Think Faster Than Words: Efficient LLM Chain-of-Thought Reasoning via Dynamic Shortcut Decoding

Anonymous ACL submission

Abstract

This paper proposes Shortcut Decoding, an efficient framework for accelerating Chain-of-Thought (CoT) reasoning in Large Language Models (LLMs). Existing methods that prune or employ early stopping to reduce latency often compromise reasoning reliability. Motivated by the observation that LLMs frequently converge to correct solutions internally before completing explicit textual reasoning, we propose a dual-signal adaptive controller that integrates lightweight probes over internal hidden states with step-level entropy. This controller detects reasoning convergence during generation and adaptively selects between a fast exit path and a stability-verified path to remove redundant steps while preserving answer correctness. Experiments across multiple mathematical reasoning benchmarks demonstrate that Shortcut Decoding reduces token usage by approximately 35%, maintains accuracy comparable to full CoT decoding, and maintains final-answer accuracy comparable to the full CoT baseline, outperforming existing early-stopping methods without updating the base model. Our code is available at https://anonymous.4open.science/r/test-15A.

1 Introduction

Chain-of-Thought (CoT) prompting and its structured variants have fundamentally enhanced the reasoning capabilities of Large Language Models (LLMs) by explicitly decomposing complex problems (Wei et al., 2022; Wang et al., 2023; Kojima et al., 2022; Yao et al., 2023a; Zhou et al., 2023). While recent Large Reasoning Models (LRMs), such as OpenAI-o1 and DeepSeek-R1, scale test-time compute to achieve competition-level performance (OpenAI, 2024; DeepSeek-AI, 2025), this paradigm incurs a substantial computational cost. Empirical evidence suggests that LRMs frequently *overthink*: they continue to generate redundant checks or digressions long after the core reasoning is internally resolved, sometimes even drifting from a correct solution (Wei et al., 2025; Yang et al., 2025; Li et al., 2024). A full CoT rollout typically enumerates all reasoning steps from (1) to (n) before emitting the final answer, even when a substantial portion of the later steps is logically redundant. While system-level optimizations can alleviate latency (Kwon et al., 2023; Dao et al., 2022; Leviathan et al., 2023), they do not address this algorithmic redundancy.

Our work addresses the critical challenge of automatically truncating these redundant segments without compromising accuracy. Prior efforts to mitigate this efficiency bottleneck have primarily relied on system-level optimizations (Dao et al., 2022; Leviathan et al., 2023) or on model compression techniques such as knowledge distillation (Yu et al., 2024; Li et al., 2025). However, these approaches often require expensive retraining or fail to dynamically adapt to the varying difficulty of individual queries. While heuristic-based early stopping (e.g., checking output entropy) offers a training-free alternative (Mao et al., 2025; Laaouach, 2025), it frequently suffers from the confident error problem, where models maintain low uncertainty even while hallucinating.

To overcome these limitations, we distinguish our approach by grounding it not on speculative heuristics, but on the **empirical phenomenon** supported by recent probing studies (Zhang et al., 2025; Turpin et al., 2023): LLMs exhibit a *thinking faster than they speak* behavior. Specifically, the high-dimensional internal representation often converges to the correct answer well before the textual generation concludes. Motivated by this misalignment between internal belief saturation and external realization, we propose a **Shortcut Decoding** framework. Prior research indicates that explicit CoT text does not always faithfully reflect the model's latent state (Lyu et al., 2023; Lightman et al., 2024; Turpin et al., 2023; Manakul et al.,

2023). While external signals such as semantic entropy quantify confusion, they often fail to detect "confidently wrong" errors (Liu et al., 2025; Farquhar et al., 2024; Kossen et al., 2024; Mao et al., 2025; Laaouach, 2025; Li et al., 2025). Conversely, internal probes on hidden states can effectively predict reasoning correctness well before the generation concludes (Fu et al., 2025; Shen et al., 2025; Xu et al., 2025).

Utilizing these insights, we define a task-specific reasoning convergence point as an intermediate step at which the model's hidden states already capture a correct solution, even though the model may continue generating additional CoT text subsequently. Our goal is to detect such a step $S_{i^\star}$ during decoding and then switch directly to final answer generation, thereby skipping the remaining reasoning steps. To achieve this, we propose a dual-signal early-exit framework built atop a frozen reasoning model, without updating the base model parameters. After each reasoning step S_i, the controller collects two complementary signals. The first is an internal confidence score $S_{\text{probe}}(S_i)$, predicted from the step representation by a lightweight MLP probe g_ϕ. The second is an external uncertainty score $\bar{H}_i$, computed as the step-averaged output entropy. These two signals are jointly used to determine whether to terminate reasoning. Specifically, a fast exit is triggered when either $S_{\text{probe}}(S_i)$ is very high or $H_{\text{avg}}(S_i)$ reaches an extreme low level, indicating strong convergence. When $S_{\text{probe}}(S_i)$ is high but $\bar{H}_i$ remains moderate, a stable exit strategy is applied, which requires the signals to remain consistent over multiple consecutive steps before exiting.

