Deadline-Valid Replay: Anytime-Valid Testing for LLM Inference Policies

Rui Gao [1]

Abstract

Large language model systems increasingly choose among inference policies rather than fixed models: they may answer greedily, sample multiple completions, branch when confidence is low, or abstain under uncertainty. Under latency deadlines, standard leaderboards do not distinguish a descriptive utility gap from a statistically certified policy advantage. We formulate deadline-constrained LLM policy evaluation as a paired sequential testing problem. Our protocol replays frozen real-model traces, assigns correct-before-deadline utility, and monitors policy comparisons using an anytime-valid mixture betting e-process. In a four-slice Qwen2.5 trace-bank study spanning GSM8K, BBH, ARC, and CoLA, descriptive leaders vary across tasks and deadlines. However, no configured comparison crosses the unadjusted $\alpha = 0.05$ e-value rejection threshold; the largest final e-value is 3.675, below the threshold of 20, and it is based on only three discordant prompts out of 120. In the full four-policy, four-slice, three-deadline grid, a one-sided family-wise Bonferroni-over-e-values rule would require threshold 1440, far above all observed e-values. These results show that deadline-sensitive leaderboard differences can be visible while remaining statistically uncertified, motivating calibrated sequential evidence as a discipline layer for LLM policy evaluation.

1. Introduction

LLM evaluation usually reports model or system quality under a fixed decoding setup (Vaswani et al., 2017; Bommasani et al., 2021). Many deployed systems, however, are better described as inference policies. A policy decides how much computation to spend, whether to sample multiple attempts, when to fall back to a more deliberate trace, and whether to abstain. These choices matter most when an answer must arrive before a deadline.

Deadline-constrained evaluation changes the statistical problem. A policy can be more accurate when allowed to finish, but still less useful under a tight latency budget. Conversely, a fast policy may become the descriptive leader at short deadlines even if it loses in a no-time-limit reference setting. The right comparison is therefore paired and deadline-conditioned: on the same prompt, under the same deadline, did policy A produce a correct answer on time when policy B did not?

This setting also creates a sequential evidence problem. Evaluation traces are often expensive, and investigators may inspect results as prompt budgets grow even when the final paper reports only a fixed table. A fixed-horizon p-value does not by itself support repeated interim looks, and naive repeated peeking can overstate evidence (Wald, 1945; Howard et al., 2021; Ramdas et al., 2023). We use an anytime-valid e-process–a nonnegative process whose conditional expectation under the null is at most its previous value–so that sequential monitoring remains calibrated (Vovk & Wang, 2021; Grünwald et al., 2024). The practical lesson of the case study is not that prior LLM benchmarks are guilty of optional stopping; it is that well-calibrated tests can reveal when observed policy gaps are too small, too tied, or too underpowered to support a winner claim.

The contribution of this paper is not a claim that the case study has established a certified directional advantage. The contribution is a reproducible evaluation workflow that separates descriptive policy rankings from statistically certified directional claims. In this case study, descriptive rankings vary by task regime and deadline, while the sequential test correctly refuses to certify directional advantages when the evidence remains weak.

Our goal is not to introduce a new e-process. Rather, we show how standard anytime-valid testing tools can be made operational for a concrete and increasingly common LLM evaluation problem: comparing inference policies under latency deadlines while keeping descriptive rankings, effective paired sample size, and statistical certification separate.

We contribute a paired deadline-conditioned testing formula-

[1]MBZUAI. Correspondence to: Rui Gao <rui.gao@mbzuai.ac.ae>.

Accepted at the ICML 2026 Workshop on Hypothesis Testing, July 11, 2026, The COEX Convention & Exhibition Center, Seoul, South Korea. Non-archival workshop paper; not published in official proceedings. *Copyright 2026 by the author(s).*

tion, a frozen-trace replay design that preserves real outputs and latency metadata, and a four-slice Qwen2.5 trace-bank study over GSM8K, BBH, ARC, and CoLA where observed leaders vary by slice and deadline but no tracked comparison crosses the anytime-valid rejection threshold (Qwen Team, 2024; Cobbe et al., 2021; Suzgun et al., 2023; Clark et al., 2018; Warstadt et al., 2019).

