SUPERVISE WHAT YOU CERTIFY: STOP-HARM PREDICTION FOR EARLY STOPPING IN REASONING MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Reasoning models gain accuracy by generating long chains of thought, but incur substantial inference cost and often continue reasoning after reaching a correct answer. Early stopping aims to remove redundant computation while limiting accuracy loss. Existing learned stoppers use supervision based on current-answer correctness, answer agreement, or later answer stability. These targets do not directly capture lost-correct events: stopping yields a wrong answer although full reasoning would answer correctly. The probability of this event upper-bounds the accuracy drop, yet the loss used for certification differs from these supervision targets. We call this event *stop harm* and introduce StopHarm to supervise it directly at each checkpoint, aligning the learning target with the policy loss at the actual stopping point. StopHarm learns a lightweight predictor from paired stopping and full-reasoning outcomes offline, and stops online when the predicted stop harm stays below a certified threshold. Independent request-level certification bounds lost-correct risk, guaranteeing an accuracy drop of at most α with confidence $1 - \delta$ under i.i.d. sampling. We prove that even Bayes-optimal proxy predictions can discard stop-harm information that no score-only post-processing can recover. Across three reasoning models and five benchmarks, StopHarm reduces generated tokens by **25.4–38.3%** with average accuracy drops of only **0.27–0.73** points and **1.71–1.77**$\times$ single-request speedups.

1 INTRODUCTION

Reasoning models gain accuracy by generating long chains of thought, but incur substantial inference cost. The benefit of additional computation varies across problems (Snell et al., 2025), and models often verify or re-derive solutions after reaching a correct answer (Chen et al., 2025; Caldarella et al., 2026). Early stopping targets this redundancy while limiting accuracy loss.

Using only the current prefix, a stopper decides at intermediate checkpoints whether to terminate reasoning and produce a final answer. Learned stoppers use several supervision targets: LYNX and BLADE learn current-answer correctness or prefix sufficiency (Akgül et al., 2025; Fu et al., 2026); Thought Calibration includes agreement with the full-reasoning answer (Wu et al., 2025); convergence-based supervision predicts later answer stability (Liu & Wang, 2025); and future-outcome supervision predicts whether the reasoning will eventually succeed (Afzal et al., 2025).

To limit accuracy loss, we take complete reasoning under a fixed decoding protocol as the reference. A *lost-correct* event occurs when stopping yields a wrong answer although the reference answers correctly. Bounding its probability by α guarantees an accuracy drop of at most α (Dong et al., 2026). Lost-correct risk depends on the joint stopping and reference outcomes, which these proxy targets need not capture. For example, two requests can both have wrong stopping answers while only one has a correct reference answer. Even LearnStop (Dong et al., 2026), which certifies lost-correct risk, trains on current-answer correctness. When proxy scores omit risk distinctions relevant to stopping decisions, they can limit savings under the certified risk budget.

We call the lost-correct event at a checkpoint *stop harm* and introduce **StopHarm** to supervise it directly. Under a consistent stop-and-answer protocol, its label at the actual stopping point equals the policy's request-level lost-correct loss (Proposition 1), so the predictor learns the event that certi-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

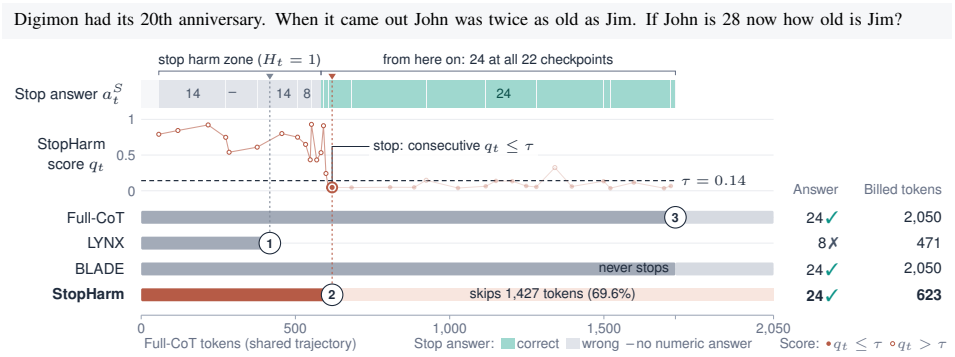


Figure 1: Qwen3-14B on GSM8K test question 1188; the stop-answer strip and faded post-stop scores are retrospective. LYNX stops early and answers 8, losing Full-CoT's correct 24. StopHarm answers 24 once the reasoning establishes $J = 24$, with 69.6% fewer billed tokens than Full-CoT.

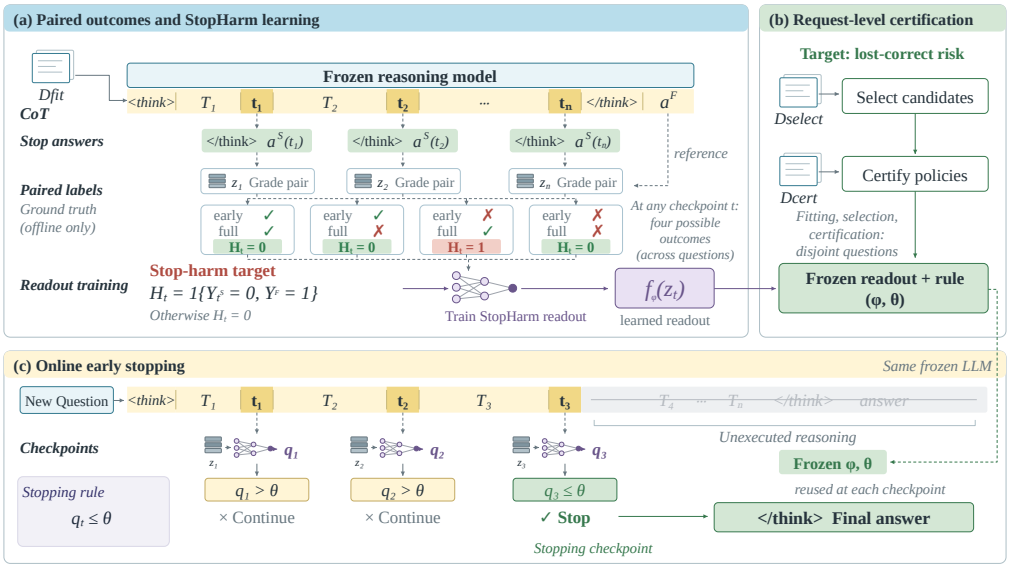


Figure 2: Overview of StopHarm.

fication controls. Offline, StopHarm trains a lightweight predictor on paired stopping and reference outcomes from the same trajectory, keeping the reasoning model frozen; its objective supervises stop harm at every checkpoint and, through the lowest harmful score of each trajectory, at the request level. Online, it reads hidden states from normal decoding and stops when the predicted score stays below a certified threshold. Learn-then-Test certification on independent requests bounds the complete policy's lost-correct risk at α with confidence $1 - \delta$ under i.i.d. sampling (Angelopoulos et al., 2025).

Even Bayes-optimal scores for current correctness, answer agreement, later answer stability, and reference success, taken together, can assign identical values to groups with different stop-harm probabilities; no score-only post-processing recovers this distinction (Theorem 1). For a fixed checkpoint distribution and risk–compute trade-off, this information has positive decision value precisely when states sharing proxy scores and cost information require opposite optimal actions with positive probability (Theorem 2). When this value exceeds a prediction-error penalty that depends on the learned score only through its ordering of checkpoints, a threshold on the learned StopHarm score achieves higher local utility than every controller restricted to those scores and cost information (Theorem 3).

A held-out trajectory, selected after inspecting test outcomes, illustrates a premature stop and continued reasoning after the correct answer is available (Figure 1).

Our contributions are threefold:

- **Supervision of the certified loss.** We derive supervision from the policy's lost-correct loss at both the checkpoint and the request level, linking direct stop-harm prediction to causal stopping and independent request-level certification.
- **Information gaps and decision value.** We prove that Bayes-optimal proxy scores can omit stop-harm information and establish when direct supervision improves local utility despite prediction error.
- **Computational gains and mechanism validation.** StopHarm saves 25.4–38.3% of generated tokens with average accuracy drops below one point, and harm supervision better separates recoverable errors from reference failures, saving more where the reference fails.

2 Problem Formulation

Fix a reasoning model, a decoding protocol with budget B, and a grader. The reference process F completes generation without early stopping. At checkpoint t, after n_t reasoning tokens, a policy either invokes a fixed stop-and-answer protocol S, or continues to the next checkpoint and decides again. Let $Y^F, Y_t^S \in \{0,1\}$ denote reference and stopping-answer correctness. Labeling and deployment use the same answering protocols.

A policy π uses the decoding history available at each checkpoint. It returns the stopping answer at its stopping time τ, or the reference answer if no stop occurs, with correctness Y^π. Its generated-token cost is $C(\pi) = n_\tau + \ell_\tau^S$ upon stopping, where ℓ_τ^S counts generated answer tokens, and the full reference generation cost otherwise. We seek

$$\min_{\pi \in \Pi} \mathbb{E}[C(\pi)] \quad \text{subject to} \quad R_{\mathrm{LC}}(\pi) = \mathbb{P}(Y^F = 1, Y^\pi = 0) \le \alpha, \tag{1}$$

where Π contains policies using only the available history. Since $\mathrm{Acc}(\pi) = \mathrm{Acc}(F) - R_{\mathrm{LC}}(\pi) + R_{\mathrm{RC}}(\pi)$, with $R_{\mathrm{RC}}(\pi) = \mathbb{P}(Y^F = 0, Y^\pi = 1)$ reported separately, this constraint bounds the accuracy drop by α (Dong et al., 2026).

Definition 1 (Stop harm)**.** The stop-harm label at checkpoint t is $H_t = Y^F(1 - Y_t^S)$.

Proposition 1 (Supervision at the stopping checkpoint)**.** *Let $A_{\pi,t} = \mathbf{1}\{\tau = t\}$. For every request run,*

$$\mathbf{1}\{Y^F = 1, Y^\pi = 0\} = \sum_t A_{\pi,t} H_t, \qquad R_{\mathrm{LC}}(\pi) = \mathbb{E} \sum_t A_{\pi,t} H_t. \tag{2}$$

At the stopping checkpoint, the stop-harm label equals the request's lost-correct loss, so StopHarm supervises at every checkpoint the event that certification controls. See appendix C for proof.

3 Method

StopHarm derives each component from Proposition 1: a probe predicts the stop-harm event from decoding features, its objective supervises this event at the checkpoint and request levels, and certification tests request-level loss. Figure 2 shows training, certification, and online execution.

3.1 Stop-harm probe and training objective

Offline, StopHarm completes each training reference trajectory and executes S at every checkpoint of that trajectory, pairing each stopping outcome with the reference outcome it would forgo to obtain H_t. It trains a small MLP on these labels while keeping the reasoning model frozen. Following the hidden-state read-out approach of semantic entropy probes (Kossen et al., 2024), the online probe reads features from the ongoing decoding process. It maps Z_t through two hidden layers of widths 128 and 32 to a logit a_t, with stop-harm score $q_t = f_\phi(Z_t) = \sigma(a_t)$.

For a minibatch $\mathcal{M}$ of complete training trajectories with at least one checkpoint, let K_i be the checkpoint count of trajectory i, $\mathcal{T}_i^+ = \{t : H_{it} = 1\}$ its harmful checkpoints, and $\mathcal{M}_+ = \{i \in$

Table 1: Supervision targets and their Bayes-optimal predictions under checkpoint-level cross-entropy on ν; method-specific label constructions may differ.

Supervision target	Label U	Bayes-optimal prediction
Current correctness / sufficiency	$1 - Y^S$	$\mathbb{P}(Y^S = 0 \mid Z)$
Agreement of current and reference outputs	$\mathbf{1}\{a^S \neq a^F\}$	$\mathbb{P}(a^S \neq a^F \mid Z)$
Stability of later answers	later answer switch occurs	$\mathbb{P}(\text{later switch} \mid Z)$
Final reference outcome	Y^F	$\mathbb{P}(Y^F = 1 \mid Z)$
StopHarm	$Y^F(1 - Y^S)$	$\mathbb{P}(Y^S = 0,\, Y^F = 1 \mid Z)$

$\mathcal{M} : \mathcal{T}_i^+ \neq \varnothing\}$. We minimize

$$\widehat{\mathcal{L}} = \underbrace{\frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \frac{1}{K_i} \sum_{t=1}^{K_i} \ell_{\mathrm{BCE}}(H_{it}, q_{it})}_{\widehat{\mathcal{L}}_{\mathrm{pt}}} + \rho \underbrace{\frac{1}{|\mathcal{M}_+|} \sum_{i \in \mathcal{M}_+} \log\bigl(1 + e^{-m_i}\bigr)}_{\widehat{\mathcal{L}}_{\mathrm{traj}}}, \tag{3}$$

where the trajectory term is zero when $\mathcal{M}_+$ is empty, and

$$m_i = -\beta \log\Bigl(\frac{1}{|\mathcal{T}_i^+|} \sum_{t \in \mathcal{T}_i^+} e^{-a_{it}/\beta}\Bigr), \qquad \rho = 1, \quad \beta = 0.5. \tag{4}$$

The pointwise term learns stop harm at the checkpoint level. It is the empirical cross-entropy under the request-balanced checkpoint distribution ν, which samples a request with at least one checkpoint and then one of its checkpoints uniformly, so each request carries equal total weight; its Bayes predictor is $\mathbb{P}_\nu(H = 1 \mid Z)$ (Table 1). Reference-failed trajectories contribute negative pointwise labels, distinguishing an incorrect stopping answer from the loss of a correct reference answer.

The trajectory term acts at the request level. The policy stops only at a checkpoint with $q_t \leq \theta$ (Section 3.2), so by Proposition 1 request i can incur stop harm only if $\min_{t \in \mathcal{T}_i^+} q_{it} \leq \theta$: the request is protected at every threshold below its lowest harmful score. The normalized soft-min m_i lies between the minimum and the mean of the harmful logits and concentrates its gradient on the lowest ones, so the trajectory term raises the scores that determine this protection.

At paragraph checkpoints, Z_t contains the last-token hidden state from one intermediate decoder layer and six scalars: n_t, $\log(1 + n_t)$, the checkpoint gap, mean top-20 normalized token entropy over the most recent eight decoding steps, $\|\Delta h_t\|_2$, and $\cos(h_t, \Delta h_t)$, where $\Delta h_t = h_t - h_{t-1}$. Features use only the available decoding history; standardization and model-specific settings are given in Appendix F.

3.2 Stopping and risk certification

At each checkpoint, the policy tests $q_t \leq \theta$ and applies a fixed confirmation rule to these tests (Section 5.1); it stops at a checkpoint that passes the test and answers there with S. Training and development data fix the probe, the threshold grid, and the testing order. On independent calibration requests, fixed-sequence Learn-then-Test (Angelopoulos et al., 2025) tests the cumulative request-level harm envelope of the ordered candidate policies. Among the certified prefix and the reference fallback, we select the policy with the lowest calibration generation cost. Under consistent protocols and i.i.d. calibration and deployment requests, this procedure yields $R_{\mathrm{LC}}(\widehat{\pi}) \leq \alpha$ with probability at least $1 - \delta$. Appendix E specifies the envelope, tie-breaking rule, and proof.

4 Theoretical Analysis

We establish when supervising stop harm provides information that proxy scores lose, when that information changes stopping decisions, and how prediction error limits its value. Condition on training and fix the request-balanced checkpoint distribution ν of Section 3.1. All predictors share Z; write $\eta(Z) = \mathbb{P}_\nu(H = 1 \mid Z)$ and take expectations under ν. Full proofs are in Appendix A.

4.1 Information lost by proxy scores

Let $Q = Q(Z)$ be a fixed proxy score or score vector. Any recalibration or stop-harm predictor reading only Q is a map $\psi(Q)$.

Theorem 1 (Stop-harm information gap)**.** *The best score-only approximation satisfies*

$$\inf_{\psi} \mathbb{E}[(\eta - \psi(Q))^2] = \mathbb{E}[\mathrm{Var}(\eta \mid Q)] =: \mathcal{D}_Q. \tag{5}$$

It is attained by $\mathbb{P}(H = 1 \mid Q)$, and $\mathcal{D}_Q = 0$ exactly when η is almost surely a function of Q. The corresponding excess log loss is $I(H; Z \mid Q)$.

