Provable Test-Time Scaling for Beam Search in LLM Reasoning

Qijia He[*]
The Ohio State University
he.2806@osu.edu

Yu Huang[*]
University of Pennsylvania
yuh42@wharton.upenn.edu.

Yuan Cheng[*]
National University of Singapore
Address
yuan.cheng@u.nus.edu

Yuxin Chen
University of Pennsylvania
yuxinc@wharton.upenn.edu

Yingbin Liang
The Ohio State University
liang.889@osu.edu

Abstract

Beam-search–based test-time methods provide an effective way to improve large language model (LLM) performance on long-horizon generation by pruning invalid reasoning paths early, leading to significantly improved reasoning efficiency and more favorable test-time cost scaling. Despite strong empirical success, the theoretical understanding of beam search remains limited. In this paper, we study the test-time compute guarantee of the commonly used beam search framework that uses the model's internal log-likelihood for intermediate scoring, while relying on an external reward model only after a complete response is generated. We first establish a lower bound for vanilla beam search, showing that at least $\Omega(C^\star(x)^2)$ samples are required for the optimal response to survive, where $C^\star(x)$ is the token-level coverage coefficient for prompt x. This motivates our modified confidence-filtered beam search (CF-Beam), which reduces the sufficient coverage dependence from quadratic to nearly linear under prefix competitiveness, for fixed horizon, gap, and target accuracy. We then show that the regret of CF-Beam is upper-bounded by the probability of rare failure events and the reward estimation error scaled by a path-level coverage coefficient, where the rare-failure term vanishes as per-step sampling increases. Our results highlight a fundamental advantage of beam search over sequence-level inference methods such as *Best-of-N* and *Best-of-Majority*. While the guarantees of these approaches typically involve coverage coefficients that grow exponentially with the horizon L, CF-Beam controls the dominant search-induced term through a token-level coverage coefficient that scales polynomially with L. Our numerical experiments further confirm that beam search is more robust on hard instances and under increasing reasoning horizons.

1 Introduction

Test-time computing has emerged as a powerful paradigm for improving the performance of large language models (LLMs) by allocating additional computation at inference time. Unlike post-training

[*]equal contribution

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

approaches such as supervised fine-tuning and reinforcement learning from human feedback, which improve performance by updating the model's parameters, test-time methods operate on a fixed model and improve accuracy via repeated sampling, aggregation, and selection. Due to their simplicity in implementation and striking performance scaling behavior, test-time methods have been widely deployed in real-world systems (Zaharia et al., 2024; Team et al., 2023).

Existing and widely studied test-time methods include Best-of-N sampling (BoN), majority voting, chain-of-thought (CoT) reasoning, and their corresponding variants (Jinnai et al., 2024; Ichihara et al., 2025; Wang et al., 2022; Liu et al., 2025a; Di et al., 2025). Despite their empirical success, applying these approaches to complex real world tasks can be prohibitively expensive, as BoN and voting-based schemes require a rapidly increasing number of samples to maintain accuracy as context length grows. Moreover, the presence of imperfect verifiers can exacerbate selection errors by over-trusting spurious candidates or discarding correct but low-scoring solutions (Huang et al., 2025).

To improve the reasoning capabilities of LLMs on long reasoning chains and mitigate the compounding errors in final answers that arise from accumulated small mistakes or model imperfections across intermediate steps (Xu et al., 2025; Xie et al., 2023), *beam search* has been found to be a promising test-time approach for sparse-success and long-horizon settings. This method expands candidate responses token by token and applies a step-level verifier, which can be either an external reward model (Lightman et al., 2023; Setlur et al., 2024) or an internal self-evaluation metric (Xie et al., 2023), to prune low-quality continuations at each step. By aggregating candidate trajectories and pruning inferior paths, beam search allocates inference-time computation to the most promising prefixes toward desired outputs, thereby increasing the probability of identifying correct solutions (Yu et al., 2025a; Zhu et al., 2024) while simultaneously reducing unnecessary test-time computation.

Despite a rich body of empirical evidence demonstrating the effectiveness of beam search, theoretical analysis of beam search remains limited. Meister et al. (2020) studied what implicit objective function beam search optimizes. Recently, Fadeeva et al. (2025) studied beam search for self-consistency-based uncertainty quantification. To the best of our knowledge, there has not been a principled theoretical understanding of its test-time scaling performance, which raises the following fundamental research questions:

1. *How does beam search scale at test time, particularly when test-time compute is constrained and verifiers are imperfect?*
2. *Compared to recent test-time guarantees for Best-of-Majority Di et al. (2025) and best-of-N Huang et al. (2025), does beam search enjoy provable advantages?*

From the analysis standpoint, the main technical obstacle is that step-by-step pruning in beam search induces strong token-level dependencies over time, making the resulting dynamics harder to analyze than in sequence-level methods. Moreover, beam search often relies on heuristic intermediate verifiers whose step-wise scores are not aligned with a single, globally consistent objective.

In this paper, we answer the above questions by analyzing the test-time performance of beam search in a sampling-access setting, where token probabilities are estimated from sampled continuations and used as an internal verifier, while an external reward model is accessible only after the complete response is generated.

- **Vanilla-Beam vs. CF-Beam.** We establish a lower bound of sample complexity, showing that vanilla beam search (Vanilla-Beam) requires the sample size to be at least $\Omega(C^\star(x)^2)$ for the optimal response to survive, where $C^\star(x)$ is the token-level coverage coefficient for prompt $x \in \mathcal{X}$. This motivates us to filter the low-probability samples via the estimated sampling policy, resulting in the confidence-filtered beam search (CF-Beam) as shown in Figure 1, which reduces the sufficient coverage dependence from quadratic to nearly linear under prefix competitiveness, for fixed horizon, gap, and target accuracy.
- **Regret Analysis.** We show that the regret of CF-Beam is upper bounded by the probability of rare failure events and the reward estimation error scaled by a path-level coverage coefficient, where the rare-failure term vanishes as per-step sampling increases. Our results highlight a fundamental advantage of beam search over sequence-level inference methods such as *Best-of-N* and *Best-of-Majority*. While the guarantees of those approaches involve coverage coefficients that may potentially grow *exponentially* with the horizon L, CF-Beam controls the dominant search-induced term through a token-level coverage coefficient that scales only *polynomially* with L.

- **Reward-Model-Free Regret.** We further analyze the regret for a fully self-consistent CF-Beam. In contrast to standard CF-Beam, whose guarantees depend on the error of an external reward model, the regret bound for the self-consistent version is driven solely by the probability of rare failure events. Since the regret is decoupled from the reward-model error, self-consistent CF-Beam admits a sufficient total query budget growing only *linearly* in L, up to logarithmic factors, for fixed beam width, token-level coverage, prefix gap, and target accuracy.
- **Numerical experiments.** Our numerical results compare CF-Beam against vanilla beam search, BoN, Majority voting and Best-of-Majority, demonstrating that CF-Beam achieves better performance on hard instances and is more robust to increasing horizon length.

2 Preliminaries

Problem Setting and Decoding. Test-time algorithms are decoding strategies that use extra inference-time compute to map an input prompt under a fixed LLM model to an improved output response. Formally, given a prompt $x \in \mathcal{X}$, the goal of the algorithm is to produce a response y, which is represented as a length-L token sequence $y = (a_1, \ldots, a_L) \in \mathcal{A}^L$, based on a fixed reference LLM policy $\pi_{\text{ref}} : \mathcal{X} \to \Delta(\mathcal{A}^L)$. Here, $\mathcal{A}$ denotes the token vocabulary, and $\Delta(\mathcal{A}^L)$ is the set of probability distributions over $\mathcal{A}^L$.

The quality of the complete response y is evaluated by a *ground-truth* outcome reward $r^\star : \mathcal{X} \times \mathcal{Y} \to [0, 1]$, which is only revealed after the entire response is generated. Such sparse outcome rewards are standard in LLM settings, e.g., in reinforcement learning with verifiable feedback (RLVR) (Chen et al., 2025; He et al., 2025). In practice, $r^\star$ is typically unavailable, and the algorithm designer instead relies on an approximate reward model $\hat{r} : \mathcal{X} \times \mathcal{Y} \to [0, 1]$ that assigns a surrogate score $\hat{r}(x, y)$ to assess the quality of response y for any prompt x.

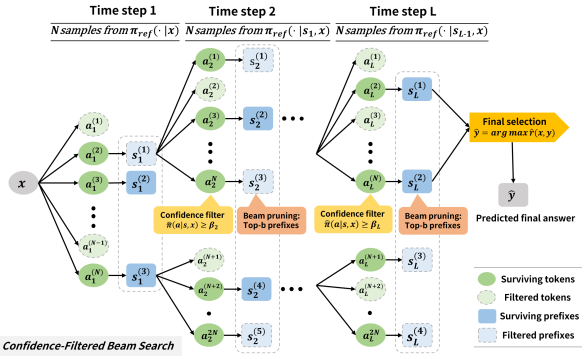


Figure 1: Confidence-Filtered Beam Search Pipeline

The reference policy π_{ref} further admits a standard autoregressive factorization by steps. Let $s_{t-1} := (a_1, \ldots, a_{t-1})$ denote the token prefix generated up to step $t-1$ (with $s_0 = \emptyset$). Then we have

$$\pi_{\text{ref}}(y \mid x) = \prod_{t=1}^{L} \pi_{\text{ref}}(a_t \mid x, s_{t-1}),$$

Here, on the right-hand side, with a slight abuse of notation, we use π_{ref} to also denote a token-level reference policy $\mathcal{X} \times \mathcal{S} \to \Delta(\mathcal{A})$, where $\mathcal{S}$ is the set of all token prefixes from steps 1 to L. Since beam search algorithms aggregate and process information at the token level, we focus throughout the remainder of the paper on the token-level policy $\pi_{\text{ref}} : \mathcal{X} \times \mathcal{S} \to \Delta(\mathcal{A})$. We assume black-box sampling access to the reference policy, without access to full logits or exact next-token probabilities, so token probabilities must be estimated from sampled continuations.

Goal and Evaluation. Given a prompt x, the goal of parallel test-time computing is to design an algorithm `Alg` that interacts with the base policy $\pi_{\text{ref}}(\cdot \mid s, x)$ through sampling and aggregation and outputs a response $y \sim \pi_{\text{Alg}}(\cdot \mid x)$, aiming to maximize the expected ground-truth reward $\mathbb{E}_{y \sim \text{Alg}(\cdot|x)}\left[r^\star(x, y)\right]$.

Let $y^\star := \arg\max_{y \in \mathcal{Y}} r^\star(x, y)$ denote the optimal response that maximizes the *true* outcome reward. We define the regret on instance x as

$$\text{Reg}(x) = r^\star(x, y^\star) - \mathbb{E}_{y \sim \pi_{\text{Alg}}(\cdot|x)}[r^\star(x, y)]\,. \tag{1}$$

Computational Complexity. We measure the test-time computational cost of an algorithm `Alg` by the number of *token-level policy queries* it makes to the base model. Concretely, each time `Alg`

queries $\pi_{\text{ref}}(\cdot \mid s, x)$ to sample from the next-token distribution at a prefix s, we count one query. We use N to denote the (per-prefix) sample complexity, i.e., the number of such queries issued at each prefix s, and let $B(x)$ denote the total number of such queries on instance x. We refer to $B(x)$ as the total (query) budget of `Alg`.

Coverage Coefficient. To quantify the ability of the base policy π_{ref} to generate the optimal response, following Huang et al. (2025), we introduce coverage coefficients that measure how difficult it is for the base policy π_{ref} to sample the optimal trajectory.

Definition 1 (Token- and sequence-level coverage)**.** Fix an instance x and let $y^\star(x) = (a_1^\star, \ldots, a_L^\star)$ be the unique optimal response, with optimal prefixes $s_{t-1}^\star = a_{1:t-1}^\star$. Let $\pi^\star$ denote the oracle policy that deterministically follows the optimal token at each step, i.e., $\pi^\star(a_t^\star \mid s_{t-1}^\star, x) = 1$ for all $t \in [L]$. The token-level coverage coefficient is defined as

$$C_t^\star(x) := \frac{\pi^\star(a_t^\star \mid s_{t-1}^\star, x)}{\pi_{\text{ref}}(a_t^\star \mid s_{t-1}^\star, x)} = \frac{1}{\pi_{\text{ref}}(a_t^\star \mid s_{t-1}^\star, x)}. \tag{2}$$

We further define $C_{\max}^\star(x) := \max_{t \in [L]} C_t^\star(x)$ and $\overline{C}(x) := \prod_{t=1}^{L} C_t^\star(x)$ as the sequence-level coverage coefficient.

The quantity $C_t^\star(x)$ captures the inverse probability of sampling the optimal token at depth t under π_{ref}, while $\overline{C}(x)$ reflects the compounding difficulty along an autoregressive trajectory. This notion is consistent with the L_1 coverage coefficient studied in Huang et al. (2025); Di et al. (2025).

For notational convenience, we omit the dependence on the instance x when no ambiguity arises.

3 Algorithm

Beam search is a search-based decoding algorithm for LLM inference, which explores diverse responses and applies a step-level verifier to prune low-quality continuations at each step, aiming to ultimately concentrate on the most promising prefixes and reduce unnecessary test-time computational cost.

Vanilla Beam Search (Vanilla-Beam). Vanilla-Beam, as shown in Algorithm 1 (Xie et al., 2023; Yu et al., 2025a), generally maintains a beam of b trajectories. At each step, it expands each trajectory by enumerating multiple next-token candidates (Lines 6-11), scores the newly formed trajectories under a chosen metric (Line 18), and prunes these trajectories to the top b candidates, which forms the beam $\mathcal{B}$ for the next step (Line 23). After reaching the end-of-sequence or length limit, it selects the final output by maximizing an external reward model over the top-b estimated log-likelihood candidates (Line 25).

Algorithm 1 **Vanilla-Beam** vs **CF-Beam**: CF-Beam adds highlighted lines **(Line 14-15)**

1: **Input:** Base policy π_{ref}, beam width $b \geq 1$, depth L, sample sizes N, threshold $\{\beta_t\}_{t=1}^L$, reward estimator $\hat{r}$, and prompt $x \in \mathcal{X}$.
2: $\mathcal{B}_0 \leftarrow \{\emptyset\}$ {initial beam}
3: $Q(s_0) \leftarrow 0$ {initial score over trajectory $s_0 = \emptyset$}
4: **for** $t = 1$ **to** L **do**
5: $\quad \mathcal{D}_t \leftarrow \emptyset$ {candidate prefixes at depth t}
6: $\quad$ **for all** $s_{t-1} \in \mathcal{B}_{t-1}$ **do**
7: $\quad\quad$ initialize $N_t(s_{t-1}, a_t) \leftarrow 0$ for all $a_t \in \mathcal{A}$
8: $\quad\quad$ **for** $i = 1$ **to** N **do**
9: $\quad\quad\quad$ sample $a_t^{(i)} \sim \pi_{\text{ref}}(\cdot \mid s_{t-1}, x)$
10: $\quad\quad\quad$ $N_t(s_{t-1}, a_t^{(i)}) \leftarrow N_t(s_{t-1}, a_t^{(i)}) + 1$
11: $\quad\quad$ **end for**
12: $\quad\quad$ **for all** $a_t \in \mathcal{A}$ **with** $N_t(s_{t-1}, a_t) \geq 1$ **do**
13: $\quad\quad\quad$ $\hat{\pi}(a_t \mid s_{t-1}, x) \leftarrow N_t(s_{t-1}, a_t)/N$
14: $\quad\quad\quad$ **if** $\hat{\pi}(a_t \mid s_{t-1}, x) < \beta_t$ **then**
15: $\quad\quad\quad\quad$ **skip** `//CF-Beam only`
16: $\quad\quad\quad$ **else**
17: $\quad\quad\quad\quad$ $s_t' \leftarrow s_{t-1} \oplus a_t$ {append a_t to s_{t-1}}
18: $\quad\quad\quad\quad$ $Q(s_t') \leftarrow Q(s_{t-1}) + \log \hat{\pi}(a_t \mid s_{t-1}, x)$
19: $\quad\quad\quad\quad$ $\mathcal{D}_t \leftarrow \mathcal{D}_t \cup \{s_t'\}$
20: $\quad\quad\quad$ **end if**
21: $\quad\quad$ **end for**
22: $\quad$ **end for**
23: $\quad \mathcal{B}_t \leftarrow$ top-$b\{s_t' \in \mathcal{D}_t$ ranked by $Q(s_t')\}$
24: **end for**
25: $\hat{y} \leftarrow \arg\max_{y \in \mathcal{B}_L} \hat{r}(x, y)$
26: **Output:** $\hat{y}$

In particular, the log-likelihood is used as the scoring metric. For each time step $t \in [L]$, we draw N fresh i.i.d. samples $a_t^{(1)}, \ldots, a_t^{(N)} \sim \pi_{\text{ref}}(\cdot \mid s_{t-1}, x)$, and then estimate the action distribution at step t by the empirical frequencies: $\hat{\pi}(a_t | s_{t-1}, x) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}(a_t^{(i)} = a_t)$. The score of a trajectory, denoted as $Q(s_t)$ is defined as the sum of estimated log-probabilities: $Q(s_t) = \sum_{i=1}^{t} \log \hat{\pi}(a_i | s_{i-1}, x)$. Samples at distinct prefixes use independent randomness. We keep

only the top-b trajectories for the next round, i.e., $\mathcal{B}_t = \text{top-}b\,\{s_t \in \mathcal{D}_t : Q(s_t)\}$. If filtering leaves an empty beam, the algorithm returns any fixed response; ties are resolved by a fixed rule.

In practice, retaining only the top-b trajectories can be fragile in Vanilla-Beam. Since the action space $\mathcal{A}$ is typically enormous, most tokens lie in the low-probability tail and are rarely sampled. Yet occasional tail samples amplified by finite-sample estimation and hard top-b pruning can introduce spurious candidates and thereby *crowd out* the optimal prefix $s^\star$. Since beam search is path-dependent, eliminating $s^\star$ at an early step is irreversible and prevents recovery even if subsequent steps are estimated accurately. The following theorem formalizes this phenomenon by constructing a hard instance where Vanilla-Beam fails (i.e., the correct trajectory is discarded with a constant probability), without a sufficiently large sample size. Detailed proof of constructing the hard instance is deferred to Appendix C.

Theorem 1. *There **exists an** LLM instance $\pi \in \Pi$ satisfying $50 \le C_i^\star(x) := C^\star(x) \ll (|\mathcal{A}|)^{1/4}$ for all $i \in [L]$, such that if we set $N \le (C^\star(x))^2/16$, the probability of eliminating the correct answer $y^\star$ using Vanilla-Beam with $2 \le b \le N/2$ is at least 0.5.*

Theorem 1 shows that without the sample size N to be at least $N > \Omega((C^\star_{\max})^2)$, Vanilla-Beam cannot survive the correct response to the end of the process.