2. Related Work

Our work connects LLM evaluation and test-time compute with sequential hypothesis testing. GSM8K, BBH, ARC, and CoLA expose different regimes (Cobbe et al., 2021; Suzgun et al., 2023; Clark et al., 2018; Warstadt et al., 2019); chain-of-thought and self-consistency show that inference-time computation can change accuracy (Wei et al., 2022; Wang et al., 2023); and selective prediction adds abstention as a policy dimension (Geifman & El-Yaniv, 2017). Sequential analysis, time-uniform concentration, e-values, and safe testing provide tools for valid inference under optional stopping (Wald, 1945; Howard et al., 2021; Vovk & Wang, 2021; Grünwald et al., 2024; Ramdas et al., 2023). We combine these threads by treating deadline-constrained LLM policy evaluation as a paired sequential testing problem rather than a fixed-horizon leaderboard comparison.

3. Deadline-Valid Policy Testing

Deadline utility. For prompt i and deadline τ, define deadline utility as

$$U_{i,\tau} = \mathbb{1}\{\text{answer is correct and finished before } \tau\}. \quad (1)$$

Directed paired comparison. For a fixed policy pair (A, B) and deadline τ, the paired comparison is

$$D_{i,\tau} = U^A_{i,\tau} - U^B_{i,\tau}. \quad (2)$$

The directional test is

$$H_0 : \mathbb{E}[D_{i,\tau}] \leq 0, \quad (3)$$

$$H_1 : \mathbb{E}[D_{i,\tau}] > 0. \quad (4)$$

Each prompt yields a win, loss, or tie for policy A against policy B. Ties are retained in the trace and reported, while the sequential evidence is driven by discordant outcomes.

Anytime-valid testing. Let $D_t = U^A_{t,\tau} - U^B_{t,\tau} \in \{-1, 0, 1\}$ be the paired outcome for the tth monitored prompt. The directed null is

$$H_0 : \mathbb{E}[D_t \mid \mathcal{F}_{t-1}] \leq 0,$$

against the alternative that policy A has positive paired advantage. For a betting fraction $\lambda \in [0, 1]$, define

$$E_t(\lambda) = \prod_{s=1}^{t} (1 + \lambda D_s).$$

Under H_0, $E_t(\lambda)$ is a nonnegative supermartingale. The implementation uses an equal-weight finite mixture over $\Lambda = \{0.10, 0.20, \ldots, 0.90\}$:

$$E_t = \sum_{\lambda \in \Lambda} w_\lambda E_t(\lambda), \qquad \sum_{\lambda \in \Lambda} w_\lambda = 1.$$

Thus E_t is an e-process and the rule "reject once $E_t \geq 1/\alpha$" is valid under optional stopping. We use $\alpha = 0.05$, so the rejection threshold is 20. We view prompts as sampled from an evaluation population within each slice; the frozen trace bank is the observed sample on which replayed policies are compared. With fixed policies, fixed prompt order, and replayed traces, D_t is a deterministic function of the tth prompt record, so the conditional-null statement used for the supermartingale proof reduces to the marginal mean null under the usual iid or exchangeable sampling interpretation within a slice.

Policy thresholds, unordered policy pairs, deadlines, prompt order, and the betting grid were fixed before monitoring. The direction labels displayed in Tables 3 and 7 are chosen after observing the sign of the paired mean difference and are reported only as compact effect-size diagnostics. They are not used to define a valid one-sided discovery, and no row is claimed as a certified directional advantage. Since no displayed e-process reaches even the unadjusted threshold of 20, the formal conclusion is non-certification. A future rejecting analysis should either pre-specify the direction of every one-sided comparison before monitoring, or use a two-sided/equal-direction correction.

