

Adaptive Test-Time Compute for Efficient Reasoning in Language Models

Stanford CS224N {Custom} Project

Ryan Tan

Department of Computer Science
Stanford University
tanryan@stanford.edu

Abstract

Recent work shows that reasoning accuracy in language models improves when multiple reasoning paths are sampled and aggregated. However, fixed self-consistency allocates equal compute to all problems regardless of difficulty, leading to inefficiency. In this project, we investigate adaptive test-time compute allocation through the design and evaluation of three stopping criteria: a novel heuristic method combining consensus, margin, and stability rules with lexicographic priority; an entropy-based criterion over the full parsed vote distribution; and a Bayesian posterior confidence method using a uniform Dirichlet prior with Monte Carlo estimation. We evaluate all three against fixed self-consistency baselines on GSM8K using Qwen2.5-7B-Instruct across budget $B \in \{5, 10, 20, 40\}$ and a threshold sensitivity analysis. All three adaptive methods match the fixed self-consistency accuracy ceiling ($\approx 95\%$) while using 3.2–6.3 samples on average, compared to 40 for the fixed-budget baseline. Stopping criterion choice has little effect on aggregate accuracy when the base model is strong as all methods converge because vote distributions concentrate rapidly. However, on the hard subset of problems where greedy decoding fails ($n=22$), meaningful differences emerge: entropy-based stopping achieves a flat 36.4% hard accuracy regardless of budget while spending up to 24.86 samples, mirroring the vote-reinforcement failure of Fixed- $N=40$. Bayesian stopping peaks at 45.5% hard accuracy at $B = 10$ (7.45 avg samples). Heuristic stopping improves with budget and reaches 45.5% at $B = 40$ with only 6.18 samples, the best compute–accuracy tradeoff on hard examples. These findings clarify when stopping criterion design matters, and motivate evaluation on harder benchmarks where model uncertainty is pervasive.

1 Key Information to include

- Mentor: Arpandeeep Khatua
- External Collaborators: No
- Sharing project: No

2 Introduction

Scaling inference compute at test time has emerged as a complementary axis to training-time scaling for improving language model reasoning. Self-consistency [Wang et al., 2023] is the canonical realization of this idea that samples N independent chain-of-thought [Wei et al., 2023] trajectories and returns the plurality answer. Accuracy increases with N because diverse reasoning paths produce independent errors that cancel under majority vote, but compute is allocated identically to every input. Therefore, easy problems that the model reliably solves in one sample receive $N-1$ unnecessary draws; hard problems where the model is genuinely uncertain might benefit from far more than N .

Adaptive self-consistency addresses this by sampling sequentially and stopping as soon as a criterion fires or a maximum budget B is reached, allocating compute proportionally to per-instance difficulty. The central design question is what that stopping criterion should be. Prior work has explored heuristic rules [Aggarwal et al., 2023], information-theoretic stopping [Li et al., 2024], and Bayesian confidence [Huang et al., 2026], but a controlled comparison across all three families under identical conditions has not been conducted.

In this paper, we design novel stopping criteria in each of these families and characterize when their choice matters and when it does not. Our contributions are:

- We design three adaptive stopping criteria in a unified framework: a novel heuristic combining consensus, margin, and stability rules with lexicographic priority (ADAPTIVE); an entropy-based criterion over the full parsed vote distribution (ENTROPY); and a Bayesian posterior confidence method using a uniform Dirichlet prior with Monte Carlo estimation (BAYESIAN). All three are evaluated on GSM8K with Qwen2.5-7B-Instruct across $B \in \{5, 10, 20, 40\}$ and threshold sensitivity sweeps (§4, §5).
- We show all three methods achieve 84–91% compute reduction relative to Fixed- $N=40$ at equivalent accuracy.
- We show that on hard examples, criteria diverge substantially, and that Fixed- $N=40$ underperforms Fixed- $N=5$ due to vote reinforcement of systematic errors (§6).
- We characterize a consistent mode-vs-gold accuracy gap across all methods, revealing that the primary bottleneck is model error rather than stopping error (§6).

3 Related Work

Wei et al. [2023] demonstrated that intermediate reasoning steps substantially improve accuracy on multi-step benchmarks. Wang et al. [2023] showed that sampling multiple chain-of-thought trajectories and selecting the plurality answer outperforms any single path, establishing the fixed self-consistency baseline against which all adaptive methods are evaluated.