Confidence-Filtered Beam Search (CF-Beam). In order to reduce the sample complexity of Vanilla-Beam, inspired by Di et al. (2025), we introduce a *thresholding* step (Lines 14-15 in Algorithm 1), which filters low-probability samples based on the estimation of their probability of sampling. We call this modified algorithm Confidence-Filtered Beam Search (CF-Beam). To be specific, for each prefix $s_t,\ t \in [L]$, after generating N actions, we keep only those satisfying $\hat{\pi}_t(a_t \mid s_{t-1}, x) \ge \beta_t$, and then select the top-b trajectories from such a filtered action set. We will show in Section 4.2 (Theorem 2 and Corollary 1) that this simple modification yields a sample complexity bound of $\widetilde{O}(C^\star_{\max})$ for fixed horizon, prefix gap, and target accuracy, thereby improving the sample complexity from the quadratic to nearly linear dependence on $C_t^\star(x)$.

4 Regret Analysis

4.1 Technical Assumptions

In this subsection, we introduce the technical assumptions to establish the performance guarantee for the beam search method.

First, while the performance metric is defined with respect to the true reward $r^\star$, the algorithm typically has access only to an approximate reward model $\hat{r}$. We capture the quality of $\hat{r}$ via a distributional estimation error under π_{ref}, which is a commonly adopted assumption in the literature (Huang et al., 2025; Di et al., 2025).

Assumption 1 (Reward estimation error)**.** For each instance $x \in \mathcal{X}$, the reward model satisfies

$$\mathbb{E}_{y\sim\pi_{\text{ref}}(\cdot|x)}\left[(r^\star(x,y) - \hat{r}(x,y))^2\right] \le \epsilon^2_{\text{RM}}(x). \tag{3}$$

Assumption 2 (Globally optimal response)**.** For each instance x, there exists a unique optimal response

$$y^\star(x) = \arg\max_{y\in\mathcal{Y}} r^\star(x,y), \tag{4}$$

with $r^\star(x, y^\star(x)) = 1$. Here, $y^\star(x) = (a_1^\star, \ldots, a_L^\star)$ captures the entire optimal path. Moreover, the reward model error at the optimum is bounded as

$$\left|r^\star(x, y^\star(x)) - \hat{r}(x, y^\star(x))\right| \le \epsilon_{\text{opt}}(x). \tag{5}$$

It is obvious that the reward model error is crucial for the success of inference time algorithms, and building an accurate reward model has been extensively studied in the Reinforcement Learning with Human Feedback (RLHF) framework (Wang et al., 2026; Sun et al., 2025b; Duan et al., 2025; Yu et al., 2025b). Since our work mainly focuses on sampling strategy of the base model, we directly assume that we have access to a reward model with the estimation errors controlled by ϵ^2_{RM} and ϵ_{opt}.

Furthermore, since the reward signal is only available at the *sequence level*, the algorithm can evaluate a completed response y using $\hat{r}(x, y)$, but does not observe intermediate-step rewards for partial

prefixes. In such settings, inference-time decisions must rely on signals intrinsic to the base policy, such as its self-likelihood and uncertainty over candidate continuations. As having been justified by prior analyses (Cordero-Encinar and Duncan, 2025; Feng et al., 2025), we impose the following prefix separability condition.

Assumption 3 (Prefix competitiveness)**.** There exists a common $\kappa \in (0, 1]$ such that, for each depth $t \in [L]$ and any prefix $s = (a_1, \ldots, a_t) \in \mathcal{A}^t$ with $s \neq s_t^\star = (a_1^\star, \ldots, a_t^\star)$, we have

$$\frac{1}{t} \sum_{i=1}^{t} \log \left(\frac{\pi_{\mathrm{ref}}(a_i^\star \mid s_{i-1}^\star, x)}{\pi_{\mathrm{ref}}(a_i \mid s_{i-1}, x)} \right) \geq \kappa, \tag{6}$$

where $s_{i-1} = a_{1:i-1}$ and $s_{i-1}^\star = a_{1:i-1}^\star$.

Any larger positive gap lower bound also implies the condition with $\kappa = 1$, so restricting the chosen lower bound to $(0, 1]$ does not restrict the model class.

Such a gap condition is needed whenever intermediate pruning decisions rely on likelihood-based signals rather than direct reward feedback. Related gap-dependent analyses have also appeared in self-consistency and certification-based decoding settings; see, e.g., Cordero-Encinar and Duncan (2025); Aeeneh et al. (2024). The following proposition shows that, in our beam-search setting, this prefix-competitiveness condition is not merely technical: without it, likelihood-based pruning can irreversibly discard the optimal trajectory even with an arbitrarily large sampling budget.

Proposition 1 (Necessity of Assumption 3)**.** *Without Assumption 3, there exists an LLM instance* $(\pi_{\mathrm{ref}}, r^\star)$ *on which, for every sample size* $N \geq 1$*, beam search with beam width* $b = 1$ *prunes the optimal trajectory* $y^\star$ *from the final beam with probability at least* $3/4$*.*

The proof is deferred to Appendix D, and additional discussion is provided in Appendix B.

4.2 Regret Upper Bound

The regret arises from three factors: (i) the accuracy of the reward model used for the final selection, (ii) the capacity of the base model π_{ref}, and (iii) search failures, i.e., events under which the optimal response is eliminated during inference. The first factor is captured by Assumptions 1 and 2, and the second by the coverage coefficients in Definition 1. To control the third factor, we define good events that separate two failure modes: the optimal action fails to pass confidence filtering, or the optimal prefix passes filtering but is discarded by top-b pruning.

Pre-sampled histograms. Since the algorithm samples only at prefixes in the current beam, the empirical frequency $\hat{\pi}(a_t^\star \mid s_{t-1}^\star, x)$ is undefined once an optimal prefix has been discarded. To define it for every t, we use an equivalent description of the sampling process. Before the algorithm runs, we independently draw, for every prefix s of depth less than L, a histogram of N i.i.d. samples from $\pi_{\mathrm{ref}}(\cdot \mid s, x)$; the histogram of s is revealed only when the algorithm expands s. Since each prefix is expanded at most once, the output of the algorithm has the same distribution as under fresh sampling, and the construction is used only in the analysis. Under this coupling, the empirical frequencies along the optimal path are defined whether or not the path survives, and histograms at distinct prefixes are mutually independent.

Definition 2 (Good events)**.** We define the following events.

(i) Filtering event. The optimal action passes confidence filtering at every depth:

$$\mathcal{E}_\beta := \left\{ \forall t \in [L] : \ \hat{\pi}(a_t^\star \mid s_{t-1}^\star, x) \geq \beta_t \right\}.$$

(ii) Beam-survival event. Let $\mathrm{rank}_t(s)$ denote the rank of $s \in \mathcal{D}_t$ when $\mathcal{D}_t$ is sorted by $Q(\cdot)$ in descending order, with ties broken by the rule of the algorithm. Define

$$\mathcal{E}_b := \left\{ \forall t \in [L] : \ s_t^\star \in \mathcal{D}_t \ \Rightarrow \ \mathrm{rank}_t(s_t^\star) \leq b \right\}.$$

The overall good event is $\mathcal{E} := \mathcal{E}_\beta \cap \mathcal{E}_b$.

By induction over the depths, $\mathcal{E}$ holds if and only if $y^\star \in \mathcal{B}_L$: on $\mathcal{E}_\beta$, every optimal prefix in the beam generates its optimal extension as a candidate, and on $\mathcal{E}_b$, top-b pruning retains this extension. We refer to $\mathbb{P}(\mathcal{E}^c)$ as the *search error*.

Now we present our main result on the regret of the CF-Beam algorithm.

Theorem 2 (Regret of CF-Beam)**.** *Suppose $L \geq 2$, $N \geq \frac{48C^\star_{\max}}{\kappa^2}\max\{2, \log(2C^\star_{\max})\}$, and $\beta_t = (1-1/L)/C^\star_t(x)$ for all $t \in [L]$. Under Assumptions 1, 2, and 3, the regret of CF-Beam in Algorithm 1 satisfies*

$$\mathrm{Reg}(x) \leq \mathbb{P}(\mathcal{E}^c) + \underbrace{\epsilon_{\mathrm{opt}}(x) + 2\sqrt{\overline{C}(x)\epsilon^2_{\mathrm{RM}}(x)}}_{\textit{reward-model error}}, \tag{7}$$

where the search error satisfies

$$\mathbb{P}(\mathcal{E}^c) \leq \underbrace{L\exp\left(-\frac{N}{2C^\star_{\max}L^2}\right)}_{\textit{filtering error}} + \underbrace{\frac{2C^\star_{\max}}{b}\exp\left(-\frac{\kappa^2 N}{48C^\star_{\max}}\right)}_{\textit{pruning error}}. \tag{8}$$

Theorem 2 separates the regret into a search error and a reward-model error. The search error accounts for filtering out an optimal action or pruning its prefix from the beam. Both probabilities decay exponentially with the per-prefix sample size N; their rates depend on the token-level coverage $C^\star_{\max}$, the horizon L for filtering, and the prefix gap κ for pruning. A wider beam reduces the pruning term, since discarding the optimal prefix requires at least b filtered competitors to match or exceed its score. Once the optimal response survives, the remaining regret comes from the final reward-based selection. Its bound involves the sequence-level coefficient $\overline{C}(x)$, which arises when transferring the reward-model error guarantee under π_{ref} to the selected output; this term does not decrease with N or b. The threshold β_t depends on the unknown $C^\star_t(x)$, so this result gives an oracle guarantee rather than a directly implementable rule. For practical use, Appendix H.1 describes an empirical variant of CF-Beam that replaces the unknown optimal-token probability $1/C^\star_t(x)$ with the largest empirical next-token frequency at the current prefix, and sets the threshold to $\beta_t(s) = \gamma \max_{a\in\mathcal{A}} \hat{\pi}(a \mid s, x)$ for a constant $\gamma \in (0, 1)$. This proxy is reasonable when the optimal continuation is locally competitive, in which case the optimal-token probability is of the same order as the largest next-token probability.

Proof sketch. We first bound the probability that the optimal response is lost during search. At each depth, the empirical count of the optimal action is binomial; a Chernoff lower-tail bound and a union bound over depths control the filtering error. The threshold lies a relative $1/L$ below the mean frequency, which yields the L^2 factor in this bound. For pruning, we count the filtered competitors whose empirical scores match or exceed that of the optimal prefix. Truncated binomial moment bounds control their score differences even when low-probability actions happen to pass the filter. Assumption 3 supplies the true likelihood gap, and summing over competitors uses their total probability mass rather than their number. If the optimal prefix passes filtering but is pruned, at least b competitors must match or exceed its score; Markov's inequality then controls the pruning error.

On $\mathcal{E}$, the optimal response reaches the final beam, so the remaining regret comes from reward-based selection. For each response y, the joint probability $\Pr(\hat{y} = y, \mathcal{E})$ is at most $\pi_{\mathrm{ref}}(y \mid x)\prod_{t=1}^{L}\beta_t^{-1} \leq 4\overline{C}(x)\pi_{\mathrm{ref}}(y \mid x)$. This comparison and Cauchy–Schwarz transfer the reward-model error bound under π_{ref} to the selected response, while the pointwise error at $y^\star$ contributes $\epsilon_{\mathrm{opt}}(x)$. We provide the comprehensive proof in Appendix E.

Corollary 1 (Sample complexity)**.** *Under the assumptions and threshold choice of Theorem 2, for $\delta \in (0, 1)$, it suffices to take*

$$N \gtrsim C^\star_{\max}\max\left\{L^2\log\frac{2L}{\delta},\quad \frac{1}{\kappa^2}\left(\log(eC^\star_{\max}) + \left[\log\frac{1}{b\delta}\right]_+\right)\right\} \tag{9}$$

to achieve regret at most $\delta + \epsilon_{\mathrm{opt}}(x) + 2\sqrt{\overline{C}(x)\epsilon^2_{\mathrm{RM}}(x)}$, where $[u]_+ := \max\{u, 0\}$ and the implicit constant is universal.

Corollary 1 shows that the sufficient per-prefix sample size grows linearly in $C^\star_{\max}$ and quadratically in L, up to logarithmic factors, and only logarithmically in $1/\delta$. Since the total query budget is at most LbN, the total budget is $\widetilde{O}(bL^3C^\star_{\max})$ for fixed κ and δ. The beam width enters only through $\log(1/(b\delta))$: a wider beam relaxes the requirement from the pruning error until the filtering error or the baseline condition dominates.

Benefit of confidence filtering. For fixed L, κ, and δ, Corollary 1 shows that $N = \widetilde{O}(C^\star_{\max})$ samples per prefix suffice for the search error to be at most δ. In contrast, Theorem 1 exhibits an instance satisfying Assumption 3 on which Vanilla-Beam with $2 \le b \le N/2$ and $N \le (C^\star)^2/16$ discards $y^\star$ with probability at least $1/2$. Confidence filtering therefore reduces the per-prefix sample requirement from quadratic to nearly linear in the token-level coverage.

4.3 Comparison with Other Test-time Methods

In this subsection, we compare the search-induced component of CF-Beam with other commonly adopted test-time algorithms for which regret guarantees have recently been established.

First, Di et al. (2025) proposed the *Best-of-Majority* algorithm, which first applies majority voting (self-consistency) to discard low-frequency candidates and then selects the final response from the remaining set using an external reward model. They established a regret bound given by $\mathrm{Reg}(x) \lesssim \epsilon_{\mathrm{opt}}(x) + \sqrt{\overline{C}(x)\,\epsilon^2_{\mathrm{RM}}(x)} + \overline{C}(x)\exp\Big(-\widetilde{N}/\overline{C}(x)\Big)$, where $\overline{C}(x) := \prod_{t=1}^{L} C^\star_t(x)$ captures the coverage along the entire trajectory in our path-based notation, and $\widetilde{N}$ refers to the number of complete responses sampled. To make the search-induced term at most δ, the sample size $\widetilde{N}$ needs to satisfy $\widetilde{N} \gtrsim \overline{C}(x)\log\big(\overline{C}(x)/\delta\big)$, which leads to the total computation budget $B = L\widetilde{N}$ on the order of $B \gtrsim L\,\overline{C}(x)\log\big(\overline{C}(x)/\delta\big)$. Whenever $C^\star_{\min} := \min_{1\le t\le L} C^\star_t(x) > 1$, this budget can scale **exponentially** with the horizon length L. This exponential scaling of the coverage-dependent term also appears in other test-time algorithms such as *Best-of-N* Huang et al. (2025), where the regret bound takes the form of $O\left(\overline{C}(x)\cdot\log(1/\varepsilon_{\mathrm{RM}}(x))/\widetilde{N} + \sqrt{\widetilde{N}\cdot\varepsilon^2_{\mathrm{RM}}(x)}\right)$. Similarly, the dependence on $\overline{C}(x)$ can lead to exponentially large sample complexity as L increases.

In contrast, our CF-Beam method controls the rare-failure term with a budget $B = LbN$ that satisfies Corollary 1, which scales only **polynomially** in L (up to the beam-width factor b). Hence, beam-pruning combined with β-filtering effectively replaces the potentially exponential dependence on $\overline{C}(x)$ with a polynomial dependence on L in the search-induced component, yielding substantially reduced computation-cost scaling in long-horizon settings when the reward-model error is controlled.

The above comparison suggests that CF-Beam can be more favorable on hard instances. Compared to sequence-level parallel sampling methods, a key advantage of CF-Beam is that its search-induced guarantee scales with $LC^\star_{\max}$ rather than the product $\overline{C}(x) = \prod_{t=1}^{L} C^\star_t(x)$. When all $C^\star_t(x)$ are close to 1 and L is moderate, sequence-level methods such as Best-of-N or Best-of-Majority may be more sample-efficient. In contrast, when some $C^\star_t(x)$ are large, $\overline{C}(x)$ grows rapidly, and CF-Beam becomes more favorable due to its dependence on $LC^\star_{\max}$ in the search-induced term. This qualitative trend is consistent with the empirical scaling study of Snell et al. (2024).

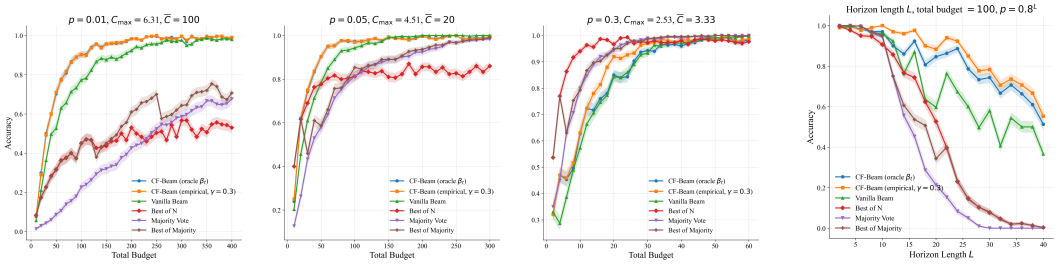
4.4 Regret Lower Bound

In this subsection, we establish the lower bound for the beam search algorithm.

Theorem 3. *For the CF-Beam algorithm, when $b \ge 2$, there exist hard instances with a sufficiently small constant $\kappa' > 0$ and $C^\star_t(x) = |\mathcal{A}|/(1+|\mathcal{A}|\kappa')$ for all $t \in [L]$, such that the regret can be lower bounded as $\Omega\left(\sqrt{\overline{C}(x)\epsilon^2_{RM}(x)}\right)$.*

This lower bound matches the reward model error term in the upper bound in Theorem 2, demonstrating the minimax optimality of the regret with respect to the reward estimation error $\epsilon^2_{\mathrm{RM}}(x)$. To be specific, the lower bound is determined by (i) the difficulty of the problem $\overline{C}$ and (ii) the accuracy of the reward model. These two dependencies are necessary, since outputting the optimal response requires both the model's ability to generate it and the reward model's ability to identify it.

Di et al. (2025) established a general lower bound for the pass@k inference problem with a similar structure. However, their construction cannot be directly applied to our setting, as it does not satisfy Assumption 3. Our lower bound instead constructs a family of instances with an arbitrarily small prefix gap κ, under which it is information-theoretically difficult for any intermediate pruning strategy to eliminate all suboptimal prefixes with high probability before the final step. The incurred regret is



(a) $p = 0.01$ (b) $p = 0.05$ (c) $p = 0.3$

Figure 2: Algorithm performance on different difficulty-level problems.

Figure 3: Algorithm performance vs. horizon length.

then driven by the inability of the imperfect reward model to always select the true optimum among the surviving candidates.

