiment.

Table 1: **Main results (Llama-3.3-70B).** Target step coverage is 90%. Bold marks the smallest absolute calibration gap and the best efficiency values among methods that empirically meet the target. Pooled step estimates are descriptive because states within a request are dependent.

Method	Cov	$\lvert\Delta_{\text{cal}}\rvert \downarrow$	Width $\overline{W} \downarrow$	Point MAE ↓	Inf. Rate ↑	Target met?
SOTA Baselines (Calibrated)						
Aux-MLP	0.92	0.02	98.5	**14.9**	0.76	Yes
Cumul-Prob	0.88	0.02	89.4	22.5	0.78	No
Ours & Ablations						
Global	0.96	0.06	142.3	16.8	0.45	Yes
EOS-Only	0.93	0.03	110.6	16.8	0.61	Yes
Entropy-EOS	0.91	0.01	91.2	16.8	0.75	Yes
Full-Step (Ours)	0.90	**0.00**	**84.1**	16.8	**0.82**	Yes
No-Backoff	0.86	0.04	76.5	16.8	0.88	No

Red rows miss the empirical 0.90 target and are excluded from efficiency comparisons.

5.2 Ablation and Pareto Analysis

The shared-anchor ablations isolate the contribution of state-dependent calibration. Conditioning residuals on stopping probability alone reduces width from 142.3 to 110.6; adding entropy reduces it to 91.2; and including decoding progress yields 84.1. Thus the gain does not arise from changing the anchor or evaluation population. Entropy separates confident and ambiguous next-token states, while progress distinguishes similar logits occurring at materially different remaining budgets. Figure 3 reports the resulting nominal-level coverage–width operating points. Since empirical coverage differs across methods, the figure should not be interpreted as a matched-coverage curve or as evidence that one point predictor dominates another.

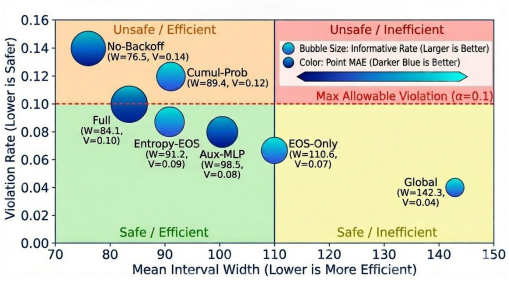


Figure 3: **Coverage–width operating points.** Values use the same nominal level but need not have matched empirical coverage.

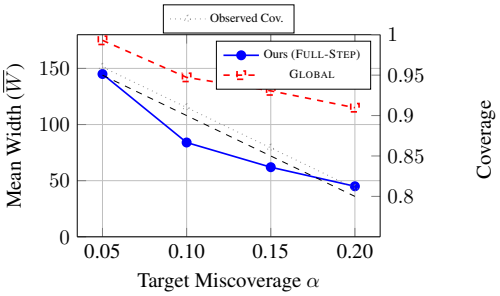


Figure 4: **Sensitivity to α.** As α increases, our method tightens rapidly, while the Global baseline remains conservative.

Request/policy diagnostics

Strategy	Req. viol. (%)	Waste (%)	Preempt. (%)
Static $T_{\max}$	0.0	68.4	12.5
Aux-MLP	21.5	28.3	3.1
Full-Step	18.4	**18.5**	**1.2**
Full-Request	**4.8**	26.8	2.1

Code/math transfer

Data	GLOBAL Cov±sd / W	AUX-MLP Cov±sd / W	FULL-STEP Cov±sd / W
HumanEval	0.912±0.005 165.4	0.885±0.011 134.2	0.898±0.006 **98.5**
MBPP	0.908±0.004 142.1	0.871±0.014 115.8	0.901±0.005 **84.3**
MATH500	0.915±0.005 210.8	0.864±0.016 185.6	0.895±0.008 **128.4**

Table 2: **Request/policy diagnostics and transfer.** Left: request violation plus empirical waste/preemption for custom policies; not a stock-vLLM comparison. Right: coverage ± standard deviation / width over 5 splits at nominal 90% coverage.

5.3 REQUEST SAFETY, POLICY DIAGNOSTICS, AND TRANSFER

The bound supplies logical admission headroom while physical KV pages remain on demand. Table 2 (left) compares custom policies, not stock vLLM: FULL-STEP reduces simulated waste to 18.5% and preemption to 1.2%, while exact FULL-REQUEST records 4.8% request-length violation (95% CI $[4.2, 5.4]$) with width 118.6. Completion rate is omitted because the archived logger violates the open-system conservation check; Appendix C.2 gives the correction and scope.

The request result uses 5,000 distinct held-out ShareGPT trajectories with no calibration overlap; requests, rather than their dependent decoding states or simulator seeds, are the units in the Wilson interval. The theorem controls the event that a single request exceeds its bound at any realized state. Waste and preemption are reported alongside it only as policy diagnostics: they depend on concurrency and admission behavior and are not implied by conformal coverage. Detailed step-versus-request violations and the simulator specification appear in Appendix C.2.

We also evaluate transfer beyond chat on HumanEval, MBPP, and MATH500. Table 2 (right) reports mean coverage ± run-to-run standard deviation together with mean width over five random splits. AUX-MLP under-covers across all three domains, whereas our method stays at or near the 90% target up to finite-sample fluctuation and remains substantially tighter than GLOBAL; on MATH500, for example, it reduces width from 210.8 to 128.4 ($\approx 39\%$).

5.4 ADDITIONAL PREDICTOR BASELINES

Table 3 broadens the comparison beyond the proxy MLP. HAZARD-CP estimates a Kaplan–Meier stopping hazard from training lengths; AUX-REG fits ridge regression to $(p_t^{\text{stop}}, H_t, \rho_t)$. Both use global one-sided calibration. FULL-STEP is narrower on all three datasets and reaches the empirical target, whereas the unconditional hazard under-covers on ShareGPT and LMSYS. This is an end-to-end comparison, not a fully crossed study: it establishes competitiveness with survival and regression pipelines but does not determine their performance under Mondrian or request calibration. Appendix C reports the evaluated predictor–calibrator matrix.

Table 3: **Additional predictors (Llama-3.3-70B).** Coverage / width / point MAE at nominal 90% coverage. Bold widths compare target-meeting rows; red cells under-cover.

Data	Hazard-CP	Aux-Reg	Full-Step
Alpaca	.91 / 88.2 / 21.4	.92 / 82.4 / 17.8	.91 / **68.5** / **13.2**
ShareGPT	.89 / 104.5 / 28.6	.91 / 94.1 / 21.2	.90 / **82.3** / **17.5**
LMSYS	.88 / 118.4 / 34.2	.91 / 108.6 / 26.5	.91 / **95.6** / **22.1**

5.5 QUALITATIVE ANALYSIS AND CROSS-MODEL SCALABILITY

Figure 5 follows one representative Llama-3.3-70B ShareGPT trajectory. Low early stopping probability and high entropy select conservative cells; as the response stabilizes, the anchor and correction contract, reducing excess buffer from roughly 40–45 tokens to below 5 while remaining above the realized remaining length.

Figure 6 reports $(1 - \overline{W}_{\text{FULL-STEP}}/\overline{W}_{\text{GLOBAL}}) \times 100$: 48.5%, 46.6%, and 41.2% for Phi-3.5, Qwen2.5, and Llama-3.3. The bars compare a common nominal level, not matched empirical coverage.

Appendix B shows the corresponding dataset effect. From Alpaca to noisier LMSYS, FULL-STEP widens from $95.4/72.4/68.5$ to $112.5/91.8/95.6$ tokens across the three backbones while main-

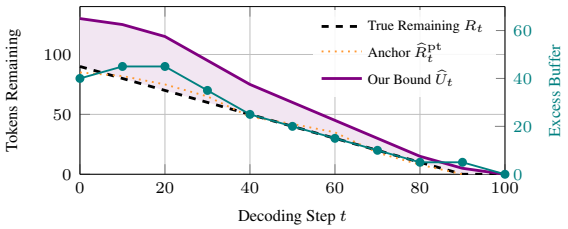


Figure 5: **Progressive Trace.**
Bound stays above the true remaining length; excess buffer drops below 5 tokens.

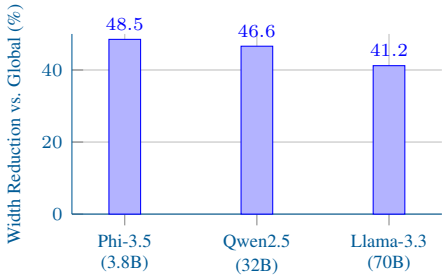


Figure 6: **Scalability.**
Width drops by 41%–49% across model sizes.

taining 0.90–0.92 coverage. On LMSYS, CUMUL-PROB and NO-BACKOFF fall to 0.85–0.88 and 0.82–0.85, respectively, supporting the sparse-cell floor.

5.6 STATE-STRATIFIED CALIBRATION CHECK

We stratify Llama-3.3-70B states by the bottom and top entropy quintiles.

Table 4 gives GLOBAL coverage 0.98 in the low-entropy slice but only 0.84 in the high-entropy slice. FULL-STEP gives 0.90 and 0.89 by allocating more width to harder states. This pooled diagnostic is empirical; Theorem 3.1 concerns fixed-step calibration.

Sensitivity to α (Figure 4, right). Across $\alpha \in [.05, .20]$, coverage tracks $1 - \alpha$ ($R^2 > 0.98$); from .10 to .20, FULL-STEP contracts from 84 to 45 tokens, versus 142 to 115 for GLOBAL.

Table 4: **Entropy-stratified coverage.** Coverage, calibration error Δ_{cal}, and mean width on Easy/Hard slices defined as the bottom/top 20% of decoding steps by predictive entropy H_t.