Aggarwal et al. [2023] proposed adaptive-consistency, which halts sampling as soon as the leading answer’s vote share exceeds a fixed threshold τ or its margin over the runner-up exceeds δ . Their single-rule criterion uses the same threshold across all problems, requiring per-model tuning, but showed that self-consistency gains saturate well before the maximum budget. Li et al. [2024] proposed ESC, which stops when a sliding window of w consecutive samples all agree. Because ESC operates on recent draws rather than the full vote distribution, it is sensitive to local streaks rather than global concentration and can fire prematurely when a wrong answer appears in a streak by chance. Huang et al. [2026] maintained a Dirichlet posterior over answer probabilities and stopped when posterior confidence in the plurality exceeds γ , showing that a three-component posterior, tracking top answer mass, second answer mass, and a collapsed “other” mass, is asymptotically optimal.

Snell et al. [2024] and Brown et al. [2024] study test-time scaling more broadly via process reward models, best-of- N sampling, and beam search, achieving higher accuracy ceilings but requiring trained verifiers. Adaptive self-consistency needs no verifier, making it appealing when only a language model is available. Our work designs novel stopping criteria within each of these families and, by evaluating them under identical conditions, addresses the open question of how they compare and when their behavior diverges.

4 Approach

4.1 Problem Setup

Let x be an input question. A language model $\mathcal{M}$ with temperature $\tau > 0$ generates stochastic chain-of-thought trajectories $y \sim \mathcal{M}(\cdot \mid x)$. Each trajectory is a natural language reasoning chain ending in a numeric answer, parsed by an extractor ϕ to produce $a = \phi(y) \in \mathcal{A} \cup \{\emptyset\}$, where $\mathcal{A}$ is the set of all possible numeric answers and $\emptyset$ denotes a parse failure.

After t draws $y_1, \dots, y_t$, define the vote count for answer a as:

$$c_t(a) = \sum_{i=1}^t \mathbf{1}[\phi(y_i) = a]$$

and let $n_t = \sum_{a \in \mathcal{A}} c_t(a)$ be the number of successfully parsed answers. The set of observed distinct answers is $\mathcal{A}_t = \{a \in \mathcal{A} : c_t(a) > 0\}$, and the current plurality answer is:

$$a_t^* = \arg \max_{a \in \mathcal{A}_t} c_t(a)$$

with ties broken by first occurrence in the sampling history. The adaptive sampling problem is to select a stopping time $T \leq B$, where B is the maximum budget, and return $\hat{a} = a_T^*$ as the final prediction.

All three adaptive methods share a common *minimum-steps guard*: no stopping criterion is evaluated until $t \geq t_{\min} = 3$ and $n_t \geq n_{\min} = 3$. This prevents premature stopping before sufficient evidence has accumulated.

4.2 Baselines

Greedy decoding. A single trajectory is generated at temperature $\tau = 0$. This is the minimum-compute baseline and serves as the difficulty baseline: examples where greedy decoding fails are designated hard; all others are easy.

Fixed- N self-consistency. N trajectories are generated at temperature $\tau = 0.7$ and the plurality answer is returned:

$$\hat{a} = \arg \max_{a \in \mathcal{A}} \sum_{i=1}^N \mathbf{1}[\phi(y_i) = a]$$

We sweep $N \in \{1, 5, 10, 20, 40\}$ to construct the fixed-compute Pareto frontier.

4.3 Adaptive Stopping Criteria

Heuristic stopping (ADAPTIVE). Three rules are checked in priority order at each step $t \geq t_{\min}$; sampling stops as soon as any one fires.

Consensus: Stop if the plurality answer holds a supermajority of all draws:

$$\frac{c_t(a_t^*)}{t} \geq \tau_c \quad (1)$$

Margin: Stop if the gap between the top and runner-up vote counts is large:

$$c_t(a_t^*) - c_t(a_t^{(2)}) \geq \delta \quad (2)$$

where $a_t^{(2)} = \arg \max_{a \neq a_t^*} c_t(a)$, set to 0 if only one answer has been observed.

Stability: Stop if the plurality answer has been unchanged for the last s consecutive parsed answers:

$$a_{n_t-s+1}^* = a_{n_t-s+2}^* = \dots = a_{n_t}^* \quad (3)$$

where the sequence is indexed over parsed answers in arrival order. We use $\tau_c = 0.9$, $\delta = 3$, and $s = 3$ throughout.