4.5 Extension to Reward Model Free Setting

In the case when a reliable reward model is unavailable (or severely noisy), finding the most probable path becomes a natural metric to identify the optimality of the path. To this end, we can derive a reward-model-free variant of CF-Beam by replacing the final selection rule, namely, instead of selecting the candidate with the largest estimated reward, we define $Q_L(y) := \sum_{t=1}^{L} \log \hat{\pi}(a_t \mid s_{t-1}, x)$ and output $\hat{y} \leftarrow \arg\max_{y \in \mathcal{B}_L} Q_L(y)$, i.e., the most "self-consistent" candidate within the final beam according to the intrinsic score Q_L. This modification yields Algorithm 2, which can be viewed as a reward-model-free variant of CF-Beam in Algorithm 1.

Algorithm 2 Self-consistent CF-Beam

1: **Given:** base policy π_{ref}, beam width $b \geq 1$, depth L, sample sizes N, threshold $\{\beta_t\}_{t=1}^{L}$.
2: **Input:** prompt $x \in \mathcal{X}$.
3: Execute Algorithm 1, Lines 2–24.
4: $\hat{y} \leftarrow \arg\max_{y \in \mathcal{B}_L} Q_L(y)$.
5: **return** $\hat{y}$

The same filtered-score analysis controls this variant, with an additional comparison at the final step: the selected complete response must have the largest empirical likelihood, rather than merely remain in the beam. The proof is deferred to Appendix E.4.

Corollary 2. *Suppose $y^\star$ is an optimal response with $r^\star(x, y^\star) = 1$, Assumption 3 holds, $N \geq \frac{48 C^\star_{\max}}{\kappa^2} \max\{2, \log(2C^\star_{\max})\}$, and $\beta_t = 1/(2C^\star_t(x))$. The regret of Algorithm 2 satisfies*

$$\mathrm{Reg}(x) \leq L \exp\left(-\frac{N}{8C^\star_{\max}}\right) + \frac{2C^\star_{\max}}{b} \exp\left(-\frac{\kappa^2 N}{48 C^\star_{\max}}\right) + \left[C^\star_{\max} \exp\left(-\frac{\kappa^2 N}{48 C^\star_{\max}}\right)\right]^L.$$

In particular, $N \gtrsim (C^\star_{\max}/\kappa^2) \log(eLC^\star_{\max}/\delta)$ suffices to achieve regret at most $\delta \in (0, 1)$. No reward-model accuracy assumption is needed.

The first two terms account for filtering and intermediate beam pruning, and the last controls final selection by the empirical score. Since the guarantee does not involve reward-model error, the threshold multiplier can be fixed at $1/2$ independently of L. This removes the L^2 factor from the filtering contribution to the sufficient per-prefix sample size in Corollary 1. The resulting total query budget is $\widetilde{O}(LbC^\star_{\max}/\kappa^2)$, with only logarithmic dependence on L in the per-prefix sample requirement.

5 Experiments

We validate our theoretical findings by comparing two implementations of CF-Beam, Oracle CF-Beam and Empirical CF-Beam, against four baselines: Vanilla beam search, Best-of-N (Huang et al., 2025), Majority Voting, and Best-of-Majority (Di et al., 2025). Oracle CF-Beam uses the threshold in Theorem 2, while Empirical CF-Beam sets $\beta_t(s) = \gamma \max_{a \in \mathcal{A}} \hat{\pi}(a \mid s, x)$ with $\gamma = 0.3$, without requiring the optimal-token probability. Following the RLVR setting, we evaluate each algorithm on

a synthetic LLM simulator paired with a noisy reward model of error level $\epsilon_{\mathrm{RM}}^2 = 0.01$, and report the accuracy of identifying the optimal response. Baseline descriptions, the experimental protocol, simulator construction, and parameter setups are deferred to Appendix H. We also evaluate Empirical CF-Beam on a pretrained LLM in Appendix H.2.

Figure 2 compares the algorithms under varying total budgets across three difficulty levels, parameterized by the per-rollout success probability $p \in \{0.01, 0.05, 0.3\}$ (larger p = easier). CF-Beam algorithms achieve the largest advantage in the hardest setting ($p = 0.01$, Figure 2a), where sparse coverage and tail fluctuations are most pronounced—precisely the regime that amplifies the failure mode of Theorem 1. As p increases (Figures 2b–2c), the gap narrows and Vanilla-Beam catches up under sufficiently large budgets. This trend mirrors the relationship between $\overline{C}$ and $C^{\star}_{\max}$ shown in each panel: as p grows, $\overline{C}$ approaches $C^{\star}_{\max}$, reducing CF-Beam's edge. These observations align with our theoretical discussion in Section 4.3.

Figure 3 reports accuracy as the horizon L grows, with per-step success probability 0.8. Sequence-level baselines (Best-of-N, Best-of-Majority, Majority Voting) degrade rapidly, while beam-based methods remain robust, and CF-Beam consistently attains the highest accuracy. This is consistent with our theory: CF-Beam's dominant sample-complexity term scales only polynomially in L, whereas sequence-level methods can suffer exponential dependence on L.

6 Related Works

In this section, we focus on related work in theoretical studies of test-time scaling. Additional related work is deferred to Appendix A.

Despite extensive literature on test-time computing (Wang et al., 2022; Kang et al., 2025; Jinnai et al., 2024; Ichihara et al., 2025; Wei et al., 2022; Ghosal et al., 2025; Liu et al., 2025a), especially beam search (Xu et al., 2025; Xie et al., 2023; Zhu et al., 2024), theoretical analyses have only emerged recently. Hu et al. (2024) decomposed the statistical error of Chain-of-Thought (CoT) estimators and showed it vanishes with more demonstrations. Huang et al. (2024) analyzed *Best-of-N* and showed that larger N does not necessarily improve performance. Cordero-Encinar and Duncan (2025) and Aeeneh et al. (2024) established convergence rates for Majority Voting, while Di et al. (2025) proposed *Best-of-Majority*, which selects the top-k response from a set of highly self-consistent candidates with non-degrading sample complexity. Yu et al. (2025c) combined Best-of-N with a rewind-and-repeat strategy (Kim et al., 2024; Beirami et al., 2024) to derive stronger guarantees.

The existing theoretical studies on *beam search* methods remains rather limited. Meister et al. (2020) characterized the implicit objective optimized by beam search, and Fadeeva et al. (2025) analyzed beam search for self-consistency-based uncertainty quantification. To our knowledge, no prior work studies the test-time scaling guarantees of beam search; Our work fills this research gap and provides the first test-time computing guarantee for beam search type algorithms.

7 Conclusion and Limitations

This paper studied test-time guarantees for beam search in LLM reasoning, where model likelihood scores intermediate trajectories and an external reward model is queried only after full responses are generated. We showed that vanilla beam search can be theoretically suboptimal, and proposed *Confidence-Filtered Beam Search* to filter low self-consistency tokens before beam selection. CF-Beam improves the search-induced scaling over sequence-level methods on hard and long-horizon instances. The limitation is that this advantage relies on prefix alignment between likelihood and final reward; otherwise, reward-optimal prefixes may be pruned before final reward evaluation. Future work includes prompt-adaptive choices between beam-based and sequence-level methods, and settings with external rewards or process verifiers available before full responses are generated.

Acknowledgments and Disclosure of Funding

Q. He and Y. Liang were supported in part by the U.S. National Science Foundation under the grants MFAI-2601957 and ECCS-2413528. Chen is supported in part by the Alfred P. Sloan Research Fellowship, the NSF grants IIS-2218773 and CIF2221009, the ONR grants N00014-22-1-2354 and

N00014- 25-1-2344, the Wharton AI & Analytics Initiative's AI Research Fund, and the Amazon Research Award.

References

Aeeneh, S., Zlatanov, N., and Yu, J. (2024). New bounds on the accuracy of majority voting for multiclass classification. *IEEE Transactions on Neural Networks and Learning Systems*, 36(4):6014–6028.

Aggarwal, P., Madaan, A., Yang, Y., et al. (2023). Let's sample step by step: Adaptive-consistency for efficient reasoning and coding with llms. *arXiv preprint arXiv:2305.11860*.

Beirami, A., Agarwal, A., Berant, J., D'Amour, A., Eisenstein, J., Nagpal, C., and Suresh, A. T. (2024). Theoretical guarantees on the best-of-n alignment policy. *arXiv preprint arXiv:2401.01879*.

Chen, F., Jia, Z., Rakhlin, A., and Xie, T. (2025). Outcome-based online reinforcement learning: Algorithms and fundamental limits. *arXiv preprint arXiv:2505.20268*.

Chen, X., Aksitov, R., Alon, U., Ren, J., Xiao, K., Yin, P., Prakash, S., Sutton, C., Wang, X., and Zhou, D. (2023). Universal self-consistency for large language model generation. *arXiv preprint arXiv:2311.17311*.

Cordero-Encinar, P. and Duncan, A. B. (2025). Certified self-consistency: Statistical guarantees and test-time training for reliable reasoning in llms. *arXiv preprint arXiv:2510.17472*.

Di, Q., Ji, K., Li, X., Zhao, H., and Gu, Q. (2025). Best-of-majority: Minimax-optimal strategy for pass@ k inference scaling. *arXiv preprint arXiv:2510.03199*.

Duan, K., Liu, Z., Mao, X., Pang, T., Chen, C., Chen, Q., Shieh, M. Q., and Dou, L. (2025). Efficient process reward model training via active learning. *arXiv preprint arXiv:2504.10559*.

Fadeeva, E., Goloburda, M., Rubashevskii, A., Vashurin, R., Shelmanov, A., Nakov, P., Sachan, M., and Panov, M. (2025). Don't throw away your beams: Improving consistency-based uncertainties in llms via beam search. *arXiv preprint arXiv:2512.09538*.

Feng, A., Alonso, M., and Odonnat, A. (2025). Optimal self-consistency for efficient reasoning with large language models. *arXiv preprint arXiv:2511.12309*.

Feng, X., Wan, Z., Wen, M., McAleer, S. M., Wen, Y., Zhang, W., and Wang, J. (2023). Alphazero-like tree-search can guide large language model decoding and training. *arXiv preprint arXiv:2309.17179*.

Ghosal, S. S., Chakraborty, S., Reddy, A., Lu, Y., Wang, M., Manocha, D., Huang, F., Ghavamzadeh, M., and Bedi, A. S. (2025). Does thinking more always help? understanding test-time scaling in reasoning models. *arXiv preprint arXiv:2506.04210*.

He, S., Xia, T., Zhou, X., and Wei, H. (2025). Response-level rewards are all you need for online reinforcement learning in llms: A mathematical perspective. *arXiv preprint arXiv:2506.02553*.

Hu, X., Zhang, F., Chen, S., and Yang, Z. (2024). Unveiling the statistical foundations of chain-of-thought prompting methods. *arXiv preprint arXiv:2408.14511*.

Huang, A., Block, A., Foster, D. J., Rohatgi, D., Zhang, C., Simchowitz, M., Ash, J. T., and Krishnamurthy, A. (2024). Self-improvement in language models: The sharpening mechanism. *arXiv preprint arXiv:2412.01951*.

Huang, A., Block, A., Liu, Q., Jiang, N., Krishnamurthy, A., and Foster, D. J. (2025). Is best-of-n the best of them? coverage, scaling, and optimality in inference-time alignment. *arXiv preprint arXiv:2503.21878*.

Ichihara, Y., Jinnai, Y., Morimura, T., Ariu, K., Abe, K., Sakamoto, M., and Uchibe, E. (2025). Evaluation of best-of-n sampling strategies for language model alignment. *arXiv preprint arXiv:2502.12668*.

Jinnai, Y., Morimura, T., Ariu, K., and Abe, K. (2024). Regularized best-of-n sampling to mitigate reward hacking for language model alignment. In *ICML 2024 Workshop on Models of Human Feedback for AI Alignment*.

Joag-Dev, K. and Proschan, F. (1983). Negative association of random variables with applications. *The Annals of Statistics*, pages 286–295.

Kang, Z., Zhao, X., and Song, D. (2025). Scalable best-of-n selection for large language models via self-certainty. *arXiv preprint arXiv:2502.18581*.

Kim, M., Thonet, T., Rozen, J., Lee, H., Jung, K., and Dymetman, M. (2024). Guaranteed generation from large language models. *arXiv preprint arXiv:2410.06716*.

Lee, H., Oh, S., Kim, J., Shin, J., and Tack, J. (2025). Revise: Learning to refine at test-time via intrinsic self-verification. *arXiv preprint arXiv:2502.14565*.

Lightman, H., Kosaraju, V., Burda, Y., Edwards, H., Baker, B., Lee, T., Leike, J., Schulman, J., Sutskever, I., and Cobbe, K. (2023). Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Liu, X., Liang, T., He, Z., Xu, J., Wang, W., He, P., Tu, Z., Mi, H., and Yu, D. (2025a). Trust, but verify: A self-verification approach to reinforcement learning with verifiable rewards. *arXiv preprint arXiv:2505.13445*.

Liu, Z., Chen, Y., Shoeybi, M., Catanzaro, B., and Ping, W. (2025b). Acemath: Advancing frontier math reasoning with post-training and reward modeling. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3993–4015.

Meister, C., Cotterell, R., and Vieira, T. (2020). If beam search is the answer, what was the question? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2173–2185. Association for Computational Linguistics.

Pinelis, I. (2021). Best lower bound on the probability of a binomial exceeding its expectation. *Statistics & Probability Letters*, 179:109224.

Setlur, A., Nagpal, C., Fisch, A., Geng, X., Eisenstein, J., Agarwal, R., Agarwal, A., Berant, J., and Kumar, A. (2024). Rewarding progress: Scaling automated process verifiers for llm reasoning. *arXiv preprint arXiv:2410.08146*.

Snell, C., Lee, J., Xu, K., and Kumar, A. (2024). Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*.

Stahlberg, F. and Byrne, B. (2019). On nmt search errors and model errors: Cat got your tongue? In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 3356–3362.

Sun, Y., Liang, Y., Zhang, Z., and Teng, J. (2025a). Theoretical modeling of llm self-improvement training dynamics through solver-verifier gap. *arXiv preprint arXiv:2507.00075*.

Sun, Z., Guo, Y., Lin, Y., Chen, X., Qi, Q., Tang, X., Wen, J.-R., et al. (2025b). Uncertainty and influence aware reward model refinement for reinforcement learning from human feedback. In *The Thirteenth International Conference on Learning Representations*.

Team, G., Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., et al. (2023). Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., and Zhou, D. (2022). Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Wang, Z., Wang, R., Wu, Y., Yu, Y., Zhang, P., Sun, S., Yang, Y., and Li, Y. (2026). Reward modeling from natural language human feedback. *arXiv preprint arXiv:2601.07349*.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Wu, Y., Sun, Z., Li, S., Welleck, S., and Yang, Y. (2024). Inference scaling laws: An empirical analysis of compute-optimal inference for problem-solving with language models. *arXiv preprint arXiv:2408.00724*.

Xie, Y., Kawaguchi, K., Zhao, Y., Zhao, J. X., Kan, M.-Y., He, J., and Xie, M. (2023). Self-evaluation guided beam search for reasoning. *Advances in Neural Information Processing Systems*, 36:41618–41650.

Xu, Y., Ren, S., and Zhang, J. (2025). Collaborative beam search: Enhancing llm reasoning via collective consensus. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 11409–11421.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. (2025). Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., and Narasimhan, K. (2023). Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822.

Yu, F., Li, Y., and Wang, B. (2025a). Scaling flaws of verifier-guided search in mathematical reasoning. *arXiv preprint arXiv:2502.00271*.

Yu, Y., Chen, Z., Zhang, A., Tan, L., Zhu, C., Pang, R. Y., Qian, Y., Wang, X., Gururangan, S., Zhang, C., et al. (2025b). Self-generated critiques boost reward modeling for language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11499–11514.

Yu, Y., Di, Q., Gu, Q., and Zhou, D. (2025c). On the limits of test-time compute: Sequential reward filtering for better inference. *arXiv preprint arXiv:2512.04558*.

Zaharia, M., Khattab, O., Chen, L., Davis, J. Q., Miller, H., Potts, C., Zou, J., Carbin, M., Frankle, J., Rao, N., and Ghodsi, A. (2024). The shift from models to compound ai systems. `https://bair.berkeley.edu/blog/2024/02/18/compound-ai-systems/`.

Zhao, X., Kang, Z., Feng, A., Levine, S., and Song, D. (2025). Learning to reason without external rewards. *arXiv preprint arXiv:2505.19590*.

Zhu, T., Zhang, K., Xie, J., and Su, Y. (2024). Deductive beam search: Decoding deducible rationale for chain-of-thought reasoning. *arXiv preprint arXiv:2401.17686*.

Table 1: Summary of Notations Used in the Paper

Notation	Description
$\pi_{\text{ref}}(\cdot\|x)$	a fixed LLM model that maps $\mathcal{X} \to \Delta(\mathcal{A}^L)$
L	Response length (horizon) of $y = (a_1, \ldots, a_L)$.
a	An action/token from the vocabulary $\mathcal{A}$.
s_t	Prefix (trajectory) up to step t: $s_t = (a_1, \ldots, a_t)$.
b	Beam width, i.e., the number of trajectories retained at each step.
B	Test-time compute budget, i.e., the total number of samples drawn from π_{ref}.
N	Number of samples drawn per prefix $s_t \in \mathcal{B}_t$ at each step.
$\mathcal{D}_t$	Candidate set after confidence filtering at step t.
$\mathcal{B}_t$	Beam set at step t, consisting of at most b selected prefixes (i.e., $\|\mathcal{B}_t\| \le b$).
$Q(s_t)$	Estimated score of prefix s_t, e.g., $Q(s_t) = \sum_{i=1}^{t} \log \hat{\pi}(a_i \mid s_{i-1}, x)$.
κ	Prefix gap parameter between optimal and suboptimal trajectories; see Assumption 3.
ϵ_{RM}^2	Reward-model estimation error; see Assumption 1.
ϵ_{opt}	Reward-model error on the optimal response; see Assumption 2.
$C_t^\star(x)$, $C_{\max}^\star(x)$, $\overline{C}(x)$	Coverage coefficients; see Definition 1.
β_t	Confidence-filtering threshold at step t.

Notations. We use $\text{Bin}(N, p)$ to denote the binomial distribution with parameters $N \in \mathbb{N}^+$ and $p \in [0, 1]$.