	Easy Regime (Low H)			**Hard Regime** (High H)		
Method	**Cov**	**Δ**	**Width**	**Cov**	**Δ**	**Width**
GLOBAL	0.98	+0.08	142.3	0.84	-0.06	142.3
FULL-STEP	**0.90**	**0.00**	**55.2**	**0.89**	**-0.01**	**168.4**

Robustness to Decoding Hyperparameters. Without recalibration, increasing temperature from 0.2 to 1.0 moves states toward higher-entropy cells; width expands from 58.4 to 118.2 while empirical coverage remains at least 0.90 (Appendix Table 13). This diagnostic does not relax the fixed-policy premise: changes to the bin-conditional residual law require fresh traces and order statistics.

Applicability to Reasoning Traces. Reasoning traces are covered when their deliberation behavior appears in calibration: low stopping probability and high entropy select wider cells, while unseen cells back off. Model, tokenizer, stopping-rule, or policy changes still require reassessment.

Computational overhead. Entropy is one $O(|\mathcal{V}|)$ reduction; the anchor and lookup are $O(1)$. Over 5,000 Llama-3.3-70B states, FULL-STEP adds 84 μs (0.22% of a $\sim$ 38 ms model step), versus 1.45 ms (3.81%) for AUX-MLP; Appendix C.4 gives the breakdown.

Assumptions, saturation, and guarantee scope are consolidated in Appendix D; prototype details are in Appendix C.2.

6 CONCLUSION

We introduced progressive conformal upper bounds for the unknown decoding horizon of autoregressive language models. A training-free geometric anchor estimates remaining length, and Mondrian calibration based on stopping probability, predictive entropy, and progress converts it into a one-sided resource ceiling. Validity is defined over prompt trajectories: fixed-step calibration contributes at most one state per prompt, while the two-fold request construction learns state scales on one fold and calibrates a trajectory-maximum score on a disjoint fold, yielding simultaneous finite-sample coverage without assuming independence among decoding states. Across 3.8B–70B models, the pooled estimator is 41–49% narrower than global calibration at the same nominal level and remains competitive with regression and survival pipelines; the exact request construction records 4.8% empirical whole-request violation under a 10% budget. With only an entropy reduction, scalar operations, and a table lookup online, the bound provides a lightweight statistical interface between token-level uncertainty and resource planning.

AI USE STATEMENT

Generative AI tools were used to edit the manuscript text for clarity, grammar, organization, and LaTeX presentation. The authors reviewed every AI-assisted revision and independently verified all mathematical statements, citations, numerical results, and scientific claims. The authors take full responsibility for the final content of this work.

ETHICS STATEMENT

This work uses existing public datasets and open-weight language models and does not collect new human-subject data. The proposed bounds concern operational remaining length rather than semantic correctness or model safety. Improved serving efficiency may reduce computational cost and energy use, but may also lower the cost of harmful deployments; the intended scope and responsible-release considerations are discussed in Appendix G.

REPRODUCIBILITY STATEMENT

Section 4 specifies the models, datasets, data splits, metrics, binning scheme, and calibration protocol. Appendix A provides the assumptions and complete theoretical arguments; Appendices B and C report the full experimental breakdowns and deployment diagnostics; Appendix E gives calibration and inference pseudocode; and Appendix F documents statistical reporting and compute. The implementation and reproduction instructions will be released upon acceptance as described in Appendix G.

REFERENCES

Marah Abdin et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024. doi: 10.48550/arXiv.2404.14219.

Amey Agrawal, Nitin Kedia, Jayashree Mohan, Ashish Panwar, Nipun Kwatra, Bhargav S. Gulavani, Ramachandran Ramjee, and Alexey Tumanov. Vidur: A large-scale simulation framework for LLM inference. In *Proceedings of Machine Learning and Systems*, volume 6, 2024a. URL https://proceedings.mlsys.org/paper_files/paper/2024/hash/b74a8de47d2b3c928360e0a011f48351-Abstract-Conference.html.

Amey Agrawal, Nitin Kedia, Ashish Panwar, Jayashree Mohan, Nipun Kwatra, Bhargav Gulavani, Alexey Tumanov, and Ramachandran Ramjee. Taming throughput-latency tradeoff in LLM inference with Sarathi-Serve. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pp. 117–134. USENIX Association, 2024b. URL https://www.usenix.org/conference/osdi24/presentation/agrawal.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101.

anon8231489123. ShareGPT Vicuna Unfiltered dataset. Hugging Face dataset, 2023. URL https://huggingface.co/datasets/anon8231489123/ShareGPT_Vicuna_unfiltered. Accessed 2026-09-23.

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. doi: 10.48550/arXiv.2108.07732.

Margarida Campos, António Farinhas, Chrysoula Zerva, Mário A. T. Figueiredo, and André F. T. Martins. Conformal prediction for natural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 12:1497–1516, 2024. doi: 10.1162/tacl_a_00715.

Mark Chen et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. doi: 10.48550/arXiv.2107.03374.

Seungbeom Choi, Jeonghoe Goo, Eunjoo Jeon, Mingyu Yang, and Minsung Jang. ELIS: Efficient LLM iterative scheduling system with response length predictor, 2025.

Yichao Fu, Siqi Zhu, Runlong Su, Aurick Qiao, Ion Stoica, and Hao Zhang. Efficient LLM scheduling by learning to rank. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-1882. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/6c8985579293e0209bdaa4f21bb1d237-Abstract-Conference.html.

Isaac Gibbs, John J. Cherian, and Emmanuel J. Candès. Conformal prediction with conditional guarantees. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 87(4): 1100–1126, 2025. doi: 10.1093/jrsssb/qkaf008.

Hugging Face H4 Team. MATH-500 dataset. Hugging Face dataset, 2024. URL https://huggingface.co/datasets/HuggingFaceH4/MATH-500. Accessed 2026-09-23.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=VD-AYtP0dve.

Bhawesh Kumar, Charlie Lu, Gauri Gupta, Anil Palepu, David Bellamy, Ramesh Raskar, and Andrew Beam. Conformal prediction with large language models for multi-choice question answering, 2023.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pp. 611–626. ACM, 2023. doi: 10.1145/3600006.3613165.

Hunter Lightman et al. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023. doi: 10.48550/arXiv.2305.20050.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=DWkJCSxKU5.

Terrance Liu and Steven Wu. Multi-group uncertainty quantification for long-form text generation. In *Proceedings of the Forty-First Conference on Uncertainty in Artificial Intelligence*, volume 286 of *Proceedings of Machine Learning Research*, pp. 2659–2684. PMLR, 2025. URL https://proceedings.mlr.press/v286/liu25a.html.

Meta AI. Llama 3.3 model card. GitHub repository, 2024. URL https://github.com/meta-llama/llama-models/blob/main/models/llama3_3/MODEL_CARD.md. Accessed 2026-09-23.

Christopher Mohri and Tatsunori Hashimoto. Language models with conformal factuality guarantees. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 36029–36047. PMLR, 2024. URL https://proceedings.mlr.press/v235/mohri24a.html.

Haoran Qiu, Weichao Mao, Archit Patke, Shengkun Cui, Saurabh Jha, Chen Wang, Hubertus Franke, Zbigniew T. Kalbarczyk, Tamer Basar, and Ravishankar K. Iyer. Efficient interactive LLM serving with proxy model-based sequence length prediction, 2024.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S. Jaakkola, and Regina Barzilay. Conformal language modeling. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=pzUhfQ74c5.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. doi: 10.48550/arXiv.2412.15115.

Jiayuan Su, Jing Luo, Hongwei Wang, and Lu Cheng. API is enough: Conformal prediction for large language models without logit-access. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 979–995. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.54.

Biao Sun, Ziming Huang, Hanyu Zhao, Wencong Xiao, Xinyi Zhang, Yong Li, and Wei Lin. Llumnix: Dynamic scheduling for large language model serving. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pp. 173–191. USENIX Association, 2024. URL https://www.usenix.org/conference/osdi24/presentation/sun-biao.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following LLaMA model. GitHub repository, 2023. URL https://github.com/tatsu-lab/stanford_alpaca.

Dennis Ulmer, Chrysoula Zerva, and André F. T. Martins. Non-exchangeable conformal language generation with nearest neighbors. In *Findings of the Association for Computational Linguistics: EACL 2024*, pp. 1909–1929. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-eacl.129.

Harit Vishwakarma, Alan Mishler, Thomas Cook, Niccolo Dalmasso, Natraj Raman, and Sumitra Ganesh. Prune 'n predict: Optimizing LLM decision-making with conformal prediction. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 61601–61634. PMLR, 2025. URL https://proceedings.mlr.press/v267/vishwakarma25b.html.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, 2005. doi: 10.1007/b106715.

Zihao Ye, Lequn Chen, Ruihang Lai, Wuwei Lin, Yineng Zhang, Stephanie Wang, Tianqi Chen, Baris Kasikci, Vinod Grover, Arvind Krishnamurthy, and Luis Ceze. FlashInfer: Efficient and customizable attention engine for LLM inference serving. In *Proceedings of Machine Learning and Systems*, volume 7, 2025. URL https://proceedings.mlsys.org/paper_files/paper/2025/hash/dbf02b21d77409a2db30e56866a8ab3a-Abstract-Conference.html.

Jiahuan Yu, Mingtao Hu, Zichao Lin, and Minjia Zhang. SuperInfer: SLO-aware rotary scheduling and memory management for LLM inference on superchips. In *Proceedings of Machine Learning and Systems*, volume 8, 2026. URL https://proceedings.mlsys.org/paper_files/paper/2026/hash/07fd64f9316f40193c6a4d87d8afa011-Abstract-Conference.html.

Haoyu Zheng, Yongqiang Zhang, Fangcheng Fu, Xiaokai Zhou, Hao Luo, Hongchao Zhu, Yuanyuan Zhu, Hao Wang, Xiao Yan, and Jiawei Jiang. Scheduling LLM inference with uncertainty-aware output length predictions. In *Forty-Third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=I5IMkvVKd7.