Score-only post-processing averages risks within each score group; it cannot recover distinctions lost by Q. Even the four Bayes-optimal proxy predictions in Table 1 can coincide while η differs: Appendix B constructs equally likely groups with $\eta = 0.1$ and 0.2, giving $\mathcal{D}_Q = 1/400$. StopHarm's pointwise term instead has η as its Bayes predictor (Lemma A.1), directly supervising these distinctions.

4.2 When information changes the decision

Relative to completing F, stopping saves expected computation $b(Z) > 0$, with $\mathbb{E}[b] < \infty$, and loses a correct reference answer with probability $\eta(Z)$. For risk price $\lambda > 0$, the local utility of a stopping rule $s(Z) \in [0, 1]$ and its maximizer are

$$J_\lambda(s) = \mathbb{E}[(b - \lambda\eta)s], \qquad s^*(Z) = \mathbf{1}\{\eta(Z) < b(Z)/\lambda\}, \tag{6}$$

the Lagrangian form of maximizing $\mathbb{E}[bs]$ subject to $\mathbb{E}[\eta s] \le \alpha$. StopHarm's test $q_t \le \theta$ has this form for constant b, with θ in the role of b/λ. Both controllers below observe b.

Theorem 2 (Decision value of stop-harm information)**.** *Let J_Z^* and J_V^* be the optimal utilities over all measurable, possibly randomized rules observing Z and $V = (Q, b)$, respectively. Then $\mathcal{G}_Q(\lambda) := J_Z^* - J_V^* \ge 0$, with strict inequality exactly when, on a set of V of positive probability, both $\mathbb{P}(\eta < b/\lambda \mid V) > 0$ and $\mathbb{P}(\eta > b/\lambda \mid V) > 0$.*

Thus omitted risk information has value precisely when it separates strictly opposite optimal actions within the same proxy-and-cost group; $\mathcal{D}_Q > 0$ alone does not suffice. In the example above, $b = 1$ and $\lambda = 20/3$ place the boundary at 0.15: stopping only in the low-risk group gives $\mathcal{G}_Q = 1/6$; under the risk budget $\alpha = 1/20$, distinguishing the two groups raises the attainable savings from $1/3$ to $1/2$ (Appendix B).

4.3 Realizing the value with a learned score

For any learned score f taking values in $(0, 1)$, define its population pointwise excess cross-entropy as $\mathcal{E}_{\mathrm{pt},\nu}(f) = \mathbb{E}[\ell_{\mathrm{BCE}}(H, f(Z)) - \ell_{\mathrm{BCE}}(H, \eta(Z))]$.

Theorem 3 (Local utility of the learned rule)**.** *For $s_f(Z) = \mathbf{1}\{f(Z) < b(Z)/\lambda\}$, using natural logarithms,*

$$J_\lambda(s_f) - J_V^* \ge \mathcal{G}_Q(\lambda) - \lambda\sqrt{\mathcal{E}_{\mathrm{pt},\nu}(f)/2}. \tag{7}$$

The learned rule outperforms every proxy-and-cost controller when the information value exceeds this prediction-error penalty. The bound applies to any trained score; it does not assume that the combined objective in Eq. (3) is a proper probability loss.

For constant b, a nondecreasing map ψ makes $s_{\psi\circ f}$ a threshold rule on f. Hence the supremum utility over thresholds satisfies the same bound with $\mathcal{E}_\nu^\uparrow(f) := \inf_{\psi\ \mathrm{nondecreasing}} \mathcal{E}_{\mathrm{pt},\nu}(\psi \circ f)$ (Corollary A.1). This error depends only on score ordering, including ties, linking the bound to StopHarm's threshold rule without requiring raw-score calibration. By Theorem 1, no post-processing of Q recovers η when $\mathcal{D}_Q > 0$; for the StopHarm score, the bound requires only that a monotone post-processing of f approximate η. In the one-checkpoint example, the combined objective has an optimal StopHarm score strictly increasing in η, so $\mathcal{E}_\nu^\uparrow = 0$ and a threshold realizes the full decision value; the optimal proxy scores still coincide.

Table 2: Risk–cost comparison under MATH-source training. Acc. is accuracy (%); TR is token reduction relative to the matched Full-CoT run (%). Average gives benchmarks equal weight.

		GSM8K		MATH500		AIME 2024		AIME 2025		GPQA D.		Average	
Model	**Method**	**Acc.**	**TR%**	**Acc.**	**TR%**	**Acc.**	**TR%**	**Acc.**	**TR%**	**Acc.**	**TR%**	**Acc.**	**TR%**
Qwen3-4B	**Full-CoT**	94.5	0.00	91.8	0.00	60.0	0.00	53.3	0.00	56.1	0.00	71.2	0.00
	$TC_{Correct}$	94.0	11.33	89.0	21.81	46.7	13.24	50.0	7.25	48.0	45.40	65.5	19.81
	$TC_{Consistent}$	93.3	25.03	89.6	14.24	56.7	5.50	53.3	1.04	53.5	10.16	69.3	11.19
	LYNX	94.0	24.46	88.8	33.51	56.7	11.52	50.0	23.50	56.1	0.00	69.1	18.60
	BLADE	92.2	56.27	88.8	32.90	50.0	23.41	50.0	20.85	56.1	5.05	67.4	27.70
	StopHarm	93.8	55.67	89.2	24.50	60.0	6.78	53.3	5.86	58.1	34.08	70.9	25.38
DeepSeek-7B	**Full-CoT**	91.6	0.00	92.4	0.00	50.0	0.00	40.0	0.00	38.4	0.00	62.5	0.00
	$TC_{Correct}$	91.8	15.56	87.0	12.52	50.0	6.74	40.0	9.32	43.9	19.59	62.6	12.75
	$TC_{Consistent}$	91.4	12.35	85.2	11.91	50.0	8.20	43.3	9.47	42.9	26.20	62.6	13.63
	LYNX	90.5	40.77	86.4	28.22	46.7	3.59	36.7	11.45	39.4	11.10	59.9	19.03
	BLADE	89.8	54.63	87.0	27.14	46.7	3.19	33.3	12.76	39.9	12.77	59.4	22.10
	StopHarm	91.1	54.26	87.0	33.95	46.7	30.93	36.7	34.44	49.5	38.08	62.2	38.33
Qwen3-14B	**Full-CoT**	96.2	0.00	91.8	0.00	66.7	0.00	60.0	0.00	63.6	0.00	75.7	0.00
	$TC_{Correct}$	94.1	40.84	86.0	21.65	63.3	3.79	60.0	9.83	56.1	60.26	71.9	27.27
	$TC_{Consistent}$	94.4	36.86	89.0	11.96	60.0	8.94	56.7	10.70	59.6	18.69	71.9	17.43
	LYNX	94.8	30.53	88.0	37.39	66.7	16.41	60.0	16.38	63.6	2.49	74.6	20.64
	BLADE	96.2	18.93	90.2	32.89	60.0	28.37	53.3	27.60	61.6	21.40	72.3	25.84
	StopHarm	95.5	47.33	87.8	29.35	66.7	6.25	60.0	4.94	64.7	45.66	74.9	26.71

From local value to deployment. Theorems 2 and 3 evaluate one stopping decision under ν against completing F, and the order-invariant bound refers to the best threshold. Along a trajectory, harm requires a harmful checkpoint at or below θ, motivating the trajectory term of Section 3.1. Appendix C bounds the error of risk estimates at adaptively selected stopping checkpoints and stopping-time divergence under shared execution logic, and independent request-level certification bounds the deployed policy's risk (Proposition E.1).

5 Experiments

5.1 Experimental setup

Models, data, and evaluation protocol. We use Qwen3-4B, DeepSeek-R1-Distill-Qwen-7B (DeepSeek-7B), and Qwen3-14B, with frozen model parameters. All stopping methods use 1,400 training and 700 calibration questions from the Hendrycks MATH training split; model selection uses only the training set. We evaluate the frozen policies on MATH500, GSM8K, AIME 2024, AIME 2025, and GPQA Diamond (Appendix F.1). Full-CoT defines the reference process F. Within each model, methods share the test questions, reference trajectories, stop-and-answer protocol S, and grader G. Accuracy and cost are computed at each policy's actual answer endpoint.

Baselines. We compare with Thought Calibration (TC; Wu et al., 2025), LYNX (Akgül et al., 2025), and BLADE (Fu et al., 2026). Their supervision targets are current correctness or answer agreement, forced-answer correctness, and prefix sufficiency, respectively. We retain each method's checkpoint, probe, and calibration procedure, and report the two TC variants separately (Appendix F.2). StopHarm follows Section 3.2 with $\delta = 0.05$ and default $\alpha = 0.05$.

5.2 Risk–cost trade-off

Table 2 shows that StopHarm saves 25.4%, 38.3%, and 26.7% of tokens on Qwen3-4B, DeepSeek-7B, and Qwen3-14B, respectively, for average accuracy losses of only 0.3, 0.3, and 0.7 points relative to Full-CoT. Its operating point lies on the accuracy–token Pareto frontier among tested methods for each model. At similar savings on the Qwen models, BLADE has 3.5 and 2.7 points lower accuracy;

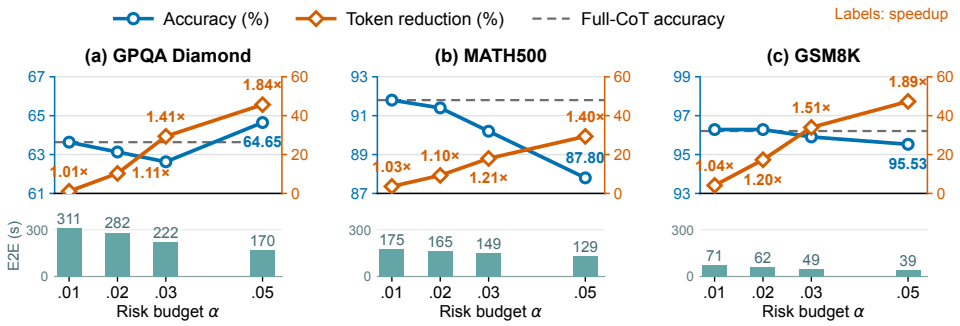


Figure 3: Risk-budget sweep for Qwen3-14B: full-test accuracy, generated-token reduction with measured speedups, and weighted end-to-end seconds per request (Full-CoT: 313.7, 180.8, 74.4 s).

Table 3: Supervision-target comparison under each model's frozen MATH-source configuration.

Model	Target	GSM8K ΔAcc.	GSM8K Red%	MATH500 ΔAcc.	MATH500 Red%	AIME 2024 ΔAcc.	AIME 2024 Red%	AIME 2025 ΔAcc.	AIME 2025 Red%	GPQA D. ΔAcc.	GPQA D. Red%	Average ΔAcc.	Average Red%
DeepSeek 7B	Correctness	−2.65	46.27	−5.00	25.84	−3.33	6.73	−3.33	5.89	+1.52	5.24	−2.56	17.99
	Consistency	−1.90	54.92	−6.20	30.76	−3.33	13.78	−3.33	13.18	+2.53	15.23	−2.45	25.57
	Last-switch	−2.58	57.29	−5.40	26.92	−6.67	13.06	−3.33	5.40	+1.52	15.64	−3.29	23.66
	StopHarm	−0.45	54.26	−5.40	33.95	−3.33	30.93	−3.33	34.44	+11.11	38.08	−0.28	38.33
Qwen3 14B	Correctness	−0.08	16.84	−1.80	18.42	0.00	1.14	0.00	0.43	−0.51	7.21	−0.48	8.81
	Consistency	−0.15	14.17	−2.00	6.29	0.00	0.00	0.00	0.00	0.00	0.00	−0.43	4.09
	Last-switch	−0.38	37.63	−2.20	7.99	0.00	0.00	0.00	0.00	0.00	2.09	−0.52	9.54
	StopHarm	−0.68	47.33	−4.00	29.35	0.00	6.25	0.00	4.94	+1.01	45.66	−0.73	26.71

on DeepSeek-7B, the largest baseline saving is 22.1%. On GPQA Diamond, StopHarm saves 34.1–45.7% while exceeding Full-CoT accuracy for all three models. On AIME, it preserves Full-CoT accuracy for both Qwen models with smaller savings (4.9–6.8%). This pattern is consistent with stopping reference failures early while retaining computation where a reference success is at risk. Sections 5.4 and 5.5 test this supervision-target explanation.

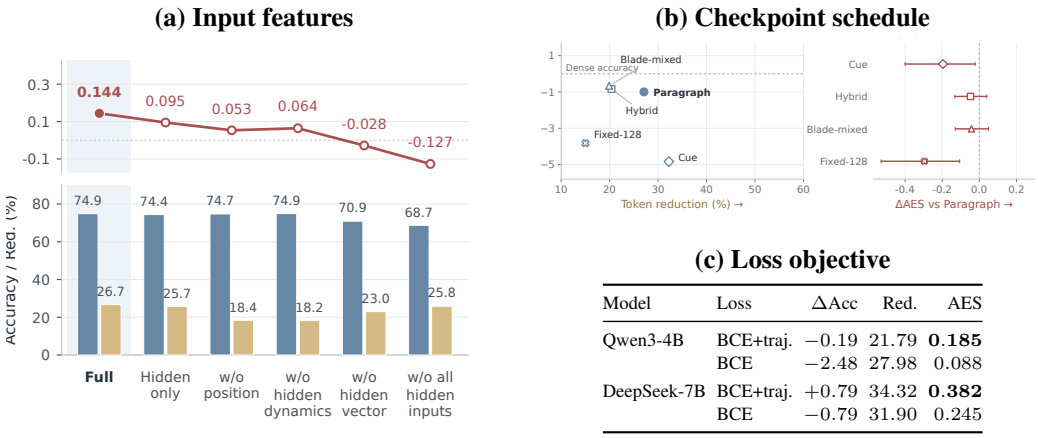
5.3 Risk-budget control of accuracy and latency

Without retraining, raising α from 0.01 to 0.05 on Qwen3-14B (Figure 3) monotonically increases the measured speedup from 1.01–1.04× to 1.40–1.89×, with the accuracy decrease below α throughout. Probing and answer-handoff overheads are negligible: the latency reduction tracks the token reduction within 0.6 points. The added savings concentrate on GSM8K and GPQA Diamond, where many stops are harm-free; on GPQA Diamond, StopHarm is 1.84× faster than Full-CoT and more accurate. At the operating points of Table 2, StopHarm attains 1.71–1.77× single-request speedups on the same three benchmarks, above every timed baseline (at most 1.60×; Appendix F.4).

5.4 Controlled comparison of supervision targets

We test the supervision-target explanation by ablating four design choices of StopHarm, starting with the **supervision target**. Correctness, Consistency, and Last-switch use the three proxy targets in Table 1 but share the model-specific reference trajectories, checkpoints, features, predictor architecture, loss, and training budget of the frozen StopHarm probe from Table 2. Harm supervision unlocks savings that these proxies forgo (Table 3): on DeepSeek-7B, StopHarm saves 38.3% rather than 18.0–25.6% of tokens while losing 0.28 rather than 2.45–3.29 points, and on Qwen3-14B, the proxies stay within 0.52 points of Full-CoT only by saving 4.1–9.5%, against 26.7% for StopHarm at a 0.73-point decrease. The gap peaks on GPQA Diamond, where Full-CoT often fails: StopHarm saves 38.1% and 45.7% of tokens, against at most 15.6% and 7.2%, and gains the most accuracy.