Our contributions are summarized as follows:

- Empirical Validation of "Thinking Faster than Speaking" Hypothesis: We provide empirical evidence that the internal hidden states of LLMs offer predictive signals for final correctness significantly earlier than explicit CoT completion, laying a foundation for early stopping based on internal states.

- Entropy-Probe Dual-Signal Framework: We propose a novel framework that unifies internal semantic consistency and external uncertainty. This dual-signal approach enables robust and adaptive early stopping, addressing the limitations of relying solely on entropy or internal probes.

- Adaptive Controller Design: An adaptive controller dynamically evaluates internal probe scores and step-level entropy during reasoning. It takes different actions based on the reasoning state (e.g., rapid convergence, gradual convergence, or confusion), achieving efficient inference under strict accuracy constraints.

- Significant Efficiency-Accuracy Trade-Off: Extensive experiments on mathematical reasoning benchmarks show that our method substantially reduces token usage while maintaining or improving accuracy, outperforming state-of-the-art dynamic early-exit baselines (Wei et al., 2025; Yang et al., 2025; Li et al., 2024). This effectively mitigates semantic drift and redundancy in long CoT reasoning.

2 Related Work

To address the high computational cost and overthinking behavior of LLMs on complex reasoning tasks, prior work has mainly followed two directions: (i) explicit self-correction and structured CoT methods that improve robustness at the cost of longer reasoning; and (ii) dynamic, training-free early stopping that intervenes during inference without changing base model parameters.

2.1 Self-Correction and Structured CoT

A primary line of research enhances reasoning robustness by structuring the generation process. Agentic and reflective paradigms, such as ReAct and Reflexion, organize reasoning into iterative "act-and-reflect" cycles to revise answers based on feedback (Yao et al., 2023b; Shinn et al., 2023). Similarly, structured prompting schemes, including Self-Consistency, Tree-of-Thoughts, Least-to-Most, and step-wise verification, explicitly decompose problems or explore diverse reasoning paths to ensure reliability (Wang et al., 2023; Yao et al., 2023a; Zhou et al., 2023; Kojima et al., 2022; Lyu et al., 2023; Lightman et al., 2024).

However, these extensive reasoning chains face two critical issues: faithfulness and efficiency. From a faithfulness perspective, models often generate explanations decoupled from their internal states, and self-correction is not guaranteed to rectify logical errors (Lyu et al., 2023; Turpin et al., 2023; Manakul et al., 2023). From an efficiency perspective, empirical studies show that accuracy

gains often saturate while models overthink or spiral into redundancy (Wei et al., 2025; Yang et al., 2025; Li et al., 2024). Recent latent-space analyses suggest a solution: essential reasoning logic is often compressed in hidden states well before the textual CoT concludes, as demonstrated by probe-based and continuous CoT studies (Fu et al., 2025; Shen et al., 2025; Xu et al., 2025).

2.2 Dynamic Inference-Time Intervention and Early Stopping

A second direction focuses on dynamic intervention: monitoring the evolving CoT and terminating generation once convergence is detected (Wei et al., 2025; Yang et al., 2025; Li et al., 2024). These methods typically rely on two types of signals.

The first group utilizes surface-level patterns in the generated text. Approaches like Dynamic Early Exit and Escape Sky-high Cost monitor stability or specific markers (e.g., "Wait") to truncate redundant explanations (Yang et al., 2025; Li et al., 2024). While training-free, these rely on handcrafted heuristics that may not be generalized.

The second group leverages uncertainty and entropy. Step-level entropy methods aggregate next-token distributions to gauge confidence, triggering early exits when entropy remains low (Mao et al., 2025; Laaouach, 2025; Li et al., 2025). Broader research on semantic entropy further quantifies hallucination risks (Liu et al., 2025; Farquhar et al., 2024; Kossen et al., 2024). However, a key limitation is the "confidently wrong" phenomenon: low entropy does not guarantee objective correctness (Turpin et al., 2023). Although early-exit mechanisms have been successfully applied in encoder-style or adaptive architectures (Xin et al., 2020; Schuster et al., 2022), adapting them to long-form CoT requires more robust signals.

Our framework addresses these limitations by combining external uncertainty with internal hidden-state probes (Fu et al., 2025), creating a unified controller that distinguishes true convergence from confident errors.

3 Methodology

This section proposes a Shortcut Decoding framework, a plug-and-play method designed to accelerate the reasoning of LLMs by pruning redundant Chain-of-Thought (CoT) steps, without requiring fine-tuning model parameters. An overall framework is shown in Figure 1. The framework constructs a three-layer hierarchical architecture atop the frozen base model.