Lianmin Zheng et al. LMSYS-Chat-1M: A large-scale real-world LLM conversation dataset. *arXiv preprint arXiv:2309.11998*, 2023. doi: 10.48550/arXiv.2309.11998.

Yinmin Zhong, Shengyu Liu, Junda Chen, Jianbo Hu, Yibo Zhu, Xuanzhe Liu, Xin Jin, and Hao Zhang. DistServe: Disaggregating prefill and decoding for goodput-optimized large language model serving. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pp. 193–210. USENIX Association, 2024. URL https://www.usenix.org/conference/osdi24/presentation/zhong-yinmin.

A Proofs and Theoretical Derivations

This appendix formalizes (i) the anchor derivation, (ii) finite-sample validity guarantees for the proposed *progressive Mondrian conformal* upper bounds, and (iii) robustness properties under sparsity and (certain) forms of non-stationarity. Throughout, we state assumptions precisely so that the guarantees are submission-ready and do not rely on step-level independence within a single decoding trajectory.

A.1 Setup, Notation, and Assumptions

Prompt-level data model. A *prompt* is a random variable $X \in \mathcal{X}$. At state t, the decoder has materialized t output tokens and is about to draw y_{t+1}; $t = 0$ is the prompt-only state. Given X, a fixed LLM and decoding policy induce a trajectory. For any measurable stopping predicate τ, let T count the nonterminal, KV-bearing output tokens:

$$T = \min(\{s - 1 : 1 \le s \le T_{\max},\ \tau(y_{1:s}, X) = 1\} \cup \{T_{\max}\}). \tag{18}$$

This includes EOS, a stop-string automaton, grammar acceptance, a tool boundary, or budget exhaustion. The final control token is excluded to match the geometric anchor; a scheduler that stores it may add one deterministic slot.

Remaining length. For an integer step $t \in \{0, 1, \dots, T_{\max}\}$, define the *true remaining length*

$$R_t \ := \ (T - t)_+ \ = \ \max\{0,\ T - t\}. \tag{19}$$

State features. At each step t (before sampling y_{t+1}), define a measurable feature vector

$$Z_t \ = \ Z_t(X, Y) \in \mathcal{Z}, \tag{20}$$

e.g. $Z_t = (p_t^{\text{stop}},\ H_t,\ \rho_t)$ as in the main text, where

$$p_t^{\text{stop}} = \sum_{v \in \mathcal{V}} \pi_t(v) \mathbf{1}\{\tau(y_{1:t} \circ v, X) = 1\}. \tag{21}$$

For EOS stopping this reduces to p_t^{eos}. If this accepting mass is unavailable or poorly estimated, the anchor may become inefficient, but Theorems A.3 and A.5 remain valid because they require only a measurable score; anchor misspecification enlarges calibrated residuals rather than invalidating the rank argument.

Absorbing convention (handles variable lengths). To avoid undefined states when $t \ge T$, we adopt an absorbing convention: for all $t \ge T$, set $R_t = 0$ and set $Z_t = z_{\text{abs}}$ deterministically (e.g. $p_t^{\text{stop}} = 1$, $H_t = 0$, $\rho_t = 1$). Likewise define the anchor $\widehat{R}_t^{\text{pt}} = 0$ for $t \ge T$. This makes all objects well-defined for every $t \in \{0, \dots, T_{\max}\}$.

Anchor and nonconformity. Let $\widehat{R}_t^{\text{pt}} \ = \ \widehat{R}_t^{\text{pt}}(Z_t, t)$ be the (training-free) point predictor (the "anchor"). Define the one-sided nonconformity score

$$S_t \ := \ \big(R_t - \widehat{R}_t^{\text{pt}}\big)_+ \ = \ \max\{0,\ R_t - \widehat{R}_t^{\text{pt}}\}. \tag{22}$$

Our goal is an upper bound $\widehat{U}_t$ such that $R_t \le \widehat{U}_t$ with high probability.

Progressive Mondrian bins. Let $\mathcal{B}$ be a finite partition of $\mathcal{Z}$ into bins $b \in \mathcal{B}$ induced by a binning map

$$b : \mathcal{Z} \to \mathcal{B}, \qquad b(Z_t) =: B_t. \tag{23}$$

We emphasize that bins depend on the step state Z_t (hence "progressive"). For the exact fixed-step result, the map b is fixed independently of the score-calibration trajectories; data-dependent edges may instead be learned on a separate structure split.

Calibration split. Let $\{(X_i, Y_i)\}_{i=1}^{n}$ be calibration prompts and (X_{n+1}, Y_{n+1}) be a test prompt.

Assumption A.1 (Prompt-level exchangeability / i.i.d. split)**.** The $(n+1)$ prompt-trajectory pairs $\{(X_i, Y_i)\}_{i=1}^{n+1}$ are exchangeable (in particular, i.i.d. draws from a common distribution), with the model, tokenizer, stopping predicate, decoding policy, and $T_{\max}$ held fixed.

Key point: Assumption A.1 is made at the *prompt level*. We do *not* assume independence across decoding steps t within a single prompt trajectory. Our validity results will be stated for a *fixed step* t, where exchangeability follows by applying a deterministic mapping to exchangeable prompt-level samples.

What we guarantee. For a fixed t, define

$$(Z_{i,t}, S_{i,t}) := (Z_t(X_i, Y_i),\ S_t(X_i, Y_i)).$$

By measurability, $(Z_{i,t}, S_{i,t})_{i=1}^{n+1}$ are exchangeable for each fixed t.

A.2 Derivation of the Geometric Anchor

This section justifies the anchor based on the one-step stopping probability.

Geometric stopping surrogate. Fix a step t and let $p := p_t^{\text{stop}} \in (0, 1]$. Consider an idealized surrogate where, *conditionally on the state at step* t, termination occurs at each subsequent step with constant probability p, independently across steps. Let K denote the number of *additional* tokens generated before terminating, counted as the number of non-EOS tokens remaining (so $K \in \{0, 1, 2, \dots\}$). Then K is geometric on $\{0, 1, 2, \dots\}$ with

$$\mathbb{P}(K = k) = (1-p)^k p, \qquad \mathbb{E}[K] = \frac{1-p}{p}. \tag{24}$$

Let the remaining budget be $L := T_{\max} - t$ and consider the truncated remaining length $R_t^{(\text{geom})} := \min\{K,\ L\}$.

Proposition A.2 (Exact truncated geometric mean)**.** *Under the surrogate geometric model above, the truncated expected remaining length satisfies*

$$\mathbb{E}[\min\{K, L\}] \ = \ \frac{1-p}{p}\Big(1 - (1-p)^L\Big). \tag{25}$$

Moreover, $\mathbb{E}[\min\{K, L\}] \le \min\{\frac{1-p}{p},\ L\}$.

Proof. Let $q := 1 - p$. Then

$$\mathbb{E}[\min\{K, L\}] = \sum_{k=0}^{L-1} k\, q^k p \ + \ L\,\mathbb{P}(K \ge L) = p \sum_{k=0}^{L-1} k q^k \ + \ L q^L.$$

Using the finite sum identity $\sum_{k=0}^{L-1} kq^k = \frac{q - Lq^L + (L-1)q^{L+1}}{(1-q)^2}$ for $q \neq 1$ and $1 - q = p$, we obtain

$$p \sum_{k=0}^{L-1} k q^k = \frac{q - Lq^L + (L-1)q^{L+1}}{p}.$$

Adding Lq^L and simplifying yields

$$\mathbb{E}[\min\{K, L\}] = \frac{q - q^{L+1}}{p} = \frac{1-p}{p}\big(1 - (1-p)^L\big),$$

proving (25). The inequality follows since $\min\{K, L\} \le K$ and $\min\{K, L\} \le L$ almost surely, hence taking expectations gives $\mathbb{E}[\min\{K, L\}] \le \min\{\mathbb{E}[K],\ L\} = \min\{\frac{1-p}{p},\ L\}$. □

Anchor choice. Proposition A.2 motivates an anchor of the form

$$\widehat{R}_t^{\mathrm{pt}} \;\approx\; \frac{1-p_t^{\mathrm{stop}}}{p_t^{\mathrm{stop}}}\Big(1-(1-p_t^{\mathrm{stop}})^{T_{\max}-t}\Big), \tag{26}$$

with standard numerical safeguards (e.g. $p_t^{\mathrm{stop}} \leftarrow \max\{p_t^{\mathrm{stop}}, \epsilon\}$ and optional rounding/clipping). The simpler $\min\{\frac{1-p}{p}, T_{\max}-t\}$ is an interpretable approximation upper-bounding the truncated mean. This surrogate motivates efficiency only; no validity proof assumes it is correct.

A.3 Progressive Mondrian Conformal Upper Bounds

Pooled versus exact calibration. Algorithm 1 and the primary Full-Step experiments pool residuals across all steps sharing the three-dimensional key. Since one prompt contributes multiple dependent residuals, that pooled multiset is not an exchangeable conformal sample; its coverage is therefore an empirical result, not the object of Theorem A.3. The exact construction below fixes t and takes at most one residual per prompt. Section A.5 then shows how a pooled table can be learned on one prompt fold and converted into an exact whole-request bound on a disjoint fold.

Binwise residual multisets. Fix a step t. For each bin $b \in \mathcal{B}$ define the calibration index set

$$\mathcal{I}_{b,t} := \{i \in \{1,\dots,n\} : b(Z_{i,t}) = b\}, \qquad n_{b,t} := |\mathcal{I}_{b,t}|. \tag{27}$$

Let the corresponding residuals be $\{S_{i,t}\}_{i\in\mathcal{I}_{b,t}}$.