With the harm target fixed, Figure 4 compares the remaining choices by the accuracy–efficiency score (AES; Luo et al., 2026; Appendix F.1). For the **training objective**, adding the request-level



(c) Loss objective

Model	Loss	ΔAcc	Red.	AES
Qwen3-4B	BCE+traj.	−0.19	21.79	**0.185**
	BCE	−2.48	27.98	0.088
DeepSeek-7B	BCE+traj.	+0.79	34.32	**0.382**
	BCE	−0.79	31.90	0.245

Figure 4: (a) Input-feature, (b) checkpoint-schedule, and (c) loss-objective ablations.

trajectory term to BCE raises AES from 0.088 to 0.185 on Qwen3-4B and from 0.245 to 0.382 on DeepSeek-7B at the main seeds. On Qwen3-14B, the **input features** show that stop harm is read mainly from the hidden state: the hidden vector alone retains AES 0.095 of the full input's 0.144, whereas removing it lowers accuracy by 4.0–6.2 points and AES below zero, and the position and hidden-dynamics scalars add savings at unchanged accuracy. On the same model, paragraphs are a robust default **checkpoint schedule**: after retraining and recalibration, no alternative among LYNX reflection cues (Cue), BLADE sentence-and-cue points (Blade-mixed), their union with paragraph and fixed-budget points (Hybrid), and a 128-token grid (Fixed-128) improves AES with a paired bootstrap 95% interval above zero, and the sparser Cue and Fixed-128 fall 3.8–4.8 points below Full-CoT, against 1.0 for Paragraph. Appendix F.3.1 reports additional models. The target thus accounts for StopHarm's advantage over proxy targets, and the objective and features help realize it.

5.5 Mechanism of harm supervision

Setup. To locate the source of the gains in Table 3, we compare the frozen StopHarm probe with a Correctness probe trained with the same recipe and seed on $U = 1 - Y^S$. By the correctness of its stopping and reference answers, each checkpoint is in state CC, CW, WC, or WW. U marks WC and WW states as positive, whereas H marks only WC states, where stopping loses a correct reference answer: the two targets differ *only* on WW states. On 1,200 held-out MATH questions, we sweep each score's threshold and average the largest token reduction with empirical lost-correct rate at most x over $x \in [0, 5\%]$, in percentage points (pp) of Full-CoT tokens. This frontier area splits into a harm-free channel (YF0) from reference-failed runs ($Y^F = 0$), where stopping cannot lose a correct reference answer, and a reference-correct channel (YF1), where the two targets coincide. Ranges span the three models; brackets are 95% bootstrap intervals (Appendix F.5).

Harm supervision separates harmful from harmless errors. On MATH500, Correctness barely separates WC from WW states ($\mathrm{AUC}_{H|U}$, the AUROC for H given $U = 1$, is 0.50–0.63). StopHarm raises this AUROC by 0.08–0.21, whereas Correctness predicts the current error better by 0.03–0.07 (Figure 5a). Each probe thus learns what its target specifies. A fitted mapping of the three proxy scores does not recover this information: adding the StopHarm score lowers its held-out log loss for H on MATH500, GSM8K, and GPQA Diamond, evidence of the information gap in Theorem 1. The separation gain holds for four training seeds (12 of 12 intervals above zero), under cross-fitting, and at every token position (Appendix F.6).

The separation comes from the WW labels. Starting from the Correctness labels, we relabel as $H = 0$ the WW rows of a random fraction p of the reference-failed training questions and keep everything else fixed; $p = 1$ reproduces StopHarm exactly. Both $\mathrm{AUC}_{H|U}$ and the harm-free frontier area rise with p (Figure 5b). Relabeling as many positive rows at random (RF) or reweighting the positive class (CW) leaves both near their values at $p = 0$, and StopHarm exceeds RF in the harm-free channel by 0.52–1.72 pp, with all intervals above zero. The WW labels, not the number or weight of positive labels, thus supply both the separation and the harm-free savings.

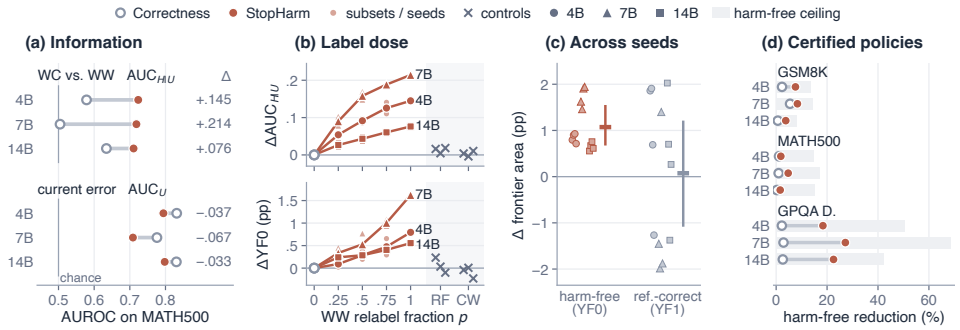


Figure 5: Mechanism of harm supervision (4B, 7B, 14B: Qwen3-4B, DeepSeek-7B, Qwen3-14B). **(a)** MATH500 AUROC for WC versus WW states (top) and the current error (bottom). **(b)** Change from $p = 0$ when the WW rows of a fraction p of reference-failed training questions are relabeled ($p = 1$: StopHarm); light points: subset draws; RF, CW: controls. **(c)** StopHarm minus Correctness frontier area by channel for 12 model–seed pairs, with the pooled mean and 95% interval. **(d)** Token reduction on reference-failed runs at certified operating points; gray tracks: harm-free ceiling.

The separation yields stable harm-free savings. At matched empirical risk, the harm-free channel of StopHarm exceeds that of Correctness by 0.55–1.62 pp, so StopHarm saves 1.5–3.1 times as many tokens on reference-failed runs. This gain and its slope in p, our preregistered primary endpoints, hold for every model after Holm correction. The gain is positive for all 12 model–seed pairs and pools to 1.07 [0.67, 1.55] pp over seeds and 1.09 [0.85, 1.38] pp under cross-fitting (Figure 5c). The reference-correct channel shows no systematic loss (0.07 $[-1.08, 1.21]$ pp). In the preregistered cross-fitted confirmatory test, the gain in total frontier area is 2.56 [1.72, 3.39] pp. At operating points certified under the same risk budget, StopHarm also saves more tokens on reference-failed runs for all nine model–benchmark pairs, most on GPQA Diamond (18.4–27.2% versus 2.2–2.9% of Full-CoT tokens; Figure 5d). Appendix F.6 traces these differences along the reasoning trajectory and reports the harm-free gain under 111 perturbations, all with intervals above zero.

6 Related Work

Training-based methods shorten reasoning by training the generator (Aggarwal & Welleck, 2025; Xia et al., 2025); early stopping instead ends a single trajectory of a frozen generator. Signal-based stoppers threshold the confidence, consistency, or entropy of trial answers (Yang et al., 2026; Fu et al., 2025b; Wang et al., 2026a), and only some failed confidence- or consistency-based exits are stop harm, as full reasoning also gets the rest wrong (Min et al., 2026). Learned stoppers and hidden-state probes predict current-answer correctness or sufficiency (Zhang et al., 2025; Akgül et al., 2025; Fu et al., 2026; Dong et al., 2026), arrival at the final answer (Wu et al., 2025; Nagle et al., 2026; Liu & Wang, 2025), or eventual success (Afzal et al., 2025), i.e., the proxy targets of Table 1; optimal-stopping analysis shows correctness thresholds can be suboptimal under an accuracy–length reward (Ehab et al., 2026). Distribution-free risk control (Angelopoulos et al., 2025; 2024) lets stoppers bound premature, wrong, or hopeless exits and probe errors (Akgül et al., 2025; Xie et al., 2025; Wang et al., 2026b; Wu et al., 2025) or lost-correct risk (Dong et al., 2026). All of them threshold given signals or proxy-trained scores; Appendix G details these and adjacent methods.

7 Conclusion

We introduced StopHarm, which directly supervises stop harm: the event that stopping yields a wrong answer although full reasoning would answer correctly. This label equals the request's lost-correct loss at the stopping checkpoint, aligning training with certification. Theoretically, even Bayes-optimal proxy scores can merge states that require opposite stopping actions, and direct supervision improves local utility when the learned score orders their risks accurately enough. Across three models and five benchmarks, StopHarm reduces generated tokens by 25.4–38.3% with average accuracy drops below one point. In controlled comparisons, harm supervision better separates recoverable errors from reference failures, unlocking harm-free savings that proxy targets forgo.

AI USE STATEMENT

In this work, we used generative AI tools to assist with code modification and with language polishing and grammatical correction of author-written text. We did not use generative AI tools to generate synthetic data, to develop theoretical models or conceptual frameworks, to formulate mathematical claims or supply critical ingredients of their proofs, to propose or refine hypotheses, to design research methodology or experiments, to produce figures or experimental results, or to interpret results; the remaining research subtasks for which disclosure is required are not applicable to this work. All experimental design, implementation decisions, analyses, results, and conclusions were produced and verified by the authors. All AI-assisted code modifications were manually reviewed and tested for correctness, and all language edits were reviewed to ensure that they faithfully preserved the authors' intended meaning. We take responsibility for the final content of this work, including any text or code modified with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

Section 2 and Appendix C define the protocols, stopping policies, and lost-correct risk. Appendix A states the theoretical results with their assumptions and proofs, Appendix B constructs the separation example, and Appendix E specifies the certification procedure and its guarantee. Section 5.1 and Appendix F.1 give the models, data splits, generation and answering protocols, and metrics; Appendices F.2, F.4, and F.5 give the baseline implementations, the timing protocol and hardware, and the mechanism-analysis protocol. The supplementary material provides the code, configurations, and scripts for checkpoint extraction, feature computation, probe training, calibration, certification, and evaluation that reproduce the results in Section 5 and Appendix F.

REFERENCES

Anum Afzal, Florian Matthes, Gal Chechik, and Yftah Ziser. Knowing before saying: LLM representations encode information about chain-of-thought success before completion. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 12791–12806, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.662. URL https://aclanthology.org/2025.findings-acl.662/.

Shivani Agarwal. Surrogate regret bounds for bipartite ranking via strongly proper losses. *Journal of Machine Learning Research*, 15(49):1653–1674, 2014. URL https://jmlr.org/papers/v15/agarwal14b.html.

Pranjal Aggarwal and Sean Welleck. L1: Controlling how long a reasoning model thinks with reinforcement learning. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=4jdIxXBNve.

Ömer Faruk Akgül, Yusuf Hakan Kalaycı, Rajgopal Kannan, Willie Neiswanger, and Viktor Prasanna. LYNX: Learning dynamic exits for confidence-controlled reasoning. arXiv preprint arXiv:2512.05325, 2025. URL https://arxiv.org/abs/2512.05325.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=33XGfHLtZg.

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then Test: Calibrating predictive algorithms to achieve risk control. *The Annals of Applied Statistics*, 19(2):1641–1662, 2025. doi: 10.1214/24-AOAS1998. URL https://arxiv.org/abs/2110.01052.

Andrea Banino, Jan Balaguer, and Charles Blundell. Pondernet: Learning to ponder. In *8th ICML Workshop on Automated Machine Learning (AutoML)*, 2021. URL https://openreview.net/forum?id=1EuxRTe0WN.

Han Bao. Proper losses, moduli of convexity, and surrogate regret bounds. In *Proceedings of the Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning*

Research, pp. 525–547. PMLR, 2023. URL https://proceedings.mlr.press/v195/bao23a.html.

Jochen Bröcker. Reliability, sufficiency, and the decomposition of proper scores. *Quarterly Journal of the Royal Meteorological Society*, 135(643):1512–1519, 2009. doi: 10.1002/qj.456.

Simone Caldarella, Davide Talon, Rahaf Aljundi, Elisa Ricci, and Massimiliano Mancini. Thinking past the answer: Evaluating harmful overthinking in large reasoning models. arXiv preprint arXiv:2606.02835, 2026. URL https://arxiv.org/abs/2606.02835.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do NOT think that much for 2+3=? on the overthinking of long reasoning models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=MSbU3L7V00.

C. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970. doi: 10.1109/TIT.1970.1054406.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Joey David. Temporal predictors of outcome in reasoning language models, 2025. URL https://arxiv.org/abs/2511.14773.

Zhe Dong, Fang Qin, and Manish Shah. When does learning to stop help? A cost-aware study of early exits in reasoning models. arXiv preprint arXiv:2606.30852, 2026. URL https://arxiv.org/abs/2606.30852.

Mohammed Ehab, Aymane El Gadarri, Vivek F. Farias, Adam Jozefiak, and Ciamac C. Moallemi. OS-Pruner: Pruning chains-of-thought of reasoning models via optimal stopping. arXiv preprint arXiv:2607.11089, 2026. URL https://arxiv.org/abs/2607.11089.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(53):1605–1641, 2010. URL http://jmlr.org/papers/v11/el-yaniv10a.html.

Adam N. Elmachtoub and Paul Grigas. Smart "predict, then optimize". *Management Science*, 68 (1):9–26, 2022. doi: 10.1287/mnsc.2020.3922.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625–630, 2024. doi: 10.1038/s41586-024-07421-0. URL https://doi.org/10.1038/s41586-024-07421-0.

Keshu Fu, Keqin Peng, Jun Bai, Shuhan Qin, Chen Li, Junzhu Liang, Yefei Chen, Jiaqi Li, and Yuanxin Ouyang. BLADE: Boundary-expanded and layer-adaptive dynamic exit for efficient LLM reasoning. arXiv preprint arXiv:2607.28966, 2026. URL https://arxiv.org/abs/2607.28966.

Yichao Fu, Junda Chen, Siqi Zhu, Zheyu Fu, Zhongdongming Dai, Yonghao Zhuang, Yian Ma, Aurick Qiao, Tajana Rosing, Ion Stoica, and Hao Zhang. Efficiently scaling LLM reasoning programs with certaindex. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a. URL https://openreview.net/forum?id=nn51ewu5k2.

Yichao Fu, Junda Chen, Yonghao Zhuang, Zheyu Fu, Ion Stoica, and Hao Zhang. Reasoning without self-doubt: More efficient chain-of-thought through certainty probing. In *ICLR 2025 Workshop on Foundation Models in the Wild*, 2025b. URL https://openreview.net/forum?id=wpK4IMJfdX.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/4a8423d5e91fda00bb7e46540e2b0cf1-Paper.pdf.

Alex Graves. Adaptive computation time for recurrent neural networks, 2017. URL https://arxiv.org/abs/1603.08983.

Bairu Hou, Yang Zhang, Jiabao Ji, Yujian Liu, Kaizhi Qian, Jacob Andreas, and Shiyu Chang. Thinkprune: Pruning long chain-of-thought of LLMs via reinforcement learning. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=V51gPu1uQD.

Liwei Kang, Yue Deng, Yao Xiao, Zhanfeng Mo, Wee Sun Lee, and Lidong Bing. First try matters: Revisiting the role of reflection in reasoning models, 2025. URL https://arxiv.org/abs/2510.08308.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. Semantic entropy probes: Robust and cheap hallucination detection in LLMs. In *ICML 2024 Workshop on Foundation Models in the Wild*, 2024. URL https://arxiv.org/abs/2406.15927.

Meelis Kull and Peter Flach. Novel decompositions of proper scoring rules for classification: Score adjustment as precursor to calibration. In *Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2015)*, Lecture Notes in Computer Science, pp. 68–85. Springer, 2015. doi: 10.1007/978-3-319-23528-8_5.

Yassir Laaouach. 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*, 2025. URL https://openreview.net/forum?id=CX5c7C1CZa.

Yiwei Li, Peiwen Yuan, Shaoxiong Feng, Boyuan Pan, Xinglin Wang, Bin Sun, Heda Wang, and Kan Li. Escape sky-high cost: Early-stopping self-consistency for multi-step reasoning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ndR8Ytrzhh.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Xin Liu and Lu Wang. Answer convergence as a signal for early stopping in reasoning. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 17896–17907, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.904. URL https://aclanthology.org/2025.emnlp-main.904/.

Haotian Luo, Haiying He, Yibo Wang, Shiwei Liu, Wei Li, Xiaochun Cao, Dacheng Tao, Naiqiang Tan, and Li Shen. O1-pruner: Length-harmonizing fine-tuning for o1-like reasoning pruning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 14242–14257, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.697. URL https://aclanthology.org/2026.findings-acl.697/.

Minjia Mao, Bowen Yin, Yu Zhu, and Xiao Fang. Early stopping chain-of-thoughts in large language models, 2026. URL https://arxiv.org/abs/2509.14004.

Dehai Min, Giovanni Vaccarino, Huiyi Chen, Yongliang Wu, Gal Yona, and Lu Cheng. Stop when reasoning converges: Semantic-preserving early exit for reasoning models, 2026. URL https://arxiv.org/abs/2605.17672.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling, 2025. URL https://arxiv.org/abs/2501.19393.