A Related Works

Test-time computing. Beyond post-training scaling laws, recent work has increasingly focused on scaling computation at inference time to improve the quality of a fixed base model. Existing work mainly falls into two approaches: parallel and sequential test-time computation. For parallel methods, the compute budget is used to generate multiple candidate trajectories simultaneously, increasing the probability of finding a high-quality response. Existing such type of methods include Best-of-N (BoN) (Huang et al., 2025; Jinnai et al., 2024; Ichihara et al., 2025), majority voting (Wang et al., 2022; Kang et al., 2025), and beam search (Xie et al., 2023; Yu et al., 2025a). In sequential test-time computation, computation is allocated adaptively through iterative refinement or search, using methods like Chain-of-Thought (Wei et al., 2022; Ghosal et al., 2025; Hu et al., 2024) and self-verification (Liu et al., 2025a; Lee et al., 2025). Some recent studies aimed to bridge the gap between existing test-time computation methods. Di et al. (2025) proposes Best-of-Majority, which used majority voting as a filtering mechanism and an external reward model to select the final answer. Yu et al. (2025c) included a sequential technique for BoN sampling. The beam search method we study here mainly falls into the first category, since candidates are generated in parallel rather than sequentially.

Beam search. Beam search algorithms have been widely used as a decoding strategy for large language models. Xu et al. (2025) and Xie et al. (2023) mainly used self-consistency as the evaluation metric for top-b selection; Zhu et al. (2024) introduced beam search to increase the successful rate of chain of thought; Yu et al. (2025a) empirically compared beam search algorithms with the methods like BoN and majority voting, and pointed out the weakness and strengths of beam search-based algorithms. Despite the empirical success of beam search and its many variants, there has been rather limited theoretical studies of beam search algorithms. Meister et al. (2020) studied what implicit objective function beam search optimizes. A recent work Fadeeva et al. (2025) studied beam search for consistency-based uncertainty quantification and provided a theoretical comparison between beam-based candidate generation and multinomial sampling estimators. To our best knowledge, there has not been any work theoretically analyzing the test-time scaling performance of beam search. Our work provides the first test-time computing guarantee for beam-search decoding algorithms in modern LLM settings.

Self-consistency. Self-consistency improves reasoning by sampling multiple independent solution paths and selecting the answer that appears most consistently across them. Existing works have demonstrated the effectiveness of self-consistency, when it is used in self-evaluated post-training (Fadeeva et al., 2025; Liu et al., 2025a; Zhao et al., 2025; Sun et al., 2025a) and test-time computing (Aggarwal et al., 2023; Chen et al., 2023). Cordero-Encinar and Duncan (2025); Feng et al. (2025) and Aeeneh et al. (2024) provided theoretical analysis of the convergence rate of self-consistency methods. Our CF-Beam is related to self-consistency methods in that it uses a self-consistency score to prune beams at intermediate steps.

B Discussion of Assumptions and Computational Costs

We clarify how response uniqueness and prefix competitiveness enter our guarantees, and distinguish the sampling budget from the cost of maintaining and processing candidate prefixes.

B.1 Uniqueness of the Optimal Response

Assumption 2 makes the optimal response unique to simplify the path-based presentation. The analysis can accommodate several reward-optimal responses by tracking one that satisfies the required prefix and reward-model error conditions. As seen in the following remark.

Remark 1 (Multiple reasoning trajectories leading to the same answer)**.** *Multiple reasoning trajectories yielding the same correct answer are a special case of nonunique reward-optimal responses. Let $\mathcal{Y}^\star(x) := \{y \in \mathcal{A}^L : r^\star(x, y) = 1\}$ be nonempty, and fix an optimal witness $y^\dagger = (a_1^\dagger, \ldots, a_L^\dagger) \in \mathcal{Y}^\star(x)$ with prefixes $s_t^\dagger := a_{1:t}^\dagger$. Suppose Assumption 3 holds with $s_t^\star$ replaced by $s_t^\dagger$, and $|r^\star(x, y^\dagger) - \hat{r}(x, y^\dagger)| \le \epsilon_{\mathrm{opt}}(x)$. The prefix gap must hold against every distinct competing prefix, including prefixes of other correct trajectories.*

Define $C_t^\dagger(x) := 1/\pi_{\mathrm{ref}}(a_t^\dagger \mid s_{t-1}^\dagger, x)$, $C_{\max}^\dagger(x) := \max_t C_t^\dagger(x)$, and $\overline{C}^\dagger(x) := \prod_{t=1}^L C_t^\dagger(x)$. With oracle thresholds $\beta_t = (1 - 1/L)/C_t^\dagger(x)$, the guarantees in Theorem 2 and Corollary 1 extend under Assumption 1 and the same sample-size conditions, with the optimal path and its coverage coefficients replaced by those of the witness. The filtering and ranking arguments track this path, and, on its survival event, the selected response $\hat{y}$ satisfies $\hat{r}(x, \hat{y}) \ge \hat{r}(x, y^\dagger)$, so the reward-error argument applies with $y^\dagger$ as an optimal comparator. This extension does not exploit the aggregate probability mass or shared-prefix structure of all correct trajectories; sharper bounds that do so remain an interesting direction.

B.2 Prefix Competitiveness and Its Relaxation

Assumption 3 formalizes alignment between the likelihood of an optimal prefix and the final outcome reward. A correct trajectory can be difficult to sample in full because of limited token-level coverage or a long horizon, while its prefixes still have a likelihood advantage over competing prefixes. This is the regime in which likelihood-based pruning can exploit token-level coverage without sampling the entire trajectory independently.

When the base policy instead assigns higher likelihood to suboptimal prefixes, intermediate pruning can irreversibly discard the optimal trajectory. Proposition 1 exhibits such an instance with a constant failure probability for every sample size. Thus, increasing the sampling budget alone cannot compensate for arbitrary misalignment between policy likelihood and outcome reward. Sequence-level methods such as Best-of-N or majority voting may be more competitive in these regimes because they postpone selection until complete responses are available (Huang et al., 2024); the relative benefit of beam search depends on the instance.

Remark 2 (Relaxing prefix competitiveness)**.** *For reward-based CF-Beam, the gap condition can allow a limited number of exceptions. Suppose that, at each depth, at most $k - 1$ distinct competing prefixes violate the gap inequality relative to the optimal prefix, where $1 \le k \le b$, and every other competing prefix satisfies the same lower bound κ. If the optimal prefix passes filtering but is pruned, at least $b - k + 1$ competitors satisfying the gap inequality must match or exceed its empirical score. Restricting the probability-mass sum in Lemma 5 to these competitors gives the same ranking bound with b replaced by $b - k + 1$. The filtering and selected-output bounds remain unchanged, so Theorem 2 extends with this replacement in the ranking term. We use the top-1 condition in the*

main text for simplicity. The reward-free final selection in Corollary 2 still requires the top-1 gap: a higher-likelihood suboptimal response could otherwise be selected even with exact scores.

B.3 Sampling Budget and Candidate Processing

Remark 3 (Beam expansion and query cost)**.** *At each depth, Algorithm 1 samples N next tokens for each active prefix, applies confidence filtering, and then retains at most b candidate prefixes. Consequently, $|\mathcal{B}_t| \le b$ and the total query budget satisfies $B(x) \le N + (L-1)bN$, since the initial beam contains only the empty prefix. Before top-b pruning, the candidate pool contains at most $N|\mathcal{B}_{t-1}| \le bN$ children. The confidence filter can only reduce this pool and introduces no additional branching. Thus, for fixed N and b, the total number of sampling queries and processed candidates grows linearly with L, although the full output space has size $|\mathcal{A}|^L$ (Stahlberg and Byrne, 2019).*

These bounds count token-level policy draws, as defined in the main text. They do not directly measure neural-network forward passes, wall-clock latency, or KV-cache memory: several draws may reuse a computed next-token distribution, and implementation costs depend on batching and cache management. Comparisons with tree-search methods such as Tree-of-Thoughts or MCTS-based decoding (Yao et al., 2023; Feng et al., 2023; Wu et al., 2024) therefore require specifying both the search configuration and the computational cost measure.

C Proof of Theorem 1

Proof of Theorem 1. We consider the following LLM instance, where $50 \le C_i^\star(x) := C^\star(x) \ll (|\mathcal{A}|)^{1/4}$. For time step $t = 1$, there exists a unique optimal action $a_1^\star$ with

$$\pi_{\text{ref}}(a_1^\star \mid x) = \frac{1}{C^\star(x)},$$

and all other actions $a' \in \mathcal{A}\backslash\{a_1^\star\}$ have equal probability

$$\pi_{\text{ref}}(a'|x) = \frac{1 - 1/C^\star(x)}{|\mathcal{A}| - 1}.$$

For time step $t = 2$, conditioned on the optimal prefix $a_1^\star$, the next-step distribution satisfies

$$\pi_{\text{ref}}(a_2^\star \mid a_1^\star, x) = \frac{1}{C^\star(x)}, \quad \pi_{\text{ref}}(a_2' \mid a_1^\star, x) = \frac{1 - 1/C^\star(x)}{|\mathcal{A}| - 1},$$

where $a_2' \in \mathcal{A}\backslash\{a_2^\star\}$. Otherwise, if conditioned on any suboptimal prefix $a_1' \ne a_1^\star$, there exist two actions $a_{2,1}^\star$ and $a_{2,2}^\star$, such that

$$\pi_{\text{ref}}(a_{2,1}^\star \mid a_1', x) = \pi_{\text{ref}}(a_{2,2}^\star \mid a_1', x) = \frac{1}{2}, \quad \pi_{\text{ref}}(a_2' \mid a_1', x) = 0,$$

where $a_2' \in \mathcal{A}\backslash\{a_{2,1}^\star, a_{2,2}^\star\}$. Notice that this constructed π_{ref} does not violate our Assumption 3, since $\pi_{\text{ref}}(s_2^\star) = 1/(C^\star(x))^2 \ge 1/(2(|\mathcal{A}| - 1)) > \pi_{\text{ref}}(s_2')$ for any $s_2' \ne s_2^\star$. For time step $t > 2$, we let $\pi_{\text{ref}}(\cdot|s_{t-1}, x)$ be any distribution such that Assumption 3 holds.

Case 1: $N \le 34$**.** Under this case, we show that the probability that the optimal prefix will not even be sampled is at least $1/2$:

$$\begin{aligned}\Pr(y^\star \text{ is eliminated}) &\ge \Pr(y^\star \text{ is eliminated in the first round}) \\ &\ge \left(1 - \frac{1}{C^\star(x)}\right)^N \ge \left(\frac{49}{50}\right)^{34} > 0.5.\end{aligned}$$

Case 2: $N \ge 35$**.** Before proceeding, we define the following events. Let E_1 denote the event that the optimal trajectory $s^\star$ is eliminated in the first round, and let E_2 denote the event that $s^\star$ is eliminated in the second round. Let $\mathcal{B}_1$ be the top-b beam (candidate set) retained after the first round, and define $B_\star := \{ |\mathcal{B}_1| = b\}$, which implies the event that we sample at least b distinct trajectories from x.

By the definition of our events, we have

$$\begin{aligned}
\Pr(s^\star \text{ is eliminated}) &= \Pr(E_1) + \Pr(E_1^c)\Pr(E_2|E_1^c) \\
&\geq \Pr(E_1) + \Pr(E_1^c)\Pr((E_2 \cap B_\star)|E_1^c) \\
&= \Pr(E_1) + \Pr(E_1^c)\Pr(B_\star|E_1^c)\Pr(E_2|(B_\star \cap E_1^c)) \\
&\geq \Pr(E_1)\Pr(B_\star|E_1^c)\Pr(E_2|(B_\star \cap E_1^c)) \\
&\quad + (1 - \Pr(E_1))\Pr(B_\star|E_1^c)\Pr(E_2|(B_\star \cap E_1^c)) \\
&= \Pr(B_\star|E_1^c)\Pr(E_2|(B_\star \cap E_1^c)).
\end{aligned}$$

We next bound each of the term correspondingly.

2.1 Bounding $\Pr(B_\star \mid E_1^c)$. Let

$$K := \#\{\text{samples equal } a_1^\star\} \sim \mathrm{Bin}(N, p), \qquad p := \pi_{\text{ref}}(a_1^\star \mid x) = 1/C^\star.$$

We first justify that suboptimal actions are (essentially) all distinct when the action space is large. Conditioned on K, there are $N - K$ suboptimal draws. Since by our construction, the suboptimal mass is sufficiently spread out (e.g., $\max_{a \neq a_1^\star} \pi_{\text{ref}}(a \mid x) \leq 1/(|\mathcal{A}| - 1)$), then by a union bound,

$$\Pr(\exists \text{ a repeated suboptimal action} \mid K) \;\leq\; \binom{N-K}{2} \max_{a \neq a_1^\star} \pi_{\text{ref}}(a \mid x) \;\leq\; \frac{N(N-1)}{2(|\mathcal{A}| - 1)}.$$

Hence, under the regime $|\mathcal{A}| \gg N^2$, the above collision probability is negligible. In the sequel, we work on this high-probability "no-collision" event, so that the $N - K$ suboptimal samples correspond to $N - K$ *distinct* suboptimal actions.

Step 1: Relating E_1^c and K. Under the no-collision condition, every suboptimal action appears at most once, and hence has empirical frequency $1/N$. If $K \geq 2$, then $a_1^\star$ appears at least twice and thus has empirical frequency at least $2/N$, strictly larger than any suboptimal action. Therefore, $a_1^\star$ must be retained in the top-b set after the first round, i.e.,

$$\{K \geq 2\} \subseteq E_1^c.$$

Step 2: Characterizing $B_\star^c$. Recall $B_\star$ is the event that the first-round beam $\mathcal{B}_1$ is fully populated by b *distinct* candidates. On the no-collision event, there are exactly $N - K$ distinct suboptimal candidates available. Since one slot is occupied by the optimal prefix (on E_1^c), the beam fails to be fully populated only if there are fewer than $b - 1$ distinct suboptimal candidates, namely,

$$B_\star^c \;\subseteq\; \{\, N - K < b - 1 \,\}.$$

Consequently,

$$\Pr(B_\star^c \mid E_1^c) = \frac{\Pr(B_\star^c \cap E_1^c)}{\Pr(E_1^c)} \;\leq\; \frac{\Pr(N - K < b - 1)}{\Pr(E_1^c)} \;\leq\; \frac{\Pr(N - K < b - 1)}{\Pr(K \geq 2)},$$

where the last inequality follows because $\{K \geq 2\} \subseteq E_1^c$.

Step 3: Bounding the ratio. If $b \leq N/2$, then $N - K < b - 1$ implies $K \geq N - b + 2 \geq \lceil N/2 \rceil$. Hence

$$\Pr(B_\star^c \mid E_1^c) \leq \frac{\Pr(K \geq \lceil N/2 \rceil)}{\Pr(K \geq 2)}.$$

For the numerator, since $p < 1$,

$$\Pr(K \geq \lceil N/2 \rceil) = \sum_{k=\lceil N/2 \rceil}^{N} \binom{N}{k} p^k (1-p)^{N-k} \leq \sum_{k=\lceil N/2 \rceil}^{N} \binom{N}{k} p^k \leq p^{N/2} \sum_{k=0}^{N} \binom{N}{k} = 2^N p^{N/2}.$$

For the denominator, we lower bound $\Pr(K \geq 2)$ by a single term:

$$\Pr(K \geq 2) \geq \Pr(K = 2) = \binom{N}{2} p^2 (1-p)^{N-2}.$$

Combining the above together yields

$$\Pr(B_\star^c \mid E_1^c) \le \frac{2^N p^{N/2}}{\binom{N}{2} p^2 (1-p)^{N-2}} = \frac{2^N}{\binom{N}{2}} \cdot \frac{p^{\frac{N}{2}-2}}{(1-p)^{N-2}} =: f(N,p).$$

In particular, for any fixed $p \le 1/50$, $f(N,p)$ is monotonically decreasing with N. For any fixed N, $f(N,p)$ is monotonically increasing with p. Then we have

$$f(N,p) \le f(35,p) \le f(35,1/50) \le \frac{2^{35}}{\binom{35}{2}} \cdot \frac{(1/50)^{\frac{35}{2}-2}}{(49/50)^{35-2}} \le 5.21 \times 10^{-19}.$$

As a result

$$\Pr(B_\star^c \mid E_1^c) \le f(35) \le 5.21 \times 10^{-19}.$$

Therefore,

$$\Pr(B_\star \mid E_1^c) = 1 - \Pr(B_\star^c \mid E_1^c) \ge 1 - 5.21 \times 10^{-19} \approx 1.$$

Interpretation. Under a large action space, suboptimal samples are almost surely all distinct. So the only way for the top-b beam $\mathcal{B}_1$ to *not* be fully populated by b distinct candidates is that the optimal action $a_1^\star$ is sampled *too many times* (so that $N - K$ is too small to provide $b-1$ distinct suboptimal candidates). Since $K \sim \mathrm{Bin}(N,p)$ with small $p = 1/C^\star$, such a large-deviation event $K \ge N/2$ has exponentially small probability, leading to $\Pr(B_\star \mid E_1^c) \approx 1$, which implies that we sample at least b distinct trajectories from x almost surely.

2.2 Bounding $\Pr(E_2 \mid B^\star \cap E_1^c)$. For the analysis in this part, we introduce an auxiliary path-likelihood quantity

$$\widetilde{Q}(s_t) \;:=\; \prod_{i=1}^{t} \hat{\pi}(a_i \mid s_{i-1}, x),$$

which differs from the additive score $Q(\cdot)$ used by the algorithm in Section 3 and is used solely within this proof. We further define the good event $\mathcal{E}$ as

$$\mathcal{E} := \Big\{ \widetilde{Q}(s_2^\star) < \frac{1}{4N} \Big\} \cap \Big\{ \exists \text{ at least } b \text{ suboptimal trajectories } s_2' \in \mathcal{S}_2 \text{ such that } \widetilde{Q}(s_2') \ge \frac{1}{4N} \Big\},$$

where $\mathcal{S}_2$ is the set of all trajectories generated by $\mathcal{B}_1$.