For a future one-sided analysis with pre-specified directions over the full four-policy, four-slice, three-deadline grid, the comparison family has

$$m = 4 \times 3 \times \binom{4}{2} = 72$$

policy-slice-deadline comparisons. A simple family-wise Bonferroni-over-e-values rule would declare a discovery only when an individual e-process reaches

$$m/\alpha = 72/0.05 = 1440.$$

The maximum final e-value in this run is 3.675, far below both the unadjusted threshold 20 and the family-wise threshold 1440. If both directions of every unordered pair are monitored, the family size should be doubled, or an explicitly two-sided e-process should be used.

The released reproducibility repository at `https://github.com/ray-suton/Deadline_Valid` contains the frozen replay configs, trace-bank manifests, validation metadata, and replay outputs used here. The relevant repository paths include `configs/benchmarks_four_slice_actual_`

qwen3b_v3.json, data/benchmarks, and validation/trace_bank_validation.json. These files are integrity anchors for the released replay package, not an external pre-registration. Future rejecting analyses should pre-register directions, family size, stopping rule, betting grid, and multiplicity rule before trace collection and monitoring.

Tied prompts contribute a multiplicative factor of one, so evidence accumulates only through discordant prompts while preserving the original paired replay record. Fixed-horizon p-values are one-sided exact sign-test values over discordant paired outcomes and are reported only as descriptive diagnostics when the displayed direction is chosen post hoc. The mixture grid is a safe hedge over medium-to-large alternatives, but it is not optimized for the very small observed effects in this case study. The largest observed paired mean difference is only 0.025, and the strongest row has only three discordant prompts, so the current grid should be interpreted as a medium-effect diagnostic rather than a power-optimized test for the observed regime. We keep the grid fixed to preserve the released repository outputs rather than retuning it after observing the results; denser low-λ grids or beta-mixture betting are natural future refinements.

Proposition 3.1. *For any fixed policy pair and deadline, suppose the prompt order is fixed in advance or randomized independently of the observed outcomes, and suppose that under the directed null,*

$$\mathbb{E}[D_t \mid \mathcal{F}_{t-1}] \leq 0 \quad \textit{for all } t.$$

Then the mixture process E_t above satisfies

$$\mathbb{P}_{H_0}\left(\sup_t E_t \geq 1/\alpha\right) \leq \alpha.$$

The proof follows from the supermartingale property of each $E_t(\lambda)$ and Ville's inequality; see Section A.

4. Frozen-Trace Replay Design

Frozen trace construction. Each benchmark row stores the prompt, answer, task metadata, and multiple model traces with raw output, extracted answer, correctness-relevant answer field, latency, confidence, token counts, model id, and decoding configuration. Stored extracted answers are scored by exact match after lowercasing and whitespace normalization. This paper uses a Qwen2.5-3B-Instruct trace bank (Qwen Team, 2024). The stored confidence score is the exponentiated mean transition log-probability of the generated completion tokens, clipped to $[0, 1]$; for API-style traces it is computed analogously from returned token log-probabilities. It is a replay feature for policy control, not a calibrated probability of correctness.