Entropy stopping (ENTROPY). The Shannon entropy of the empirical vote distribution over parsed answers is computed in nats:

$$H_t = - \sum_{a \in \mathcal{A}_t} \frac{c_t(a)}{n_t} \ln \frac{c_t(a)}{n_t} \quad (4)$$

$H_t = 0$ when all parsed answers agree perfectly; $H_t = \ln |\mathcal{A}_t|$ when votes are uniformly distributed. Sampling stops when $H_t \leq \varepsilon$. At $\varepsilon = 0.3$: in the two-answer case, $-p \ln p - (1-p) \ln(1-p) = 0.3$ implies $p \approx 0.95$, so the threshold corresponds to approximately 95% plurality share. We sweep $\varepsilon \in \{0.1, 0.3, 0.5, 0.7\}$ at $B = 40$ for sensitivity analysis and hyperparameter tuning.

Bayesian stopping (BAYESIAN). We model the latent probability vector over answers with a Dirichlet prior. Let $K = |\mathcal{A}_t|$ be the number of distinct observed answers, and define an augmented probability vector $\theta = (\theta_1, \dots, \theta_K, \theta_0)$ where θ_k corresponds to answer $a_k \in \mathcal{A}_t$ and θ_0 captures the residual probability mass on any answer not yet observed. We place a symmetric Dirichlet prior with concentration $\alpha = 1.0$ on observed answers and $\alpha_0 = 1.0$ on the “other” bucket. After observing $\mathbf{c}_t$, the posterior is:

$$\theta \mid \mathbf{c}_t \sim \text{Dir}(\alpha + c_t(a_1), \dots, \alpha + c_t(a_K), \alpha_0) \quad (5)$$

The posterior confidence that a_t^* is the true plurality answer is estimated via Monte Carlo with $M = 2000$ draws:

$$q_t = P\left(\theta_{a_t^*} = \max_k \theta_k \mid \mathbf{c}_t\right) \approx \frac{1}{M} \sum_{m=1}^M \mathbf{1}\left[\arg \max_k \theta_k^{(m)} = a_t^*\right], \quad \theta^{(m)} \sim \text{Dir}(\alpha + c_t(a_1), \dots, \alpha + c_t(a_K), \alpha_0) \quad (6)$$

with Monte Carlo standard error bounded by $\sqrt{q_t(1-q_t)/M} \leq 0.011$. Sampling stops when $q_t \geq \gamma$. We use $\gamma = 0.95$ as the canonical threshold, selected from a sensitivity sweep over $\gamma \in \{0.80, 0.90, 0.95, 0.99\}$.

5 Experiments

5.1 Data and Evaluation

We evaluate on GSM8K [Cobbe et al., 2021], a benchmark of 8,500 grade-school arithmetic word problems (7,473 train / 1,319 test), each requiring multi-step reasoning over 2–8 arithmetic operations and having a unique numeric answer. We use a fixed random subset of 300 test examples with a fixed seed for reproducibility across all 23 runs.

We report four metrics: **gold accuracy** (exact match after normalization), **mode accuracy** (agreement with Fixed-N=40 plurality, separating stopping errors from model errors), **average samples** (inference cost proxy), and **% max budget** (fraction of examples that exhaust full budget B).

5.2 Experimental Details

All experiments use Qwen2.5-7B-Instruct with temperature 0.7 and top-p 0.95. The system prompt instructs the model to produce step-by-step reasoning and conclude with `<number>`. We run three experiment groups: (1) a **fixed-N self consistency sweep** with $N \in \{1, 5, 10, 20, 40\}$ establishing the reference frontier; (2) an **adaptive budget sweep** for each of the three methods at $B \in \{5, 10, 20, 40\}$ using canonical thresholds ($\tau_c=0.9$, $\varepsilon=0.3$, $\gamma=0.95$); and (3) a **threshold sensitivity analysis** at $B = 40$ sweeping $\varepsilon \in \{0.1, 0.3, 0.5, 0.7\}$ for ENTROPY and $\gamma \in \{0.80, 0.90, 0.95, 0.99\}$ for BAYESIAN.