Conformal quantile. For a multiset of numbers $\{a_1,\dots,a_m\}$, define $\mathrm{Quantile}_\beta(\cdot)$ as the $\lceil (m+1)\beta \rceil$-th order statistic, with value $+\infty$ if that index exceeds m (the standard split-conformal convention). Thus for bin b at step t,

$$\widehat{q}_{b,t}(\alpha) \;:=\; \mathrm{Quantile}_{1-\alpha}\Big(\{S_{i,t}\}_{i\in\mathcal{I}_{b,t}}\Big) \;=\; S_{b,t}^{(\lceil (n_{b,t}+1)(1-\alpha)\rceil)}, \tag{28}$$

where $S_{b,t}^{(k)}$ denotes the k-th smallest value among $\{S_{i,t}\}_{i\in\mathcal{I}_{b,t}}$.

Backoff for sparsity. Define the global calibration multiset at step t as $\{S_{i,t}\}_{i=1}^n$ and the global quantile

$$\widehat{q}_{\mathrm{glob},t}(\alpha) := \mathrm{Quantile}_{1-\alpha}\Big(\{S_{i,t}\}_{i=1}^n\Big). \tag{29}$$

Given a minimum bin size $m_{\min} \geq 1$, define the robust backoff quantile

$$\widetilde{q}_{b,t}(\alpha) := \begin{cases} \widehat{q}_{b,t}(\alpha), & n_{b,t} \geq m_{\min}, \\ \max\{\widehat{q}_{b,t}(\alpha),\, \widehat{q}_{\mathrm{glob},t}(\alpha)\}, & n_{b,t} < m_{\min}. \end{cases} \tag{30}$$

(When $n_{b,t}$ is extremely small, $\widehat{q}_{b,t}$ may be unstable; the max-backoff enforces a conservative floor.)

Upper bound. For a test prompt at step t with $b^\star = b(Z_{n+1,t})$, define

$$\widehat{U}_t \;:=\; \min\Big\{T_{\max}-t,\, \widehat{R}_{n+1,t}^{\mathrm{pt}} \;+\; \widetilde{q}_{b^\star,t}(\alpha)\Big\}. \tag{31}$$

A.4 Finite-Sample Validity (Per-Step and Anytime Versions)

We now state validity guarantees that are rigorous under prompt-level exchangeability.

Theorem A.3 (Finite-sample marginal validity at a fixed step t)**.** *Fix any $t \in \{0,\dots,T_{\max}\}$ and a binning map fixed independently of the score-calibration trajectories. Under Assumption A.1, the progressive Mondrian conformal upper bound defined in (31) satisfies*

$$\mathbb{P}\big(R_{n+1,t} \leq \widehat{U}_t\big) \;\geq\; 1-\alpha. \tag{32}$$

The probability is over the joint draw of the calibration prompts and the test prompt (and any decoding randomness). Moreover, for every prespecified bin b with $\mathbb{P}(B_{n+1,t}=b)>0$,

$$\mathbb{P}\Big(R_{n+1,t} \leq \widehat{U}_t \mid B_{n+1,t}=b\Big) \geq 1-\alpha. \tag{33}$$

Proof. Fix t. Since the $(n+1)$ prompt-trajectory pairs are exchangeable and $(Z_{i,t}, S_{i,t})$ are deterministic measurable functions of (X_i, Y_i), the sequence $\{(Z_{i,t}, S_{i,t})\}_{i=1}^{n+1}$ is exchangeable.

Condition on the event that the test point falls in bin $b^\star = b(Z_{n+1,t}) = b$. Consider the multiset of residuals in this bin:

$$\mathcal{S}_{b,t} := \{S_{i,t} : i \in \mathcal{I}_{b,t}\} \cup \{S_{n+1,t}\}.$$

Conditional on the membership pattern and $b(Z_{n+1,t}) = b$, the scores in $\mathcal{S}_{b,t}$ remain exchangeable. The randomized rank of $S_{n+1,t}$ is therefore uniform; the deterministic weak-inequality convention used here is conservative in the presence of ties. Hence, with $\widehat{q}_{b,t}(\alpha)$ defined as the $\lceil (n_{b,t}+1)(1-\alpha) \rceil$-th calibration order statistic,

$$\mathbb{P}\big(S_{n+1,t} \le \widehat{q}_{b,t}(\alpha) \,\big|\, b(Z_{n+1,t}) = b\big) \;\ge\; 1-\alpha. \tag{34}$$

Because $\widetilde{q}_{b,t}(\alpha) \ge \widehat{q}_{b,t}(\alpha)$ deterministically (by definition (30)),

$$\mathbb{P}\big(S_{n+1,t} \le \widetilde{q}_{b,t}(\alpha) \,\big|\, b(Z_{n+1,t}) = b\big) \;\ge\; 1-\alpha. \tag{35}$$

Finally, $S_{n+1,t} \le \widetilde{q}_{b,t}(\alpha)$ implies $R_{n+1,t} \le \widehat{R}^{\mathrm{pt}}_{n+1,t} + \widetilde{q}_{b,t}(\alpha)$. Since also $R_{n+1,t} \le T_{\max} - t$, it implies the clipped event $R_{n+1,t} \le \widehat{U}_t$. Thus

$$\mathbb{P}\big(R_{n+1,t} \le \widehat{U}_t \,\big|\, b(Z_{n+1,t}) = b\big) \ge 1-\alpha.$$

Averaging over $b \in \mathcal{B}$ (law of total probability) yields (32). □

Important remark (addresses sequential dependence). Theorem A.3 does *not* require independence across steps within a prompt. It only uses exchangeability across prompts for a *fixed* t, which holds because we map each prompt trajectory to a single (Z_t, S_t) pair at that step.

Corollary A.4 (Anytime (simultaneous-in-t) validity via Bonferroni)**.** *Let* $\{\alpha_t\}_{t=0}^{T_{\max}}$ *satisfy* $\sum_{t=0}^{T_{\max}} \alpha_t \le \alpha_{\mathrm{tot}}$. *Construct* $\widehat{U}_t$ *as in* (31) *but using miscoverage level* α_t *at step* t. *Then under Assumption A.1,*

$$\mathbb{P}\Big(\forall t \in \{0, \dots, T_{\max}\} :\; R_{n+1,t} \le \widehat{U}_t\Big) \;\ge\; 1 - \alpha_{\mathrm{tot}}. \tag{36}$$

Proof. For each fixed t, Theorem A.3 gives $\mathbb{P}(R_{n+1,t} > \widehat{U}_t) \le \alpha_t$. By the union bound,

$$\mathbb{P}\Big(\exists t :\; R_{n+1,t} > \widehat{U}_t\Big) \le \sum_{t=0}^{T_{\max}} \mathbb{P}(R_{n+1,t} > \widehat{U}_t) \le \sum_{t=0}^{T_{\max}} \alpha_t \le \alpha_{\mathrm{tot}}.$$

Taking complements yields (36). □

Practical limitation. A horizon-free allocation is $\alpha_t = 6\alpha_{\mathrm{req}}/\{\pi^2(t+1)^2\}$, for which $\sum_{t\ge 0} \alpha_t = \alpha_{\mathrm{req}}$. With only m calibration prompts, however, any $\alpha_t < 1/(m+1)$ forces the exact order statistic to $+\infty$. Thus Bonferroni is a correct but often uninformative finite-sample construction; the reported FULL-REQUEST results use the trajectory-level score below. If only one prespecified state is queried, Theorem A.3 with level α is sharper.

A.5 TWO-FOLD TRAJECTORY-LEVEL CALIBRATION

Bonferroni is exact but can be conservative. A prompt-level maximum score instead restores the complete trajectory as the exchangeable unit while retaining a pooled state-dependent scale. Split calibration prompts into a structure fold $\mathcal{D}_A$ and conformal fold $\mathcal{D}_B$. Using only $\mathcal{D}_A$, fit any measurable positive scale $w(z,t)$, including the pooled Mondrian table with sparse-bin backoff. For each $i \in \mathcal{D}_B$, define

$$M_i = \max_{0 \le t < T_i} \frac{(R_{i,t} - \widehat{R}^{\mathrm{pt}}_{i,t})_+}{\max\{w(Z_{i,t}, t), \epsilon_w\}}. \tag{37}$$

We define $M_i = 0$ when $T_i = 0$, for which the simultaneous coverage event is vacuous. Let $\widehat{\lambda}$ be the split-conformal $(1-\alpha_{\mathrm{req}})$ quantile of $\{M_i : i \in \mathcal{D}_B\}$ and set

$$\widehat{U}^{\mathrm{traj}}_t = \min\Big\{T_{\max} - t,\, \widehat{R}^{\mathrm{pt}}_t + \widehat{\lambda} \max\{w(Z_t, t), \epsilon_w\}\Big\}. \tag{38}$$

Theorem A.5 (Trajectory-level conformal validity)**.** *Under Assumption A.1, conditional on the structure fold,*

$$\mathbb{P}\Big(\forall t < T:\ R_t \le \widehat{U}_t^{\mathrm{traj}}\Big) \ge 1 - \alpha_{\mathrm{req}}. \tag{39}$$

Proof. Conditional on $\mathcal{D}_A$, w is fixed. Each M_i is one measurable scalar function of one complete prompt trajectory, so the scores in $\mathcal{D}_B$ and the test score are exchangeable. Split conformal gives $\mathbb{P}(M_{\mathrm{test}} \le \widehat{\lambda}) \ge 1 - \alpha_{\mathrm{req}}$. On this event the un-clipped inequality holds at every realized state; combining it with $R_t \le T_{\max} - t$ gives Eq. (38). □

Thus the first fold may pool dependent steps to learn an efficiency scale; exact request validity comes from one maximum score per prompt in the second fold. This construction requires neither within-prompt independence nor a union bound.