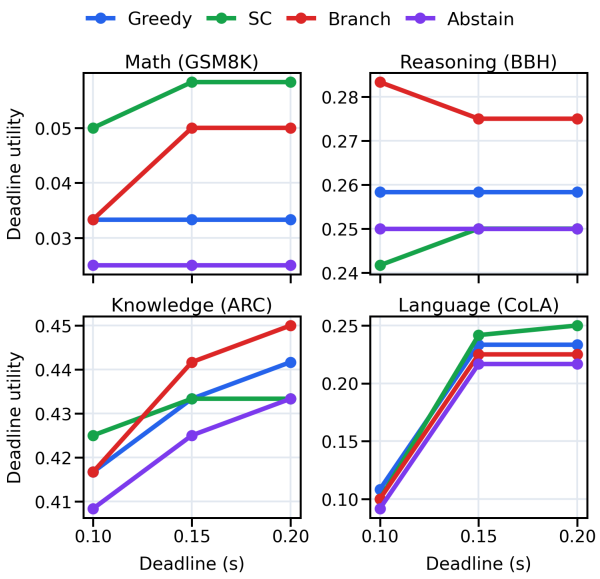

Figure 1. Deadline utility by policy and slice. SC denotes self_consistency_3, Branch denotes budget_branch, and Abstain denotes selective_abstain. The plotted source asset is generated from the main frozen replay experiment. Axes are slice-specific and zoomed; visual separations should be interpreted together with the absolute utilities in Table 2.

Policy replay. Policies consume frozen traces rather than making live API calls. This preserves reproducibility while allowing deadline-conditioned behavior. The current policy set contains:

- greedy: use the direct greedy trace.
- self_consistency_3: serially collect up to three sampled direct traces, then vote by answer count with confidence-mass and first-seen tie breaks (Wang et al., 2023).
- budget_branch: use the greedy trace when confident and fall back to a deliberate trace when confidence is low and time allows.
- selective_abstain: abstain when greedy confidence is below threshold (Geifman & El-Yaniv, 2017).

All thresholds were fixed before replay: budget branching attempts fallback when greedy confidence is below $\gamma_b = 0.80$, and selective abstention abstains below $\gamma_a = 0.75$. Selective abstention is included as a replayable control to illustrate that the framework can evaluate abstention behavior, but the main utility in Eq. (1) rewards only correct answers before deadline. Under this utility, abstention receives zero credit, so abstention results should not be interpreted as a full selective-prediction evaluation; risk–coverage utility is left to future work.

Latency semantics. The replay policy may only use traces that would have completed before the deadline. A single-

Table 1. Frozen trace slice summary. Each slice contains 120 prompts and five traces per prompt.

Slice	Source	Prompts	Traces	Answer type
Math	GSM8K	120	600	integer
Reasoning	BBH	120	600	boolean/integer
Knowledge	ARC	120	600	multiple-choice
Language	CoLA	120	600	string label

Table 2. Highest observed policy by deadline. Values in parentheses are mean deadline utility.

Slice	0.10 s	0.15 s	0.20 s
Math	SC (0.050)	SC (0.058)	SC (0.058)
Reasoning	Branch (0.283)	Branch (0.275)	Branch (0.275)
Knowledge	SC (0.425)	Branch (0.442)	Branch (0.450)
Language	Greedy (0.108)	SC (0.242)	SC (0.250)

trace policy succeeds if its latency is at most τ and its extracted answer is correct. Self-consistency is modeled as serial requests: at deadline τ, it votes over the largest completed prefix whose cumulative latency is at most τ. This is conservative because batched or parallel generation would more naturally use maximum sample latency plus overhead. We did not rerun the main replay outputs under that sensitivity model, so self-consistency results should be read under the stated serial-serving assumption and should not be interpreted as estimates of a fully parallel deployment. For budget-branching, greedy is evaluated first; if confidence is below threshold, fallback is eligible only if cumulative completion time remains within τ. When both complete on time, the policy selects the higher-confidence answer. No policy conditions on correctness or future trace latency.

Trace bank. The main replay experiment contains 480 prompts and 2400 traces across math, reasoning, knowledge, and language slices. Table 1 gives the slice provenance and metadata summary. The replay uses latency metadata stored at trace-collection time rather than remeasuring latency during analysis.