The bottom layer is the Logical Step Segmentation Layer. In this layer, the model generates the chain-of-thought in an autoregressive manner, while the system dynamically segments the continuous token stream into distinct logical reasoning steps. The middle layer is the Signal Extraction Layer, which operates at the boundary of each logical step to capture multi-dimensional indicators, including the internal cognitive state obtained via hidden state probes and the external uncertainty obtained via output entropy. The top layer is the Adaptive Decision Layer, where a dual-track controller evaluates these fused signals to identify the "Reasoning Completion Point" (RCP) and determines whether to immediately truncate the generation process.

3.1 Problem Setting

We consider a large reasoning language model f_θ that takes a natural language question q as input and produces a chain-of-thought followed by a final answer as output. The generated token sequence can be written as

$$Y = (y_1, y_2, \ldots, y_N), \tag{1}$$

where the prefix corresponds to the chain-of-thought (CoT) and the tail corresponds to the final answer. The model defines a conditional distribution over output sequences and generates them in an autoregressive fashion:

$$p_\theta(Y \mid q) = \prod_{t=1}^{N} p_\theta(y_t \mid y_{<t}, q). \tag{2}$$

During generation, the model maintains a sequence of hidden states $\{h_t\}_{t=1}^{N}$, where $h_t \in \mathbb{R}^d$ denotes the final-layer hidden representation at position t. At each position, a linear projection followed by a softmax maps h_t to a probability distribution over the vocabulary V,

$$p_t(v) = p_\theta(y_t = v \mid y_{<t}, q), \quad v \in V, \tag{3}$$

which is used both for sampling the next token and for constructing uncertainty-based signals in later sections.

To make early-stopping decisions at a more meaningful granularity than single tokens, we segment the CoT prefix into a sequence of *logical steps*:

$$C = (S_1, S_2, \ldots, S_L), \tag{4}$$

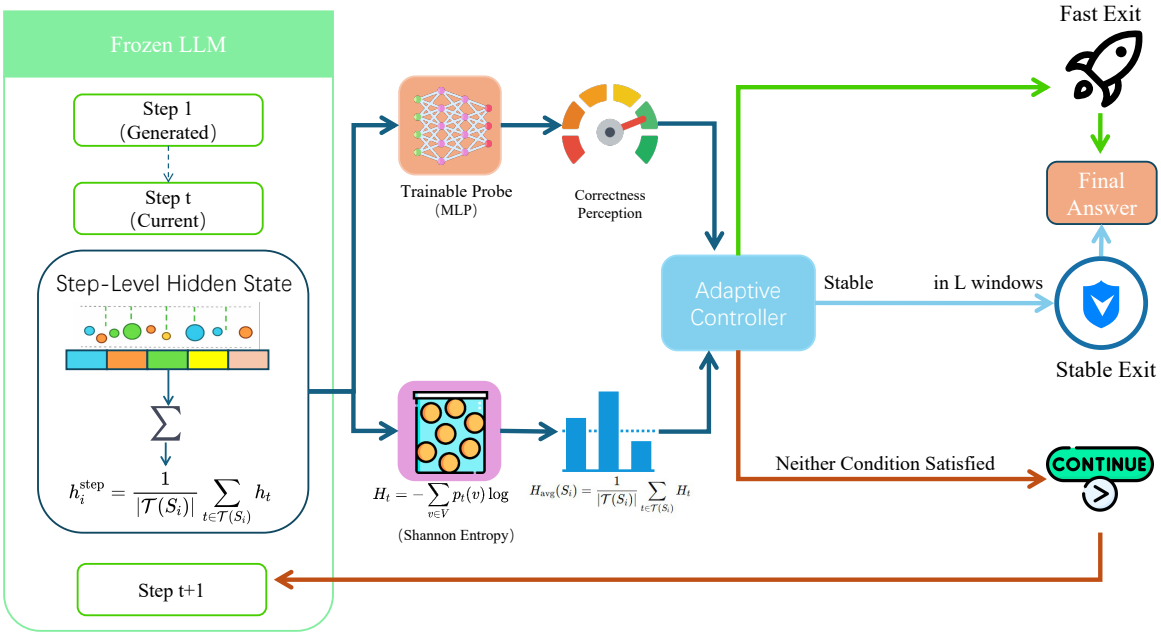

Figure 1: Overall architecture of our shortcut decoding framework. At each reasoning step, the controller reads the step-average entropy and the probe score, and chooses among fast early stop, stable early stop, or continuing CoT generation.

where each S_i is a contiguous subsequence of tokens corresponding to one interpretable reasoning step. To facilitate online boundary detection, we adopt a unified prompting template that encourages the model to write CoT in a numbered format such as "(1) ... (2) ... (3) ...". During generation, whenever the output stream contains a "newline + step index" pattern (e.g., "`\n(2)`"), we treat the tokens from the previous index up to that newline as a complete step S_i, and only at such step boundaries do we compute step-level signals and consider early-stopping decisions.