5.3 Results

Table 1 reports accuracy and average samples for all canonical runs. Greedy achieves 93.0%. Fixed self-consistency saturates quickly: Fixed-N=5 reaches 95.0%, with no further gain at $N \in \{10, 20, 40\}$. All three adaptive methods at $B = 40$ match 95.0% while using 3.66–6.29 avg samples, reductions of 84–91% relative to Fixed-N=40.

Figure 1 shows the full Pareto frontier. Fixed-N accuracy flattens above $N = 5$, demonstrating rapid saturation of the accuracy ceiling with fixed sampling. All adaptive methods at each budget lie at lower average samples than the Fixed-N point of equal or greater accuracy. ADAPTIVE is the most compute-efficient (3.66 avg samples at $B = 40$), followed by BAYESIAN (5.66) and ENTROPY (6.29). The efficiency ordering reflects criterion strictness: ADAPTIVE fires as soon as any one of three heuristic conditions is met; BAYESIAN requires posterior confidence to exceed 0.95, which demands more evidence when votes are split.

Table 1: Main results on 300 GSM8K test examples. Parse rate is the fraction of generated samples yielding a parseable answer.

Method	Config	Acc (%)	Avg Samples	Parse Rate	% Max Budget
Greedy	—	93.0	1.00	100.0	—
Fixed-N	$N = 1$	92.7	1.00	100.0	—
Fixed-N	$N = 5$	95.0	5.00	99.9	—
Fixed-N	$N = 10$	94.7	10.00	100.0	—
Fixed-N	$N = 20$	95.3	20.00	100.0	—
Fixed-N	$N = 40$	94.7	40.00	100.0	—
ADAPTIVE	$B = 5$	93.7	3.18	99.9	6.0
ADAPTIVE	$B = 10$	93.3	3.29	99.9	1.7
ADAPTIVE	$B = 20$	94.7	3.57	100.0	0.7
ADAPTIVE	$B = 40$	95.0	3.66	99.9	0.7
ENTROPY	$\varepsilon=0.3, B = 5$	93.7	3.19	100.0	9.7
ENTROPY	$\varepsilon=0.3, B = 10$	94.7	3.82	99.8	11.7
ENTROPY	$\varepsilon=0.3, B = 20$	93.7	4.71	99.9	8.7
ENTROPY	$\varepsilon=0.3, B = 40$	94.7	6.29	100.0	8.0
BAYESIAN	$\gamma=0.95, B = 5$	94.0	4.14	100.0	14.7
BAYESIAN	$\gamma=0.95, B = 10$	94.7	4.64	100.0	6.7
BAYESIAN	$\gamma=0.95, B = 20$	94.7	5.19	99.9	4.7
BAYESIAN	$\gamma=0.95, B = 40$	95.0	5.66	99.9	2.3

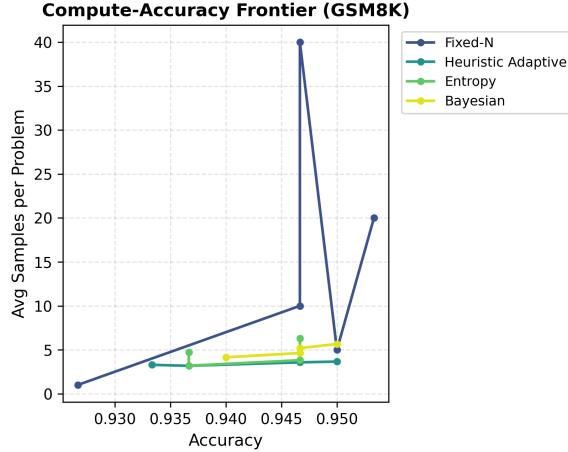


Figure 1: Accuracy vs. average samples (Pareto frontier). All adaptive methods achieve equivalent accuracy at substantially lower compute than Fixed-N SC.

6 Analysis

6.1 Convergence on a Strong Model

The aggregate Pareto curves for all three adaptive methods are nearly indistinguishable. We define examples that the greedy method fails on "hard examples", and the ones that it succeeds on "easy examples." On 278 of 300 examples, "the easy partition", the model reliably produces the correct answer, causing votes to concentrate within the first 3–4 draws regardless of stopping criterion. This is confirmed by sample distributions: ADAPTIVE stops at $t = 3$ on 271 of 300 examples (90.3%), and BAYESIAN stops at $t = 4$ on 264 of 300 (88%). The three criteria all detect the same signal, rapid vote concentration, and fire at nearly the same step. When nearly all examples are easy, aggregate accuracy and average samples are dominated by this easy majority, and stopping criterion design has minimal effect.