Deadlines. The evaluated deadlines are 0.10, 0.15, and 0.20 seconds. These deadlines are intentionally strict: they expose the gap between raw correctness and correct-on-time utility. Across 2400 individual traces, the median latency is 0.129 seconds with interquartile range 0.058–0.528 seconds; the deadline grid therefore stresses the lower-to-middle part of the observed latency distribution rather than representing universal deployment constraints. A no-time-limit reference view is also generated from the same traces so that deadline losses can be interpreted against the underlying trace quality. Difficulty metadata is tracked in the repository outputs, but the current trace bank contains only the `medium` bucket; difficulty-stratified analyses are therefore deferred to future trace banks.

5. Results

5.1. Descriptive Leaders Vary by Slice and Deadline

Figure 1 and Table 2 show the main descriptive result: policy choice is regime-sensitive. Math is led by self-consistency at all evaluated deadlines. Reasoning is led by budget-branching at all evaluated deadlines. Knowledge shifts from self-consistency at 0.10 seconds to budget-branching at 0.15 and 0.20 seconds. Language shifts from greedy at 0.10 seconds to self-consistency at 0.15 and 0.20 seconds. These leader differences are small relative to the 120-prompt slice size and should be read as descriptive rankings, not confidence intervals or certified policy wins. A single globally certified policy advantage is therefore not the right summary for this benchmark.

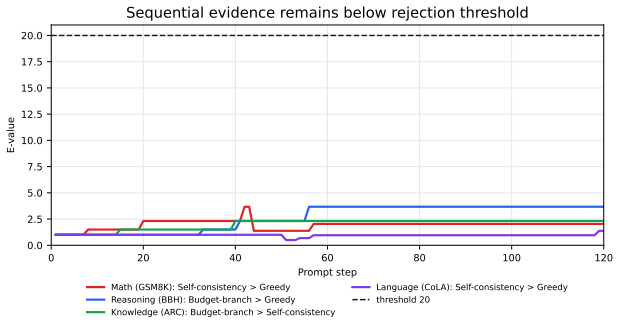

Figure 2. Largest diagnostic e-value trajectory per slice. SC denotes `self_consistency_3` and Branch denotes `budget_branch`. All trajectories remain below the $\alpha = 0.05$ rejection threshold of 20.

5.2. Sequential Evidence Does Not Certify Advantages

Table 3 reports representative nonzero diagnostics from the configured pairwise comparisons, and Figure 2 plots the largest e-value trajectory per slice. Every sequential decision remains `continue`. As a descriptive summary, the largest final e-value is observed for `budget_branch` over `greedy` on the reasoning slice at deadline 0.10 seconds, with mean utility difference 0.025, final e-value 3.675, fixed-horizon $p = 0.125$, and sequential decision `continue`. This row has only three discordant prompts and 117 ties, so the effective paired sample size is far smaller than the nominal 120 prompts. Since the unadjusted rejection threshold is $1/\alpha = 20$, even the strongest observed trajectory remains far from rejection. Under the full one-sided 72-comparison family, the Bonferroni-over-e-values threshold would be 1440, making the non-certification conclusion even stronger.

Thus the replay identifies descriptive deadline-sensitive differences, but the anytime-valid test does not certify any directional policy advantage at the evaluated prompt budget.

Table 3. Selected nonzero pairwise diagnostics. The displayed direction is chosen after observing the sign of the paired mean difference and is reported only as a compact effect-size diagnostic, not as a certified one-sided discovery. Every sequential decision remains `continue`. SC denotes `self_consistency_3`; Branch denotes `budget_branch`.

Slice	Displayed dir.	τ	Mean diff	W/L/T	Final E_T
Reasoning	Branch > Greedy	0.10	0.025	3/0/117	3.675
Math	SC > Greedy	0.15	0.025	4/1/115	2.038
Math	SC > Greedy	0.20	0.025	4/1/115	2.038
Knowledge	Branch > SC	0.20	0.017	2/0/118	2.317
Language	SC > Greedy	0.20	0.017	3/1/116	1.380

5.3. No-Time-Limit Reference

The no-time-limit reference view in Section C distinguishes raw answer quality from strict-deadline utility. For example, reasoning has higher no-time-limit utility for budget-branching than greedy, while the deadline-constrained comparison still lacks enough paired evidence for certification.