We first control the score of the optimal trajectory under the same conditioning used in this subsection. Let

$$K_1 := N\hat{\pi}(a_1^\star \mid x), \qquad K_2 := N\hat{\pi}(a_2^\star \mid a_1^\star, x).$$

The event $B^\star \cap E_1^c$ is determined by the first-round samples, whereas K_2 is generated from the second-round samples at the optimal prefix. Hence, conditional on K_1, the random variable K_2 remains independent of $B^\star \cap E_1^c$ and satisfies $K_2 \sim \mathrm{Bin}(N, 1/C^\star(x))$. Moreover,

$$\widetilde{Q}(s_2^\star) = \hat{\pi}(a_1^\star \mid x)\hat{\pi}(a_2^\star \mid a_1^\star, x) = \frac{K_1 K_2}{N^2}.$$

Therefore, by Markov's inequality,

$$\Pr\left(\widetilde{Q}(s_2^\star) \ge \frac{1}{4N} \,\middle|\, B^\star \cap E_1^c \right) = \Pr\left(K_1 K_2 \ge \frac{N}{4} \,\middle|\, B^\star \cap E_1^c \right) \le \frac{\mathbb{E}[K_1 K_2 \mid B^\star \cap E_1^c]}{N/4}.$$

It remains to control the conditional expectation in the numerator. Since K_2 is independent of the first-round samples and $\mathbb{E}[K_2] = N/C^\star(x)$, we have

$$\mathbb{E}[K_1 K_2 \mid B^\star \cap E_1^c] = \mathbb{E}[K_1 \mid B^\star \cap E_1^c] \, \frac{N}{C^\star(x)}.$$

On the no-collision event considered in Section 2.1, the event E_1^c can only add outcomes with $K_1 = 1$ beyond the event $\{K_1 \ge 2\}$, while the event $B^\star$ only removes outcomes with too many repeated samples of the optimal action. Hence

$$\mathbb{E}[K_1 \mid B^\star \cap E_1^c] \le \mathbb{E}[K_1 \mid K_1 \ge 2]\,.$$

For a binomial random variable, conditioning on at least two successes is stochastically dominated by adding two deterministic successes to an independent binomial count over the remaining trials. Thus

$$\mathbb{E}[K_1 \mid K_1 \geq 2] \leq 2 + \frac{N}{C^\star(x)} \leq 2 + \frac{C^\star(x)}{16} \leq \frac{C^\star(x)}{9},$$

where the second inequality uses $N \leq (C^\star(x))^2/16$, and the last one uses $C^\star(x) \geq 50$. Therefore,

$$\mathbb{E}[K_1 K_2 \mid B^\star \cap E_1^c] \leq \frac{N}{9},$$

and consequently

$$\Pr\left(\widetilde{Q}(s_2^\star) < \frac{1}{4N} \,\middle|\, B^\star \cap E_1^c\right) \geq 1 - \frac{4}{9} = \frac{5}{9}. \tag{10}$$

Next, we control the second event in $\mathcal{E}$. For any suboptimal prefix $a \in \mathcal{B}_1$, conditioned on $B^\star \cap E_1^c$, its second-step distribution has two possible offspring, $s'_{2,1} = (a, a^\star_{2,1})$ and $s'_{2,2} = (a, a^\star_{2,2})$, satisfying

$$\pi_{\mathrm{ref}}(a^\star_{2,1} \mid a, x) = \pi_{\mathrm{ref}}(a^\star_{2,2} \mid a, x) = \frac{1}{2}.$$

We can bound the probability that both generated trajectories have sufficiently large likelihood as follows:

$$\begin{aligned}
&\Pr\left(\widetilde{Q}(s'_{2,1}) > \frac{1}{4N}, \widetilde{Q}(s'_{2,2}) > \frac{1}{4N} \,\middle|\, B^\star \cap E_1^c\right) \\
&= \Pr\left(\hat{\pi}(a^\star_{2,1} \mid a, x)\hat{\pi}(a \mid x) > \frac{1}{4N}, \hat{\pi}(a^\star_{2,2} \mid a, x)\hat{\pi}(a \mid x) > \frac{1}{4N} \,\middle|\, B^\star \cap E_1^c\right) \\
&\geq \Pr\left(\hat{\pi}(a^\star_{2,1} \mid a, x) > \frac{1}{4}, \hat{\pi}(a^\star_{2,2} \mid a, x) > \frac{1}{4}\right) \\
&= \Pr\left(\frac{1}{4} < \hat{\pi}(a^\star_{2,1} \mid a, x) < \frac{3}{4}\right) \\
&= 1 - \Pr\left(\left|\hat{\pi}(a^\star_{2,1} \mid a, x) - \frac{1}{2}\right| \geq \frac{1}{4}\right) \\
&\geq 1 - 2\exp\left(-\frac{N}{8}\right),
\end{aligned}$$

where the first inequality holds because every selected first-step prefix has empirical probability at least $1/N$, i.e., $\hat{\pi}(a \mid x) \geq 1/N$; the second equality follows from $\hat{\pi}(a^\star_{2,1} \mid a, x) + \hat{\pi}(a^\star_{2,2} \mid a, x) = 1$; and the last inequality follows from Hoeffding's inequality applied to $N\hat{\pi}(a^\star_{2,1} \mid a, x) \sim \mathrm{Bin}(N, 1/2)$.

Then, at the second step, for each such suboptimal prefix $s_1 \in \mathcal{B}_1$, let $\mathsf{Succ}(s_1)$ denote the event that its two offspring trajectories $s'_{2,1}$ and $s'_{2,2}$ satisfy $\widetilde{Q}(s'_{2,1}) \geq 1/(4N),\ \widetilde{Q}(s'_{2,2}) \geq 1/(4N)$. By construction,

$$\Pr(\mathsf{Succ}(s_1) \mid B^\star \cap E_1^c) \geq 1 - 2\exp(-N/8),$$

and these events are independent across different suboptimal prefixes. Hence the number of successful suboptimal prefixes conditioned on $B^\star \cap E_1^c$ stochastically dominates

$$Z \sim \mathrm{Bin}\left(b - 1,\ 1 - 2\exp\left(-\frac{N}{8}\right)\right).$$

In particular, there exists b suboptimal trajectories s'_2 such that $\widetilde{Q}(s'_2) \geq 1/(4N)$ is implied by $\{Z \geq \lceil b/2 \rceil\}$, and therefore

$$\begin{aligned}
&\Pr\left(\exists \text{ at least } b \text{ suboptimal trajectories } s'_2 \text{ with } \widetilde{Q}(s'_2) \geq \frac{1}{4N} \,\middle|\, B^\star \cap E_1^c\right) \\
&\quad \geq \Pr\left(Z \geq \left\lceil \frac{b}{2} \right\rceil\right) = \Pr\left(\mathrm{Bin}\left(b - 1,\ 1 - 2\exp\left(-\frac{N}{8}\right)\right) \geq \left\lceil \frac{b}{2} \right\rceil\right).
\end{aligned} \tag{11}$$

By Lemma 1 (see Appendix C.1), the above binomial tail probability is nondecreasing along the even subsequence $\{b = 2, 4, 6, \ldots\}$ and along the odd subsequence $\{b = 3, 5, 7, \ldots\}$. Consequently,

$$\begin{aligned}
&\Pr\left(\mathrm{Bin}\left(b-1,\, 1-2\exp\left(-\frac{N}{8}\right)\right) \geq \left\lceil \frac{b}{2} \right\rceil\right) \\
&\quad \geq \min\left\{\Pr\left(\mathrm{Bin}\left(1,\, 1-2\exp\left(-\frac{N}{8}\right)\right) \geq 1\right),\ \Pr\left(\mathrm{Bin}\left(2,\, 1-2\exp\left(-\frac{N}{8}\right)\right) \geq 2\right)\right\} \\
&\quad = \left(1-2\exp\left(-\frac{N}{8}\right)\right)^2. \qquad (12)
\end{aligned}$$

Using a union bound under the conditioning $B^\star \cap E_1^c$, we obtain

$$\Pr(\mathcal{E} \mid B^\star \cap E_1^c) \geq \left(1-2\exp\left(-\frac{N}{8}\right)\right)^2 - \frac{4}{9}.$$

Since $N \geq 35$ in Case 2, we have

$$\left(1-2\exp\left(-\frac{N}{8}\right)\right)^2 - \frac{4}{9} \geq \left(1-2\exp\left(-\frac{35}{8}\right)\right)^2 - \frac{4}{9} = 0.506$$

Under the good event $\mathcal{E}$, there exist at least b suboptimal trajectories s_2' such that

$$\widetilde{Q}(s_2^\star) < \frac{1}{4N} \leq \widetilde{Q}(s_2').$$

Therefore, the optimal response is crowded out of the top-b candidates by b suboptimal trajectories. Combining this result with the bound of $\Pr(B^\star \mid E_1^c)$ yields

$$\Pr(s^\star \text{ is eliminated}) \geq 0.5.$$

The proof is completed. □

C.1 Technical Lemmas

In the subsection, we show that $\Pr\big(\mathrm{Bin}(b-1,p) \geq \frac{b}{2}\big)$ is monotone along the odd and even subsequences of b, respectively. This result is useful in bounding Equation (11) in the proof of Theorem 1.

Lemma 1 (Monotonicity in b for odd/even subsequences)**.** *Let $N > 30$ and let $p := 1 - \frac{4}{N} > \frac{1}{2}$. Define, for integers $b \geq 2$,*

$$F(b) := \Pr\left(\mathrm{Bin}(b-1,p) \geq \frac{b}{2}\right),$$

where for odd b the threshold is understood in the usual way: $\{\mathrm{Bin}(b-1,p) \geq b/2\} = \{\mathrm{Bin}(b-1,p) \geq \lceil b/2 \rceil\}$. Then $F(b)$ is nondecreasing along the even subsequence $\{b = 2, 4, 6, \ldots\}$ and also nondecreasing along the odd subsequence $\{b = 3, 5, 7, \ldots\}$. In particular,

$$F(2) \leq F(4) \leq F(6) \leq \cdots, \qquad F(3) \leq F(5) \leq F(7) \leq \cdots.$$

Proof. Fix $p > \frac{1}{2}$. We consider the following two cases.

Even b. Let $b = 2k$ with $k \geq 1$ and let $S_k \sim \mathrm{Bin}(2k-1, p)$. Then $F(2k) = \Pr(S_k \geq k)$. We further have

$$S_{k+1} = S_k + B_1 + B_2,$$

where $B_1, B_2 \overset{i.i.d.}{\sim} \mathrm{Bern}(p)$ are independent of S_k. A direct decomposition over $S_k \in \{k-1, k, \geq k+1\}$ gives

$$F(2k+2) - F(2k) = -(1-p)^2 \Pr(S_k = k) + p^2 \Pr(S_k = k-1).$$

For $S_k \sim \mathrm{Bin}(2k-1, p)$,

$$\frac{\Pr(S_k = k)}{\Pr(S_k = k-1)} = \frac{\binom{2k-1}{k}}{\binom{2k-1}{k-1}} \cdot \frac{p}{1-p} = \frac{k}{k} \cdot \frac{p}{1-p} = \frac{p}{1-p}.$$

Hence $\Pr(S_k = k-1) = \Pr(S_k = k)\frac{1-p}{p}$. Substituting yields

$$F(2k+2) - F(2k) = \Pr(S_k = k)\,(1-p)\,(2p-1) \geq 0,$$

which implies that $\{F(2k)\}_{k\geq 1}$ is nondecreasing.

Odd b. Let $b = 2k+1$ with $k \geq 1$ and let $T_k \sim \mathrm{Bin}(2k, p)$. Then $F(2k+1) = \Pr(T_k \geq k+1)$. We then have $T_{k+1} = T_k + B_1 + B_2$ with the same B_1, B_2. Similarly,

$$F(2k+3) - F(2k+1) = -(1-p)^2 \Pr(T_k = k+1) + p^2 \Pr(T_k = k).$$

For $T_k \sim \mathrm{Bin}(2k, p)$,

$$\frac{\Pr(T_k = k+1)}{\Pr(T_k = k)} = \frac{\binom{2k}{k+1}}{\binom{2k}{k}} \cdot \frac{p}{1-p} = \frac{k}{k+1} \cdot \frac{p}{1-p}.$$

Hence, $\Pr(T_k = k) = \Pr(T_k = k+1)\frac{k+1}{k}\frac{1-p}{p}$. Therefore,

$$F(2k+3) - F(2k+1) = \Pr(T_k = k+1)\,(1-p)\left(\frac{k+1}{k}p - (1-p)\right).$$

Since $p > \frac{1}{2}$ implies $\frac{k+1}{k}p \geq p > 1-p$, the bracketed term is positive. Hence, $F(2k+3) \geq F(2k+1)$ for all $k \geq 1$. This proves monotonicity on both subsequences. Finally, the assumption $N > 30$ ensures $p = 1 - \frac{4}{N} > \frac{1}{2}$, and hence the above applies. □

D Proof of Proposition 1

Proof of Proposition 1. We exhibit an instance with horizon $L = 1$ that violates Assumption 3 and bound the survival probability of $y^\star$ via Markov's inequality.

Construction. Let $\mathcal{A} = \{a^\star, a_1, \ldots, a_{|\mathcal{A}|-1}\}$ and define

$$\pi_{\mathrm{ref}}(a^\star \mid x) = \tfrac{1}{8}, \qquad \pi_{\mathrm{ref}}(a_1 \mid x) = \tfrac{7}{8}, \qquad \pi_{\mathrm{ref}}(a_i \mid x) = 0 \ \text{ for } i \geq 2.$$

Set the ground-truth reward as $r^\star(x, a^\star) = 1$ and $r^\star(x, a) = 0$ for all $a \neq a^\star$, so that $y^\star = a^\star$ is the unique optimal response.

Violation of Assumption 3. At depth $t = 1$, the suboptimal prefix $s = a_1 \neq s_1^\star = a^\star$ satisfies

$$\log\left(\frac{\pi_{\mathrm{ref}}(a^\star \mid x)}{\pi_{\mathrm{ref}}(a_1 \mid x)}\right) = \log\frac{1/8}{7/8} = -\log 7 < 0,$$

so no $\kappa > 0$ can satisfy the prefix-competitiveness condition on this instance.

Survival probability. Fix any $N \geq 1$. Let

$$X^\star := N\,\hat{\pi}(a^\star \mid x) \sim \mathrm{Bin}\big(N, \tfrac{1}{8}\big), \qquad X_1 := N\,\hat{\pi}(a_1 \mid x) \sim \mathrm{Bin}\big(N, \tfrac{7}{8}\big).$$

Since $a^\star$ and a_1 are the only tokens with positive mass under $\pi_{\mathrm{ref}}(\cdot \mid x)$, we have $X^\star + X_1 = N$ almost surely. With $b = 1$, the algorithm retains the sampled candidate with the largest score $Q(a) = Q(s_0) + \log\hat{\pi}(a \mid x) = \log\hat{\pi}(a \mid x)$. Since the logarithm is strictly increasing, this is equivalent to retaining the candidate with the largest empirical frequency $\hat{\pi}(a \mid x)$. Hence $y^\star = a^\star$ is retained in the final beam only if $\hat{\pi}(a^\star \mid x) \geq \hat{\pi}(a_1 \mid x)$, equivalently $X^\star \geq X_1$. Since $X^\star + X_1 = N$, this is equivalent to $X^\star \geq N/2$.

Applying Markov's inequality to the nonnegative random variable $X^\star$ with $\mathbb{E}[X^\star] = N/8$,

$$\Pr\big(X^\star \geq \tfrac{N}{2}\big) \;\leq\; \frac{\mathbb{E}[X^\star]}{N/2} \;=\; \frac{N/8}{N/2} \;=\; \frac{1}{4}.$$

Consequently,

$$\Pr(y^\star \text{ retained in the final beam}) \;\leq\; \Pr\big(X^\star \geq \tfrac{N}{2}\big) \;\leq\; \frac{1}{4},$$

so the probability of pruning $y^\star$ is at least $1 - \frac{1}{4} = \frac{3}{4}$, completing the proof. □

E Regret Upper Bound: Proof of Theorem 2 and Corollary 2

E.1 Technical Lemmas: Filtering and Beam Survival

We first control the distribution of the final output and the probability of filtering out an optimal action. We then bound moments of filtered score differences and use them to control the rank of the optimal prefix. The binomial moment inequalities used below are proved in Lemma 6 in the Auxiliary Lemmas section.

Fix x throughout this section. As in Definition 2, independently generate an N-sample histogram at every prefix of depth less than L and reveal it only if the algorithm visits that prefix. Along a fixed path, these histograms belong to distinct nodes and are independent; paths with shared ancestors still have dependent scores. For a prefix $s_t = (a_1, \dots, a_t)$, write $\mathcal{F}(s_t) := \{\hat{\pi}(a_i \mid s_{i-1}, x) \geq \beta_i,\ i \in [t]\}$. On this event, $Q(s_t) = \sum_{i=1}^{t} \log \hat{\pi}(a_i \mid s_{i-1}, x)$ is finite. Expressions involving inverse empirical probabilities or differences of scores are defined to be zero outside the indicated filtering events. In particular, $\mathcal{E}_\beta = \mathcal{F}(s_L^\star)$. This construction is used only for analysis and does not change the algorithm's queries.