Alliot Nagle, Jakhongir Saydaliev, Dhia Garbaya, Michael Gastpar, Ashok Vardhan Makkuva, and Hyeji Kim. Terminator: Learning optimal exit points for early stopping in chain-of-thought reasoning, 2026. URL https://arxiv.org/abs/2603.12529.

Alexandre Perez-Lebel, Marine Le Morvan, and Gaël Varoquaux. Beyond calibration: Estimating the grouping loss of modern neural networks. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.16315.

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

Tal Schuster, Adam Fisch, Jai Gupta, Mostafa Dehghani, Dara Bahri, Vinh Q. Tran, Yi Tay, and Donald Metzler. Confident adaptive language modeling. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://arxiv.org/abs/2207.07061.

Amrith Setlur, Chirag Nagpal, Adam Fisch, Xinyang Geng, Jacob Eisenstein, Rishabh Agarwal, Alekh Agarwal, Jonathan Berant, and Aviral Kumar. Rewarding progress: Scaling automated process verifiers for LLM reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=A6Y7AqlzLW.

Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4FWAwZtd2n.

Surat Teerapittayanon, Bradley McDanel, and H. T. Kung. Branchynet: Fast inference via early exiting from deep neural networks. *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 2464–2469, 2016. URL https://api.semanticscholar.org/CorpusID:2916466.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-Shepherd: Verify and reinforce LLMs step-by-step without human annotations. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9426–9439, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.510. URL https://aclanthology.org/2024.acl-long.510/.

Xi Wang, James McInerney, Lequn Wang, and Nathan Kallus. Entropy after </Think> for reasoning model early exiting. arXiv preprint arXiv:2509.26522, 2026a. URL https://arxiv.org/abs/2509.26522.

Xi Wang, Anushri Suresh, Alvin Zhang, Rishi More, William Jurayj, Benjamin Van Durme, Mehrdad Farajtabar, Daniel Khashabi, and Eric Nalisnick. Conformal thinking: Risk control for reasoning on a compute budget. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://arxiv.org/abs/2602.03814.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Menghua Wu, Cai Zhou, Stephen Bates, and Tommi Jaakkola. Thought calibration: Efficient and confident test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 14291–14305, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.722. URL https://aclanthology.org/2025.emnlp-main.722/.

Heming Xia, Chak Tou Leong, Wenjie Wang, Yongqi Li, and Wenjie Li. TokenSkip: Controllable chain-of-thought compression in LLMs. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3351–3363, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.165. URL https://aclanthology.org/2025.emnlp-main.165/.

Violet Xiang, Chase Blagden, Rafael Rafailov, Nathan Lile, Sang Truong, Chelsea Finn, and Nick Haber. Just enough thinking: Efficient reasoning with adaptive length penalties reinforcement learning, 2025. URL https://arxiv.org/abs/2506.05256.

Yangxinyu Xie, Tao Wang, Soham Mallick, Yan Sun, Georgy Noarov, Mengxin Yu, Tanwi Mallick, Weijie J Su, and Edgar Dobriban. Statistical early stopping for reasoning models. In *NeurIPS 2025 Workshop on Efficient Reasoning*, 2025. URL https://openreview.net/forum?id=cmi4FBheRq.

Ji Xin, Raphael Tang, Jaejun Lee, Yaoliang Yu, and Jimmy Lin. DeeBERT: Dynamic early exiting for accelerating BERT inference. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2246–2251, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.204. URL https://aclanthology.org/2020.acl-main.204/.

Chenxu Yang, Qingyi Si, Yongjie Duan, Zheliang Zhu, Chenyu Zhu, Qiaowei Li, Minghui Chen, Zheng Lin, and Weiping Wang. Dynamic early exit in reasoning models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=NpU7ZXafRi.

Anqi Zhang, Yulin Chen, Jane Pan, Chen Zhao, Aurojit Panda, Jinyang Li, and He He. Reasoning models know when they're right: Probing hidden states for self-verification. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=O6I0Av7683.

Qingjie Zhang, Yujia Fu, Yang Wang, Liu Yan, Tao Wei, Ke Xu, Minlie Huang, and Han Qiu. Stop before you fail: Operational capability boundaries for mitigating unproductive reasoning in large reasoning models, 2026. URL https://arxiv.org/abs/2509.24711.

Wangchunshu Zhou, Canwen Xu, Tao Ge, Julian McAuley, Ke Xu, and Furu Wei. Bert loses patience: Fast and robust inference with early exit. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 18330–18341. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/d4dd111a4fd973394238aca5c05bebe3-Paper.pdf.

A Full Statements and Proofs of the Theoretical Results

Throughout this appendix, expectations are taken with respect to the request-balanced checkpoint distribution ν of Section 3.1 unless marked otherwise, and all predictors share the features Z, the stop-and-answer protocol, and the reference labels. Here Z denotes the available compressed features of the online history. Let P cover the requests and the generation randomness, let K be the number of checkpoints of a request, and let $p_+ = \mathbb{P}_P(K > 0) > 0$. For a quantity g_t defined on the checkpoints of a request,

$$\mathbb{E}_\nu[g] = \frac{1}{p_+}\mathbb{E}_P\Big[\mathbf{1}\{K > 0\}\frac{1}{K}\sum_{t=1}^{K} g_t\Big]. \tag{A.1}$$

A.1 Supervision targets and conditional predictors

Lemma A.1 (Bayes predictor of a supervision task)**.** *For any binary label U, let*

$$\mathcal{L}_U(f) = \mathbb{E}\big[-U\log f(Z) - (1-U)\log(1-f(Z))\big]. \tag{A.2}$$

Over all measurable probabilistic predictors, the optimal predictor is $\eta_U(Z) = \mathbb{P}_\nu(U = 1 \mid Z)$*, and*

$$\mathcal{L}_U(f) - \mathcal{L}_U(\eta_U) = \mathbb{E}\ \mathrm{KL}\big(\mathrm{Bern}(\eta_U(Z)) \,\|\, \mathrm{Bern}(f(Z))\big). \tag{A.3}$$

This property follows from the strict propriety of the log loss: given Z, the excess cross-entropy risk is exactly the KL divergence between two Bernoulli distributions, which is nonnegative and vanishes only when $f = \eta_U$; taking the expectation over Z yields Eq. (A.3). Related theory can be found in Agarwal (2014) and Bao (2023). By Eq. (A.1), $\mathcal{L}_U$ is the population form of the pointwise term of Eq. (3) with label U; taking $U = H$ defines $\mathcal{L}_{\mathrm{pt},\nu} := \mathcal{L}_H$, with Bayes predictor η.

The joint event also admits the factorization

$$\eta(Z) = \mathbb{P}(Y^S = 0 \mid Z)\, \mathbb{P}(Y^F = 1 \mid Y^S = 0, Z). \tag{A.4}$$

Accurately estimating the two conditional factors in Eq. (A.4) also recovers the stop-harm probability. The second factor describes the probability of reference success given that stopping now is wrong, and thus retains the conditional dependence between the two outcomes.

A.2 Full statement and proof of Theorem 1

Let $m_Q(Q) = \mathbb{E}[\eta(Z) \mid Q] = \mathbb{P}(H = 1 \mid Q)$. In addition to Eq. (5), the same information gap appears in the Brier loss on the observed label H:

$$\inf_\psi \mathbb{E}\big[(H - \psi(Q))^2\big] - \mathbb{E}\big[(H - \eta(Z))^2\big] = \mathcal{D}_Q. \tag{A.5}$$

For the log loss,

$$\inf_\psi \mathcal{L}_{\mathrm{pt},\nu}(\psi \circ Q) - \mathcal{L}_{\mathrm{pt},\nu}(\eta) = \mathbb{E}\ \mathrm{KL}\big(\mathrm{Bern}(\eta) \,\|\, \mathrm{Bern}(m_Q(Q))\big) = I(H; Z \mid Q). \tag{A.6}$$

Proof. Conditional expectation yields the orthogonal decomposition

$$\mathbb{E}\big[(\eta - \psi(Q))^2\big] = \mathbb{E}\big[(\eta - m_Q(Q))^2\big] + \mathbb{E}\big[(m_Q(Q) - \psi(Q))^2\big]. \tag{A.7}$$

The minimum is attained at $\psi = m_Q$, and the first term is $\mathcal{D}_Q$; moreover, $\mathcal{D}_Q = 0$ if and only if $\eta = m_Q$ almost surely, that is, if and only if η is measurable with respect to Q. Decomposing the Brier loss on the label in the same way, the common conditional Bernoulli noise $\eta(1 - \eta)$ cancels, which gives Eq. (A.5). Under the log loss, the best score post-processing is again m_Q, and Lemma A.1 yields the KL form. The two optimal log losses are the conditional entropies of H given Q and given Z, respectively; since Q is a function of Z, their difference is $I(H; Z \mid Q)$. □

When $\mathcal{D}_Q = 0$, a score post-processing that allows non-monotone maps suffices to recover the risk. When hidden states, positions, histories, or other scores are added to the input of the post-processor, the corresponding information gap is defined with the augmented Q.

A.3 Full statement and proof of Theorem 2

Here $b(Z) > 0$ is the conditional expected computation saved by stopping relative to completing the reference process F, with $\mathbb{E}[b] < \infty$. Both controllers share this local benefit. Because the cost itself can also distinguish states, the proxy controller observes b as well. The utility J_λ in Eq. (6) is obtained from the Lagrangian of the risk-constrained problem $\max_s \mathbb{E}[bs]$ subject to $\mathbb{E}[\eta s] \le \alpha$ by dropping the policy-independent constant.

Let $m_V(V) = \mathbb{E}[\eta \mid V]$. When each controller may use any measurable randomized stopping rule of its available information, the optimal utilities of the full-feature controller and of the proxy controller are, respectively,

$$J_Z^* = \mathbb{E}\big[(b - \lambda\eta)_+\big], \qquad J_V^* = \mathbb{E}\big[(b - \lambda m_V(V))_+\big], \tag{A.8}$$

where $(x)_+ = \max\{x, 0\}$.

Proof. The objective is linear in s, so pointwise maximization gives the optimal full-feature rule $s^*(Z) = \mathbf{1}\{b > \lambda\eta(Z)\}$. For rules that depend only on V, both b and s are V-measurable; replacing η by its conditional mean m_V and maximizing pointwise in the same way yields J_V^*. Since b is known given V, the conditional Jensen inequality applied to the positive-part function gives $\mathbb{E}[(b - \lambda\eta)_+ \mid V] \geq (b - \lambda m_V)_+$, and taking expectations yields $\mathcal{G}_Q(\lambda) \geq 0$. Strictness comes from the different slopes of the positive-part function on the two sides of zero. If, given V, the quantity $b - \lambda\eta$ takes both positive and negative values with positive probability, the conditional Jensen inequality is strict; if its values lie almost surely on one side, equality holds. Hence the two-sided condition holds on a set of values of V with positive probability if and only if $\mathcal{G}_Q(\lambda) > 0$. □

In Theorems 2 and 3, b denotes the expected computation saved relative to completing F. StopHarm's test $q_t \leq \theta$ corresponds to a constant benefit with θ in the role of b/λ, and Corollary A.1 extends this correspondence to any monotone post-processing of the score; at equality, either action may be chosen by a prespecified rule.

A.4 Proof of Theorem 3 and order invariance

By Lemma A.1 with $U = H$, the pointwise excess risk is

$$\mathcal{E}_{\mathrm{pt},\nu}(f) = \mathbb{E}\ \mathrm{KL}\big(\mathrm{Bern}(\eta) \,\|\, \mathrm{Bern}(f)\big). \tag{A.9}$$

Theorem 3 follows from the local regret bound

$$J_Z^* - J_\lambda(s_f) \leq \lambda\mathbb{E}|f - \eta| \leq \lambda\sqrt{\mathcal{E}_{\mathrm{pt},\nu}(f)/2}. \tag{A.10}$$

Proof. Regret arises only where f and η fall on different sides of the boundary b/λ, and its magnitude there is $|b - \lambda\eta|$. Since the boundary then lies between f and η, we have $|b - \lambda\eta| \leq \lambda|f - \eta|$. Taking expectations and applying the Bernoulli Pinsker inequality and the Cauchy–Schwarz inequality yields Eq. (A.10). Eq. (7) then follows from $J_\lambda(s_f) - J_V^* = \mathcal{G}_Q(\lambda) - (J_Z^* - J_\lambda(s_f))$. □

The bound holds for every read-out f, including one trained with the complete objective of Eq. (3): $\mathcal{E}_{\mathrm{pt},\nu}(f)$ is a population quantity relative to η, whereas the training loss measures empirical fit. For constant benefit, the best-threshold bound below is invariant to strictly increasing score transformations; the raw pointwise excess risk need not be.

Corollary A.1 (Order invariance)**.** *For any nondecreasing $\psi : (0,1) \to [0,1]$, the rule $s_{\psi\circ f}(Z) = \mathbf{1}\{\psi(f(Z)) < b(Z)/\lambda\}$ satisfies*

$$J_\lambda(s_{\psi\circ f}) - J_V^* \geq \mathcal{G}_Q(\lambda) - \lambda\sqrt{\mathcal{E}_{\mathrm{pt},\nu}(\psi \circ f)/2}. \tag{A.11}$$

If b is constant, every $s_{\psi\circ f}$ is a threshold rule on f, so the supremum of J_λ over threshold rules on f exceeds J_V^* by at least $\mathcal{G}_Q(\lambda) - \lambda\sqrt{\mathcal{E}_\nu^\uparrow(f)/2}$, where $\mathcal{E}_\nu^\uparrow(f) = \inf_\psi \mathcal{E}_{\mathrm{pt},\nu}(\psi \circ f)$. The threshold class includes open and closed sublevel sets and the constant rules. Restricting to closed sublevel sets gives the same supremum, since an open cutoff is approached from below and $\mathbb{E}[b] < \infty$. The quantity $\mathcal{E}_\nu^\uparrow(f)$ is invariant under strictly increasing transformations of f and vanishes when η is a nondecreasing function of f.

Proof. Theorem 3 and Eq. (A.10) use only that the score takes values in $[0,1]$; applying them to $\psi \circ f$ gives Eq. (A.11). For constant b, the set $\{u : \psi(u) < b/\lambda\}$ is an initial segment of $(0,1)$, so $s_{\psi\circ f}$ stops exactly when f lies below a threshold. If h is strictly increasing, the nondecreasing post-processings of $h \circ f$ and of f generate the same set of scores, which gives the invariance. If $\eta = \psi_0 \circ f$ for a nondecreasing ψ_0, then $\mathcal{E}_{\mathrm{pt},\nu}(\psi_0 \circ f) = 0$. □

A.5 Corresponding result under a fixed risk budget

For the risk-constrained problem $\max_s \mathbb{E}[bs]$ subject to $\mathbb{E}[\eta s] \leq \alpha$, the class of full-feature policies contains every policy that depends only on V, so under the same true risk budget the optimal savings of the former weakly dominate those of the latter. For a learned predictor with a known bound

$\mathbb{E}|f-\eta| \le e$, the predicted-risk constraint can be tightened to $\alpha - e$. Since every stopping rule s satisfies $|\mathbb{E}[fs]-\mathbb{E}[\eta s]| \le e$, provided that $\alpha \ge 2e$ and the corresponding population optimization is solved, the selected rule has true risk at most α, and its savings are at least the full-feature optimum under budget $\alpha - 2e$. If this value exceeds the optimal savings of rules depending only on V under budget α, the learned rule strictly dominates under the same true risk budget. By Eq. (A.10), one may take $e = \sqrt{\mathcal{E}_{\mathrm{pt},\nu}(f)/2}$. This comparison presupposes a known bound on the population error and the solution of the population constrained problem. The finite-candidate certification of Section 3.2 instead directly tests the request-level losses of each complete candidate policy.

B Example 1: Identical Proxy Scores with Different Stop Harm

Example 1. Consider two compressed feature groups A and B, each with probability $1/2$. Every trajectory has a single available checkpoint, followed by the end of the reference. Table B.1 lists the conditional probabilities in the two groups.