5.4. Prompt-Budget Simulations Explain Non-Rejection

Section D gives the prompt-budget simulation context. Under the null row, the anytime-valid procedure stays near nominal false-positive rate while naive repeated peeking rejects in 0.170 of simulations. At 120 prompts, simulated mean differences 0.195 and 0.260 yield sequential reject rates 0.604 and 0.876, respectively. The observed paired differences in Table 3, all at most 0.025 and with high tie rates, are therefore below the range where 120 prompts would provide reliable sequential certification.

For the strongest observed row, an optimistic plug-in extrapolation treats future prompts as following the empirical reasoning Branch-over-Greedy distribution at deadline 0.10 seconds: win probability $3/120 = 0.025$, loss probability 0, and tie probability $117/120 = 0.975$. Under this zero-loss extrapolation, the current mixture first exceeds the unadjusted threshold 20 after seven all-positive discordances, and reaches 0.80 probability of doing so at roughly 362 prompts. For the one-sided 72-comparison family-wise threshold 1440, the corresponding optimistic 0.80-power budget is roughly 679 prompts. These numbers should be read as favorable lower-bound diagnostics, because they assume the future loss rate remains zero and the prompt distribution is stable.

6. Conclusion

Deadline-constrained LLM evaluation should compare inference policies under deployment timing, not just rank models. We provide a frozen-trace, paired, anytime-valid workflow. In this Qwen2.5 case study, leaders vary by slice and deadline, yet no tracked comparison is certified. Reports should separate descriptive leaderboards, discordant-prompt counts, and anytime-valid evidence. Any rejecting study should pre-specify directions, family size, betting grid, stopping rule, and multiplicity control before monitoring.

Limitations and Future Work This frozen-replay case study uses one Qwen2.5-3B trace bank, four task slices, and strict deadlines; it does not capture queueing, hosted-model drift, online adaptation, or universal policy rankings. The current self-consistency replay uses a serial latency model, which likely disadvantages self-consistency relative to batched or parallel serving at short deadlines. The betting grid is safe but not power-optimized for the very small observed paired effects. Future work should evaluate broader models and serving regimes, report serial and parallel latency sensitivities, add wider deadline grids, tune branching thresholds under multiplicity control, use denser low-λ or beta-mixture betting for small effects, add risk–coverage utility for abstention, and pre-register multiplicity-controlled certification targets.

Code and Reproducibility Statement

Code, frozen replay configs, trace-bank manifests, validation metadata, tests, and generated analysis outputs are released at `https://github.com/ray-suton/Deadline_Valid`. The repository is replay-based: it supports reproducing the reported analysis from frozen traces and does not require live model calls.

Ethical Statement

This work studies evaluation methodology for inference-policy comparisons under latency deadlines. It uses frozen benchmark traces and does not collect personal data, interact with users, or deploy a model. The main ethical risk is overstating small or post hoc leaderboard differences as certified advantages. We mitigate this by separating descriptive rankings from anytime-valid certification, reporting discordant-prompt counts and non-rejections, and making the replay package public for inspection.

References

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., and Tafjord, O. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, Ł., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Geifman, Y. and El-Yaniv, R. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Grünwald, P., de Heide, R., and Koolen, W. M. Safe testing. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(5):1091–1128, 2024. doi: 10.1093/jrsssb/qkae011.

Howard, S. R., Ramdas, A., McAuliffe, J., and Sekhon, J. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021.

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

Ramdas, A., Grünwald, P., Vovk, V., and Shafer, G. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 38(4):576–601, 2023.