Lemma 2 (Output probabilities under confidence filtering)**.** *For any thresholds $\beta_i \in (0, 1]$ and any fixed $y \in \mathcal{A}^L$, the output y^{beam} of CF-Beam satisfies*

$$\Pr(y^{\mathrm{beam}} = y, \mathcal{E}) \leq \pi_{\mathrm{ref}}(y \mid x) \prod_{i=1}^{L} \beta_i^{-1}.$$

Proof. On $\mathcal{E}$, the final beam is nonempty, and any output $y = (a_1, \dots, a_L)$ has passed filtering along all its edges. Hence

$$\begin{aligned}\Pr(y^{\mathrm{beam}} = y, \mathcal{E}) \leq \Pr(\mathcal{F}(y)) &= \prod_{i=1}^{L} \Pr\big(\hat{\pi}(a_i \mid s_{i-1}, x) \geq \beta_i\big) \\ &\leq \prod_{i=1}^{L} \frac{\mathbb{E}[\hat{\pi}(a_i \mid s_{i-1}, x)]}{\beta_i} = \pi_{\mathrm{ref}}(y \mid x) \prod_{i=1}^{L} \beta_i^{-1}.\end{aligned}$$

The equality uses independence of the histograms along the fixed path, and the inequality follows from Markov's inequality and unbiasedness of each empirical frequency. The argument does not condition the histograms on the output-selection event. □

Lemma 3 (Optimal actions pass filtering)**.** *For $\beta_t = \alpha / C_t^\star(x)$ with $0 < \alpha < 1$,*

$$\Pr(\mathcal{E}_\beta^c) \leq L \exp\left(-\frac{N(1-\alpha)^2}{2C_{\max}^\star}\right).$$

Proof. For each t, the count $N\hat{\pi}(a_t^\star \mid s_{t-1}^\star, x)$ has distribution $\mathrm{Bin}(N, 1/C_t^\star(x))$. Applying the lower-tail Chernoff bound in Lemma 8 with relative deviation $1 - \alpha$, and then a union bound, gives

$$\begin{aligned}\Pr(\mathcal{E}_\beta^c) &\leq \sum_{t=1}^{L} \Pr\left(N\hat{\pi}(a_t^\star \mid s_{t-1}^\star, x) < \frac{\alpha N}{C_t^\star(x)}\right) \\ &\leq \sum_{t=1}^{L} \exp\left(-\frac{N(1-\alpha)^2}{2C_t^\star(x)}\right) \leq L \exp\left(-\frac{N(1-\alpha)^2}{2C_{\max}^\star}\right).\end{aligned}$$

□

Lemma 4 (Moments of filtered score differences)**.** *Let $\beta_i = \alpha / C_i^\star(x)$ with $1/2 \leq \alpha < 1$. For every $s_t \neq s_t^\star$ and integer $1 \leq r \leq N\alpha/(4C_{\max}^\star)$,*

$$\mathbb{E}\Big[e^{r(Q(s_t) - Q(s_t^\star))} \mathbf{1}_{\mathcal{F}(s_t) \cap \mathcal{F}(s_t^\star)}\Big] \leq \left(\frac{\pi_{\mathrm{ref}}(s_t \mid x)}{\pi_{\mathrm{ref}}(s_t^\star \mid x)}\right)^r \exp\left(\frac{3tC_{\max}^\star r^2}{N\alpha}\right).$$

Proof. For each depth i, let $K_i = \lceil N\beta_i \rceil \geq N\alpha/C^\star_{\max}$, so $2r \leq K_i/2$. Apply the positive and inverse moment bounds of Lemma 6 with order $2r$ to each edge. Independence along each fixed path yields

$$\mathbb{E}\Big[e^{2rQ(s_t)}\mathbf{1}_{\mathcal{F}(s_t)}\Big] \leq \pi_{\mathrm{ref}}(s_t \mid x)^{2r} \exp\left(\sum_{i=1}^{t} \frac{(2r)(2r-1)}{2(K_i - 2r + 1)}\right),$$
$$\mathbb{E}\Big[e^{-2rQ(s^\star_t)}\mathbf{1}_{\mathcal{F}(s^\star_t)}\Big] \leq \pi_{\mathrm{ref}}(s^\star_t \mid x)^{-2r} \exp\left(\sum_{i=1}^{t} \frac{(2r)(2r+1)}{2K_i}\right).$$

All optimal edge probabilities are positive by the finiteness of $C^\star_{\max}$. A competing path of true probability zero has zero probability of passing filtering, so its contribution is zero. Cauchy–Schwarz now gives

$$\begin{aligned}
&\mathbb{E}\Big[e^{r(Q(s_t)-Q(s^\star_t))}\mathbf{1}_{\mathcal{F}(s_t)\cap\mathcal{F}(s^\star_t)}\Big] \\
&\leq \Big(\mathbb{E}[e^{2rQ(s_t)}\mathbf{1}_{\mathcal{F}(s_t)}]\mathbb{E}[e^{-2rQ(s^\star_t)}\mathbf{1}_{\mathcal{F}(s^\star_t)}]\Big)^{1/2} \\
&\leq \left(\frac{\pi_{\mathrm{ref}}(s_t \mid x)}{\pi_{\mathrm{ref}}(s^\star_t \mid x)}\right)^r \exp\left[\sum_{i=1}^{t}\left(\frac{(2r)(2r-1)}{4(K_i - 2r + 1)} + \frac{(2r)(2r+1)}{4K_i}\right)\right] \\
&\leq \left(\frac{\pi_{\mathrm{ref}}(s_t \mid x)}{\pi_{\mathrm{ref}}(s^\star_t \mid x)}\right)^r \exp\left(\sum_{i=1}^{t}\frac{3r^2}{K_i}\right) \leq \left(\frac{\pi_{\mathrm{ref}}(s_t \mid x)}{\pi_{\mathrm{ref}}(s^\star_t \mid x)}\right)^r \exp\left(\frac{3tC^\star_{\max}r^2}{N\alpha}\right).
\end{aligned}$$

For the penultimate inequality, $K_i - 2r + 1 \geq K_i/2$ bounds the two terms by $(2r^2 - r)/K_i$ and $(r^2 + r/2)/K_i$, respectively. In particular, Cauchy–Schwarz controls the common optimal score and shared-prefix randomness without requiring independent path scores. □

Lemma 5 (Survival under top-b pruning)**.** *Suppose Assumption 3 holds, $\beta_i = \alpha/C^\star_i(x)$ with $1/2 \leq \alpha < 1$, and $N \geq \frac{48C^\star_{\max}}{\kappa^2}\max\{2, \log(2C^\star_{\max})\}$. Then*

$$\Pr(\mathcal{E}_\beta \cap \mathcal{E}_b^c) \leq \frac{2C^\star_{\max}}{b}\exp\left(-\frac{\kappa^2 N}{48C^\star_{\max}}\right).$$

More precisely, let Z_t count the filtered nonoptimal prefixes in the full depth-t tree whose scores are at least $Q(s^\star_t)$, and set $Z_t = 0$ if $\mathcal{F}(s^\star_t)$ fails. For every integer $m \geq 1$,

$$\Pr(Z_t \geq m) \leq \frac{1}{m}\left[C^\star_{\max}\exp\left(-\frac{\kappa^2 N}{48C^\star_{\max}}\right)\right]^t. \tag{13}$$

Proof. For any admissible integer $r \geq 1$ in Lemma 4, each competitor counted by Z_t contributes at least one to the corresponding likelihood-ratio moment. Since the depth-t tree is finite, summing the lemma gives

$$\begin{aligned}
\mathbb{E}[Z_t] &\leq \sum_{s_t \neq s^\star_t} \mathbb{E}\Big[e^{r(Q(s_t)-Q(s^\star_t))}\mathbf{1}_{\mathcal{F}(s_t)\cap\mathcal{F}(s^\star_t)}\Big] \\
&\leq \exp\left(\frac{3tC^\star_{\max}r^2}{N\alpha}\right)\sum_{s_t \neq s^\star_t}\left(\frac{\pi_{\mathrm{ref}}(s_t \mid x)}{\pi_{\mathrm{ref}}(s^\star_t \mid x)}\right)^r \\
&\leq \exp\left(\frac{3tC^\star_{\max}r^2}{N\alpha} - (r-1)t\kappa\right)\frac{\sum_{s_t \neq s^\star_t}\pi_{\mathrm{ref}}(s_t \mid x)}{\pi_{\mathrm{ref}}(s^\star_t \mid x)} \\
&\leq \left[C^\star_{\max}\exp\left(-(r-1)\kappa + \frac{3C^\star_{\max}r^2}{N\alpha}\right)\right]^t.
\end{aligned}$$

The third line uses Assumption 3, which implies $\pi_{\mathrm{ref}}(s_t \mid x) \leq e^{-t\kappa}\pi_{\mathrm{ref}}(s^\star_t \mid x)$, and $r - 1 \geq 0$. The last line uses total prefix probability at most one and $\pi_{\mathrm{ref}}(s^\star_t \mid x)^{-1} = \prod_{i=1}^t C^\star_i(x) \leq (C^\star_{\max})^t$. Thus the sum is controlled by true probability mass rather than by the number of adaptively selected candidates.

Choose $r = \lfloor N\alpha\kappa/(6C^\star_{\max})\rfloor$. The sample-size condition and $1/2 \le \alpha < 1$, $0 < \kappa \le 1$ imply $N\alpha\kappa^2/C^\star_{\max} \ge 48$ and $1 \le r \le N\alpha/(4C^\star_{\max})$. Using the two bounds on the floor function,

$$-(r-1)\kappa + \frac{3C^\star_{\max} r^2}{N\alpha} \le -\frac{N\alpha\kappa^2}{12C^\star_{\max}} + 2\kappa \le -\frac{N\alpha\kappa^2}{24C^\star_{\max}} \le -\frac{N\kappa^2}{48C^\star_{\max}}.$$

The middle inequality follows from $N\alpha\kappa^2/C^\star_{\max} \ge 48 \ge 48\kappa$. Markov's inequality applied to Z_t proves (13).

On $\mathcal{E}_\beta$, if the optimal path is first discarded by beam pruning at depth t, it is a generated candidate and at least b other generated candidates have scores at least its score. These candidates are among the full-tree filtered prefixes, so $Z_t \ge b$. Counting all ties against the optimal prefix makes this implication valid for any tie-breaking rule. Since the sample-size condition also gives $C^\star_{\max} e^{-N\kappa^2/(48C^\star_{\max})} \le 1/2$, a union bound and a geometric sum yield

$$\begin{aligned}\Pr(\mathcal{E}_\beta \cap \mathcal{E}_b^c) &\le \sum_{t=1}^{L} \Pr(Z_t \ge b) \le \frac{1}{b}\sum_{t=1}^{L}\left[C^\star_{\max} e^{-N\kappa^2/(48C^\star_{\max})}\right]^t \\ &\le \frac{C^\star_{\max} e^{-N\kappa^2/(48C^\star_{\max})}}{b\left(1 - C^\star_{\max} e^{-N\kappa^2/(48C^\star_{\max})}\right)} \le \frac{2C^\star_{\max}}{b} e^{-N\kappa^2/(48C^\star_{\max})}.\end{aligned}$$

This bounds the intersection with $\mathcal{E}_\beta$ directly; no concentration inequality is applied after conditioning on a future good event. □

E.2 Proof of Theorem 2

Proof. For $L \ge 2$, the threshold multiplier $1 - 1/L$ lies in $[1/2, 1)$. Applying Lemmas 3 and 5 with $\alpha = 1 - 1/L$ gives

$$\begin{aligned}\Pr(\mathcal{E}^c) &= \Pr(\mathcal{E}_\beta^c) + \Pr(\mathcal{E}_\beta \cap \mathcal{E}_b^c) \\ &\le L\exp\left(-\frac{N}{2C^\star_{\max}L^2}\right) + \frac{2C^\star_{\max}}{b}\exp\left(-\frac{\kappa^2 N}{48C^\star_{\max}}\right).\end{aligned}$$

On $\mathcal{E}$, induction over the depths gives $y^\star \in \mathcal{B}_L$, so the final selection satisfies $\hat{r}(x, y^{\text{beam}}) \ge \hat{r}(x, y^\star)$. On $\mathcal{E}^c$, the regret is at most one because rewards lie in $[0, 1]$. Consequently,

$$\begin{aligned}\mathrm{Reg}(x) &\le \Pr(\mathcal{E}^c) + \mathbb{E}\Big[\mathbf{1}_{\mathcal{E}}\big(r^\star(x,y^\star) - \hat{r}(x,y^\star) + \hat{r}(x,y^\star) - \hat{r}(x,y^{\text{beam}}) \\ &\qquad + \hat{r}(x,y^{\text{beam}}) - r^\star(x,y^{\text{beam}})\big)\Big] \\ &\le \Pr(\mathcal{E}^c) + \epsilon_{\text{opt}}(x) + \mathbb{E}\big[\mathbf{1}_{\mathcal{E}}\big|\hat{r}(x,y^{\text{beam}}) - r^\star(x,y^{\text{beam}})\big|\big] \\ &\le \Pr(\mathcal{E}^c) + \epsilon_{\text{opt}}(x) + \left(\sum_{y\in\mathcal{A}^L} \Pr(y^{\text{beam}} = y, \mathcal{E})\big(\hat{r}(x,y) - r^\star(x,y)\big)^2\right)^{1/2} \\ &\le \Pr(\mathcal{E}^c) + \epsilon_{\text{opt}}(x) + \left[\left(1 - \frac{1}{L}\right)^{-L}\overline{C}(x)\sum_{y\in\mathcal{A}^L}\pi_{\text{ref}}(y \mid x)\big(\hat{r}(x,y) - r^\star(x,y)\big)^2\right]^{1/2} \\ &\le \Pr(\mathcal{E}^c) + \epsilon_{\text{opt}}(x) + 2\sqrt{\overline{C}(x)\epsilon^2_{\text{RM}}(x)}.\end{aligned}$$

The second inequality uses Assumption 2 and reward maximization. The third is Cauchy–Schwarz with $\Pr(\mathcal{E}) \le 1$, and the fourth applies Lemma 2. The last inequality follows from Assumption 1 and $(1 - 1/L)^{-L} \le 4$ for integer $L \ge 2$. This inequality follows because $u\log(1 - 1/u)$ is nondecreasing for $u \ge 2$: its derivative is $\log(1 - 1/u) + 1/(u - 1) \ge 0$ by $\log(1 + v) \le v$. The distribution comparison is for the joint probabilities $\Pr(y^{\text{beam}} = y, \mathcal{E})$, so no normalization by $\Pr(\mathcal{E})$ is needed. □

E.3 Proof of Corollaries 1

Proof of Corollary 1. Each search-error term in (8) is at most $\delta/2$ whenever

$$N \geq \max\left\{ \frac{48C^\star_{\max}}{\kappa^2}\max\{2,\log(2C^\star_{\max})\},\quad 2C^\star_{\max}L^2\log\frac{2L}{\delta},\quad \frac{48C^\star_{\max}}{\kappa^2}\left[\log\frac{4C^\star_{\max}}{b\delta}\right]_+ \right\}.$$

The first term is the sample-size condition in Theorem 2; the other two follow by solving the two exponential inequalities. Since $C^\star_{\max} \geq 1$, $[\log(4C^\star_{\max}/(b\delta))]_+ \leq \log(4C^\star_{\max}) + [\log(1/(b\delta))]_+$, so (9) with a sufficiently large universal constant implies all three requirements. Substitution into (7) proves the corollary. □

E.4 Proof of Corollary 2

Proof. Apply Lemma 3 with $\alpha = 1/2$ to obtain $\Pr(\mathcal{E}_\beta^c) \leq L\exp(-N/(8C^\star_{\max}))$. Unlike reward-based selection, the final maximum-score selection must also distinguish $y^\star$ from every remaining competitor. If $\mathcal{E}_\beta$ holds, $Z_t < b$ at every $t < L$, and $Z_L = 0$, then the optimal path survives the first $L-1$ rounds, has strictly larger score than every other filtered complete path, and is selected at the final round. Thus

$$\begin{aligned}
\mathrm{Reg}(x) &\leq \Pr(y^{\mathrm{beam}} \neq y^\star) \leq \Pr(\mathcal{E}_\beta^c) + \sum_{t=1}^{L-1}\Pr(Z_t \geq b) + \Pr(Z_L \geq 1) \\
&\leq L\exp\left(-\frac{N}{8C^\star_{\max}}\right) + \frac{1}{b}\sum_{t=1}^{L-1}\left[C^\star_{\max}e^{-N\kappa^2/(48C^\star_{\max})}\right]^t + \left[C^\star_{\max}e^{-N\kappa^2/(48C^\star_{\max})}\right]^L \\
&\leq L\exp\left(-\frac{N}{8C^\star_{\max}}\right) + \frac{2C^\star_{\max}}{b}\exp\left(-\frac{N\kappa^2}{48C^\star_{\max}}\right) + \left[C^\star_{\max}e^{-N\kappa^2/(48C^\star_{\max})}\right]^L.
\end{aligned}$$

The first inequality uses bounded rewards and optimal reward one. The next bounds follow from (13) and the same geometric sum as in Lemma 5; final selection is controlled with $m = 1$, while intermediate pruning is controlled with $m = b$.

For an explicit sufficient sample size, take $N \geq 96(C^\star_{\max}/\kappa^2)\log(6LC^\star_{\max}/\delta)$. This satisfies the sample-size condition of Lemma 5 and gives $C^\star_{\max}e^{-N\kappa^2/(48C^\star_{\max})} \leq \delta/(6L)$. The ranking sum is then at most $\delta/(3L)$, and its final-depth power is at most $\delta/(6L)$, since $b \geq 1$ and $\delta/(6L) < 1$. Also $N \geq 8C^\star_{\max}\log(3L/\delta)$, so the filtering term is at most $\delta/3$. Their sum is at most δ, proving the stated sufficient sample complexity. □

F Regret Lower Bound: Proof of Theorem 3

Proof of Theorem 3. Fix a horizon L and an action set $\mathcal{A} = \{1, 2, \ldots, |\mathcal{A}|\}$ with $|\mathcal{A}| \geq \frac{\log L}{\log(4/3)} + 2$. We construct an instance in which, at each time step $t \in [L]$, there is a unique optimal action along the optimal trajectory. Without loss of generality (by relabeling actions), we denote this optimal action $a_t^\star$ by $a_{t,1}$.

Let $s^\star_{t-1}$ be the state reached by following the optimal action sequence up to time $t-1$ (i.e., the "optimal-prefix" state), and let s_{t-1} denote any state that is *not* on this optimal prefix. We define a reference policy $\pi_{\mathrm{ref}}(\cdot \mid s, x)$ as follows:

$$\pi_{\mathrm{ref}}(a_{t,1} \mid s^\star_{t-1}, x) = \frac{1}{|\mathcal{A}|} + \kappa', \qquad \pi_{\mathrm{ref}}(a_{t,i} \mid s^\star_{t-1}, x) = \frac{1}{|\mathcal{A}|} - \frac{\kappa'}{|\mathcal{A}|-1},\ \ i = 2, \ldots, |\mathcal{A}|,$$

and for any off-prefix state $s_{t-1} \neq s^\star_{t-1}$,

$$\pi_{\mathrm{ref}}(a_{t,i} \mid s_{t-1}, x) = \frac{1}{|\mathcal{A}|}, \qquad i = 1, \ldots, |\mathcal{A}|.$$

By construction, $\sum_{i=1}^{|\mathcal{A}|}\pi_{\mathrm{ref}}(a_{t,i} \mid s^\star_{t-1}, x) = 1$, and we assume κ' is a sufficiently small value that will be assigned later. Next, define the true reward $r^\star$ and the estimated reward $\hat{r}$ for complete

trajectories. Let $y^\star$ denote the unique optimal trajectory induced by choosing $a_{t,1}$ at every step along the optimal prefix, and let $y \neq y^\star$ denote any other trajectory. We set

$$r^\star(x, y^\star) = 1, \qquad r^\star(x, y) = 1 - \delta \ \ (y \neq y^\star),$$

while the reward estimator is misspecified in the opposite direction:

$$\hat{r}(x, y^\star) = 1 - 2\delta, \qquad \hat{r}(x, y) = 1 - \delta \ \ (y \neq y^\star),$$

where $\delta \in (0, 1/2]$ will be chosen later, which ensures all rewards lie in $[0, 1]$. This construction satisfies Assumption 3.

Thereby, under this hard instance, if the final top-b response contains suboptimal responses, the reward model will output the incorrect result with probability one. According to the definition of Regret,

$$\mathrm{Reg}(x) = r^\star(x, y^\star) - \mathbb{E}_{y \sim \pi_{\mathrm{beam}}(\cdot|x)}[r^\star(x, y)] = \delta \pi_{\mathrm{beam}}(\mathcal{Y}'|x),$$

where $\mathcal{Y}' = \mathcal{Y} \backslash \{y^\star\}$. Since we have $b \geq 2$ and the reward model will choose a suboptimal response if the final top-b set contains $y \in \mathcal{Y}'$,

$$\pi_{\text{beam}}(\mathcal{Y}'|x) = 1 - \Pr(\text{the final top } b \text{ only contains } y^\star).$$

Providing the lower bound is to providing the upper bound for $\Pr(\text{the final top } b \text{ only contains } y^\star)$.