6.2 Hard vs. Easy Example Analysis

Partitioning by difficulty reveals substantial differences invisible in aggregate metrics. Table 2 reports accuracy and average samples for easy ($n = 278$) and hard ($n = 22$) examples across the full budget sweep.

Table 2: Performance by difficulty. Easy: greedy correct ($n = 278$). Hard: greedy incorrect ($n = 22$). Methods are indistinguishable on easy examples; differences are substantial on hard ones.

Method	Config	Easy ($n = 278$)		Hard ($n = 22$)	
		Acc (%)	Avg Samp.	Acc (%)	Avg Samp.
Fixed-N	$N = 5$	98.9	5.0	45.5	5.0
Fixed-N	$N = 10$	98.9	10.0	40.9	10.0
Fixed-N	$N = 20$	99.3	20.0	45.5	20.0
Fixed-N	$N = 40$	99.3	40.0	36.4	40.0
ADAPTIVE	$B = 5$	98.2	3.08	36.4	4.36
ADAPTIVE	$B = 10$	98.2	3.16	31.8	4.95
ADAPTIVE	$B = 20$	98.9	3.34	40.9	6.50
ADAPTIVE	$B = 40$	98.9	3.46	45.5	6.18
ENTROPY	$B = 5$	98.2	3.12	36.4	4.18
ENTROPY	$B = 10$	99.3	3.48	36.4	8.09
ENTROPY	$B = 20$	98.2	3.99	36.4	13.86
ENTROPY	$B = 40$	99.3	4.82	36.4	24.86
BAYESIAN	$B = 5$	98.9	4.09	31.8	4.77
BAYESIAN	$B = 10$	98.6	4.41	45.5	7.45
BAYESIAN	$B = 20$	98.9	4.62	40.9	12.50
BAYESIAN	$B = 40$	99.3	4.83	40.9	16.18

On easy examples, all methods achieve 98.2–99.3% accuracy in 3–5 samples. Hard examples reveal distinct failure modes.

Fixed-N vote reinforcement. Hard accuracy is non-monotone in N : 45.5% at $N \in \{5, 20\}$, 40.9% at $N = 10$, falling to 36.4% at $N = 40$. Many hard examples have a consistent but incorrect answer; as N grows the wrong plurality accumulates votes and becomes harder to dislodge.

ENTROPY is compute-insensitive on hard examples. Hard accuracy is flat at 36.4% across all budgets while avg samples grows from 4.18 to 24.86. The criterion cannot distinguish concentration around a correct answer from concentration around a wrong one. Entropy drops whenever votes accumulate, regardless of their correctness. At $B = 40$, ENTROPY spends the most compute on hard examples and achieves the worst hard accuracy of any adaptive method, mirroring Fixed-N=40.

ADAPTIVE exhibits fast commitment. Hard accuracy improves with budget (36.4% at $B = 5$ to 45.5% at $B = 40$) while avg samples grows modestly from 4.36 to 6.18. At small budgets, heuristic rules fire on early wrong pluralities; at larger budgets, competing votes occasionally prevent premature firing. At $B = 40$, ADAPTIVE achieves the highest hard accuracy at the lowest compute of any adaptive method.

BAYESIAN shows calibrated uncertainty. Hard accuracy is non-monotone: 31.8% at $B = 5$, 45.5% at $B = 10$, settling at 40.9% for $B \in \{20, 40\}$. The Dirichlet posterior requires evidence relative to the full answer space including the “other” bucket, making it harder to satisfy when votes are split. At $B = 10$ this conservatism is well-calibrated: BAYESIAN achieves 45.5% hard accuracy, matching the best fixed baseline, with only 7.45 avg samples. At larger budgets hard accuracy plateaus, suggesting the same systematic error ceiling as Fixed-N.

Table 3: Gold vs. mode accuracy across all runs. The gap measures stopping error vs. fundamental model error.