3.2 Entropy-Based External Signal and Training-Free Early Stopping

We use output-distribution entropy as an external uncertainty signal. At decoding position t, the model yields a next-token distribution p_t over the vocabulary V, and the token entropy is

$$H_t = -\sum_{v \in V} p_t(v) \log p_t(v). \tag{5}$$

To reduce token-level noise, we aggregate entropy at the reasoning-step level. Let $\mathcal{T}(S_i)$ be the set of token positions in step S_i. The step-average entropy is

$$H_{\text{avg}}(S_i) = \frac{1}{|\mathcal{T}(S_i)|} \sum_{t \in \mathcal{T}(S_i)} H_t. \tag{6}$$

Entropy-only baseline. During online decoding, at each step boundary we maintain (i) current CoT token usage under a per-problem thinking budget, and (ii) a window of the most recent step entropies. We trigger early stopping at step S_i if (a) the remaining budget is sufficient for final answer generation and (b) all step-average entropies in the recent window are below a fixed threshold. Once triggered, we terminate CoT, treat the current history as the reasoning context, and switch to an answer-only prompt to produce the final answer.

This baseline is training-free and model-agnostic, but entropy mainly reflects confidence rather than correctness: the model may follow an incorrect premise with consistently low entropy, causing premature exits on "confident-but-wrong" trajectories. This motivates adding an internal signal that better correlates with reasoning correctness and completion.

3.3 Adaptive Early Stopping with Entropy and Probe Signals

As shown in Figure 1, our framework keeps the base model frozen and adds an Adaptive Controller that decides whether to stop or continue after each reasoning step S_i. The controller takes two step-level signals as input, an internal probe score and an external entropy score.

For each step S_i, we first construct a step-level

hidden representation by averaging the final-layer hidden states within the step

$$h_i^{\text{step}} = \frac{1}{|\mathcal{T}(S_i)|} \sum_{t \in \mathcal{T}(S_i)} h_t, \tag{7}$$

which summarizes the internal reasoning state after completing step S_i. On top of h_i^{step}, we attach a lightweight probe g_ϕ implemented as a small multilayer perceptron

$$s(S_i) = g_\phi\big(h_i^{\text{step}}\big), \tag{8}$$

and obtain a step-level probe score through a sigmoid

$$S_{\text{probe}}(S_i) = \sigma\big(s(S_i)\big), \tag{9}$$

which estimates whether the current reasoning state is sufficient for safely producing the final answer. The probe is trained with step-level labels while the base model f_θ remains frozen. In parallel, the entropy branch provides the external uncertainty signal $H_{\text{avg}}(S_i)$ defined in Section 3.2.

At inference time, the Adaptive Controller reads the pair $\big(H_{\text{avg}}(S_i), S_{\text{probe}}(S_i)\big)$ at each step boundary and chooses among Fast Exit, Stable Exit, and Continue, matching the three outgoing paths in Figure 1. Fast Exit targets highly confident steps and triggers immediate termination whenever either condition holds

$$H_{\text{avg}}(S_i) \le \theta_{\text{entropy\_extreme}}, \tag{10}$$

or

$$S_{\text{probe}}(S_i) \ge \theta_{\text{aha}}. \tag{11}$$

When Fast Exit is triggered, we stop CoT generation at step S_i and directly generate the Final Answer from the current context.

Stable Exit handles gradual convergence, where $S_{\text{probe}}(S_i)$ is high but $H_{\text{avg}}(S_i)$ is not yet at an extreme low level. To avoid reacting to transient fluctuations, the controller requires step-level consistency over a window of length K and triggers Stable Exit only when the probe scores remain above θ_{stable} and the recent entropies stay in a medium-to-low range for consecutive steps. If neither Fast Exit nor Stable Exit is activated, the controller selects Continue and the Frozen LLM generates the next reasoning step, after which the signals are recomputed and the decision is repeated.

4 Experiments

4.1 Experimental Setup

Datasets and Models To assess model performance across reasoning tasks of varying difficulty, we selected three representative benchmarks: MATH-500 (covering algebra, geometry, and other competition-level domains), GSM8K (grade school math word problems), and AIME 2024/2025 (high-difficulty competition problems). For the models, we employed DeepSeek-R1-7B, DeepSeek-R1-14B, and Qwen3-8B. These models possess strong Chain-of-Thought (CoT) generation capabilities, allowing us to verify the method's generalizability across different scales and architectures.

Baselines We compare our proposed "Entropy-Probe Dual-Signal" method against several baselines: (1) Standard CoT (Baseline); (2) No-thinking (direct answer generation), serving as a performance lower bound; and (3) state-of-the-art dynamic early-stopping methods, DEER (Yang et al., 2025) and Dynasor (Fu et al., 2025). Additionally, we examine ablation variants, Pure Entropy and Pure Probe, to validate the necessity of the dual-signal design.