We next prove $\Pr(\text{the final top } b \text{ only contains } y^\star) \leq 3/4$ by cases, and hence $\pi_{\text{beam}}(\mathcal{Y}'|x) \geq 1/4$ is in a constant order.

Case 1: $N \leq \max\{C_t^\star(x)\}$**.** Under this case, let $\kappa' \leq 1/|\mathcal{A}|$. At the step where $C_t^\star(x) = C_{\max}^\star$, the probability generating the correct response can be upper bounded by

$$1 - \left(1 - \frac{1}{C_t^\star}\right)^N \leq 1 - \left(1 - \frac{1}{C_t^\star}\right)^{C_t^\star} \leq 3/4.$$

where the last inequality follows by $C_t^\star = \frac{|\mathcal{A}|}{\kappa'|\mathcal{A}|+1} \geq 2$.

Therefore, the optimal response $y^\star$ is excluded from the final top-b candidate set with probability at least $1/4$, and hence $\Pr(\text{the final top } b \text{ only contains } y^\star) \leq 3/4$.

Case 2.1: $\{N > \max\{C_t^\star(x)\}$ **and** $\exists\ \beta_t \geq 1/C_t^\star(x)\}$**.** In this case, the optimal trajectory will be eliminated with a constant probability greater than 1/4, since

$$\Pr\left(\text{Bin}(N, 1/C_t^\star(x)) < N\beta_t\right) \geq \Pr\left(\text{Bin}(N, 1/C_t^\star(x)) < \frac{N}{C_t^\star(x)}\right) \geq \frac{1}{4},$$

where the last inequality follows from Lemma 12. Thus we exclude the optimal response from the final top b candidate set with probability at least $1/4$, and hence $\Pr(\text{the final top } b \text{ only contains } y^\star) \leq 3/4$.

Case 2.2: $\{N > \max_{t\in[L]} C_t^\star(x)\}$ **and** $\forall t,\ \beta_t < 1/C_t^\star(x)\}$**.** In this case, every filtering threshold β_t lies *below* the optimal-trajectory probability $1/C_t^\star(x)$. Hence, for each $t \in [L]$, there exists some $\alpha \in (0, 1)$ such that $\beta_t < \frac{\alpha}{C_t^\star(x)}$. We choose κ' so that the probability of any fixed suboptimal next-token choice at the optimal prefix dominates the scaled optimal one, namely

$$\alpha\Big(\frac{1}{|\mathcal{A}|} + \kappa'\Big) \ \leq \ \frac{1}{|\mathcal{A}|} - \frac{\kappa'}{|\mathcal{A}| - 1}.$$

Equivalently, $\alpha\, \pi_{\text{ref}}\big(a_t^\star \mid s_{t-1}^\star, x\big) \ \leq \ \pi_{\text{ref}}\big(a_{t,i} \mid s_{t-1}^\star, x\big)\,,\ \forall i \in \{2, \ldots, |\mathcal{A}|\}$. Now fix any round t and consider any suboptimal child s_t' obtained by taking corresponding $a_{t,i}$ with $i \geq 2$ from the optimal prefix $s_{t-1}^\star$. Let

$$p := \pi_{\text{ref}}(a_{t,i} \mid s_{t-1}^\star, x) = \frac{1}{|\mathcal{A}|} - \frac{\kappa'}{|\mathcal{A}| - 1}.$$

By the choice of α and κ', we have $\beta_t < \alpha/C_t^\star(x) \le p$. Therefore, under N i.i.d. draws, this suboptimal child survives the β_t-filtering with constant probability:

$$\Pr(\mathrm{Bin}(N,p) \ge N\beta_t) \;\ge\; \Pr(\mathrm{Bin}(N,p) > N\beta_t) \;\ge\; \Pr(\mathrm{Bin}(N,p) > Np) \;\ge\; \frac{1}{4},$$

where the last inequality follows from Lemma 11. Consequently, the probability that a single suboptimal child is eliminated at round t is at most

$$\Pr(\mathrm{Bin}(N,p) < N\beta_t) \le \frac{3}{4}.$$

Since $b \ge 2$, as long as at least one suboptimal trajectory remains in the beam, the final top-b set cannot consist solely of $y^\star$ unless *all* suboptimal trajectories are eliminated at some intermediate round by the β_t-filtering (so that the beam collapses to a singleton). Formally, for each $t \in [L]$ define the event

$$E_t := \{\text{at round } t, \text{ after } \beta_t\text{-filtering, } \textit{no} \text{ suboptimal child survives}\}, \qquad E := \bigcup_{t=1}^{L} E_t.$$

Then $\{$the final top-b set contains only $y^\star\} \subseteq E$, and hence

$$\Pr(\text{the final top-}b\text{ set contains only } y^\star) \le \Pr(E) \le \sum_{t=1}^{L} \Pr(E_t).$$

Next, fix a round t and consider the N draws from the next-token distribution at the optimal prefix $s_{t-1}^\star$. Let $(N_{t,1}, \ldots, N_{t,|\mathcal{A}|})$ denote the resulting multinomial count vector with parameters N and probabilities $(p_{t,1}, \ldots, p_{t,|\mathcal{A}|})$. Since multinomial counts are negatively associated, and thus satisfy a negative lower-orthant dependence bound Joag-Dev and Proschan (1983):

$$\Pr(E_t) \le \Pr\left(\bigcap_{i=2}^{|\mathcal{A}|} \{N_{t,i} < N\beta_t\}\right) \le \prod_{i=2}^{|\mathcal{A}|} \Pr(N_{t,i} < N\beta_t).$$

Moreover, for each $i \ge 2$, $N_{t,i} \sim \mathrm{Bin}(N, p_{t,i})$, and by the previous bound $\Pr(\mathrm{Bin}(N, p_{t,i}) < N\beta_t) \le 3/4$. Therefore, $\Pr(E_t) \le \left(\frac{3}{4}\right)^{|\mathcal{A}|-1}$. Consequently,

$$\Pr(E) \le \sum_{t=1}^{L} \Pr(E_t) \le L \cdot \left(\frac{3}{4}\right)^{|\mathcal{A}|-1} \le \frac{3}{4},$$

where the last inequality follows from our choice of $|\mathcal{A}|$, namely $|\mathcal{A}| \ge \frac{\log L}{\log(4/3)} + 2$.

Taking **Case 1**, **Case 2.1** and **Case 2.2** together, $\Pr($the final top b only contains $y^\star) \le 3/4$.

Finally, we specify δ through the notation of reward model error. By the definition of ϵ_{RM}^2, we have

$$\epsilon_{\mathrm{RM}}^2 = \mathbb{E}_{y\sim\pi_{\mathrm{ref}}(\cdot|x)}\left[(r^*(x,y) - \hat{r}(x,y))^2\right] = \pi_{\mathrm{ref}}(y^\star|x) \cdot 4\delta^2 = \frac{4\delta^2}{\overline{C}}.$$

Therefore, taking $\delta = \sqrt{\overline{C}(x)\epsilon_{\mathrm{RM}}^2(x)}/2$, the regret can be lower bounded as

$$\begin{aligned}
\mathrm{Reg}(x) &= \delta\pi_{\mathrm{beam}}(\mathcal{Y}'|x) \\
&= \sqrt{\frac{\overline{C}(x)\epsilon_{\mathrm{RM}}^2(x)}{4}}(1 - \Pr(\text{the final top } b \text{ only contains } y^\star)) \\
&\ge \frac{1}{4}\sqrt{\frac{\overline{C}(x)\epsilon_{\mathrm{RM}}^2(x)}{4}}.
\end{aligned}$$

The proof is completed. □

G Auxiliary Lemmas

Lemma 6 (Truncated binomial moments)**.** *Let $X \sim \mathrm{Bin}(N,p)$ and let $K \in \{1,\dots,N\}$. For every integer $1 \le q \le K$,*

$$\mathbb{E}\big[(X/N)^q \mathbf{1}_{\{X \ge K\}}\big] \le p^q \exp\!\left(\frac{q(q-1)}{2(K-q+1)}\right).$$

If $p > 0$, then for every integer $q \ge 1$,

$$\mathbb{E}\big[(Np/X)^q \mathbf{1}_{\{X \ge K\}}\big] \le \exp\!\left(\frac{q(q+1)}{2K}\right),$$

where the integrand is zero on $\{X < K\}$.

Proof. Write $(X)_q = X(X-1)\cdots(X-q+1)$. On $X \ge K \ge q$, the inequality $\log(1+u) \le u$ gives

$$\begin{aligned}
\log \frac{X^q}{(X)_q} &= \sum_{j=0}^{q-1} \log\left(1 + \frac{j}{X-j}\right) \le \sum_{j=0}^{q-1} \frac{j}{X-j} \le \frac{q(q-1)}{2(K-q+1)}, \\
\mathbb{E}[(X/N)^q \mathbf{1}_{\{X \ge K\}}] &\le \frac{e^{q(q-1)/(2(K-q+1))}}{N^q} \mathbb{E}[(X)_q] = \frac{(N)_q}{N^q} p^q e^{q(q-1)/(2(K-q+1))} \\
&\le p^q e^{q(q-1)/(2(K-q+1))}.
\end{aligned}$$

The factorial-moment identity follows by expressing $(X)_q$ as the sum over ordered q-tuples of distinct Bernoulli trials of the product of their success indicators. This also covers $p = 0$.

For the inverse moment, take $p > 0$. Expanding the binomial probability mass function and substituting $y = x + q$ yields

$$\begin{aligned}
\mathbb{E}\frac{1}{(X+1)\cdots(X+q)} &= \sum_{x=0}^{N} \frac{N! p^x (1-p)^{N-x}}{(x+q)!(N-x)!} \\
&= \frac{N!}{(N+q)! p^q} \sum_{y=q}^{N+q} \binom{N+q}{y} p^y (1-p)^{N+q-y} \\
&= \frac{\Pr\{\mathrm{Bin}(N+q,p) \ge q\}}{p^q (N+1)\cdots(N+q)} \le \frac{1}{p^q (N+1)\cdots(N+q)}.
\end{aligned}$$

For $p = 1$, the same identity follows directly from $X = N$. On $X \ge K$, the product $\prod_{j=1}^{q}(1 + j/X)$ is at most $\exp(q(q+1)/(2K))$. Consequently,

$$\begin{aligned}
\mathbb{E}[(Np/X)^q \mathbf{1}_{\{X \ge K\}}] &\le (Np)^q e^{q(q+1)/(2K)} \mathbb{E}\frac{1}{(X+1)\cdots(X+q)} \\
&\le \frac{N^q}{(N+1)\cdots(N+q)} e^{q(q+1)/(2K)} \le e^{q(q+1)/(2K)},
\end{aligned}$$

which proves the inverse-moment bound without a lower bound on p. □

Definition 3 (Bernoulli KL)**.** For $p, q \in (0,1)$, the Bernoulli KL divergence is defined as

$$\mathrm{KL}(p \,\|\, q) := p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q}.$$

Lemma 7 (Markov's inequality)**.** *Let $X \ge 0$ be a nonnegative random variable with $\mathbb{E}[X] < \infty$. Then for any $a > 0$,*

$$\mathbb{P}(X \ge a) \le \frac{\mathbb{E}[X]}{a}.$$

Lemma 8 (Chernoff Inequality under Binomial Setting)**.** *Let $X_i \sim$ Ber(p), $S = \sum_{i=1}^{n} X_i \sim$ Bin(n, p), $\mu = \mathbb{E}[S]$, then $\forall \delta > 0$,*

$$\Pr\left(S \geq (1+\delta)\mu\right) \leq \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu} \leq \exp\left(-\frac{\mu\delta^2}{2+\delta}\right),$$

$$\Pr\left(S \leq (1-\delta)\mu\right) \leq \exp\left(-\frac{\mu\delta^2}{2}\right).$$

A KL-form Chernoff bound is:

$$\Pr\left(\frac{S}{n} \geq \alpha\right) \leq \exp(-n\ \mathrm{KL}(\alpha\|p)) \quad \textit{for } \alpha > p,$$

$$\Pr\left(\frac{S}{n} \leq \alpha\right) \leq \exp(-n\ \mathrm{KL}(\alpha\|p)) \quad \textit{for } \alpha < p,$$

where

$$\mathrm{KL}(\alpha\|p) = \alpha \log\frac{\alpha}{p} + (1-\alpha)\log\frac{1-\alpha}{1-p}.$$

Lemma 9. *Let $X_{i,1}, \ldots, X_{i,N} \overset{\text{i.i.d.}}{\sim} \mathrm{Ber}(\pi(a_i \mid s_{i-1}, x))$ for some $\pi(a_i \mid s_{i-1}, x) \in (0,1)$, and define $\hat{\pi}(a_i \mid s_{i-1}, x) \triangleq \frac{1}{N}\sum_{j=1}^{N} X_{i,j}$. Then for any $\varepsilon \in (0,1)$,*

$$\Pr\left(\left|\log\frac{\hat{\pi}(a_i \mid s_{i-1}, x)}{\pi(a_i \mid s_{i-1}, x)}\right| \geq \log\left(\frac{1}{1-\varepsilon}\right)\right) \leq 2\exp\left(-\frac{N\pi(a_i \mid s_{i-1}, x)\varepsilon^2}{3}\right).$$

Proof. By Lemma 8, we have

$$\Pr\big(\big|\hat{\pi}(a_i \mid s_{i-1}, x) - \pi(a_i \mid s_{i-1}, x)\big| \geq \varepsilon\, \pi(a_i \mid s_{i-1}, x)\big) \leq 2\exp\left(-\frac{N\pi(a_i \mid s_{i-1}, x)\varepsilon^2}{3}\right).$$

That is, with probability at least $1 - 2\exp\left(-\frac{N\pi(a_i|s_{i-1},x)\varepsilon^2}{3}\right)$, we have

$$(1-\varepsilon)\pi(a_i \mid s_{i-1}, x) \leq \hat{\pi}(a_i \mid s_{i-1}, x) \leq (1+\varepsilon)\pi(a_i \mid s_{i-1}, x).$$

This implies

$$\log\frac{\hat{\pi}(a_i \mid s_{i-1}, x)}{\pi(a_i \mid s_{i-1}, x)} \in [\log(1-\varepsilon), \log(1+\varepsilon)].$$

Hence,

$$\left|\log\frac{\hat{\pi}(a_i \mid s_{i-1}, x)}{\pi(a_i \mid s_{i-1}, x)}\right| \leq \max\{-\log(1-\varepsilon), \log(1+\varepsilon)\} = \log\left(\frac{1}{1-\varepsilon}\right).$$

Therefore,

$$\Pr\left(\left|\log\frac{\hat{\pi}(a_i \mid s_{i-1}, x)}{\pi(a_i \mid s_{i-1}, x)}\right| \geq \log\left(\frac{1}{1-\varepsilon}\right)\right) \leq 2\exp\left(-\frac{N\pi(a_i \mid s_{i-1}, x)\varepsilon^2}{3}\right).$$

□

Lemma 10. *Let $\alpha \in (0,1)$ and $p \in (0,1)$. Then the following two lower bounds hold:*

$$\mathrm{KL}(p\|\alpha p) \geq (\log(1/\alpha) - (1-\alpha))\, p, \tag{14}$$

$$\mathrm{KL}(\alpha p\|p) \geq ((1-\alpha) - \alpha\log(1/\alpha))\, p. \tag{15}$$

Proof. We prove Eqs. (14) and (15) separately.

Proof of Eq. (14). By the definition of Bernoulli KL,

$$\mathrm{KL}(p\|\alpha p) = p\log\frac{p}{\alpha p} + (1-p)\log\frac{1-p}{1-\alpha p} = p\log\frac{1}{\alpha} - (1-p)\log\frac{1-\alpha p}{1-p}.$$

Let $u := \frac{(1-\alpha)p}{1-p} > 0$, then

$$\frac{1-\alpha p}{1-p} = 1 + u.$$

Using $\log(1+u) \le u$, we obtain

$$-(1-p)\log\frac{1-\alpha p}{1-p} = -(1-p)\log(1+u) \ge -(1-p)u = -(1-\alpha)p.$$

Combining the last display with the expression of $\mathrm{KL}(p\|\alpha p)$ yields

$$\mathrm{KL}(p\|\alpha p) \ge \big(\log(1/\alpha) - (1-\alpha)\big)p.$$

Proof of Eq. (15). Again by the definition of Bernoulli KL,

$$\mathrm{KL}(\alpha p\|p) = \alpha p\log\frac{\alpha p}{p} + (1-\alpha p)\log\frac{1-\alpha p}{1-p} = \alpha p\log\alpha + (1-\alpha p)\log\frac{1-\alpha p}{1-p}.$$

Let $x := \frac{(1-\alpha)p}{1-p} > 0$, so that

$$\frac{1-\alpha p}{1-p} = 1 + x.$$

Using $\log(1+x) \ge \frac{x}{1+x}$, we get

$$(1-\alpha p)\log\frac{1-\alpha p}{1-p} = (1-\alpha p)\log(1+x) \ge (1-\alpha p)\frac{x}{1+x}.$$

Note that

$$\frac{x}{1+x} = \frac{\frac{(1-\alpha)p}{1-p}}{1+\frac{(1-\alpha)p}{1-p}} = \frac{(1-\alpha)p}{1-\alpha p}.$$

Hence

$$(1-\alpha p)\frac{x}{1+x} = (1-\alpha)p.$$

Therefore,

$$\mathrm{KL}(\alpha p\|p) \ge \alpha p\log\alpha + (1-\alpha)p = \big((1-\alpha) - \alpha\log(1/\alpha)\big)p.$$

This completes the proof. □

Lemma 11. *Let $X \sim \mathrm{Bin}(N,p)$ with $N \ge 1$ and $p \in (0,1)$. If*

$$p \ge \frac{c}{N}, \qquad \textit{where } c := \ln(4/3) = 0.28768\ldots,$$

then

$$\Pr\big(X > \mathbb{E}[X]\big) \ge \frac{1}{4}.$$

Proof. See Theorem 1 of Pinelis (2021). □

Lemma 12. *Let $X \sim \mathrm{Bin}(N,p)$ with $N \ge 1$ and $p \in (0,1/2]$. Then*

$$\Pr\big(X < \mathbb{E}[X]\big) = \Pr(X < Np) \ge \frac{1}{4}.$$

Proof. Let $q := 1-p \in [1/2,1)$ and define $Y := N - X$. Then $Y \sim \mathrm{Bin}(N,q)$ and $\mathbb{E}[Y] = Nq$. Moreover,

$$\{X < Np\} \iff \{N - X > N - Np\} \iff \{Y > Nq\} \iff \{Y > \mathbb{E}[Y]\},$$

so $\Pr(X < Np) = \Pr(Y > \mathbb{E}[Y])$.