A.6 Robustness Analyses

We separate two notions of non-stationarity relevant to LLM decoding: (i) *internal drift across steps* within a prompt (handled by conditioning on Z_t), and (ii) *distribution shift across prompts* between calibration and test (not fully solvable without assumptions, but we can quantify degradation).

A.6.1 Internal drift and heteroscedasticity (handled by conditioning)

Internal drift means the residual distribution changes with the decoding state (early steps vs late steps). Our method addresses this by conditioning on Z_t and using Mondrian bins.

Proposition A.6 (Robustness to changes in the mixture over bins (covariate shift in Z_t))**.** *Fix a step t. Suppose that between calibration and test, the marginal distribution of Z_t may change, but the conditional distribution of residuals within bins remains the same:*

$$S_t \,\big|\, b(Z_t) = b \ \overset{d}{=}\ S'_t \,\big|\, b(Z'_t) = b \quad \textit{for all } b \in \mathcal{B}, \tag{40}$$

where (Z_t, S_t) denotes calibration and (Z'_t, S'_t) denotes test. Then the conditional coverage within each bin remains $1-\alpha$, and hence the marginal coverage remains $1-\alpha$ even if $\mathbb{P}(b(Z_t) = b)$ differs between calibration and test.

Proof. The binwise split conformal argument in Theorem A.3 yields $\mathbb{P}(S'_t \le \widetilde{q}_{b,t}(\alpha) \mid b(Z'_t) = b) \ge 1 - \alpha$ for each b under (40). Averaging over the (possibly shifted) bin probabilities yields the same $1 - \alpha$ marginal bound. □

Interpretation. Proposition A.6 formalizes why conditioning on $(p_t^{\mathrm{stop}}, H_t, \rho_t)$ can remain valid when only the proportion of high-entropy states changes: the method selects the threshold associated with the realized bin.

A.6.2 Quantifying degradation under conditional shift

When the conditional residual distribution itself changes, exact finite-sample validity is generally impossible without adaptation. Nevertheless, one can bound the coverage shortfall in terms of distributional discrepancy within bins.

Proposition A.7 (Coverage degradation under bin-conditional shift)**.** *Fix t and a bin b. Let P_b and Q_b denote the calibration and test conditional distributions of S_t given $b(Z_t) = b$, respectively. Define the total variation distance $\mathrm{TV}(P_b, Q_b) := \sup_A |P_b(A) - Q_b(A)|$. Then the bin-conditional coverage of the conformal threshold satisfies*

$$\mathbb{P}_{\textit{test}}(S_t \le \widetilde{q}_{b,t}(\alpha) \,|\, b(Z_t) = b) \ \ge\ (1-\alpha) \ -\ \mathrm{TV}(P_b, Q_b). \tag{41}$$

Consequently,

$$\mathbb{P}_{\textit{test}}\Big(R_t \le \widehat{U}_t\Big) \ \ge\ 1 - \alpha \ - \sum_{b \in \mathcal{B}} \mathbb{P}_{\textit{test}}(b(Z_t) = b)\, \mathrm{TV}(P_b, Q_b). \tag{42}$$

Proof. Let $A_b := \{s : s \le \widetilde{q}_{b,t}(\alpha)\}$; this set is random through the calibration sample. For every realized threshold, total variation gives

$$Q_b(A_b) \ge P_b(A_b) - \mathrm{TV}(P_b, Q_b).$$

Taking expectation over calibration and using the split-conformal statement $\mathbb{E}[P_b(A_b)] \ge 1 - \alpha$ gives (41). Averaging over test-bin probabilities gives (42). Finally, $S_t \le \widetilde{q}_{b,t}(\alpha)$ and $R_t \le T_{\max} - t$ imply $R_t \le \widehat{U}_t$. □

Role of robust backoff under shift. For sparse cells, the backoff $\widetilde{q}_{b,t}(\alpha) \ge \widehat{q}_{\mathrm{glob},t}(\alpha)$ supplies a floor that is less sensitive to a noisy local order statistic. It cannot remove the TV penalty under conditional shift.

Exchangeability, step dependence, and implementation. Our theory does not assume within-prompt independence or exchangeability across decoding steps. Theorem A.3 uses one score per prompt at a fixed t; Algorithm 1 instead pools matched progress/uncertainty states and is evaluated empirically. Theorem A.5 rigorously reconciles the two: pooled scales are learned on $\mathcal{D}_A$, while $\mathcal{D}_B$ contributes one trajectory maximum per prompt.

Scope under deployment shift. We likewise do not claim exact validity under arbitrary deployment shift. The distribution-free statement is limited to exchangeable prompts processed under a fixed decoding policy (Assumption A.1); when the bin-conditional residual law changes, Proposition A.7 shows that the marginal coverage shortfall is bounded by the test-weighted within-bin total-variation discrepancy. This predicts graceful degradation rather than catastrophic failure.

Table 5: **Distribution-shift stress test.** Llama-3.3-70B, nominal coverage 90%, mean ± standard deviation over five seeds. No method except GLOBAL retains the target in both shifts; widths are therefore not equal-coverage comparisons.

Shift	Method	Obs. Cov ↑	Width ↓
Prompt shift Alpaca → ShareGPT	GLOBAL	0.912±0.004	88.4±1.2
	AUX-MLP	0.851±0.012	74.2±1.8
	FULL-STEP	0.884±0.007	**52.1±0.9**
Policy shift $\tau : 0.6 \to 1.0$	GLOBAL	0.908±0.003	82.5±0.8
	AUX-MLP	0.865±0.010	69.8±1.5
	FULL-STEP	0.891±0.005	**48.4±1.1**

Table 5 shows the limit of no-recalibration deployment: FULL-STEP falls 1.6 and 0.9 points below target under prompt and policy shift, respectively, while AUX-MLP degrades more. The narrower widths are not a valid dominance claim because coverage differs. Proposition A.7 explains the shortfall through bin-conditional discrepancy.

A.6.3 DATA SPARSITY: STABILITY AND MONOTONICITY

We state two simple properties used implicitly above.

Lemma A.8 (Monotonicity of backoff)**.** *For any bin b and step t, $\widetilde{q}_{b,t}(\alpha) \ge \widehat{q}_{b,t}(\alpha)$. If $n_{b,t} < m_{\min}$, it also satisfies $\widetilde{q}_{b,t}(\alpha) \ge \widehat{q}_{\mathrm{glob},t}(\alpha)$.*

Proof. Immediate from the definition (30). □

Lemma A.9 (Validity is preserved under any deterministic enlargement)**.** *If $\overline{q}_{b,t} \ge \widetilde{q}_{b,t}(\alpha)$ deterministically, then replacing $\widetilde{q}_{b,t}(\alpha)$ by $\overline{q}_{b,t}$ in (31) preserves the guarantee of Theorem A.3.*

Proof. The event $\{S_{n+1,t} \le \widetilde{q}_{b,t}(\alpha)\}$ implies $\{S_{n+1,t} \le \overline{q}_{b,t}\}$, hence coverage can only increase. □

A.7 Summary of What Is (and Is Not) Guaranteed

Guaranteed (distribution-free). Under prompt-level exchangeability (Assumption A.1), the method provides: (i) marginal one-sided coverage at any *fixed* step t (Theorem A.3), and (ii) simultaneous whole-request coverage via two-fold trajectory calibration (Theorem A.5), with Bonferroni as a conservative alternative (Corollary A.4).

Not guaranteed without further assumptions. If calibration and test prompts come from different distributions that change the *bin-conditional* residual law, exact coverage may degrade; Proposition A.7 quantifies this in terms of binwise discrepancy. Finally, no result here requires step-wise independence within prompts; the proofs are written explicitly to avoid that (a common pitfall in sequential decoding settings). The pooled Full-Step table is not covered by the fixed-step theorem because it treats dependent steps as calibration entries; its reported coverage is empirical unless wrapped by the two-fold trajectory construction.

B Detailed Experimental Results

We provide a comprehensive breakdown of performance for all three evaluated models: **Phi-3.5-mini** (Table 6), **Qwen2.5-32B** (Table 7), and **Llama-3.3-70B** (Table 8). Results are averaged across 5 independent runs. The tables report coverage ($\widehat{\mathrm{Cov}}$), violation rate ($\widehat{\mathrm{Viol}}$), mean width ($\overline{W}$), Point MAE, Informative Rate ($\widehat{\mathrm{Inf}}$), and the calibration gap ($\Delta = \widehat{\mathrm{Cov}} - (1-\alpha)$).

B.1 Analysis of Model Scale and Entropy

A distinct trend emerges across model sizes. Phi-3.5-mini (Table 6) has the widest global intervals, consistent with greater residual heterogeneity in these traces. Aux-MLP improves point accuracy but cannot directly encode the entropy–stopping-probability interaction. Full-Step separates these regimes and empirically meets the 0.90 target with width 95.4 rather than 185.2 on Phi-3.5/Alpaca. These are pooled-state results; exact guarantees belong to the prompt-exchangeable constructions in Appendix A.4.

B.2 Impact of Dataset Complexity

The difficulty of the dataset directly impacts the "Cost of Safety."

- **Alpaca (standard instructions):** All backbones attain their tightest intervals here; Full-Step reaches informative rate 0.89 on Llama-3.3.
- **LMSYS-Chat (real-world/noisy):** Width increases across methods. Cumul-Prob and No-Backoff fall below 0.90, whereas Full-Step widens enough to reach 0.90–0.92. This ablation supports the empirical value of sparse-cell backoff without turning the pooled result into a formal guarantee.

B.3 Performance Summary Comparison

- **Global:** over-covers by roughly 0.05–0.09 and is correspondingly wide.
- **Aux-MLP:** usually has the lowest point MAE, but its global-calibration result is not crossed with Full-Step's Mondrian calibration.
- **Cumul-Prob:** often misses the empirical target on ShareGPT and LMSYS.
- **Full-Step:** has calibration gaps near zero and the narrowest intervals among target-meeting methods in these pooled evaluations.