Evaluation Metrics The evaluation metrics primarily include final answer Accuracy (ACC), Average Token Consumption (Avg Tokens), and Compression Ratio (CR). The compression ratio quantifies efficiency gains and is defined as:

$$\text{CR} = \frac{\sum_{i=1}^{N} T_{\text{early}}^{(i)}}{\sum_{i=1}^{N} T_{\text{full}}^{(i)}} \tag{12}$$

where $T_{\text{early}}^{(i)}$ and $T_{\text{full}}^{(i)}$ represent the token counts for the early-stopped and full reasoning paths, respectively. A lower CR value indicates greater computational savings.

Inference Implementation Strategy For evaluation, we run the early-exit decoding and the full-CoT decoding separately under the same prompts and decoding settings. When simulating an early exit, we reuse the KV cache at the selected step boundary to start answer-only decoding from the same prefix, ensuring a fair comparison without information leakage. All token statistics reported for efficiency are computed on the early-exit path only.

Method	GSM8K			MATH-500			AIME 24			AIME 25		
	Acc (%)↑	Tokens↓	CR↓	Acc (%)↑	Tokens↓	CR↓	Acc (%)↑	Tokens↓	CR↓	Acc (%)↑	Tokens↓	CR↓
DeepSeek-R1-Distill-Qwen-7B												
Baseline	89.5%	1512.3	100.0%	90.8%	3661.9	100.0%	43.3%	10611.3	100.0%	36.7%	10309.0	100.0%
No-thinking	86.9%	**302.1**	**20.0%**	80.1%	**585.7**	**16.0%**	10.0%	**2289.5**	**21.6%**	10.0%	**1755.3**	**17.0%**
Dynasor	89.5%	1293.3	85.5%	87.7%	2896.3	79.1%	43.3%	10231.6	96.4%	36.7%	9125.4	88.5%
DEER	89.8%	1167.9	77.2%	87.8%	2347.3	64.1%	43.3%	10611.3	100.0%	33.3%	11432.8	111.0%
Ours	**90.6%**	1036.5	68.5%	**91.2%**	2193.5	59.9%	**53.3%**	6583.5	62.0%	**40.0%**	5937.0	57.6%
DeepSeek-R1-Distill-Qwen-14B												
Baseline	**93.2%**	1519.6	100.0%	92.1%	3992.1	100.0%	51.7%	11252.1	100.0%	36.7%	12103.0	100.0%
No-thinking	91.2%	**293.9**	**19.3%**	86.4%	**1321.8**	**33.1%**	26.7%	**5639.4**	**50.1%**	13.3%	**6128.3**	**50.6%**
Dynasor	93.1%	1242.5	81.8%	91.0%	2294.4	57.5%	46.7%	8921.9	79.3%	33.3%	9834.6	81.3%
DEER	93.0%	1183.2	77.9%	91.6%	2364.3	59.2%	53.3%	8214.2	73.0%	**43.3%**	10313.6	85.2%
Ours	**93.2%**	1003.7	66.1%	**92.4%**	2102.6	52.7%	**63.3%**	8219.3	73.0%	**43.3%**	10625.7	87.8%
Qwen3-8B												
Baseline	93.8%	2194.5	100.0%	93.4%	5216.3	100.0%	63.3%	13120.3	100.0%	53.3%	11923.8	100.0%
No-thinking	89.2%	**452.3**	**20.6%**	92.5%	**1269.6**	**24.3%**	26.7%	**3189.7**	**24.3%**	16.7%	**4323.0**	**36.3%**
Dynasor	92.4%	1823.5	83.1%	91.1%	3069.3	58.8%	60.0%	11250.6	85.7%	46.7%	9281.5	77.8%
DEER	95.3%	1662.7	75.8%	89.2%	3064.9	58.8%	63.3%	10383.1	79.1%	53.3%	11284.3	94.6%
Ours	**95.8%**	1513.3	69.0%	**96.3%**	2753.8	52.8%	**66.7%**	8123.3	61.9%	**60.0%**	8039.9	67.4%

Table 1: Accuracy–cost trade-off on GSM8K, MATH-500, AIME 24, and AIME 25. Acc is the final-answer accuracy (%), Tokens is the average number of CoT tokens generated (excluding the final answer), and CR is the compression ratio relative to the full CoT baseline (lower CR indicates fewer tokens retained, i.e., higher efficiency).

4.2 Does the Model "Think Faster Than It Speaks"?

Before building an early-stopping mechanism, we first verify the core hypothesis of this work: the internal hidden states of large reasoning models contain forward-looking signals that can predict the correctness of the final answer before the explicit chain-of-thought (CoT) text is fully generated. If this holds, then success or failure can be judged from intermediate internal representations, providing a solid basis for shortcut decoding.

We attach a lightweight MLP probe on top of the frozen base model. At the end of each logical step in the CoT, the probe reads the corresponding step-level hidden representation and is trained, under binary supervision, to predict whether the current reasoning path will eventually lead to a correct final answer. The probe output is mapped through a sigmoid to obtain an estimated probability that "continuing from this step will yield a correct answer."