If $N = 1$, then $Y \in \{0,1\}$ and $\mathbb{E}[Y] = q \in (1/2,1)$, hence

$$\Pr(Y > \mathbb{E}[Y]) = \Pr(Y = 1) = q \ge \frac{1}{2}.$$

If $N \ge 2$, since $q \ge 1/2$ and $c = \ln(4/3) < 1/2$, we have $q \ge 1/2 \ge c \ge c/N$. Thus the condition of Lemma 11 holds for Y, and so

$$\Pr(Y > \mathbb{E}[Y]) \ge \frac{1}{4}.$$

Combining both cases yields $\Pr(X < Np) = \Pr(Y > \mathbb{E}[Y]) \ge 1/4$. □

H Experiment Details and Supplemental Experiments

H.1 Details of Numerical Experiments

Baselines. We compare CF-Beam against four representative test-time methods: (i) *Vanilla beam search*, which follows the same beam pipeline as CF-Beam but without the confidence-filtering step; (ii) *Best-of-N* (Huang et al., 2025), which samples N responses independently and selects the one with the highest reward; (iii) *Majority Voting*, which samples N responses independently and selects the one with the highest frequency; and (iv) *Best-of-Majority* (Di et al., 2025), which first filters candidates via majority voting (self-consistency) and then chooses the highest-scoring response under an external reward model.

Remark 4 (Oracle CF-Beam and Empirical CF-Beam)**.** *The threshold used in Theorem 2 is*

$$\beta_t = \frac{1 - 1/L}{C_t^\star(x)} = \Big(1 - \frac{1}{L}\Big)\pi_{\mathrm{ref}}(a_t^\star \mid s_{t-1}^\star, x),$$

which depends on the unknown optimal token probability along the optimal prefix. Therefore, this choice is oracle and is generally not directly implementable in practice. In the experiments, we include this version as Oracle CF-Beam*, which serves as a diagnostic reference for the theory-prescribed threshold.*

For a practical implementation, we instead use a prefix-adaptive empirical threshold

$$\beta_t(s_{t-1}) = \gamma\, \hat{p}_{\max,t}(s_{t-1}), \qquad \hat{p}_{\max,t}(s_{t-1}) := \max_{a \in \mathcal{A}} \hat{\pi}(a \mid s_{t-1}, x),$$

where $\gamma \in (0,1)$ is a tuning parameter. This rule is motivated by the fact that the oracle threshold is a constant fraction of the optimal token probability. In likelihood-aligned regimes, where the optimal continuation is locally competitive under the reference policy, the optimal token probability is of the same order as the local maximum probability. Thus $\hat{p}_{\max,t}(s_{t-1})$ provides a scale-adaptive proxy for the unknown quantity $1/C_t^\star(x)$, while avoiding the need to know either $C_t^\star(x)$ or the optimal token $a_t^\star$.

This empirical rule preserves the main purpose of the oracle threshold: it filters out low-frequency tail tokens whose empirical probabilities are small relative to the local mode, while retaining multiple plausible high-probability continuations. In our experiments, we set $\gamma = 0.3$ and refer to this implementation as Empirical CF-Beam*. Figures 2 and 3 show that Empirical CF-Beam achieves performance comparable to, and sometimes slightly better than, Oracle CF-Beam.*

Experimental protocol. In all experiments, we use an LLM simulator specified by the collection of conditional next-token distributions $\{\pi_{\mathrm{ref}}(\cdot \mid s, x)\}_{s \in \mathcal{A}^{\le L-1}}$, which fully characterizes the rollout distribution $\pi_{\mathrm{ref}}(\cdot \mid x)$ over trajectories $y \in \mathcal{A}^L$ for a fixed prompt x. The optimal output is defined as $y^\star := \arg\max_{y \in \mathcal{A}^L} \pi_{\mathrm{ref}}(y \mid x)$. Following the setting of Reinforcement Learning with Verifiable Reward (RLVR), we define the ground-truth reward $r^\star(x, y^\star) = 1$ and $r^\star(x, y) = 0$ for all $y \in \mathcal{A}^L \setminus \{y^\star\}$, and evaluate each algorithm by its accuracy in identifying $y^\star$. The noisy reward model $\hat{r}$ with error level $\epsilon_{\mathrm{RM}}^2 = 0.01$ is implemented as symmetric label noise: $\hat{r}(x, y)$ flips $r^\star(x, y)$ independently with probability 0.01. For each instance, we run each algorithm for 300 independent trials and report the average accuracy of returning the correct answer.

Construction of the LLM simulator. We construct a history-dependent token policy $\pi_{\mathrm{ref}}(\cdot \mid s, x)$ over an action set $\mathcal{A}$ with horizon L, optimal-suboptimal gap κ', and target optimal-trajectory probability $p = \frac{1}{C(x)} \in (0,1)$. The construction ensures: (i) there exists a unique optimal trajectory $s_L^\star$ with $\pi_{\mathrm{ref}}(s_L^\star | x) = p$, and (ii) for every state s_{t-1} at depth $t-1$, the best action $a_t^\star(s_{t-1})$ satisfies the log-gap condition $\log \pi_{\mathrm{ref}}(s_t^\star \mid x) - \log \pi_{\mathrm{ref}}(s_t \mid x) \ge 7\kappa'$ for all $s_t \ne s_t^\star$. To be specific, we first generate the optimal trajectory's distribution $\{C_t^\star(x)\}_{t=1}^L$, and then generate the suboptimal action's distribution at each time step $t \in [L]$. The key steps are illustrated as follows.

Step 1 (optimal path). Since the probability of generating the correct answer using one single rollout is $1/\overline{C}$, the geometric mean per-step probability along the optimal trajectory is $(1/\overline{C})^{1/L}$. To add randomness towards the language model, we draw $(\xi_1, \ldots, \xi_L)$ from a zero-sum Gaussian distribution (e.g., $\tilde{\xi}_t \sim \mathcal{N}(0, \sigma^2)$ i.i.d. and $\xi_t = \tilde{\xi}_t - \frac{1}{L}\sum_{j=1}^L \tilde{\xi}_j$), and set

$$\log \frac{1}{C_t^\star(x)} = \frac{1}{L}\log p + \xi_t, \qquad t = 1, \ldots, L.$$

The larger the value of σ^2, the greater the variability in per-step difficulty.

Step 2 (suboptimal actions distribution). Since the optimal trajectory is set, we now focus on designing the distribution for suboptimal actions over optimal and suboptimal trajectories. First, for any suboptimal trajectories $s_{t-1} \neq s^\star_{t-1}$, $t \in [L]$, we assign the probability of generating local best action with $\pi(a^\star_t(s_{t-1}) \mid s_{t-1}, x) = \pi(a^\star_t \mid s^\star_{t-1}, x) = \frac{1}{C^\star_t(x)}$. Second, for suboptimal actions on both optimal and suboptimal trajectories, we sample their probability mass vector using a Dirichlet distribution:

$$(\pi_{\mathrm{ref}}(a \mid s_{t-1}, x))_{a \in \mathcal{A} \setminus \{a^\star(s_{t-1})\}} \sim \mathrm{Dirichlet}(\alpha \mathbf{1}),$$

where α controls the concentration of the suboptimal action distribution. Finally, we normalize the distribution over suboptimal actions such that $\sum_{a \neq a^\star_t(s)} \pi_{\mathrm{ref}}(a \mid s, x) = 1 - \frac{1}{C^\star_t(x)}$ and it does not violate our unique optimal trajectory assumption. Therefore, given the basic parameters $|\mathcal{A}|$, L and p, together with the hyperparameters σ and α, we construct next-token distributions $\pi_{\mathrm{ref}}(\cdot \mid s, x)$ for all prefixes s of length at most $L-1$, thereby simulating the LLM sampling process.

Parameter setup.

Figure 2: $|\mathcal{A}|$=100, L=10, $p = [0.01, 0.05, 0.3]$, $\kappa = 0.01$, $\alpha = 0.5$, $\sigma = 1$.
Figure 3: $|\mathcal{A}|$=100, $L \in [2, 40]$, $p = 0.8^L$, $\kappa = 0.01$, $\alpha = 0.1$, $\sigma = 1$.

Detailed empirical findings. We expand here on the discussion in Section 5 with a finer-grained, per-panel analysis of Figures 2 and 3.

Difficulty experiment (Figure 2). In the hardest setting ($p = 0.01$, Figure 2a), both Oracle CF-Beam and Empirical CF-Beam achieve substantially higher accuracy than the sequence-level parallel baselines (Best-of-N, Majority Voting, Best-of-Majority) and Vanilla-Beam, especially under limited budgets. This is precisely the regime where sparse coverage and finite-sample tail fluctuations are most pronounced, and hence where the failure mode identified in Theorem 1 is most amplified. As the problem becomes easier (Figure 2b), the performance gap narrows: the sequence-level baselines catch up at moderate-to-large budgets, and the improvement of CF-Beam over Vanilla-Beam becomes more moderate. In the easiest regime ($p = 0.3$, Figure 2c), most methods achieve comparable performance, and Vanilla-Beam nearly matches CF-Beam once the budget is sufficiently large. These trends suggest that the confidence filter is most beneficial in hard, sparse-coverage instances, while its marginal gain decreases when the base policy is already accurate enough for the optimal prefixes to be reliably sampled. This is also consistent with the gap between the sequence-level coefficient $\overline{C}$ and the worst-step coefficient $C^\star_{\max}$ shown in each panel: as p increases, this gap becomes less pronounced, reducing the potential advantage of CF-Beam.

Horizon-length experiment (Figure 3). We consider a setting where, at each step, the model selects the correct action with mean probability 0.8. As the horizon grows, the accuracy of Best-of-N, Best-of-Majority, and Majority Voting drops rapidly, whereas the beam-based methods are more robust. Among them, the CF-Beam variants consistently maintain the highest accuracy, indicating that confidence filtering further improves the stability of beam search in long-horizon regimes. This is consistent with our theory, which shows that the dominant sample-complexity term for CF-Beam grows only polynomially in the horizon length L. In contrast, sequence-level baselines can exhibit exponential dependence on L, causing their performance to degrade more rapidly as the horizon increases.

Remark 5 (Finite-budget performance of Empirical CF-Beam)**.** *It may seem counterintuitive that, in Figure 3, Empirical CF-Beam slightly outperforms Oracle CF-Beam. Here,* oracle *refers to knowledge of the optimal-token probability used to set the threshold; the theory-prescribed choice need not maximize accuracy at every finite budget. Writing $p^\star_t := \pi_{\mathrm{ref}}(a^\star_t \mid s^\star_{t-1}, x)$, the oracle threshold $(1 - 1/L)p^\star_t$ leaves a margin of only $p^\star_t/L$ below the mean empirical frequency. As L grows, this shrinking margin makes the filter more sensitive to downward sampling fluctuations. The sufficient condition in Corollary 1 includes $N \gtrsim C^\star_{\max} L^2 \log(2L/\delta)$ to control filtering errors, whereas the horizon experiment fixes the total sampling budget and therefore cannot increase the per-prefix sample size at this rate. With $\gamma = 0.3$, the empirical threshold $\gamma \hat{p}_{\max,t}(s^\star_{t-1})$ can provide a lower cutoff when $\hat{p}_{\max,t}(s^\star_{t-1})$ tracks $p^\star_t$. This reduces the risk of premature elimination, while potentially admitting more suboptimal candidates, and provides a plausible explanation for the observed empirical advantage.*

H.2 Additional Experiments over LLMs

To complement the simulator experiments, we evaluate CF-Beam on a matrix-multiplication task using a pretrained LLM. This experiment examines whether the empirical confidence filter improves response accuracy under a comparable sampling budget when both the reference policy and the reward model are implemented by neural networks.

Experimental setup. We evaluate Qwen3-1.7B-Base Yang et al. (2025) on 500 synthetic 4×4 integer matrix-multiplication problems. All methods sample from the same reference policy π_{ref}, given by the base model at sampling temperature 1.3. Completed responses are scored by AceMath-7B-RM (Liu et al., 2025b), which supplies the outcome reward for methods that use reward-based selection. We compare Empirical CF-Beam with the same four baselines used in the simulator experiments: Best-of-N, Majority Voting, Best-of-Majority, and Vanilla-Beam. Both beam methods follow Algorithm 1, with per-prefix sample size $N = 12$ and beam width $b = 2$. For Empirical CF-Beam, we use the prefix-adaptive threshold $\beta_t(s) = \gamma \hat{p}_{\max,t}(s)$ with $\gamma = 0.3$, as defined in the preceding subsection. We report the accuracy of the final selected response and the sampling budget, using the token-level policy-query accounting defined in the main text.

Table 2: Comparison on 500 synthetic 4×4 integer matrix-multiplication problems using Qwen3-1.7B-Base. Beam methods use $N = 12$ and $b = 2$; Empirical CF-Beam uses $\gamma = 0.3$. Budgets are reported in thousands of token-level policy queries.

	Best-of-N	Majority Voting	Best-of-Majority	Vanilla-Beam	CF-Beam
Accuracy	0.332	0.304	0.326	0.372	**0.388**
Budget ($\times 10^3$)	16.8	16.8	16.8	18.6	**16.2**

Comparison with baselines. Table 2 shows that Empirical CF-Beam achieves the highest accuracy among the evaluated methods, reaching 0.388 with a sampling budget of 16.2k. This improves on Vanilla-Beam by 1.6 percentage points while using a smaller budget (16.2k versus 18.6k). Relative to the strongest sequence-level baseline, Best-of-N, CF-Beam improves accuracy by 5.6 percentage points, again with a slightly smaller sampling budget. These results provide evidence that confidence filtering can improve the accuracy–budget tradeoff in this LLM setting, consistent with the trends observed in the simulator experiments.

Ablation over the filtering threshold and beam width. We further vary $\gamma \in \{0, 0.15, 0.3, 0.45, 0.6, 0.75, 0.9\}$ and $b \in \{2, 4, 8\}$ while keeping $N = 12$ fixed. Here, $\gamma = 0$ disables confidence filtering and recovers Vanilla-Beam. Since the response lengths and reward-model errors depend on the chosen model and task, this ablation focuses on the two algorithmic parameters γ and b.

Table 3: Accuracy under different beam widths b and filtering coefficients γ on the same LLM task, with $N = 12$. The column $\gamma = 0$ corresponds to Vanilla-Beam. Bold entries mark the highest accuracy for each beam width.

	$\gamma = 0$	0.15	0.3	0.45	0.6	0.75	0.9
$b = 2$	0.372	0.378	**0.388**	0.372	0.318	0.292	0.290
$b = 4$	0.482	**0.512**	0.472	0.392	0.374	0.330	0.302
$b = 8$	0.536	**0.550**	0.480	0.434	0.364	0.292	0.300

Table 3 shows that the best accuracy is attained at a moderate filtering coefficient: $\gamma = 0.3$ for $b = 2$ and $\gamma = 0.15$ for $b = 4$ and $b = 8$. At each beam width, the best filtered configuration improves on Vanilla-Beam. Larger thresholds lead to substantially lower accuracy, consistent with overly aggressive filtering discarding useful continuations. These observations reflect the tradeoff between removing low-frequency tail candidates and preserving plausible reasoning trajectories. Increasing the beam width also improves the best observed accuracy, from 0.388 at $b = 2$ to 0.512 at $b = 4$ and 0.550 at $b = 8$. This comparison holds the per-prefix sample size fixed; maintaining a wider beam generally requires a larger total sampling budget.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The Vanilla-Beam lower bound, nearly linear sufficient coverage dependence for CF-Beam, and search-induced regret guarantee are stated in Theorems 1 and 2 and Corollary 1. The simulator results appear in Section 5, with an additional LLM evaluation in Appendix H.2. The prefix-competitiveness condition and the reward-model contribution are stated with the guarantees.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See the last section in the main paper - conclusion and limitations

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Assumptions 1–3 are stated explicitly. Appendix E provides the regret upper-bound and corollary proofs, including the reward-free final-selection argument, with the required binomial moment bounds in the Auxiliary Lemmas section.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [No]

Justification: Appendix H describes the simulator and the main settings of the LLM experiment. The latter does not yet specify the exact problem-generation and prompting procedure, model checkpoint revisions, random seeds, or the sampling configurations of the sequence-level baselines, which are needed for faithful reproduction.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The currently available supplementary archive contains the manuscript source and figures, but no scripts or generated problems for reproducing the simulator and LLM experiments. The two pretrained model checkpoints are publicly accessible, as specified in Appendix H.2.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [No]

   Justification: Appendix H reports the simulator parameters and identifies the LLM, reward model, task size, sampling temperature, beam settings, and filtering coefficients. The LLM evaluation still lacks the exact prompt and task-generation details, model revisions, and sequence-level baseline settings; no model training is performed in this work.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: The simulator figures display shaded standard-error bands over 300 trials, but Tables 2 and 3 report accuracies on 500 problems without confidence intervals or another measure of uncertainty.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Appendix H reports token-level sampling budgets, but does not state the CPU/GPU type, memory, or runtime for either the simulator or the pretrained-LLM inference experiment.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work combines theory, simulator experiments, and evaluation on publicly available pretrained models using synthetic arithmetic problems. It involves no human subjects or personal data and releases no trained model.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The paper studies a general inference-time search method and evaluates it on simulated and synthetic arithmetic tasks. It does not introduce a deployed system or release a new model; broader downstream applications are outside the scope of the reported experiments.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The work uses existing publicly available model checkpoints but does not release new model weights, scraped data, or a deployment that would require release safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Appendix H.2 credits Qwen3-1.7B-Base (Yang et al., 2025) and AceMath-7B-RM (Liu et al., 2025b), links their checkpoints, and identifies their Apache 2.0 and CC BY-NC 4.0 licenses, respectively. AceMath is used for noncommercial research evaluation.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper introduces an inference algorithm but does not release a new model or standalone dataset. The synthetic arithmetic problems are generated for evaluation; the existing pretrained models are credited under the preceding item.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: the paper involves no crowdsourcing and no human subjects; all experiments are simulator-based.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: the paper involves no human subjects or study participants, so IRB approval is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Appendix H.2 identifies Qwen3-1.7B-Base as the sampled reference policy and AceMath-7B-RM as the final-response reward model in the LLM evaluation. These pretrained models are evaluation components; the theoretical results do not depend on either specific checkpoint.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.