Table B.1: A mathematical construction in which the Bayes-optimal predictions of the proxy tasks coincide while stop harm differs. Each group contains several distinct prefixes; the stop-and-answer outcome can be a deterministic function of the full prefix.

Feature group	Stopping wrong	Disagreement	Later switch	Reference correct	Stop harm
A	0.50	0.40	0.40	0.50	0.10
B	0.50	0.40	0.40	0.50	0.20

The vector of Bayes-optimal predictions of the four proxy targets is identical in the two groups, so the best stop-risk post-processing can only output 0.15. The Bayes-optimal stop-harm prediction, in contrast, separates $\eta(A) = 0.10$ from $\eta(B) = 0.20$; hence

$$\mathcal{D}_Q = \tfrac{1}{2}(0.10 - 0.15)^2 + \tfrac{1}{2}(0.20 - 0.15)^2 = \frac{1}{400} > 0. \tag{B.1}$$

Joint distribution. Let the ground-truth answer be 7. Within each group, the stopping answer and the reference output are distributed as in Table B.2.

Table B.2: Within-group joint distribution of Example 1.

Group	Stopping answer a^S	Reference output a^F	Within-group probability
A	7	7	0.4
A	7	5	0.1
A	5	7	0.1
A	5	5	0.2
A	5	9	0.2
B	7	7	0.3
B	7	5	0.2
B	5	7	0.2
B	5	5	0.3

Each row of the table can correspond to a distinct type of full prefix, and these types are compressed by Φ into the same group label. Both the stopping answer and the reference answer can be set as deterministic functions of the full-prefix type; the conditional probabilities in the table arise from mixing different prefixes within a group.

In A, the probability that stopping is wrong is $0.1 + 0.2 + 0.2 = 0.5$, the probability of answer disagreement is $0.1 + 0.1 + 0.2 = 0.4$, the probability of reference success is $0.4 + 0.1 = 0.5$, and the stop-harm probability is 0.1. In B, the corresponding probabilities are $0.2 + 0.3 = 0.5$, $0.2 + 0.2 = 0.4$, $0.3 + 0.2 = 0.5$, and 0.2. Since only the reference output follows the checkpoint, the event of a later switch coincides with the current output differing from the reference output. The vector of the four proxy posteriors is therefore the same in both groups:

$$Q(A) = Q(B) = \left(\tfrac{1}{2}, \tfrac{2}{5}, \tfrac{2}{5}, \tfrac{1}{2}\right), \qquad \eta(A) = \tfrac{1}{10}, \quad \eta(B) = \tfrac{1}{5}. \tag{B.2}$$

All entries of Table B.1 thus follow from the single joint distribution in Table B.2.

Decision value and fixed risk budget. Let $b(A) = b(B) = 1$ and $\lambda = 20/3$. Then $b(A) - \lambda\eta(A) = 1/3$ and $b(B) - \lambda\eta(B) = -1/3$. The full-feature rule stops in A and continues in B, so $J_Z^* = \frac{1}{2} \times \frac{1}{3} = \frac{1}{6}$. Since Q and b coincide in the two groups, $m_V = 3/20$ and $b - \lambda m_V = 0$; hence $J_V^* = 0$, and the decision value is $\mathcal{G}_Q(20/3) = 1/6$. Under the risk budget $\alpha = 1/20$, let s_A and s_B be the stopping probabilities in the two groups; the risk is then $\frac{1}{20}s_A + \frac{1}{10}s_B$, and the savings are $\frac{1}{2}(s_A + s_B)$. Stopping only in A attains a risk of exactly $1/20$ with savings $1/2$. A rule that reads only V must set $s_A = s_B = \bar{s}$, with risk $\frac{3}{20}\bar{s}$; feasibility requires $\bar{s} \le 1/3$, so the maximum savings are $1/3$.

The construction under the training objective. Each request has one checkpoint, so m_i in Eq. (4) equals the logit of that checkpoint, and the trajectory term reduces to $-\log f(Z)$ on the positive requests of a minibatch. For a binary target U with prevalence $p_U \in (0, 1)$ and minibatches of $|\mathcal{M}|$ independent requests, the set $\mathcal{M}_+$ is nonempty with probability $\gamma_U = 1 - (1 - p_U)^{|\mathcal{M}|}$, and the expected trajectory term is $-\gamma_U \mathbb{E}[\log f(Z) \mid U = 1]$. Over measurable scores, the expected objective of Eq. (3) is therefore a weighted cross-entropy with Bayes predictor

$$f_U^*(Z) = g_{w_U}(\eta_U(Z)), \qquad g_w(u) = \frac{wu}{1 - u + wu}, \qquad w_U = 1 + \frac{\rho\,\gamma_U}{p_U}. \tag{B.3}$$

The map g_w is strictly increasing. The proxy posteriors coincide in the two groups, and so do their optimal scores under Eq. (3); with $|\mathcal{M}| = 24$ and $\rho = 1$, these are $(0.75, 0.70, 0.70, 0.75)$ in both groups. The optimal StopHarm scores are 0.456 in A and 0.653 in B, a strictly increasing function of η, so their excess risk after monotone post-processing in Corollary A.1 is zero. Every threshold in $(0.456, 0.653)$ stops in A and continues in B, attaining $J_Z^* = 1/6$, whereas every controller reading the optimal proxy scores attains $J_V^* = 0$. The information gap in Eq. (B.1) and the decision value $\mathcal{G}_Q(20/3) = 1/6$ thus carry over to the training objective.

A special case with sufficient features. The construction relies on Z compressing the full prefix. If the error of stopping now is a deterministic function of Z, that is, $1 - Y^S = e(Z)$ for some $e(Z) \in \{0, 1\}$, then $\eta(Z) = e(Z)\,\mathbb{P}(Y^F = 1 \mid Z)$, and the exact posteriors of the current error and of reference success already determine stop harm jointly. More generally, the two exact conditional factors in Eq. (A.4) also determine η. The information gap of proxy scores therefore depends on their specific relationship to the given features and the joint label.

C From Checkpoint Prediction to Policy Execution

C.1 Protocols, stopping policies, and paired outcomes

Let a request and its ground-truth answer (X, Y^*) follow a target distribution P, which also covers the generation randomness. The reference process F generates the complete reasoning and the reference output a^F under the prespecified decoding protocol and budget B; termination and answer extraction upon budget exhaustion belong to the protocol, and $Y^F = G(a^F, Y^*)$ for the grading rule G. Checkpoints are paragraph boundaries of the reference trajectory, so their occurrence depends only on the content generated so far; the trajectory has K checkpoints, the t-th at reasoning-token position n_t. At checkpoint t, the stop-and-answer protocol S ends thinking, switches to answer decoding with a fixed prompt, decoding parameters, and parsing rule, and produces a_t^S with $Y_t^S = G(a_t^S, Y^*)$. Offline labeling, policy replay, and online execution use the same S.

Let $\mathcal{F}_t$ be the history available online at checkpoint t, including the prefix and the current and past features and scores. A stopping policy π has stopping time $\tau \in \{1, \dots, K\} \cup \{\star\}$, where $\star$ indicates that no early stop occurs; each $A_{\pi,t} = \mathbf{1}\{\tau = t\}$ is $\mathcal{F}_t$-measurable, and $\sum_t A_{\pi,t} \le 1$. A continue action advances along F to the next checkpoint. The policy output has correctness $Y^\pi = Y_t^S$ if $\tau = t$ and $Y^\pi = Y^F$ otherwise. Offline, each stopping branch copies the generation state of the reference prefix and uses randomness independent of reference decoding, so the reference trajectory is unperturbed and every stopping outcome is paired with the reference outcome of its own trajectory.

Proof of Proposition 1. If the policy stops at t, then $Y^\pi = Y_t^S$, and only this position contributes to the right-hand side of Eq. (2); if it never stops, then $Y^\pi = Y^F$ and both sides vanish. Taking expectations yields the risk identity. □

C.2 Full-history risk

Let $M_{\mathrm{cp}} = \mathbb{E}_P[K] \in (0, \infty)$, let $p_t = \mathbb{P}(H_t = 1 \mid \mathcal{F}_t)$ be the full-history conditional risk, and let f_t be any $\mathcal{F}_t$-measurable probability score. The existence indicator $\mathbf{1}\{K \geq t\}$ and the current checkpoint are measurable with respect to the history up to that position; later checkpoints in the analysis are potential checkpoints of the same reference process, and an online policy always uses only the history generated before it stops. The average KL error of f relative to the full-history conditional risk is

$$\mathcal{E}_{\mathrm{hist}}(f) = \frac{1}{M_{\mathrm{cp}}} \mathbb{E}_P \sum_{t=1}^{K} \mathrm{KL}\big(\mathrm{Bern}(p_t) \,\|\, \mathrm{Bern}(f_t)\big). \tag{C.1}$$

Proposition C.1 (Full-history prediction error and policy risk estimation)**.** *For any causal stopping policy π,*

$$\left| R_{\mathrm{LC}}(\pi) - \mathbb{E}_P \sum_t A_{\pi,t} f_t \right| \leq \sqrt{M_{\mathrm{cp}} \mathcal{E}_{\mathrm{hist}}(f)/2}. \tag{C.2}$$

Proof. By Eq. (2) and the $\mathcal{F}_t$-measurability of $A_{\pi,t}$, the tower property of conditional expectation gives $R_{\mathrm{LC}}(\pi) = \mathbb{E}_P \sum_t A_{\pi,t} p_t$. Since each request has at most one selected position, the Cauchy–Schwarz inequality gives $\left|\mathbb{E}_P \sum_t A_{\pi,t}(p_t - f_t)\right| \leq \left(\mathbb{E}_P \sum_t (p_t - f_t)^2\right)^{1/2}$; applying the Bernoulli Pinsker inequality to each term and substituting Eq. (C.1) yields the claim. □

If $f_t = f(Z_t)$ reads compressed features, $\mathcal{E}_{\mathrm{hist}}$ still takes the full-history posterior p_t as its reference, thereby accounting for both the history information omitted by the compressed features and the prediction error of the read-out. The actual stopping events of the deployed policy are measurable with respect to the online history, so the bound applies to it as well.

C.3 Stopping-time stability under a common decision boundary

Consider two scores on the same features, where Z_t denotes the features used by the current read-out and η is the posterior of Section 4. The two policies run along the same reference trajectory, share the stop-and-answer outcomes, a predictable boundary c_t, and checkpoint eligibility, and apply the same fixed, deterministic, causal stopping logic to the history of threshold decisions. Apart from these decisions, their inputs, initial states, and update rules are identical; the stopping logic does not separately read the differing numerical scores. Denote their stopping times by τ_f and τ_η, and define

$$d_t^f = \mathbf{1}\{f(Z_t) \leq c_t\}, \qquad d_t^\eta = \mathbf{1}\{\eta(Z_t) \leq c_t\}. \tag{C.3}$$

These decisions are evaluated only at commonly eligible checkpoints; the deployed test $q_t \leq \theta$ uses the constant boundary $c_t = \theta$. For any $\varepsilon > 0$, define the expected number of eligible checkpoints near the boundary as

$$\mathcal{B}_c(\varepsilon) = \mathbb{E}_P \sum_{t=1}^{K} \mathbf{1}\{t \text{ is eligible for a threshold decision}\}\, \mathbf{1}\{|\eta(Z_t) - c_t| \leq \varepsilon\}. \tag{C.4}$$

Proposition C.2 (Stopping-time stability under a common boundary)**.** *For any $\varepsilon > 0$,*

$$\mathbb{P}_P(\tau_f \neq \tau_\eta) \leq \min\Big\{1,\ \mathcal{B}_c(\varepsilon) + \frac{p_+}{2\varepsilon^2} \mathbb{E}_\nu\big[K\ \mathrm{KL}\big(\mathrm{Bern}(\eta) \,\|\, \mathrm{Bern}(f)\big)\big]\Big\}. \tag{C.5}$$

If $K \leq K_{\max}$ almost surely, the second term is at most $p_+ K_{\max} \mathcal{E}_{\mathrm{pt},\nu}(f)/(2\varepsilon^2)$.

Proof. If the two policies have identical threshold-decision sequences, their shared stopping logic produces identical stopping times. Hence differing stopping times imply a differing threshold decision at some commonly eligible checkpoint. If $|\eta(Z_t) - c_t| > \varepsilon$ at that checkpoint, different decisions require $|f(Z_t) - \eta(Z_t)| > \varepsilon$. Applying the union bound and Markov's inequality to these two kinds of checkpoints, then the identity $\mathbb{E}_P \sum_t g_t = p_+ \mathbb{E}_\nu[Kg]$ from Eq. (A.1) and the Bernoulli Pinsker inequality, yields Eq. (C.5). □

When $\tau_f = \tau_\eta$, the shared stop-and-answer and reference-fallback protocols give the two policies the same output correctness, so $|R_{\mathrm{LC}}(\pi_f) - R_{\mathrm{LC}}(\pi_\eta)| \le \mathbb{P}(\tau_f \neq \tau_\eta)$. If the normalized cost satisfies $\bar{C}_\pi \in [0,1]$ and the two policies incur the same cost whenever their stopping times coincide, the expected cost difference obeys the same bound; if the two read-outs have different additional overheads, the overhead difference must be accounted for separately. If there exist $\varepsilon_m \downarrow 0$ such that $\mathcal{B}_c(\varepsilon_m) \to 0$ and $\mathbb{E}_\nu[K\ \mathrm{KL}(\mathrm{Bern}(\eta) \,\|\, \mathrm{Bern}(f_m))]/\varepsilon_m^2 \to 0$, then Eq. (C.5) implies that the stopping times agree with probability tending to one, and the differences in risk and in the above expected cost also vanish. This result compares the learned rule and the ideal-score rule under a common boundary and shared execution logic; policies with different thresholds obtained by separate certification are evaluated through their respective request-level losses.

C.4 Effect of adaptive selection on risk estimation

Consider two equally likely request runs, both with a correct reference and each with four checkpoints:

Request	Checkpoint 1	Checkpoint 2	Checkpoint 3	Checkpoint 4
Path A	$A,\ H_1 = 1$	$A,\ H_2 = 1$	shared, $H_3 = 1$	shared, $H_4 = 1$
Path B	$B,\ H_1 = 0$	$B,\ H_2 = 0$	shared, $H_3 = 0$	shared, $H_4 = 0$

Each entry gives the feature Z_t and harm label H_t. Both requests have four checkpoints, so ν weights the eight checkpoints equally, and the exact posteriors are $\eta(A) = 1$, $\eta(B) = 0$, and $\eta(\mathrm{shared}) = 1/2$. With the constant boundary 0.6 and stopping after two consecutive qualifying checkpoints, path B stops at the second checkpoint and path A at the fourth. The true policy risk is $1/2$, whereas the expected score at the selected positions is $\mathbb{E}\sum_t A_{\pi,t}\eta(Z_t) = \frac{1}{2}\times 0 + \frac{1}{2}\times\frac{1}{2} = \frac{1}{4}$.

Since the posteriors above are exact, the discrepancy arises entirely from adaptive selection: only path A reaches the third checkpoint without having stopped, and this history information is not contained in the current compressed features. The full-history conditional risk captures this selection information; Proposition C.1 therefore uses it as the error reference, and request-level certification measures the loss directly at the actual stopping position.