Suzgun, M., Scales, N., Schärli, N., Gehrmann, S., Tay, Y., Chung, H. W., Chowdhery, A., Le, Q. V., Chi, E. H., Zhou, D., and Wei, J. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Vovk, V. and Wang, R. E-values: Calibration, combination and applications. *The Annals of Statistics*, 49(3):1736–1754, 2021.

Wald, A. Sequential tests of statistical hypotheses. *The Annals of Mathematical Statistics*, 16(2):117–186, 1945.

Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi, E. H., Narang, S., Chowdhery, A., and Zhou, D. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

Warstadt, A., Singh, A., and Bowman, S. R. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641, 2019.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q. V., and Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

A. E-Process Validity Proof

Proof. For each $\lambda \in [0, 1]$, the factor $1 + \lambda D_t$ is nonnegative because $D_t \in [-1, 1]$. Under the directed null, its conditional expectation is

$$\mathbb{E}[1 + \lambda D_t \mid \mathcal{F}_{t-1}] = 1 + \lambda \mathbb{E}[D_t \mid \mathcal{F}_{t-1}] \leq 1,$$

so $E_t(\lambda)$ is a nonnegative supermartingale. A convex mixture of nonnegative supermartingales is also a nonnegative supermartingale (Howard et al., 2021). Ville's inequality gives

$$\mathbb{P}_{H_0}\left(\sup_t E_t \geq 1/\alpha\right) \leq \alpha.$$

□

B. Difficulty Metadata

The trace-bank metadata includes a `difficulty` field, and the analysis path emits per-difficulty utility and leader summaries. In the current trace bank, however, every prompt is labeled `medium`: 120/120 prompts in each of math, reasoning, knowledge, and language. The trace-bank metadata includes a `difficulty` field, and the analysis path emits per-difficulty utility and leader summaries. In the current trace bank, however, every prompt is labeled `medium`: 120/120 prompts in each of math, reasoning, knowledge, and language. Therefore, this study tracks difficulty metadata but does not evaluate policy rankings across difficulty strata.

C. No-Time-Limit Reference

Table 4. No-time-limit reference leaders from the same frozen traces.

Slice	No-limit leader	Utility	Mean latency
Math	Branch	0.108	0.532
Reasoning	Branch	0.508	0.292
Knowledge	Branch	0.875	0.284
Language	Greedy	0.500	0.412

D. Prompt-Budget Simulation Details

Table 5. Prompt-budget simulation context at 120 prompts.

Win prob.	Mean diff	Seq. reject	Fixed reject	Naive reject
0.50	0.000	0.043	0.037	0.170
0.55	0.065	0.104	0.188	0.438
0.60	0.130	0.275	0.517	0.699
0.65	0.195	0.604	0.828	0.928
0.70	0.260	0.876	0.972	0.990

For the strongest observed diagnostic row, reasoning Branch-over-Greedy at deadline 0.10 seconds, the empirical W/L/T count is 3/0/117. Under the optimistic plug-in model with win probability 0.025, loss probability 0, and tie probability 0.975, the e-value after K all-positive discordances is

$$E(K) = \frac{1}{9} \sum_{\lambda \in \{0.1, \ldots, 0.9\}} (1 + \lambda)^K.$$

This mixture first exceeds the unadjusted threshold 20 at $K = 7$, where $E(K) = 28.66$, and first exceeds the one-sided 72-comparison family-wise threshold 1440 at $K = 14$, where $E(K) = 1622.40$. Under $K \sim \text{Binomial}(n, 0.025)$, the probability of reaching $K \geq 7$ is approximately 0.80 at $n = 362$, while the probability of reaching $K \geq 14$ is approximately 0.80 at $n = 679$. This calculation is intentionally optimistic because it assumes no future losses.

E. Repository Sources

All paths in this appendix are relative to the released GitHub reproducibility repository: https://github.com/ray-suton/Deadline_Valid.

Table 6. Paper claims and their source repository paths.