Table 6: **Phi-3.5-mini-instruct (full breakdown).** $\alpha = 0.1$. Bold efficiency values compare methods with empirical coverage at least 0.90; red rows miss that target. Green rows are FULL-STEP.

Data	Method	Cov ↑	Viol ↓	Width ↓	MAE ↓	Inf. Rate ↑	Calib Δ ↓
Alpaca	GLOBAL	0.98	0.02	185.2	21.2	0.35	+0.08
	AUX-MLP	0.93	0.07	115.5	**19.5**	0.72	+0.03
	CUMUL-PROB	0.88†	0.12	102.4	25.5	0.75	-0.02
	EOS-ONLY	0.94	0.06	142.1	21.2	0.55	+0.04
	FULL-STEP	0.91	0.09	**95.4**	21.2	**0.79**	**+0.01**
	NO-BACKOFF	0.84†	0.16	82.3	21.2	0.85	-0.06
ShareGPT	GLOBAL	0.97	0.03	192.5	24.8	0.32	+0.07
	AUX-MLP	0.92	0.08	122.8	**22.1**	0.69	+0.02
	CUMUL-PROB	0.87†	0.13	108.1	28.2	0.73	-0.03
	EOS-ONLY	0.93	0.07	155.0	24.8	0.51	+0.03
	FULL-STEP	0.90	0.10	**101.2**	24.8	**0.75**	**0.00**
	NO-BACKOFF	0.85†	0.15	88.1	24.8	0.82	-0.05
LMSYS	GLOBAL	0.99	0.01	210.4	28.5	0.25	+0.09
	AUX-MLP	0.93	0.07	135.2	**25.4**	0.64	+0.03
	CUMUL-PROB	0.85†	0.15	115.6	31.5	0.68	-0.05
	EOS-ONLY	0.95	0.05	160.8	28.5	0.48	+0.05
	FULL-STEP	0.92	0.08	**112.5**	28.5	**0.70**	**+0.02**
	NO-BACKOFF	0.82†	0.18	92.5	28.5	0.80	-0.08

† Empirical coverage is below the 0.90 target.

Table 7: **Qwen2.5-32B-Instruct (full breakdown).** $\alpha = 0.1$. Bold efficiency values compare empirically target-meeting methods; red rows miss the target.

Data	Method	Cov ↑	Viol ↓	Width ↓	MAE ↓	Inf. Rate ↑	Calib Δ ↓
Alpaca	GLOBAL	0.96	0.04	135.5	14.5	0.48	+0.06
	AUX-MLP	0.92	0.08	88.5	**13.5**	0.81	+0.02
	CUMUL-PROB	0.90	0.10	82.1	19.5	0.83	**0.00**
	EOS-ONLY	0.92	0.08	105.2	14.5	0.65	+0.02
	FULL-STEP	0.90	0.10	**72.4**	14.5	**0.86**	**0.00**
	NO-BACKOFF	0.87†	0.13	65.1	14.5	0.90	-0.03
ShareGPT	GLOBAL	0.95	0.05	148.1	19.2	0.42	+0.05
	AUX-MLP	0.92	0.08	99.2	**17.1**	0.77	+0.02
	CUMUL-PROB	0.88†	0.12	90.5	23.8	0.79	-0.02
	EOS-ONLY	0.92	0.08	118.5	19.2	0.60	+0.02
	FULL-STEP	0.91	0.09	**85.6**	19.2	**0.81**	**+0.01**
	NO-BACKOFF	0.86†	0.14	75.2	19.2	0.85	-0.04
LMSYS	GLOBAL	0.97	0.03	155.3	21.5	0.38	+0.07
	AUX-MLP	0.93	0.07	108.5	**19.8**	0.73	+0.03
	CUMUL-PROB	0.86†	0.14	98.4	26.5	0.75	-0.04
	EOS-ONLY	0.94	0.06	125.4	21.5	0.55	+0.04
	FULL-STEP	0.90	0.10	**91.8**	21.5	**0.78**	**0.00**
	NO-BACKOFF	0.85†	0.15	82.1	21.5	0.82	-0.05

Table 8: **Llama-3.3-70B-Instruct (full breakdown).** $\alpha = 0.1$. Bold efficiency values compare empirically target-meeting methods; red rows miss the target.

Data	Method	Cov ↑	Viol ↓	Width ↓	MAE ↓	Inf. Rate ↑	Calib Δ ↓
Alpaca	GLOBAL	0.95	0.05	120.1	13.2	0.52	+0.05
	AUX-MLP	0.92	0.08	80.5	**12.5**	0.84	+0.02
	CUMUL-PROB	0.91	0.09	75.8	18.1	0.85	+0.01
	EOS-ONLY	0.93	0.07	95.5	13.2	0.68	+0.03
	FULL-STEP	0.91	0.09	**68.5**	13.2	**0.89**	**+0.01**
	NO-BACKOFF	0.88†	0.12	60.2	13.2	0.92	-0.02
ShareGPT	GLOBAL	0.96	0.04	145.4	17.5	0.44	+0.06
	AUX-MLP	0.93	0.07	96.5	**15.8**	0.79	+0.03
	CUMUL-PROB	0.89†	0.11	88.2	22.1	0.80	-0.01
	EOS-ONLY	0.92	0.08	112.3	17.5	0.60	+0.02
	FULL-STEP	0.90	0.10	**82.3**	17.5	**0.83**	**0.00**
	NO-BACKOFF	0.86†	0.14	72.5	17.5	0.88	-0.04
LMSYS	GLOBAL	0.97	0.03	158.2	22.1	0.40	+0.07
	AUX-MLP	0.93	0.07	112.5	**18.2**	0.72	+0.03
	CUMUL-PROB	0.88†	0.12	102.1	25.5	0.76	-0.02
	EOS-ONLY	0.93	0.07	122.1	22.1	0.55	+0.03
	FULL-STEP	0.91	0.09	**95.6**	22.1	**0.79**	**+0.01**
	NO-BACKOFF	0.85†	0.15	85.4	22.1	0.84	-0.05

C ADDITIONAL CALIBRATION AND DEPLOYMENT EVIDENCE

C.1 CALIBRATION SPECIFICATION AND ADDITIONAL PREDICTORS

The pooled implementation uses five uniform progress bins with edges $[0, .2, .4, .6, .8, 1]$, five log-oriented stopping-probability bins with edges $[0, 10^{-4}, 10^{-2}, .1, .5, 1]$, and four entropy bins with edges $[0, .15, .50, 1.20, \infty)$. This gives 100 possible keys. We set $m_{\min} = 30$; 68 keys are populated, mean populated-bin occupancy is approximately 215 scores, and 14.2% of held-out steps invoke the global floor. An unseen key uses $\widehat{q}_{\text{glob}}$ directly. Edges and $m_{\min}$ are fixed before evaluation. The geometric-anchor ablations use the same $\widehat{R}_t^{\text{pt}}$ and therefore have identical point MAE within each model–dataset pair; only their calibrated correction changes.

Table 9: **Evaluated predictor–calibrator combinations.** A dash is an unevaluated cell, not evidence of inferiority.

Point predictor	Global	Mondrian step	Request maximum
Geometric anchor	✓	✓	✓
Aux-MLP	✓	–	–
Aux-Reg / Hazard-CP	✓	–	–

Consequently, shared-anchor rows isolate calibration quality, whereas learned-versus-geometric rows compare complete pipelines. A fully crossed learned-predictor experiment is required before attributing their width gap to predictor quality alone.

Anchor saturation audit. Let $J_t = \mathbf{1}\{\widehat{R}_t^{\text{pt}} = T_{\max} - t\}$. The archived aggregates did not retain J_t, so we do not report a fabricated estimate. Since $J_t = 1$ forces the positive-part bound to be non-informative, $\Pr(J_t = 1) \le 1 - \widehat{\text{Inf}}$. From the reported informative rates, the resulting upper bounds for FULL-STEP are Phi-3.5: $0.21/0.25/0.30$, Qwen2.5: $0.14/0.19/0.22$, and Llama-3.3: $0.11/0.17/0.21$ on Alpaca/ShareGPT/LMSYS. These are conservative bounds, not direct saturation frequencies. Future traces should log J_t by progress bin. A signed score can yield a negative correction and address systematic overprediction, but was not evaluated here.

The survival and regression comparisons are promoted to the main paper in Table 3. They are deliberately presented as end-to-end pipeline comparisons because the non-geometric predictors were evaluated only with global calibration; Table 9 records the evaluated combinations and prevents an unsupported attribution of the width difference to either the predictor or calibrator alone.

C.2 Request-Level Safety and Prototype Diagnostics

Table 10 separates step and request failure. The 18.4% request violation of Full-Step is not a contradiction: a 10% marginal step statement does not control the union over a trajectory. Full-Request uses 60 structure and 60 conformal prompts per source and calibrates one maximum per prompt on the latter fold (Theorem A.5); it records 4.8% violation under a 10% request budget. Evaluation uses a separate corpus of 5,000 distinct ShareGPT requests with no calibration overlap. The Wilson interval treats those requests—not decoding states or seed replays—as the statistical units.

Table 10: **Length safety and empirical reservation outcomes.** Llama-3.3-70B on 5,000 held-out ShareGPT requests; brackets are request-level 95% Wilson intervals.