D MAIN NOTATION

Symbol	Definition
F, S, G	Fixed reference process, stop-and-answer protocol, and grading rule
B, n_t, ℓ_t^S	Generation budget, reasoning tokens at checkpoint t, and generated answer tokens of S at t
$C(\pi)$	Generated-token cost of a policy at the point where it answers
K, p_+, M_{cp}	Number of checkpoints of a request, $\mathbb{P}_P(K > 0)$, and $\mathbb{E}_P[K]$
Y^F, Y_t^S, Y^π	Correctness of the reference, of stopping now, and of the policy output
$H_t, \mathcal{T}_i^+$	Joint event that stopping now is wrong and the reference is correct, and the harmful checkpoints of trajectory i
$\mathcal{F}_t, Z_t$	Full online history, and the compressed features actually read out
$\tau, A_{\pi,t}$	Stopping time, and indicator of the actual stopping position
a_t, q_t	Probe logit and stop-harm score $q_t = f_\phi(Z_t) = \sigma(a_t)$
ρ, β, m_i	Weight of the trajectory term, soft-min temperature, and normalized soft-min of the harmful logits of trajectory i
θ	Certified score threshold
$\nu, \eta(Z)$	Request-balanced checkpoint distribution, and the stop-harm probability under it, which is the Bayes predictor of the pointwise term
Q, ψ	Fixed score of a proxy predictor, and a post-processing map
$\mathcal{D}_Q$	Stop-harm information that post-processing reading only Q cannot recover (Theorem 1)
b, λ	Expected computation saved relative to completing F, and price of risk
s, s^*, s_f	Stopping rule, optimal rule under full risk information, and learned-score rule
$J_\lambda, V = (Q, b)$	Local risk–compute utility, and the score and cost information available to the proxy controller
$\mathcal{G}_Q(\lambda)$	Decision value of risk information, i.e., the optimal utility gain of full risk information over proxy information (Theorem 2)
$\mathcal{E}_{\mathrm{pt},\nu}, \mathcal{E}_\nu^\uparrow$	Pointwise excess cross-entropy of a read-out relative to η, and its value after the best monotone post-processing (Corollary A.1)
$\mathcal{E}_{\mathrm{hist}}$	KL error of a score relative to the full-history posterior (Proposition C.1)
$c_t, \mathcal{B}_c(\varepsilon)$	Common decision boundary, and expected number of eligible checkpoints within ε of it
$L_j, \bar{L}_j$	Request loss of the j-th candidate, and its cumulative envelope
α, δ	Request-level risk budget, and upper bound on the certification failure probability
$R_{\mathrm{LC}}, R_{\mathrm{RC}}$	Probability of losing a correct reference answer, and probability of recovering an answer the reference gets wrong

E CANDIDATE SELECTION AND REQUEST-LEVEL RISK CERTIFICATION

E.1 REQUEST-LEVEL LOSS AND FIXED-SEQUENCE TESTING

Continue actions advance along the reference process F, and offline stopping branches leave the reference trajectory unperturbed. Given the recorded reference trajectories, checkpoint features, and paired stopping outcomes, the stopping checkpoint, request loss, and generation cost of every frozen candidate follow by replay, so all candidates reuse the same paired records. Before the calibration data are used, the probe, feature transformations, confirmation rule, candidate thresholds $\theta_1 < \cdots < \theta_J$ taken from the scores of training requests, risk parameters, and selection rule are fixed. By Proposition 1, the loss of request X under the j-th candidate π_j is $L_j(X) = \sum_t A_{\pi_j,t} H_t$, and the tested quantity is its cumulative envelope

$$\bar{L}_j(X) = \max_{\ell \le j} L_\ell(X). \tag{E.1}$$

Each candidate stops only at a checkpoint with $q_t \le \theta_\ell$, so $\bar{L}_j(X) \le \mathbf{1}\{\min_{t \in \mathcal{T}^+} q_t \le \theta_j\}$, where $\mathcal{T}^+$ collects the harmful checkpoints of X and the minimum over an empty set is $+\infty$. The certified quantity is thus controlled by the lowest harmful score of each request, which the trajectory term of Eq. (3) raises.

For a prespecified risk budget $\alpha \in (0,1)$ and significance level $\delta \in (0,1)$, we compute on the n_{cert} independent calibration requests

$$V_j = \sum_{i=1}^{n_{\mathrm{cert}}} \bar{L}_j(X_i), \qquad p_j = \sum_{v=0}^{V_j} \binom{n_{\mathrm{cert}}}{v} \alpha^v (1-\alpha)^{n_{\mathrm{cert}}-v}. \tag{E.2}$$

Here p_j is the exact binomial lower-tail p-value for the null hypothesis $\mathbb{E}[\bar{L}_j] > \alpha$. Following fixed-sequence Learn-then-Test (Angelopoulos et al., 2025), we reject this null hypothesis and proceed if $p_j \le \delta$, and we terminate the testing sequence at the first $p_j > \delta$. Among the passing candidates and the reference policy, whose lost-correct risk is zero, the procedure returns the policy with the largest generation-token reduction on the calibration requests, breaking ties by larger stopping coverage and then by earlier position in the order. Algorithm 1 summarizes the procedure.

Algorithm 1 Certification of frozen candidate policies

Input: frozen probe and confirmation rule; thresholds $\theta_1 < \cdots < \theta_J$ fixed on training requests; independent calibration requests; risk parameters α, δ

$\widehat{\Theta}_{\mathrm{cert}} \leftarrow \emptyset$
for $j = 1, \ldots, J$ **do**
  Replay π_j on every calibration request and compute $\bar{L}_j$ by Eq. (E.1)
  Compute V_j and the exact binomial lower-tail p-value p_j by Eq. (E.2)
  if $p_j \le \delta$ **then**
    $\widehat{\Theta}_{\mathrm{cert}} \leftarrow \widehat{\Theta}_{\mathrm{cert}} \cup \{\theta_j\}$
  else
    break ▷ end the sequence; later candidates are not tested
  end if
end for
return the policy in $\widehat{\Theta}_{\mathrm{cert}}$ or the reference policy with the largest calibration generation-token reduction

E.2 Risk guarantee for the complete policy

Proposition E.1 (Risk guarantee for the complete policy)**.** *Conditional on the training and development data, suppose that the predictor, candidate policies, testing order, risk parameters, and selection rule are fixed, and that the calibration requests, together with their generation randomness, are i.i.d. samples from the deployment distribution under consistent generation and answering protocols. The policy $\widehat{\pi}$ returned by the procedure in Appendix E.1 then satisfies*

$$\mathbb{P}_{\mathrm{cal}}\big(R_{\mathrm{LC}}(\widehat{\pi}) \le \alpha\big) \ge 1 - \delta. \tag{E.3}$$

Proof. Write $\bar{R}_j = \mathbb{E}[\bar{L}_j]$. For a fixed candidate, $V_j \sim \mathrm{Binomial}(n_{\mathrm{cert}}, \bar{R}_j)$, and the stochastic monotonicity of the binomial distribution in its parameter makes the p-value in Eq. (E.2) valid whenever $\bar{R}_j > \alpha$. Since $\bar{L}_j$ is nondecreasing in j, so is $\bar{R}_j$. If some envelope is unsafe, let j_0 be the first index with $\bar{R}_{j_0} > \alpha$; accepting any unsafe envelope requires rejecting this first true null hypothesis, an event of probability at most δ. Hence, with probability at least $1-\delta$, all passing envelopes are safe simultaneously. Because $L_j \le \bar{L}_j$, every passing policy then satisfies $R_{\mathrm{LC}}(\pi_j) \le \bar{R}_j \le \alpha$. Selecting among them by calibration cost, or returning the reference policy, preserves the guarantee. □

Together with the accuracy identity of Section 2, this bounds the accuracy drop relative to the reference by α with probability at least $1-\delta$. The procedure applies fixed-sequence Learn-then-Test to the request-level envelope; its statistical unit is the request.

E.3 Risk under varying thresholds

The request loss of a stopping policy can vary non-monotonically with the threshold. Consider a reference-correct trajectory with three checkpoints whose scores are (h, l, l), where $0 < l < h$; stopping at the second checkpoint is harmless and stopping at the third is harmful. With stopping

after two consecutive qualifying checkpoints, increasing the threshold past l and then h yields reference fallback, a stop at the third checkpoint, and a stop at the second, respectively. The request loss is therefore 0, then 1, then 0. The envelope $\bar{L}_j$ is nondecreasing along the testing order by construction, so the passing candidates form a prefix, and Proposition E.1 applies to this non-monotone loss.

F Experimental Details

F.1 Data splits and evaluation protocol

Data splits. Training and calibration use only Hendrycks MATH, with question-level splits stratified by its seven subjects: 200 training and 100 calibration questions per subject. Model selection uses a subset of the training questions.

Generation and answering. Full-CoT samples with temperature 0.6, top-p 0.95, top-k 20, and a 13,000-token generation limit; reaching the limit invokes the stop-and-answer protocol. At a stopping checkpoint, S appends `\n</think>\n\n\boxed{` and greedily generates at most 48 answer tokens.

Accuracy–efficiency score (AES). Let p and L be answer accuracy and mean generated-token count, and let p_F and L_F be the corresponding paired Full-CoT values. For figures combining datasets, each quantity is first averaged equally across the datasets. We then compute

$$\mathrm{AES} = 1 - \frac{L}{L_F} + \begin{cases} 3(p - p_F)/p_F, & p \geq p_F, \\ 5(p - p_F)/p_F, & p < p_F. \end{cases} \tag{F.1}$$

The accuracy term rewards gains and penalizes losses more strongly.

F.2 Baseline implementations and operating points

Thought Calibration (TC). At reasoning-step boundaries, TC trains probes for current-answer correctness (Correct) or agreement with the maximum-budget answer (Consistent), and calibrates thresholds with Learn-then-Test.

LYNX. LYNX trains a forced-answer correctness probe at naturally occurring reflection points. It applies class-conditional split conformal calibration and stops when the prediction set first contains only the positive class.

BLADE. BLADE labels a prefix sufficient when four sampled forced answers are all correct, trains a compact probe on four adaptively selected hidden-state layers, and calibrates its threshold by a quantile rule. It stops after one threshold crossing at a self-doubt point or two consecutive crossings at sentence ends.

F.3 Supplementary results

F.3.1 Additional ablation results

For the input-feature ablation, the full input attains the highest AES on both Qwen3-4B and DeepSeek-7B (Figure 6). The hidden vector alone retains 66% and 85% of the full-input AES, respectively. Removing the hidden vector lowers accuracy by 5.2–6.0 points and leaves AES at most -0.125 on Qwen3-4B and 0.077 on DeepSeek-7B.

For the DeepSeek-7B checkpoint-schedule ablation (Figure 7), Paragraph is the only schedule that exceeds Full-CoT accuracy. Cue gains 0.05 AES relative to Paragraph but costs 2.2 accuracy points; no alternative has a paired bootstrap 95% interval for AES improvement entirely above zero.

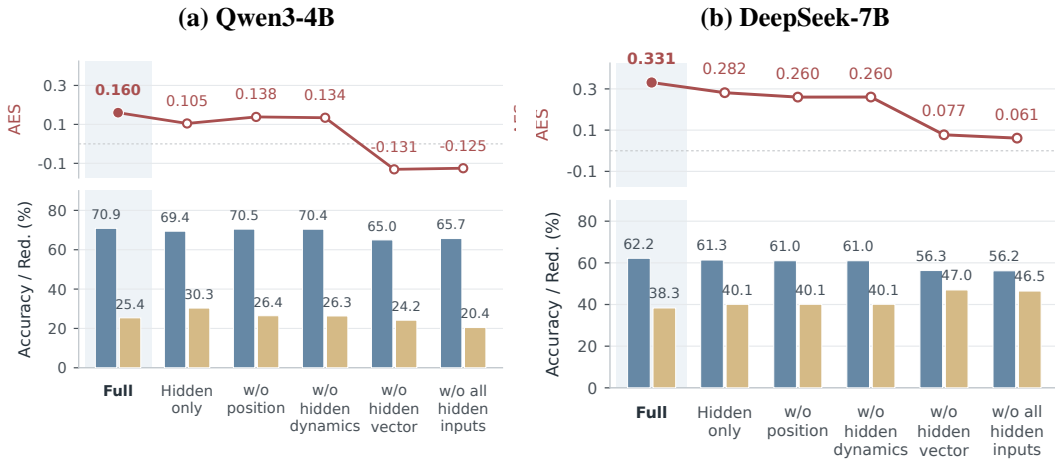


Figure 6: Input-feature ablation for the other two models. Blue bars show accuracy, gold bars show token reduction, and red lines show AES; shading marks the full input. Hidden only retains the hidden vector. Removing the hidden vector alone retains two hidden-derived scalars; removing all hidden inputs also removes those scalars.

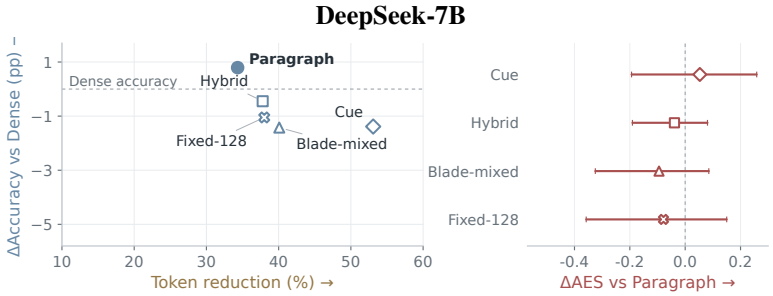


Figure 7: Checkpoint-schedule ablation for DeepSeek-7B, averaged over MATH500, AIME 2024, and GPQA Diamond. Left: accuracy change versus Full-CoT against token reduction. Right: AES change versus Paragraph with paired bootstrap 95% intervals.

F.4 End-to-end latency measurement

Protocol. We measure whether generated-token savings reduce execution time on a fixed, cost-matched workload. The three models are timed on MATH500, GPQA Diamond, and GSM8K using one NVIDIA L20 GPU per request, BF16, tensor parallelism of one, vLLM eager mode, and single-request load. At each token, the model performs a real forward pass and normal sampling computation before the frozen formal token is replayed, and every request is answered at its formal endpoint. All methods thus execute their evaluated trajectories, so latency differences reflect execution cost rather than changes in the generated trajectory. Causal features and probes run online, and the stop handoff and full-prefix recomputation for the answer branch are included. Each question is paired with its Full-CoT run on the same GPU and timed once. Hot-request timing runs from prompt encoding through final-answer decoding; model loading and preparation of the replay plan are excluded.

Weighting and metrics. For each dataset, positive question weights are fixed before timing and shared by each policy and its paired Full-CoT run, so that the weighted token reduction matches the policy's formal token reduction on the full test set. Speedup is the ratio of weighted Full-CoT time to weighted policy time on the same questions, not the mean of per-question speedups. Token reduction counts generated reasoning and answer tokens; execution time also includes control and answer-handoff overhead.

Table F.1: End-to-end timing on cost-matched formal-token workloads. TR is generated-token reduction (%); ΔAcc. is the weighted accuracy change on the timed questions (percentage points); speedup is relative to paired Full-CoT. Each entry averages dataset-level values equally over MATH500, GPQA Diamond, and GSM8K; each request is timed once under single-request load.

Model	Method	TR (%)	ΔAcc. (pp)	Speedup
Qwen3-4B	$TC_{Consistent}$	16.54	+0.46	1.21
	BLADE ($\delta = 0.03$)	31.51	+0.00	1.60
	LYNX	19.39	-3.07	1.28
	StopHarm	38.08	+1.50	1.71
DeepSeek-7B	$TC_{Consistent}$	16.90	-3.89	1.21
	BLADE ($\delta = 0.03$)	31.66	+1.67	1.58
	LYNX	26.79	-6.01	1.40
	StopHarm	42.10	+0.00	1.77
Qwen3-14B	$TC_{Consistent}$	22.63	-2.97	1.32
	BLADE ($\delta = 0.03$)	24.47	+0.00	1.34
	LYNX	23.57	-0.69	1.35
	StopHarm	40.78	-0.52	1.71

Table F.2: Timed values of the risk-budget sweep for Qwen3-14B (Figure 3). TR is generated-token reduction (%); E2E is weighted end-to-end seconds per request; speedup is relative to Full-CoT on the same weighted questions.

	GPQA Diamond			MATH500			GSM8K		
α	TR	E2E	Speedup	TR	E2E	Speedup	TR	E2E	Speedup
Full-CoT	0.00	313.7	1.00	0.00	180.8	1.00	0.00	74.4	1.00
0.01	1.10	311.0	1.01	3.61	174.8	1.03	4.13	71.5	1.04
0.02	10.30	282.2	1.11	9.26	164.7	1.10	17.31	61.7	1.20
0.03	29.39	221.9	1.41	17.97	149.2	1.21	33.94	49.4	1.51
0.05	45.66	170.5	1.84	29.35	128.8	1.40	47.33	39.4	1.89

Comparison across methods. Table F.1 averages each metric equally over the three datasets. StopHarm attains the largest token reduction and the highest speedup on every model: 1.71×, 1.77×, and 1.71× on Qwen3-4B, DeepSeek-7B, and Qwen3-14B, compared with at most 1.60×, 1.58×, and 1.35× among the baselines. ΔAcc. is computed on the timed questions with the timing weights and describes the timed workload; Table 2 reports full-test accuracy.