Claim surface	GitHub repository paths
Descriptive deadline utility	results/four_slice_actual_qwen3b_v3/analysis/utility_summary.csv; results/four_slice_actual_qwen3b_v3/analysis/leaderboard.csv; results/four_slice_actual_qwen3b_v3/analysis/utility_by_deadline.svg
Sequential non-certification	results/four_slice_actual_qwen3b_v3/analysis/comparison_summary.csv; results/four_slice_actual_qwen3b_v3/analysis/decision_summary.csv; results/four_slice_actual_qwen3b_v3/analysis/evidence_trace.svg
Complete comparison grid	72-row unordered table in results/four_slice_actual_qwen3b_v3/analysis/comparison_summary.csv
No-time-limit interpretation	results/four_slice_actual_qwen3b_v3/analysis/no_time_limit_summary.csv; results/four_slice_actual_qwen3b_v3/analysis/no_time_limit_comparison_summary.csv; results/four_slice_actual_qwen3b_v3/analysis/no_time_limit_decision_summary.csv
Difficulty coverage	results/four_slice_actual_qwen3b_v3/analysis/difficulty_utility_summary.csv; results/four_slice_actual_qwen3b_v3/analysis/difficulty_leaderboard.csv
Prompt-budget context	results/power_sim_v1/power_simulation_summary.csv; results/power_sim_v1/power_simulation_summary.md; plug-in calculation in Section D
Reverification	Independent rerun under results/four_slice_actual_qwen3b_v3_reverify/analysis
Direction and multiplicity diagnostics	configs/benchmarks_four_slice_actual_qwen3b_v3.json; results/four_slice_actual_qwen3b_v3/analysis/comparison_summary.csv
Replay provenance and validation	data/benchmarks; validation/trace_bank_validation.json; frozen replay configs under configs

F. Displayed Nonzero Pairwise Diagnostics

This appendix lists the nonzero empirically favorable directions used for compact diagnostic reporting. The complete 72-row unordered comparison table, including abstention comparisons and zero-difference rows, is provided in results/four_slice_actual_qwen3b_v3/analysis/comparison_summary.csv. The rows below are descriptive: the displayed direction is chosen after observing the sign of the paired mean difference, no row is used as a certified one-sided discovery, and every sequential decision remains continue. A future rejecting analysis should either pre-specify directions or use a two-sided/multiplicity-controlled rule as discussed in the main text.

Table 7. Displayed nonzero pairwise diagnostics. W/L/T reports wins, losses, and ties in the displayed direction. These rows are compact descriptive diagnostics, not a complete listing of all 72 configured unordered comparisons.

Slice	Displayed dir.	Deadline	Mean diff	W/L/T	Final E_T	Fixed p
Knowledge	SC > Branch	0.10	0.008	3/2/115	0.721	0.500
Knowledge	Branch > SC	0.15	0.008	2/1/117	0.958	0.500
Knowledge	Branch > SC	0.20	0.017	2/0/118	2.317	0.250
Language	Greedy > SC	0.10	0.008	3/2/115	0.721	0.500
Language	SC > Greedy	0.15	0.008	3/2/115	0.721	0.500
Language	SC > Greedy	0.20	0.017	3/1/116	1.380	0.312
Math	SC > Branch	0.10	0.017	3/1/116	1.380	0.312
Math	SC > Greedy	0.10	0.017	3/1/116	1.380	0.312
Math	SC > Branch	0.15	0.008	2/1/117	0.958	0.500
Math	SC > Greedy	0.15	0.025	4/1/115	2.038	0.188
Math	SC > Branch	0.20	0.008	2/1/117	0.958	0.500
Math	SC > Greedy	0.20	0.025	4/1/115	2.038	0.188
Reasoning	Branch > Greedy	0.10	0.025	3/0/117	3.675	0.125
Reasoning	Branch > Greedy	0.15	0.017	3/1/116	1.380	0.312
Reasoning	Branch > Greedy	0.20	0.017	3/1/116	1.380	0.312