Figure 2 reports the probe accuracy as a function of relative CoT progress. Even within the first $\sim$10% of the reasoning trajectory, the probe already reaches around 70–80% accuracy, well above chance, indicating that meaningful correctness signals emerge very early. More importantly, the accuracy rises further and reaches the mid-90% range around the late stage (roughly 75–85% of the CoT), after which it remains essentially stable. This suggests that before the model spends the last $\sim$15–25% of its computation generating detailed derivations and formatted explanations, its internal representations have largely locked onto the correct conclusion. These observations provide empirical evidence for the claim that the model thinks faster than it speaks, and motivate using internal states to trigger early stopping around the reasoning completion point.

4.3 Effects and Limitations of the Pure Entropy Early-Stopping Baseline

We analyze a pure entropy-based early-stopping baseline to show that external uncertainty alone is insufficient. At each logical step, we compute the average token-level Shannon entropy and apply a fixed threshold to decide whether to terminate CoT generation.

Figure 3 shows a representative MATH-500 case. Although the model reaches the correct answer at Step 3, it continues redundant verification. Step entropy oscillates: it spikes during linguistic hesitation (e.g., "Wait...") and drops during deterministic re-checks, making fixed thresholds unreliable for separating true completion from redundant continuation.

Method	GSM8K			MATH-500			AIME 24			AIME 25		
	Acc (%)↑	Tokens↓	CR (%)↓	Acc (%)↑	Tokens↓	CR (%)↓	Acc (%)↑	Tokens↓	CR (%)↓	Acc (%)↑	Tokens↓	CR (%)↓
DeepSeek-R1-Distill-Qwen-7B												
Pure Probe	90.7%	1092.3	72.2%	91.2%	2308.6	63.0%	53.3%	6732.8	63.5%	40.0%	6436.0	62.4%
Pure Entropy	88.3%	982.9	64.9%	88.1%	2089.3	57.1%	46.7%	6128.3	59.1%	36.7%	4770.0	46.3%
Ours	90.6%	1036.5	68.5%	91.2%	2193.5	59.9%	53.3%	6583.5	62.0%	40.0%	5937.0	57.6%
DeepSeek-R1-Distill-Qwen-14B												
Pure Probe	93.3%	1205.3	79.3%	92.4%	2552.7	63.9%	63.3%	7716.8	68.6%	46.7%	10935.4	90.4%
Pure Entropy	91.9%	983.5	64.7%	88.6%	2076.4	52.0%	46.7%	4993.8	44.4%	33.3%	9939.4	82.1%
Ours	93.2%	1003.7	66.1%	92.4%	2102.6	52.7%	63.3%	8219.3	73.0%	43.3%	10625.7	87.8%
Qwen3-8B												
Pure Probe	95.9%	1683.7	76.7%	95.6%	3512.8	67.3%	66.7%	11073.3	84.4%	60.0%	10935.7	91.7%
Pure Entropy	92.7%	1159.6	52.8%	92.7%	2493.1	47.8%	60.0%	7034.5	53.6%	53.3%	8064.0	67.6%
Ours	95.8%	1513.3	69.0%	96.3%	2753.8	52.8%	66.7%	8123.3	61.9%	60.0%	8039.9	67.4%

Table 2: Ablation study of different signaling mechanisms. Pure Probe relies solely on the internal hidden-state probe, Pure Entropy relies solely on the external step entropy, and Ours (dual-signal) combines both to balance efficiency and accuracy. Metrics: answer accuracy (Acc), average CoT tokens (Tokens), and compression ratio (CR).

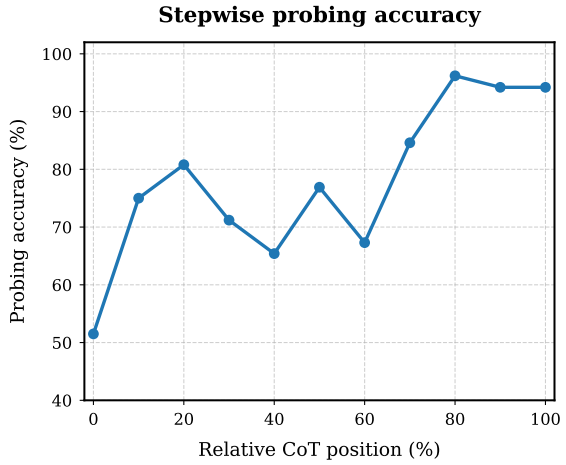

Figure 2: Stepwise probing accuracy across relative chain-of-thought positions. The probe accuracy rises quickly and saturates around 80% of the CoT, well before the final answer is written out.

Quantitatively, this baseline reduces CoT tokens by 43% but lowers accuracy from 90.8% to 88.1% (Table 2). This highlights a fundamental mismatch: step entropy reflects local next-token uncertainty rather than global reasoning completion, motivating the need for internal probe signals.