Method	Config	Gold Acc (%)	Mode Acc (%)	Gap (%)
Fixed-N	$N = 1$	92.7	95.3	2.7
Fixed-N	$N = 5$	95.0	97.7	2.7
Fixed-N	$N = 10$	94.7	99.0	4.3
Fixed-N	$N = 20$	95.3	99.3	4.0
Fixed-N	$N = 40$	94.7	100.0	5.3
ADAPTIVE	$B = 5$	93.7	96.7	3.0
ADAPTIVE	$B = 10$	93.3	98.3	5.0
ADAPTIVE	$B = 20$	94.7	98.3	3.7
ADAPTIVE	$B = 40$	95.0	98.7	3.7
ENTROPY	$\varepsilon=0.3, B = 5$	93.7	98.0	4.3
ENTROPY	$\varepsilon=0.3, B = 10$	94.7	98.7	4.0
ENTROPY	$\varepsilon=0.3, B = 20$	93.7	98.0	4.3
ENTROPY	$\varepsilon=0.3, B = 40$	94.7	99.7	5.0
BAYESIAN	$\gamma=0.95, B = 5$	94.0	99.0	5.0
BAYESIAN	$\gamma=0.95, B = 10$	94.7	98.7	4.0
BAYESIAN	$\gamma=0.95, B = 20$	94.7	98.7	4.0
BAYESIAN	$\gamma=0.95, B = 40$	95.0	98.7	3.7

6.3 Mode Accuracy vs. Gold Accuracy

Table 3 reports gold accuracy alongside mode accuracy (agreement with Fixed-N=40 plurality). The consistent 2–5% gap across all methods confirms that the dominant error source is the model producing an incorrect plurality, not the stopping criterion selecting the wrong answer from an otherwise correct distribution.

The gap grows with budget for Fixed-N (2.7% to 5.3%) and ENTROPY (4.3% to 5.0%): at large budgets ENTROPY allows votes to accumulate toward the Fixed-N=40 distribution, yielding high mode accuracy (99.7%) but the same gold ceiling (94.7%). BAYESIAN’s gap decreases from 5.0% at $B = 5$ to 3.7% at $B = 40$ while gold accuracy increases from 94.0% to 95.0. Additional samples under posterior uncertainty occasionally correct an early wrong plurality. ADAPTIVE’s gap is stable at 3.0–3.7%, reflecting early commitment regardless of budget.

6.4 Threshold Sensitivity

Figure 2 shows sensitivity at $B = 40$. For ENTROPY, accuracy is flat at 94.7% across $\varepsilon \in \{0.1, 0.3, 0.5\}$ while avg samples decreases from 7.07 to 5.20; only at $\varepsilon = 0.7$ does accuracy drop to 93.3%. For BAYESIAN, accuracy is flat at 95.0% for $\gamma \in \{0.95, 0.99\}$ with avg samples increasing from 5.66 to 8.28; accuracy drops to 94.0% and 94.3% at $\gamma \in \{0.80, 0.90\}$. Within recommended ranges ($\varepsilon \in [0.1, 0.5]$, $\gamma \in [0.95, 0.99]$), threshold choice primarily controls compute rather than accuracy.

6.5 Stopping Reason Distribution and Sample Histograms

At $B = 40$: ADAPTIVE stops via consensus on 271 examples (90.3%), margin on 17 (5.7%), stability on 10 (3.3%), and max budget on 2 (0.7%). ENTROPY stops via entropy threshold on 276 (92.0%) and max budget on 24 (8.0%). BAYESIAN fires via posterior confidence on 293 (97.7%) and max budget on 7 (2.3%). ENTROPY’s 8.0% max-budget rate is the highest, consistent with entropy remaining elevated on hard examples with persistent disagreement. ADAPTIVE concentrates at $t = 3$ for 90.3% of examples, BAYESIAN at $t = 4$ for 88%; ENTROPY has a heavier tail extending to $t = 40$ (Figures 3, 4).

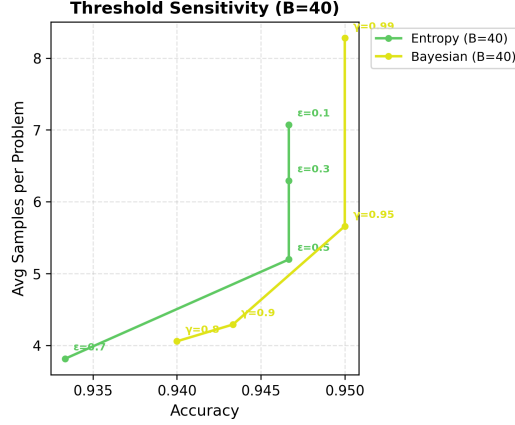


Figure 2: Threshold sensitivity at $B = 40$.