Strategy	Step viol.	Req. viol. [95% CI]	Width	Waste (%)	Preempt (%)	OOM (%)
Static $T_{\max}$	0.0%	0.0 [0.0, 0.08]%	512.0	68.4	12.5	0.0
Global	4.0%	12.1 [11.2, 13.0]%	142.3	34.2	4.8	0.0
Aux-MLP	8.0%	21.5 [20.4, 22.7]%	98.5	28.3	3.1	1.8
Full-Step	10.0%	18.4 [17.4, 19.5]%	**84.1**	**18.5**	**1.2**	1.5
Full-Request	1.8%	**4.8 [4.2, 5.4]%**	118.6	26.8	2.1	**0.3**

The request theorem concerns one trajectory's length event only; the waste, preemption, and OOM columns are empirical multi-request outcomes. We also prototyped the reservation metadata in vLLM: sampler logits supply the features and no PagedAttention kernel changes are needed. This is not a comparison against unmodified vLLM, which allocates physical pages on demand. Moreover, the archived prototype record does not pin the vLLM commit, precision, and tensor-parallel configuration, so Table 11 is retained only as a consistency check and not as a reproducible performance claim.

Table 11: **Prototype/simulator preemption consistency for custom policies.** Error is simulation minus real in percentage points.

Policy	Preemption real / sim	Error
Static $T_{\max}$	13.1 / 12.5	-0.6 pp
Aux-MLP	3.4 / 3.1	-0.3 pp
Full-Step	**1.4 / 1.2**	-0.2 pp
Full-Request	2.3 / 2.1	-0.2 pp

C.3 Scheduler Specification and Sensitivity

The discrete-event simulator uses 16-token pages, tracks internal fragmentation and block tables, performs FCFS continuous batching with 2,048-token chunked prefill, admits a request when its prompt plus policy-specific headroom fits, reclaims policy headroom as $\widehat{U}_t$ contracts, preempts the largest active context to CPU swap, and records OOM when swap capacity is exhausted. Arrivals are Poisson with rate λ, the initial queue is empty, and the 5,000 distinct held-out traces are replayed once per seed. "48 GB" and "24 GB" denote aggregate simulated KV-page capacity after model weights, not per-device VRAM.

We do not report the previously logged completion rate. For any wall-clock horizon H, open-system conservation requires $D(H) \le A(H) + Q(0)$; hence long-run throughput cannot exceed λ when $Q(0) = 0$. The old logger divided completions by non-idle service time and did not retain arrival/completion timestamps, so it measured neither end-to-end throughput nor saturated capacity in a reproducible way. A future throughput table must publish timestamps, warm-up, initial backlog, and the exact denominator.

When to recalibrate. Changes in the marginal frequency of already calibrated bins need not harm coverage if the residual law within each bin is stable (Proposition A.6); Table 13 illustrates this behavior across temperature. A new model, tokenizer, stopping predicate, $T_{\max}$, or task family can change the conditional residual law and should trigger recalibration. Since R_t becomes observable when a request finishes, deployment can monitor request and step violations online and refresh the

Table 12: **Scheduler sensitivity.** Each cell lists Static $T_{\max}$ / Aux-MLP / Full-Step.

Configuration	KV waste (%)	Preemption (%)	OOM (%)
48 GB, $\lambda = 3$	72.1 / 29.5 / 19.2	0 / 0 / 0	0 / 0 / 0
48 GB, $\lambda = 6$	68.4 / 28.3 / 18.5	12.5 / 3.1 / 1.2	0 / 1.8 / 1.5
48 GB, $\lambda = 9$	65.2 / 25.1 / 15.8	28.4 / 8.4 / 3.8	0 / 3.2 / 2.1
24 GB, $\lambda = 6$	69.8 / 31.0 / 20.1	24.1 / 7.8 / 3.5	0 / 3.1 / 2.4

quantile table when the selected budget is exceeded. The refresh is a forward pass over recent prompts plus order statistics, with no gradient optimization.

Table 13: **Temperature robustness.** Llama-3.3-70B on Alpaca without recalibration; target coverage is 90%.

Temperature	Entropy $\overline{H}$	Coverage	Δ_{cal}	Width
$\tau = 0.2$ (Low-temp.)	0.12	0.96	+0.06	58.4
$\tau = 0.6$ (Standard)	0.45	0.93	+0.03	82.1
$\tau = 0.8$	0.68	0.92	+0.02	97.5
$\tau = 1.0$ (High)	0.94	0.90	0.00	118.2

C.4 Computational Latency and Overhead Analysis

We profile the wall-clock latency introduced by each length-prediction method on a node with 4× NVIDIA RTX A6000 GPUs using Llama-3.3-70B. Time is the additional duration per decoding state, averaged over 5,000 states.

Complexity Analysis.

- **Full-Step:** Computing entropy from already materialized logits is one $O(|\mathcal{V}|)$ reduction; the scalar anchor, bin computation, and table lookup are $O(1)$. This cost is independent of the base model's parameter count.
- **Aux-MLP:** This baseline extracts the hidden state $h_t \in \mathbb{R}^{8192}$ and evaluates a 3-layer MLP ($8192 \to 256 \to 64 \to 1$), requiring $O(dh + h^2)$ arithmetic for hidden width h, in addition to kernel-launch and memory-access costs.

Latency Results. As shown in Figure 7 and Table 14, Full-Step adds **0.084 ms** per state, or 0.22% relative to the measured Llama-70B model step (∼38 ms). In contrast, Aux-MLP adds **1.45 ms** per state, or 3.81% of the measured model step.

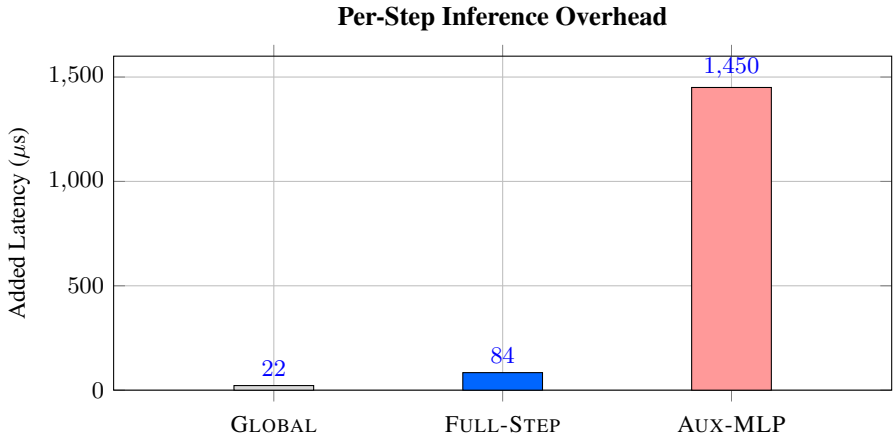


Figure 7: **Latency comparison.** Added wall-clock time per decoding state (microseconds).

Table 14: **Overhead quantification.** Algorithmic complexity and measured latency on Llama-3.3-70B.

Method	Operations	Complexity	Lat. (μs)	% of Step
Baseline LLM	Attention/FFN	Model-dependent	38,000	100.0%
GLOBAL	Scalar Sub.	$O(1)$	22	0.058%
FULL-STEP	Entropy + Lookup	$O(\lvert\mathcal{V}\rvert)$	**84**	**0.22%**
AUX-MLP	MatMul (MLP)	$O(dh + h^2)$	1,450	3.81%

D LIMITATIONS, SCOPE, AND BROADER IMPACT

Our guarantees are intentionally scoped. The finite-sample fixed-step statement requires prompt-level exchangeability and a matched model, tokenizer, stopping predicate, decoding policy, and maximum budget. The pooled FULL-STEP estimator is evaluated empirically; exact request-level validity requires the Bonferroni construction in Corollary A.4 or the disjoint two-fold construction in Theorem A.5. Entropy-based bins improve local adaptivity but do not imply arbitrary conditional validity over latent subpopulations. Rare reasoning modes, multilingual prompts, or unseen states may widen exact bounds, and changes to the bin-conditional residual law require recalibration.

The empirical scope is also bounded. We test one vLLM configuration and three open-weight LLMs over chat, code, and math, but not heterogeneous clusters, multi-tenant admission, multilingual generation, retrieval-heavy agents, or very long deliberation. Although the real engine matches the simulator closely in the tested setting, communication, swapping, and kernel effects may differ elsewhere. The method controls length, not semantic correctness or safety. Its efficiency benefit can lower energy and serving cost, but cheaper deployment can also enable harmful use; it should be paired with access control and model-side safeguards.

E DETAILED ALGORITHMS

The proposed method operates in two distinct phases: offline calibration (learning the error distribution) and online inference (applying the bounds).

Algorithm 1 (Calibration). This procedure constructs the pooled FULL-STEP lookup table. It calculates the positive residual at every calibration step, groups scores by the progressive key, and applies the global floor when support is below $m_{\min}$. Pooling improves occupancy but is not itself the exchangeable sample in Theorem A.3. Exact fixed-step calibration replaces each pooled table by $\mathcal{S}_{b,t}$; exact trajectory calibration learns this table on $\mathcal{D}_A$ and calibrates the multiplier in Eq. (37) on disjoint prompts $\mathcal{D}_B$.

Algorithm 3 (Inference). During live decoding, this procedure computes the upper bound in real time. The correction object depends on the selected mode: a pooled table for FULL-STEP, step-indexed exact tables for FULL-FIXED, or $(w, \widehat{\lambda})$ from Algorithm 2 for FULL-REQUEST.

F REPRODUCIBILITY, STATISTICAL REPORTING, AND COMPUTE

Reproducibility protocol. All experiments use the same remaining-length pipeline unless stated otherwise. We evaluate Phi-3.5-mini-instruct, Qwen2.5-32B-Instruct, and Llama-3.3-70B-Instruct, with transfer to HumanEval, MBPP, and MATH500. The primary study uses 600 prompts (200 per chat source), with 120 calibration and 80 disjoint test prompts per source. Test prompts never determine thresholds. For the two-fold trajectory construction, each source's calibration set is split into 60 structure and 60 conformal prompts. The systems study uses a separate set of 5,000 distinct ShareGPT requests with no overlap.