4.4 Entropy-Probe Dual-Signal Adaptive Early-Stopping Method

We now evaluate the proposed entropy-probe dual-signal adaptive early-stopping method on the main benchmarks and compare it with both the standard full CoT decoding and existing early-stopping baselines such as DEER and Dynasor-CoT.

Table 1 summarizes the performance of DeepSeek-R1-Distill-Qwen-7B, DeepSeek-R1-Distill-Qwen-14B, and Qwen3-8B on GSM8K, MATH-500, and AIME 2024/2025. Across models and datasets, the dual-signal controller achieves a favorable balance between efficiency and accuracy. For example, across models and benchmarks, our method reduces CoT tokens by roughly 31–47%, while maintaining accuracy comparable to or slightly better than full CoT decoding. This counter-intuitive improvement suggests that removing redundant late-stage reasoning not only saves computation but can also reduce the risk of logical drift and harmful self-correction in very long CoT trajectories.

Compared with existing inference-time early-stopping methods, the dual-signal controller offers both higher accuracy and stronger compression. Under comparable accuracy levels, it consistently achieves lower compression ratios than DEER and Dynasor-CoT, indicating more aggressive yet still safe truncation of redundant segments. When contrasted with the single-signal variants, the effect of signal combination becomes clearer: relative to the pure entropy baseline, the dual-signal method recovers most of the lost accuracy, especially on medium- and high-difficulty problems where confident but incorrect reasoning is common; relative to the pure probe baseline, it achieves substantially better compression by using entropy to identify segments where the model is confident at the token level and therefore can afford to exit earlier.

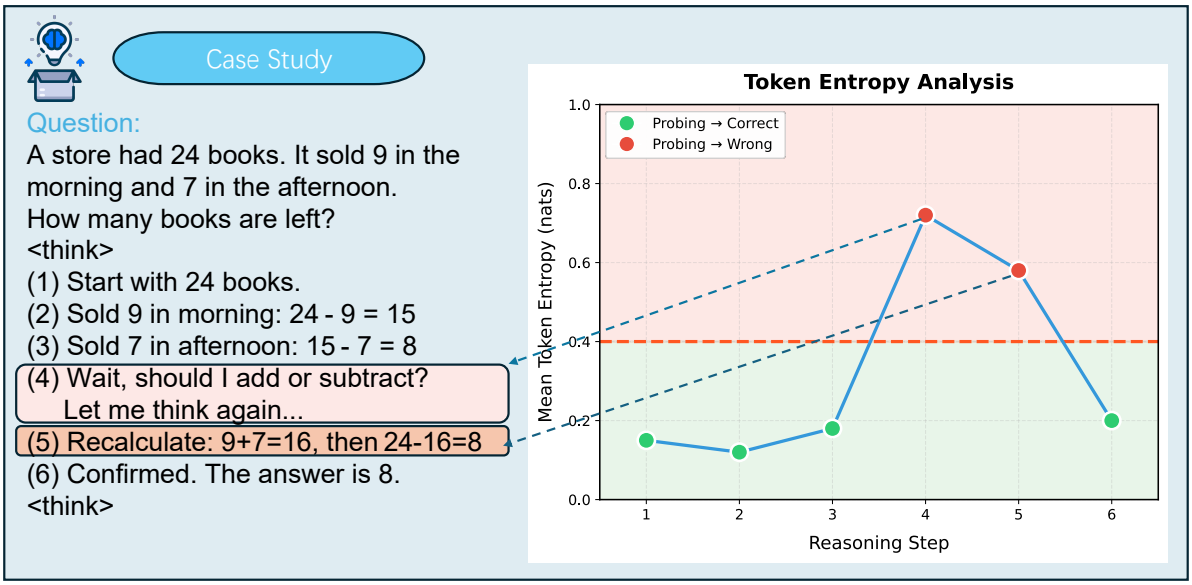

Figure 3: A typical overthinking case. The model obtains the correct answer at Step 3 but continues redundant reasoning. Step-wise entropy is high during hesitation phases and low during confident, deterministic steps, revealing the limitation of simple entropy-based stopping criteria.

Overall, these results show that combining internal probe signals with external entropy signals yields a robust and transferable early-stopping mechanism that respects accuracy constraints while significantly mitigating overthinking and high inference cost.

4.5 Ablation Studies and Error Analysis

Table 2 reports ablations with Pure Entropy and Pure Probe. Pure Entropy achieves the strongest compression but consistently harms accuracy; on MATH-500, it drops by about 2.7 points compared to full CoT, showing that uncertainty alone cannot separate true convergence from confident errors. Pure Probe largely preserves accuracy but yields weaker compression, indicating conservative stopping when relying on internal signals alone. Combining both signals yields a better balance: entropy supports earlier exits in low-uncertainty regions, while the probe helps reject semantically unreliable trajectories.