Risk-budget sweep. For Figure 3, Full-CoT and the four policies share one set of weights, fixed before timing to match each policy's formal token reduction. Accuracy uses all 198, 500, and 1,319 test questions of GPQA Diamond, MATH500, and GSM8K. Table F.2 lists the timed values. All twelve speedups exceed one, and averaging the $\alpha = 0.05$ entries over the three datasets recovers the Qwen3-14B StopHarm entry of Table F.1 (40.78% and 1.71×).

F.5 Protocol of the mechanism analysis

The analyses of Section 5.5 reuse the frozen reference trajectories, checkpoints, stop-and-answer outcomes, grader, and generated-token costs of the MATH-source experiments; no trajectory is regenerated.

Probes and data. StopHarm is the frozen MATH-source probe of Table 2. Correctness keeps the features, architecture, loss, class weighting, optimizer, training budget, checkpoints, confirmation rule, and seed of StopHarm and replaces the labels with $U = 1 - Y^S$. Both probes are trained on the 1,400 MATH training questions, of which 1,120 fit the probe and 280 select the epoch. Three further seeds (101, 202, and 303) each train a new StopHarm–Correctness pair. The held-out MATH pool consists of the 700 calibration questions and MATH500 (1,200 questions); none of them enters probe training or model selection.

States and separation. A checkpoint is in state CC, CW, WC, or WW according to the correctness of its stopping answer and of the reference answer, in that order. $\mathrm{AUC}_{H|U}$ is the AUROC for H among checkpoints with $U = 1$, which compares WC with WW states, and AUC_U is the AUROC for U over all checkpoints. Position bins group checkpoints by their token index into $[0, 512)$, $[512, 1024)$, $[1024, 2048)$, $[2048, 4096)$, and $[4096, \infty)$.

Frontier area and channels. For each score, the candidate thresholds are the 2,001 quantiles of its values at eligible checkpoints, together with a threshold that never stops. Each threshold is replayed with the model's frozen confirmation rule, which stops once k consecutive checkpoints at or beyond token position $t_{\min}$ have scores at or below the threshold ($k/t_{\min}$ is $1/0$, $3/0$, and $2/256$ for Qwen3-4B, DeepSeek-7B, and Qwen3-14B). The replay gives the empirical lost-correct rate (Lost) and the generated-token reduction TR on the evaluation set, and the frontier $\mathrm{TR}^*(x) = \max\{\mathrm{TR}(\theta) : \mathrm{Lost}(\theta) \le x\}$ is averaged over 101 equally spaced values of $x \in [0, 5\%]$. At each frontier point, the reduction splits by the reference outcome of each question into the harm-free channel (YF0, $Y^F = 0$) and the reference-correct channel (YF1, $Y^F = 1$); both are normalized by the Full-CoT tokens of the evaluation set and add up to the total. Sweeping thresholds on the evaluation set compares score orderings at matched realized risk; deployed operating points are certified as in Section 3.2. The harm-free ceiling is the reduction obtained when every reference-failed run stops at the cheapest checkpoint that its confirmation rule allows.

Label dose and controls. Starting from the Correctness labels, the currently wrong rows of a random fraction $p \in \{0.25, 0.5, 0.75\}$ of the reference-failed training questions, all of which are WW rows, are relabeled 0. Each p uses two independent draws of questions, and $p = 1$ reproduces the StopHarm probe bit for bit. The random-flip control (RF) relabels as many positive rows as $p = 1$, drawn without regard to their state. The class-weight control (CW) keeps the Correctness labels and multiplies the positive-class weight in both loss terms by the ratio of H-positive to U-positive training rows. All other settings, including the seed, stay fixed.

Cross-fitting. The 2,600 MATH-source questions (training, calibration, and MATH500) are split into two halves of 1,300 questions (1,040 for fitting and 260 for selection) that train and evaluate each other, over three random partitions. Each question is scored only by probes that have not seen it, and each estimate averages the six held-out half statistics. This preregistered confirmatory test is the one analysis whose training pool includes MATH500 questions.

Finite mappings. Let Q collect the scores of the Correctness, Consistency, and Last-switch probes (Section 5.4) and q denote the StopHarm score. On the 280 selection questions of the training split, we fit mappings $g_Q(Q)$ and $g_{Q,q}(Q, q)$ that predict H, choosing between L_2-regularized logistic regression and gradient-boosted trees of depth at most two by three-fold cross-validation over questions. The frozen mappings are evaluated on MATH500, GSM8K, and GPQA Diamond by the log loss for H pooled over checkpoints, with intervals from a question-level paired bootstrap.

Reasoning trajectory. In a reference-correct run with last currently wrong checkpoint r^*, a currently wrong checkpoint r lies $r^* + 1 - r$ checkpoints before the final correction. WC checkpoints at each distance are compared with WW checkpoints from the same position bin, and the AUROCs are averaged over bins with weights proportional to the WC checkpoints. A WW state is *locked* when its stopping answer is equivalent to the final reference answer.

Robustness families. Figure 9 collects the harm-free gain over eight families: four training seeds; logit averages over one to four seeds, averaged over all seed subsets of each size; cross-fitting; four evaluation sets (the held-out MATH pool, its two parts, and GSM8K); twelve input-feature variants, namely the full features and eleven variants that remove the hidden state, all scalars, one scalar, or a group of features, each trained with H and compared with the same Correctness probe; a logistic-regression probe trained with H; eight confirmation rules ($k = 1, \ldots, 5$, two exponential smoothings of the logit, and the mean of the last three logits), applied to both scores; and three stopping boundaries, namely a constant threshold on the raw score, a constant threshold on an isotonically calibrated risk, and a threshold on the calibrated risk divided by the remaining budget fraction $(13000 - n_t)/13000$. As in the main analysis, thresholds are swept on the evaluation set.

Table F.3: Values in Figure 5a: separation among currently wrong checkpoints and prediction of the current error on MATH500. Rows is the number of currently wrong checkpoints, with WW states in parentheses.

		$\mathrm{AUC}_{H\|U}$ (WC vs. WW)			AUC_{U} (current error)		
Model	Rows (WW)	Correctness	StopHarm	Δ	Correctness	StopHarm	Δ
Qwen3-4B	20,833 (5,941)	0.579	0.724	+0.145 [0.096, 0.193]	0.831	0.794	−0.037 [−0.056, −0.019]
DeepSeek-7B	25,503 (9,177)	0.504	0.718	+0.214 [0.160, 0.267]	0.776	0.709	−0.067 [−0.100, −0.032]
Qwen3-14B	19,043 (6,197)	0.634	0.710	+0.076 [0.052, 0.101]	0.831	0.798	−0.033 [−0.045, −0.021]

Statistics. Intervals are question-level bootstrap 95% intervals with 2,000 resamples, stratified by source on the held-out MATH pool; the three models share resampling weights, and pooled estimates resample questions jointly across models. The preregistered primary endpoints, the harm-free gain of the frozen probes and the slope of the harm-free frontier area in p, are Holm-corrected over the three models, and cross-fitting provides the preregistered confirmatory test of the total frontier area.

F.6 Additional mechanism results

Tables F.3–F.8 list the values plotted in Figure 5 and cited in Section 5.5. Differences are StopHarm minus Correctness unless noted otherwise, and brackets are 95% intervals.

Separation. Table F.3 gives the values of Figure 5a. For Qwen3-4B, DeepSeek-7B, and Qwen3-14B, StopHarm raises $\mathrm{AUC}_{H|U}$ from 0.579, 0.504, and 0.634 for Correctness by 0.145, 0.214, and 0.076, whereas Correctness predicts the current error better by 0.037, 0.067, and 0.033. The gain in $\mathrm{AUC}_{H|U}$ holds for every training seed; under lenient grading, which also accepts answers semantically equivalent to the ground-truth answer; under cross-fitting on 2,600 questions (0.126 [0.113, 0.138] pooled over models); and in every position bin, including the first 512 reasoning tokens (Table F.4). Adding the StopHarm score to a mapping of the three proxy scores lowers the held-out log loss for H on MATH500 and on both transfer benchmarks (Table F.5).

Label dose. Table F.6 gives the values of Figure 5b, which plots differences from $p = 0$. The slopes in p are 0.144, 0.208, and 0.073 for $\mathrm{AUC}_{H|U}$ and 0.79, 1.53, and 0.48 pp for the harm-free frontier area. The RF and CW controls stay within 0.02 of $p = 0$ in $\mathrm{AUC}_{H|U}$ and within 0.23 pp in the harm-free frontier area, and StopHarm exceeds RF in the harm-free channel by 0.56, 1.72, and 0.52 pp.

Channels. Table F.7 gives the values of Figure 5c together with the cross-fitted estimates. For the frozen probes, the harm-free gain is 0.79, 1.62, and 0.55 pp, so StopHarm saves 1.7, 3.1, and 1.5 times as many tokens as Correctness on reference-failed runs. Across the four seeds, the standard deviation of the gain is 0.09–0.23 pp in the harm-free channel and 1.40–1.60 pp in the reference-correct channel. The reference-correct gain averages 0.07 [−1.08, 1.21] pp over seeds and models and is 1.48 [0.72, 2.16] pp under cross-fitting.

Along the reasoning trajectory. StopHarm's gain in separating WC from WW states is largest one to two checkpoints before the reasoning finally corrects its answer (Figure 8a; 18 of 18 intervals above zero), where Correctness ranks WC states as safer than WW states (AUROC 0.39–0.44). At the frontier point with empirical Lost at most 2%, StopHarm stops more reference-failed runs than Correctness, mostly at WW states whose answer is already locked (31, 37, and 24 runs, against 18, 21, and 18; Figure 8b).

Robustness and certified operating points. All 111 estimates of the harm-free gain in Figure 9 are positive, with 95% intervals above zero. Table F.8 gives the values of Figure 5d: at the certified operating points, StopHarm saves more tokens on reference-failed runs than Correctness for all nine model–benchmark pairs.

Table F.4: Robustness of the separation gain $\Delta\mathrm{AUC}_{H|U}$. Each seed pairs the two probes trained with that seed. Lenient grading relabels 7,915, 14, and 6,211 checkpoints of the held-out MATH pool. Pooled over models, the cross-fitted gain is 0.126 [0.113, 0.138].

Evaluation	Condition	Qwen3-4B	DeepSeek-7B	Qwen3-14B
MATH500	Original seed (Table F.3)	+0.145 [0.096, 0.193]	+0.214 [0.160, 0.267]	+0.076 [0.052, 0.101]
MATH500	Seed 101	+0.140 [0.093, 0.186]	+0.196 [0.147, 0.244]	+0.077 [0.051, 0.103]
MATH500	Seed 202	+0.149 [0.100, 0.197]	+0.193 [0.148, 0.236]	+0.078 [0.051, 0.103]
MATH500	Seed 303	+0.139 [0.092, 0.186]	+0.209 [0.158, 0.259]	+0.083 [0.057, 0.109]
MATH500	Lenient grading	+0.124 [0.073, 0.176]	+0.214 [0.160, 0.267]	+0.060 [0.036, 0.084]
2,600 questions	Cross-fitting	+0.121 [0.104, 0.137]	+0.169 [0.151, 0.186]	+0.088 [0.074, 0.101]
Held-out MATH, token position	$[0, 512)$	+0.091 [0.055, 0.127]	+0.065 [0.029, 0.100]	+0.067 [0.048, 0.089]
	$[512, 1024)$	+0.070 [0.031, 0.116]	+0.065 [0.034, 0.097]	+0.048 [0.029, 0.066]
	$[1024, 2048)$	+0.075 [0.043, 0.110]	+0.051 [0.028, 0.077]	+0.050 [0.033, 0.066]
	$[2048, 4096)$	+0.115 [0.078, 0.152]	+0.073 [0.037, 0.111]	+0.074 [0.051, 0.098]
	≥ 4096	+0.163 [0.113, 0.214]	+0.127 [0.076, 0.187]	+0.068 [0.033, 0.103]

Table F.5: Information beyond three proxy scores: reduction in held-out log loss for H, $\mathrm{LL}(g_Q) - \mathrm{LL}(g_{Q,q})$, when the StopHarm score q is added to a mapping of the Correctness, Consistency, and Last-switch scores Q. The mappings are fit on MATH training questions only.

Test set	Qwen3-4B	DeepSeek-7B	Qwen3-14B
MATH500	+0.0182 [0.0066, 0.0295]	+0.0171 [0.0064, 0.0272]	+0.0105 [0.0007, 0.0208]
GSM8K	+0.0576 [0.0479, 0.0673]	+0.0128 [0.0088, 0.0165]	+0.0323 [0.0272, 0.0372]
GPQA Diamond	+0.0794 [0.0386, 0.1140]	+0.0545 [0.0419, 0.0675]	+0.0340 [0.0120, 0.0531]

Table F.6: Values in Figure 5b: $\mathrm{AUC}_{H|U}$ on MATH500 and the harm-free frontier area on the held-out MATH pool (pp) for each set of training labels. Intermediate doses list both subset draws.

	$\mathrm{AUC}_{H\|U}$			Harm-free frontier area (pp)		
Training labels	Qwen3-4B	DeepSeek-7B	Qwen3-14B	Qwen3-4B	DeepSeek-7B	Qwen3-14B
$p = 0$ (Correctness)	0.579	0.504	0.634	1.13	0.78	1.16
$p = 0.25$	0.629 / 0.636	0.607 / 0.582	0.655 / 0.667	1.15 / 1.29	1.22 / 1.00	1.38 / 1.42
$p = 0.5$	0.664 / 0.676	0.673 / 0.652	0.671 / 0.684	1.51 / 1.33	1.54 / 1.07	1.47 / 1.42
$p = 0.75$	0.689 / 0.719	0.693 / 0.693	0.694 / 0.695	1.42 / 1.79	1.85 / 1.71	1.49 / 1.64
$p = 1$ (StopHarm)	0.724	0.718	0.710	1.93	2.40	1.72
Random flip (RF)	0.595	0.523	0.639	1.36	0.68	1.20
Class weight (CW)	0.582	0.515	0.630	1.09	0.55	1.17
Slope in p	+0.144	+0.208	+0.073	+0.79	+1.53	+0.48
95% interval	[0.098, 0.190]	[0.153, 0.263]	[0.050, 0.096]	[0.40, 1.27]	[0.69, 2.56]	[0.19, 0.83]
StopHarm − RF	–	–	–	+0.56	+1.72	+0.52
95% interval	–	–	–	[0.13, 1.08]	[0.88, 2.81]	[0.09, 1.01]

Table F.7: Values in Figure 5c: frontier-area differences on the held-out MATH pool (pp, $x \in [0, 5\%]$) and the cross-fitted estimates on 2,600 questions. Pooled estimates resample questions jointly across models.