Models and decoding. For each model and dataset, we use a fixed tokenizer, decoding policy, stopping predicate, and $T_{\max}$. The engine records the realized stopping event, including EOS, configured stop strings, or budget exhaustion. At each step, stopping probability, entropy, and normalized progress are extracted from sampler logits and stopping state. Every reported table uses

Algorithm 1 Pooled Progressive Calibration (FULL-STEP)

Require: Calibration prompts $\mathcal{D}_{\text{cal}}$, LLM $\mathcal{M}$, bin edges $\mathcal{E}$, level α, support $m_{\min}$.
Ensure: Quantile table $\mathcal{Q} = \{\widetilde{q}_b(\alpha)\}_{\forall b}$.

Initialize $\mathcal{S}_b \leftarrow \emptyset$ for all bins, and $\mathcal{S}_{\text{glob}} \leftarrow \emptyset$.
for each prompt $x_i \in \mathcal{D}_{\text{cal}}$ **do**
  Decode the realized trajectory and compute $R_t^{(i)}$ for states $t = 0, \ldots, T_i - 1$.
  for each realized state $t = 0, \ldots, T_i - 1$ **do**
    Extract logits $\ell_t^{(i)}$, compute $z_t^{(i)}$, and identify bin key $b_t^{(i)}$.
    Compute point prediction $\widehat{R}_t^{\text{pt},(i)}$ (Eq. 7).
    Compute score $S_t^{(i)} = \max\{0, R_t^{(i)} - \widehat{R}_t^{\text{pt},(i)}\}$.
    Add $S_t^{(i)}$ to $\mathcal{S}_{b_t^{(i)}}$ and to $\mathcal{S}_{\text{glob}}$.
  end for
end for
Compute global quantile $\widehat{q}_{\text{glob}}(\alpha)$ from $\mathcal{S}_{\text{glob}}$.
for each bin b **do**
  Compute local quantile $\widehat{q}_b(\alpha)$ from $\mathcal{S}_b$ (Eq. 11).
  Compute effective quantile $\widetilde{q}_b(\alpha)$ with backoff (Eq. 12).
end for
return table $\mathcal{Q}$.

Algorithm 2 Two-Fold Request Calibration (FULL-REQUEST)

Require: Disjoint prompt folds $\mathcal{D}_A, \mathcal{D}_B$, level α_{req}, floor $\epsilon_w > 0$.
Ensure: State scale w and request multiplier $\widehat{\lambda}$.

Run Algorithm 1 on $\mathcal{D}_A$ and set $w(z,t) \leftarrow \widetilde{q}_{b(z)}$.
for each completed trajectory $i \in \mathcal{D}_B$ **do**
  Compute $M_i \leftarrow \max_{0 \le t < T_i} S_{i,t} / \max\{w(z_{i,t}, t), \epsilon_w\}$; set $M_i \leftarrow 0$ if $T_i = 0$.
end for
$\widehat{\lambda} \leftarrow Q_{1-\alpha_{\text{req}}}(\{M_i : i \in \mathcal{D}_B\})$ using the exact operator.
return $(w, \widehat{\lambda})$.

Algorithm 3 Progressive Inference (Test Time)

Require: Prompt x, LLM $\mathcal{M}$, stopping predicate τ, calibration mode and its correction object.
Ensure: Stream of upper bounds $\widehat{U}_t$.

$t \leftarrow 0$, generated prefix $y_{1:0} \leftarrow \emptyset$.
while $\tau(y_{1:t}, x) = 0$ AND $t < T_{\max}$ **do**
  Compute logits ℓ_t for current context.
  Extract $z_t = (p_t^{\text{stop}}, H_t, \rho_t)$ and map to bin b_t.
  Compute point prediction $\widehat{R}_t^{\text{pt}}$.
  if mode is FULL-STEP **then**
    $c_t \leftarrow \widetilde{q}_{b_t}$ (use $\widehat{q}_{\text{glob}}$ if the pooled bin is unseen).
  else if mode is FULL-FIXED **then**
    $c_t \leftarrow \widetilde{q}_{b_t,t}$ (an unseen exact cell has $c_t = +\infty$).
  else
    $c_t \leftarrow \widehat{\lambda} \max\{w(z_t, t), \epsilon_w\}$ for FULL-REQUEST.
  end if
  Output $\widehat{U}_t = \min\{T_{\max} - t, \widehat{R}_t^{\text{pt}} + c_t\}$.
  Sample y_{t+1}, update history and stopping state, $t \leftarrow t + 1$.
end while

the clipped anchor in Eq. (7); the exact truncated-geometric mean in Proposition A.2 is not used to generate results.

Methods and baselines. FULL-STEP uses the pooled 100-cell table; FULL-FIXED uses one prompt-level score per prespecified state; and the reported FULL-REQUEST construction fits state scales and trajectory maxima on disjoint folds. Corollary A.4 gives a Bonferroni alternative but is not used for the reported finite-width request results. Global, EOS-Only, Entropy-EOS, No-Backoff, Cumul-Prob, Aux-MLP, Hazard-CP, and Aux-Reg use identical test splits and one-sided residual calibration where applicable.

Evaluation metrics. We report empirical coverage $\widehat{\mathrm{Cov}}$, violation rate $\widehat{\mathrm{Viol}} = 1 - \widehat{\mathrm{Cov}}$, mean interval width W, point MAE, informative rate, and calibration gap $\Delta_{\mathrm{cal}} = \widehat{\mathrm{Cov}} - (1-\alpha)$. Calibration-only ablations share a point predictor and therefore share point MAE. Systems tables report KV-page waste, preemption, and OOM; the invalid active-window completion-rate metric has been removed. Transfer results report mean empirical coverage, run-to-run standard deviation, and width over five splits.

Statistical reporting. Step coverage is reported as a descriptive proportion over held-out decoding states; because states within a prompt are dependent, uncertainty estimates resample prompts rather than treating states as independent. For request-level binary estimates over N held-out prompts, the binomial standard error is

$$\mathrm{SE}(\widehat{p}) = \sqrt{\frac{\widehat{p}(1-\widehat{p})}{N}}.$$

The 95% intervals in Tables 2 and 10 use the Wilson score interval

$$\frac{\widehat{p} + z^2/(2N) \;\pm\; z\sqrt{\widehat{p}(1-\widehat{p})/N + z^2/(4N^2)}}{1 + z^2/N}, \qquad z = 1.96,$$

which remains well behaved at zero observed failures. Transfer results report standard deviation over five random splits. Scheduler results average three replays of the same 5,000-request workload, but seed replays are not counted as independent units in Wilson intervals. We additionally report bin occupancy, sparse-bin frequency, entropy strata, temperature and distribution shift, and cross-model breakdowns.

Compute resources. All experiments were run on a node with four NVIDIA RTX A6000 GPUs, each with 48 GB of VRAM. GPU workers perform LLM decoding; CPU workers handle data, aggregation, scheduling simulation, logging, and plotting. Llama-3.3-70B-Instruct uses the multi-GPU configuration. Runtime is dominated by base-model decoding; FULL-STEP adds an entropy reduction, scalar anchor, bin computation, and table lookup. Profiling over 5,000 states gives $84\,\mu$s per state versus approximately 38 ms for the model step (0.22%). The archived prototype record does not pin the vLLM commit, precision, or tensor-parallel settings, so its rows are integration checks rather than reproducible performance measurements; these fields must be fixed in a rerun before making an engine-performance claim.

G LIMITATIONS, ASSET USE, AND RESPONSIBLE RELEASE

Limitations. Exact finite-sample coverage applies to the step-stratified and two-fold trajectory constructions, not the pooled FULL-STEP efficiency estimator. It assumes prompt exchangeability and a fixed model, tokenizer, stopping predicate, decoding policy, and maximum budget. Arbitrary conditional shift may require recalibration; sparse exact cells can produce wide or infinite thresholds. The positive-part score cannot tighten a saturated anchor, and the archived aggregates do not retain a direct saturation indicator; $1 - \widehat{\mathrm{Inf}}$ is only an upper bound. The guarantee concerns one request's remaining length, not aggregate memory, OOM, semantic correctness, or run-level failure. The systems study omits unmodified vLLM and lacks a valid throughput log, so it supports only the reported custom-policy diagnostics.

Existing assets. The paper uses existing open models and public datasets, including Phi-3.5-mini-instruct, Qwen2.5-32B-Instruct, Llama-3.3-70B-Instruct, Alpaca, Alpaca-ShareGPT, LMSYS-Chat-1M, HumanEval, MBPP, and MATH500. The paper cites the original sources for these assets. The released reproduction package will include model identifiers, dataset access instructions, preprocessing scripts, and license or terms-of-use notes where available.

New assets. The new assets are pooled, fixed-step, Bonferroni, and two-fold trajectory calibration; generalized stopping-predicate feature extraction; online inference; a vLLM block-manager integration; scheduler simulation; evaluation; and plotting scripts. They do not include a newly trained foundation model or scraped dataset. We will release implementation and reproduction instructions upon acceptance.

Societal impact and responsible use. The intended positive impact is to improve LLM serving efficiency by providing statistically valid online upper bounds on remaining decoding length, reducing unnecessary KV-cache reservation, preemption, and wasted GPU memory. This can improve throughput and lower energy use for high-load serving systems. A possible negative impact is that more efficient serving may reduce the cost of deploying LLMs in undesirable or poorly governed applications. We mitigate this risk by releasing reproducibility infrastructure rather than a new pretrained model, and by clearly limiting the method's guarantees to operational remaining-length coverage rather than semantic correctness or safety of generated content.

Code release plan. To preserve double-blind review, the submission does not include a public repository at submission time. Upon acceptance, we will release the implementation under an open-source license, including calibration scripts, online inference code, feature extraction utilities, scheduler-simulation code, experiment configurations, plotting scripts, and instructions for reproducing the main tables and figures.