We also analyze why reducing overthinking can improve final accuracy. We split baseline failures into reasoning errors, where the model never reaches the correct answer, and overthinking errors, where a correct intermediate conclusion is later overwritten by redundant self-correction or exploration. In our analysis, overthinking errors account for a clear majority (roughly 55–65% of failures), while the remaining 35–45% are reasoning errors. This suggests many failures arise after the model has already reached a correct state. By detecting the reasoning convergence moment and truncating promptly, our controller reduces such overwriting and improves both efficiency and final-answer quality.

5 Conclusion

In this work, we address the inefficiency and overthinking issues in Large Reasoning Models (LRMs) by proposing the Shortcut Decoding framework. Grounded in the hypothesis that models think faster than they speak, our framework employs a dual-signal adaptive controller that combines internal hidden-state probes with external step-level entropy to accurately detect the reasoning convergence point. Experimental results demonstrate that our method reduces token consumption by approximately 35% while maintaining or enhancing accuracy, effectively reducing semantic drift from redundant generation. This establishes a robust paradigm for efficient reasoning.

Limitations

Our approach has limitations: it requires training task-specific probes, limiting zero-shot generalization to new tasks without additional data, and relies on white-box access to model internals, making it incompatible with closed-source systems that do not expose intermediate representations.

References

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359.

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *Preprint*, arXiv:2501.12948.

Sebastian Farquhar, Jannik Kossen, Lukas Kuhn, and 1 others. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Yichao Fu, Junda Chen, Yonghao Zhuang, Zheyu Fu, Ion Stoica, and Hao Zhang. 2025. Reasoning without self-doubt: More efficient chain-of-thought through certainty probing. In *ICLR 2025 Workshop on Foundation Models in the Wild*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems*, 35:22199–22213.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. 2024. Semantic entropy probes: Robust and cheap hallucination detection in LLMs. *Preprint*, arXiv:2406.15927.

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

Yassir Laaouach. 2025. HALT-CoT: Model-agnostic early stopping for chain-of-thought reasoning via answer entropy. In *4th Muslims in ML Workshop co-located with ICML 2025*.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. 2023. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*, pages 19274–19286.

Yiwei Li, Peiwen Yuan, Shaoxiong Feng, Boyuan Pan, Xinglin Wang, Bin Sun, Heda Wang, and Kan Li. 2024. Escape sky-high cost: Early-stopping self-consistency for multi-step reasoning. In *The Twelfth International Conference on Learning Representations*.

Zeju Li, Jianyuan Zhong, Ziyang Zheng, Xiangyu Wen, Zhijian Xu, Yingying Cheng, Fan Zhang, and Qiang Xu. 2025. Compressing chain-of-thought in LLMs via step entropy. *Preprint*, arXiv:2508.03346.

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

Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. 2025. Uncertainty quantification and confidence calibration in large language models: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 6107–6117.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful chain-of-thought reasoning. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 305–329.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017.

Minjia Mao, Bowen Yin, Yu Zhu, and Xiao Fang. 2025. Early stopping chain-of-thoughts in large language models. *Preprint*, arXiv:2509.14004.

OpenAI. 2024. OpenAI o1 system card. *Preprint*, arXiv:2412.16720.

Tal Schuster, Adam Fisch, Jai Gupta, Mostafa Dehghani, Dara Bahri, Vinh Tran, Yi Tay, and Donald Metzler. 2022. Confident adaptive language modeling. *Advances in Neural Information Processing Systems*, 35:17456–17472.

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. 2025. CODI: Compressing chain-of-thought into continuous space via self-distillation. *Preprint*, arXiv:2502.21074.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36:74952–74965.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Zihao Wei, Liang Pang, Jiahao Liu, Jingcheng Deng, Shicheng Xu, Zenghao Duan, Jingang Wang, Fei Sun, Xunliang Cai, Huawei Shen, and Xueqi Cheng. 2025. Stop spinning wheels: Mitigating LLM overthinking via mining patterns for early reasoning exit. *Preprint*, arXiv:2508.17627.

Ji Xin, Raphael Tang, Jaejun Lee, Yaoliang Yu, and Jimmy Lin. 2020. DeeBERT: Dynamic early exiting for accelerating BERT inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2246–2251.

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. 2025. SoftCoT: Soft chain-of-thought for efficient reasoning with LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23336–23351.

Chenxu Yang, Qingyi Si, Yongjie Duan, Zheliang Zhu, Chenyu Zhu, Qiaowei Li, Minghui Chen, Zheng Lin, and Weiping Wang. 2025. Dynamic early exit in reasoning models. *Preprint*, arXiv:2504.15895.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36:11809–11822.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023b. ReAct: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Ping Yu, Jing Xu, Jason Weston, and Ilia Kulikov. 2024. Distilling system 2 into system 1. *Preprint*, arXiv:2407.06023.

Anqi Zhang, Yulin Chen, Jane Pan, Chen Zhao, Aurojit Panda, Jinyang Li, and He He. 2025. Reasoning models know when they're right: Probing hidden states for self-verification. *Preprint*, arXiv:2504.05419.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V. Le, and Ed H. Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*.