Estimate	Qwen3-4B	DeepSeek-7B	Qwen3-14B	Pooled
Harm-free channel (YF0)				
Original seed (frozen StopHarm)	+0.79 [0.36, 1.29]	+1.62 [0.82, 2.66]	+0.55 [0.23, 0.94]	–
Seed 101	+0.90 [0.46, 1.40]	+1.46 [0.68, 2.44]	+0.67 [0.26, 1.14]	–
Seed 202	+0.93 [0.47, 1.49]	+1.91 [1.05, 2.94]	+0.76 [0.27, 1.33]	–
Seed 303	+0.71 [0.24, 1.31]	+1.94 [1.02, 3.13]	+0.61 [0.18, 1.15]	–
Mean over the four seeds	+0.83 [0.43, 1.33]	+1.73 [0.92, 2.74]	+0.65 [0.28, 1.10]	+1.07 [0.67, 1.55]
Standard deviation over seeds	0.10	0.23	0.09	–
Cross-fitting (2,600 questions)	+0.99 [0.73, 1.30]	+1.85 [1.32, 2.46]	+0.43 [0.24, 0.68]	+1.09 [0.85, 1.38]
Reference-correct channel (YF1)				
Original seed (frozen StopHarm)	+1.86 [−0.06, 3.93]	−1.45 [−3.58, 0.79]	+2.03 [0.12, 3.96]	–
Seed 101	+1.91 [−0.14, 4.02]	−1.99 [−4.20, 0.21]	+0.70 [−0.91, 2.28]	–
Seed 202	+0.69 [−1.83, 3.06]	+1.40 [−0.59, 3.26]	−1.38 [−3.42, 0.90]	–
Seed 303	−1.27 [−3.67, 1.17]	−1.88 [−4.12, 0.38]	+0.26 [−1.56, 2.20]	–
Mean over the four seeds	+0.80 [−1.04, 2.65]	−0.98 [−2.85, 0.90]	+0.40 [−1.11, 2.04]	+0.07 [−1.08, 1.21]
Standard deviation over seeds	1.49	1.60	1.40	–
Cross-fitting (2,600 questions)	+3.40 [2.19, 4.56]	+0.70 [−0.32, 1.67]	+0.33 [−0.78, 1.30]	+1.48 [0.72, 2.16]
Total				
Mean over the four seeds	+1.63 [−0.39, 3.67]	+0.75 [−1.53, 3.16]	+1.05 [−0.65, 2.89]	+1.15 [−0.16, 2.48]
Cross-fitting (2,600 questions)	+4.39 [3.03, 5.69]	+2.54 [1.23, 3.76]	+0.76 [−0.40, 1.88]	+2.56 [1.72, 3.39]

Table F.8: Values in Figure 5d: token reduction on reference-failed runs at the certified operating points and the harm-free ceiling (% of Full-CoT tokens).

Benchmark	Model	Harm-free ceiling	Correctness	StopHarm
GSM8K	Qwen3-4B	13.4	2.3	7.6
	DeepSeek-7B	14.4	5.5	8.4
	Qwen3-14B	8.2	0.6	3.8
MATH500	Qwen3-4B	14.8	1.1	1.8
	DeepSeek-7B	17.1	1.0	4.7
	Qwen3-14B	15.1	0.2	1.6
GPQA Diamond	Qwen3-4B	50.3	2.2	18.4
	DeepSeek-7B	68.7	2.9	27.2
	Qwen3-14B	42.1	2.6	22.5

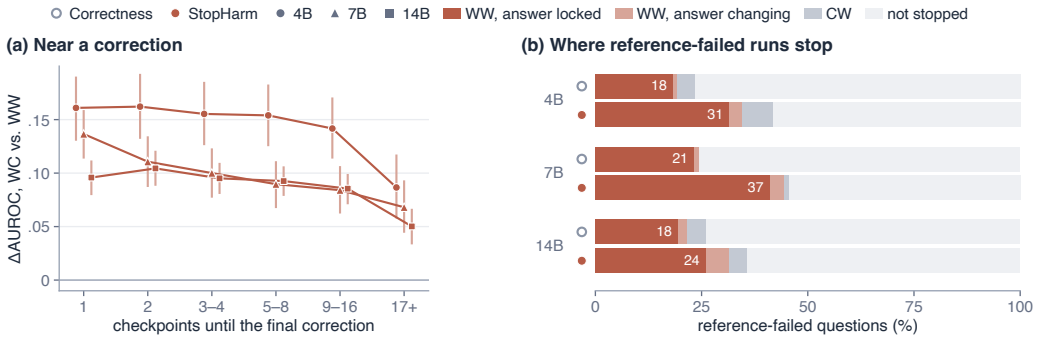


Figure 8: StopHarm along the reasoning trajectory on the held-out MATH pool. **(a)** Gain in AUROC for separating WC from WW states within the same position bin, by the number of checkpoints before the reasoning finally corrects its answer; bars are 95% intervals. **(b)** Stopping state of reference-failed questions at the frontier point with empirical Lost at most 2%; numbers count the questions stopped at locked WW states.

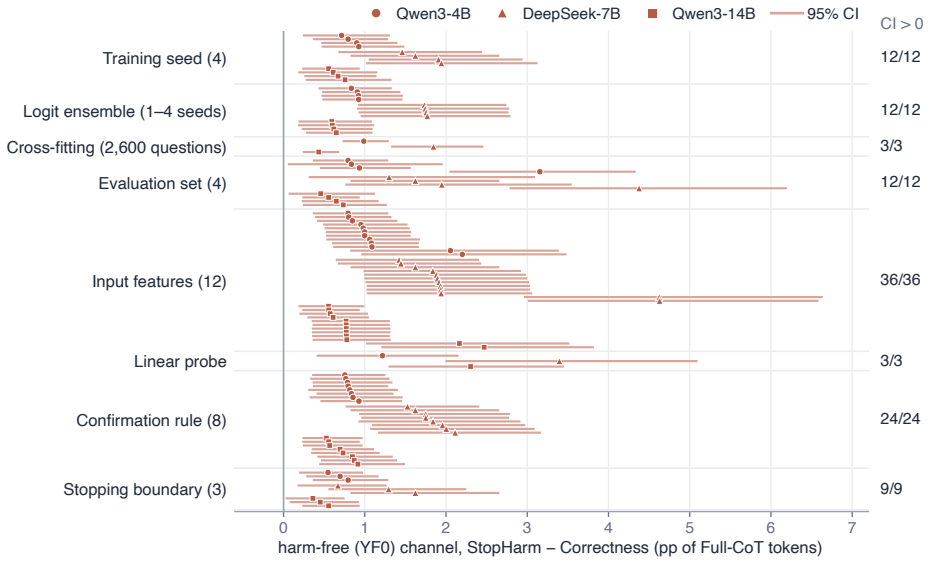


Figure 9: Harm-free gain under eight families of perturbations (Appendix F.5). Each point is one estimate with its 95% interval; the right column counts intervals above zero.

G Extended Related Work

G.1 Efficient reasoning and online stopping

Efficient reasoning methods adapt computation at several levels. Length-aware rewards and token budgets train generators to produce shorter chains (Aggarwal & Welleck, 2025; Hou et al., 2026; Xiang et al., 2025), and compressed-chain fine-tuning transfers concise reasoning into the model (Xia et al., 2025). At inference time, adaptive self-consistency controls the number of sampled solutions (Wang et al., 2023; Li et al., 2024), and certainty-based scheduling allocates computation across reasoning programs (Fu et al., 2025a). Conditional computation varies recurrent processing time (Graves, 2017; Banino et al., 2021) or network depth through intermediate exits (Teerapittayanon et al., 2016; Xin et al., 2020; Zhou et al., 2020). Stopping a single reasoning trajectory adapts its remaining generation budget as the solution develops. This setting is motivated by the coexistence of redundant verification and useful correction: models frequently continue after reaching an answer (Chen et al., 2025; Kang et al., 2025), while additional reasoning can correct an earlier error or revise a correct answer into an incorrect one (Muennighoff et al., 2025; Caldarella et al., 2026). The benefit of continuation therefore depends on the current state and the outcome that further reasoning would deliver.

Online stopping signals make this evolving state observable through answer confidence, convergence, and internal representations. DEER evaluates the confidence of trial answers at reasoning transitions (Yang et al., 2026); answer-consistency methods track agreement across successive prefixes (Liu & Wang, 2025; Mao et al., 2026); and entropy-based methods monitor uncertainty in sampled answers or in the answer distribution exposed by ending the thinking phase (Laaouach, 2025; Wang et al., 2026a). PUMA uses a learned reasoning-step redundancy detector to identify candidate exits and verifies them through trial-answer confidence and consistency (Min et al., 2026). Hidden-state read-outs provide another source of information: early reasoning representations predict eventual correctness (David, 2025), and semantic entropy probes estimate uncertainty from a single forward pass (Farquhar et al., 2024; Kossen et al., 2024). Acquiring these signals is part of the stopping cost. Trial-answer methods decode auxiliary answers at inspected checkpoints, with costs that depend on prefix reuse (Dong et al., 2026). StopHarm reads hidden states and statistics already produced by normal decoding; its answer-generation branch is invoked when the policy stops. This makes the information learned by the read-out central to the choice of which prefixes receive further computation.

G.2 Supervision targets and risk-controlled stopping

Learned stopping scores encode different properties of a reasoning prefix. LYNX trains on the correctness of forced exits at self-doubt cues (Akgül et al., 2025). BLADE expands checkpoint coverage and selects informative layers, constructing prefix-sufficiency labels from repeated forced completions: unanimously correct and unanimously incorrect completions provide the positive and negative examples (Fu et al., 2026). These objectives identify prefixes from which the model can answer correctly. Other objectives describe the evolution of the answer. Thought Calibration probes current correctness, agreement with the maximum-budget answer, and reasoning novelty (Wu et al., 2025). The learned variant of answer convergence predicts whether later prefix answers have stabilized (Liu & Wang, 2025), and TERMINATOR supervises the first position at which the trajectory logically reaches its final answer (Nagle et al., 2026). Future-outcome predictors estimate whether the completed chain will succeed (Afzal et al., 2025; Zhang et al., 2026). Outcome verifiers and process reward models similarly learn solution quality or intermediate-step quality from final outcomes and step annotations (Cobbe et al., 2021; Lightman et al., 2024; Wang et al., 2024; Setlur et al., 2025). Together, these supervision schemes expose correctness, progress, and future success. The loss of a correct reference answer depends on how stopping and reference outcomes occur jointly: an incorrect prefix answer may precede a successful correction or an equally incorrect final answer. The two cases have the same current-correctness label and different lost-correct losses. Agreement and stability labels also record changes between incorrect answers, and future-success labels specify the reference outcome. Table G.1 summarizes the signals, supervision, and operating rules of representative methods.

Statistical risk control makes the loss associated with a stopping rule explicit. Selective prediction studies the trade-off between coverage and prediction risk (Chow, 1970; El-Yaniv & Wiener, 2010;

Geifman & El-Yaniv, 2017), and Learn-then-Test and conformal risk control provide frameworks for selecting prediction rules under specified losses (Angelopoulos et al., 2025; 2024). Applications include calibrated language generation (Quach et al., 2024) and layer-wise exits that preserve full-model behavior (Schuster et al., 2022). For reasoning termination, Thought Calibration applies Learn-then-Test to the stopping threshold using correctness- or consistency-based risk formulations (Wu et al., 2025), and LYNX uses split conformal calibration for its forced-exit predictor (Akgül et al., 2025). Conformal Thinking calibrates upper and lower thresholds for terminating confident or persistently uncertain trajectories (Wang et al., 2026b). Its upper-threshold loss counts incorrect exits; its lower-threshold loss weights an exit by the fraction of correct answers from the current checkpoint through the remaining trajectory. These definitions attach risk to different aspects of termination: current answer error, consistency with extended reasoning, and correctness available along the remaining chain.

Lost-correct risk gives a direct link between early stopping and the accuracy of a fixed reference process. LearnStop trains a logistic predictor of current answer correctness from online trial-answer and prefix features, then selects its threshold with a finite-grid upper confidence bound on request-level lost-correct risk (Dong et al., 2026). This risk bounds the net accuracy decline relative to full reasoning. StopHarm uses the same protected event to construct its training labels, pairing each stopping outcome with the reference outcome of that trajectory. The label $H_t = \mathbf{1}\{Y_t^S = 0, Y^F = 1\}$ separates prefixes whose continuation recovers a correct reference answer from prefixes followed by an incorrect reference answer. It also assigns zero harm to an already correct stopping answer. This supervision brings both sources of low-harm stopping opportunities into a single prediction task. At the realized stopping checkpoint, the label equals the request's lost-correct loss (Proposition 1). StopHarm trains on that event at each checkpoint and raises the lowest harmful scores within each training trajectory; its independent calibration step certifies the resulting policy's request-level loss. Supervision and certification consequently refer to the same protected outcome at their respective checkpoint and policy levels.

The value of this supervision can be understood through the information that a score retains for a decision. Proper scoring rules distinguish calibration from refinement (Bröcker, 2009), and grouping loss characterizes risk heterogeneity among examples assigned the same score (Kull & Flach, 2015; Perez-Lebel et al., 2023). For stopping, such heterogeneity arises when identical proxy scores merge prefixes with different probabilities of losing a correct reference answer. Theorem 1 establishes that this can occur even for Bayes-optimal correctness, agreement, stability, and future-success scores taken together. Decision-focused learning connects prediction quality to the decisions it supports (Elmachtoub & Grigas, 2022; Agarwal, 2014; Bao, 2023), and OS-Pruner formulates stopping through a state-dependent continuation value under an accuracy–length reward (Ehab et al., 2026). For the local risk–compute objective studied here, Theorems 2 and 3 characterize when retained stop-harm information changes the optimal action and when a learned score realizes the resulting utility gain despite prediction error. This connects the choice of supervision to attainable computational savings under an explicit accuracy-protection objective.

H Limitations and Future Work

StopHarm is designed to preserve correct outcomes produced by a fixed reference inference procedure, and therefore defines risk relative to that reference rather than in terms of absolute correctness. This formulation explicitly distinguishes errors introduced by early stopping from errors already present in the reference trajectory, but also makes the resulting supervision dependent on the reference budget, decoding strategy, stop-and-answer protocol, and evaluation procedure. A natural extension is to consider stronger or adaptive reference procedures, as well as policies that jointly choose between stopping and verified fallback mechanisms. Such extensions would enlarge the set of computational actions available at inference time while preserving the same principle of supervising decisions with respect to the outcome being protected.

Our decision-value analysis characterizes the information retained by direct stop-harm supervision for stopping decisions under a fixed checkpoint distribution, while request-level certification controls the risk induced by the resulting sequential policy. These results do not, however, constitute a complete characterization of globally cost-optimal sequential inference. In particular, continuation changes not only the probability of future success, but also the distribution of subsequent stopping

Table G.1: Online signals, supervision targets, and operating rules for reasoning early stopping. Risk definitions are stated in the text; calibration acts on the loss specified by each method.

Method	Online signal	Supervision target	Operating rule / calibration objective
DEER (Yang et al., 2026)	Confidence of trial answers at reasoning transitions	None (training-free)	Confidence threshold
Answer convergence (Liu & Wang, 2025)	Trial-answer agreement; LSTM over hidden states	Stability of later answers (learned variant)	Agreement window; validated threshold
Thought Calibration (Wu et al., 2025)	Linear probes on step representations	Correctness; maximum-budget answer agreement; novelty	Learn-then-Test on correctness- or consistency-based risk
LYNX (Akgül et al., 2025)	Hidden-state probe at self-doubt cues	Correctness of forced exits	Split conformal calibration of forced-exit predictions
BLADE (Fu et al., 2026)	Selected-layer probe at sentence and self-doubt boundaries	Prefix sufficiency from unanimous forced completions	Calibrated thresholds; boundary-specific confirmation
TERMINATOR (Nagle et al., 2026)	Token-level hidden-state probe	First logical arrival at the final answer	Sliding-window majority vote
LearnStop (Dong et al., 2026)	Trial-answer and prefix features	Current-answer correctness	Finite-grid upper confidence bound on lost-correct risk
Conformal Thinking (Wang et al., 2026b)	Uncertainty signals	Thresholds on given signals	Upper/lower thresholds; wrong exits / remaining-chain correctness
OS-Pruner (Ehab et al., 2026)	Stopping head on fine-tuned copies of final layers	Expected accuracy–length reward over stopping times	Length-penalty weight
StopHarm	MLP on normal-decoding states and statistics	Stop harm at checkpoint and trajectory levels	Fixed-sequence Learn-then-Test on request-level lost-correct risk

opportunities and their associated computational costs. Modeling these dependencies explicitly suggests a broader class of history-aware and cost-sensitive controllers that optimize over future computation rather than applying a checkpoint-wise threshold rule. Importantly, such extensions are compatible with the same protected-event formulation and need not alter the underlying supervision target.

The current implementation also incurs an offline preparation cost because it requires paired outcome generation and access to intermediate decoding representations. More selective checkpoint annotation, trajectory reuse, and efficient branching could substantially reduce this overhead. At inference time, however, StopHarm relies only on causal features available from the ongoing decoding process and does not require evaluating counterfactual trial answers at each checkpoint. Finally, our latency results are measured using cost-matched trajectory replay under single-request execution, which provides controlled estimates of the computational savings attributable to early stopping but does not capture all effects present in production serving systems. Evaluating naturally decoded workloads under continuous batching, heterogeneous request lengths, and realistic serving concurrency would provide a more complete assessment of how these savings translate into end-to-end throughput and tail latency.