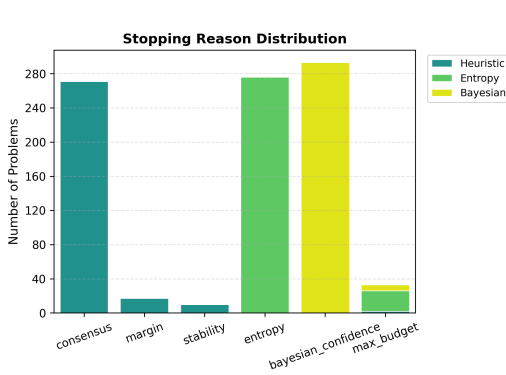


Figure 3: Stop reason distribution at $B = 40$.

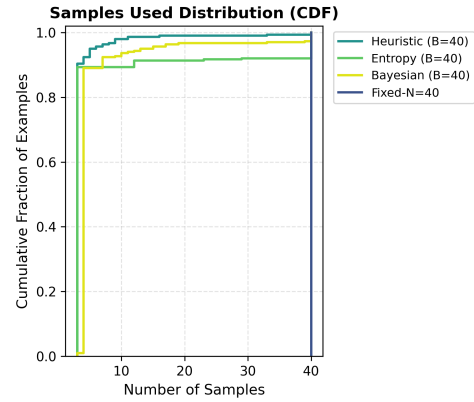


Figure 4: CDF of samples used at $B = 40$.

7 Conclusion

We implemented and evaluated three families of adaptive test-time compute strategies, heuristic, entropy-based, and Bayesian posterior stopping, for self-consistent reasoning on GSM8K. All three methods achieve 84–91% compute reduction relative to Fixed-N=40 while matching its accuracy, and stopping criterion choice has minimal effect on aggregate metrics when the base model is strong. The distinction emerges on hard examples: entropy-based stopping with a large budget mirrors the failure mode of Fixed-N=40, spending 24.86 avg samples to achieve only 36.4% accuracy on hard examples. Heuristic stopping commits earliest (6.18 avg samples, 45.5% accuracy). Bayesian stopping draws a moderate number of samples when posterior confidence is hard to achieve, achieves 40.9% accuracy on hard examples, highest among adaptive methods, and maintains the highest or joint-highest aggregate accuracy across all budget levels. A consistent mode-vs-gold accuracy gap of 2–5% across all methods confirms that model error, not stopping error, is the primary bottleneck.

The primary limitation is that evaluation uses a strong model on an accessible benchmark, concentrating nearly all examples in the easy partition. Future work should evaluate on MATH [Hendrycks et al., 2021], providing a setting where the hard partition dominates and the differences between stopping criteria are most consequential. Additional directions include learning difficulty-predictive features for pre-sampling budget allocation, and augmenting the Bayesian stopping criterion with an explicit cost signal that trades accuracy against compute in the stopping decision.

References

- P. Aggarwal, A. Madaan, Y. Yang, and Mausam. Let’s sample step by step: Adaptive-consistency for efficient reasoning and coding with llms. 2023. URL <https://arxiv.org/abs/2305.11860>.
- B. Brown, J. Juravsky, R. Ehrlich, R. Clark, Q. V. Le, C. Ré, and A. Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. 2024. URL <https://arxiv.org/abs/2407.21787>.
- K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman. Training verifiers to solve math word problems. 2021. URL <https://arxiv.org/abs/2110.14168>.
- D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt. Measuring mathematical problem solving with the math dataset. 2021. URL <https://arxiv.org/abs/2103.03874>.
- J. Huang, W. Ma, and Z. Zhou. Optimal bayesian stopping for efficient inference of consistent llm answers. 2026. URL <https://arxiv.org/abs/2602.05395>.
- Y. Li, P. Yuan, S. Feng, B. Pan, X. Wang, B. Sun, H. Wang, and K. Li. Escape sky-high cost: Early-stopping self-consistency for multi-step reasoning. 2024. URL <https://arxiv.org/abs/2401.10480>.
- C. Snell, J. Lee, K. Xu, and A. Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. 2024. URL <https://arxiv.org/abs/2408.03314>.
- X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou. Self-consistency improves chain of thought reasoning in language models. 2023. URL <https://arxiv.org/abs/2203.11171>.
- J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. 2023. URL <https://arxiv.org/abs/2201.11903>.