SEQUENTIAL CONSENSUS POLICY OPTIMIZATION FOR TEST-TIME REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Test-time policy optimization improves language models by sampling multiple rollouts per prompt and using their majority-vote answer as a pseudo-label, without any ground-truth supervision. This procedure is expensive since allocating a fixed rollout budget to every prompt wastes compute on prompts whose answer has already settled. Adaptive stopping rules help, but existing methods decide using only the current visit, discarding what earlier visits to the same prompt revealed about its answer distribution. We introduce Sequential Consensus Policy Optimization (SCPO), which maintains a Dirichlet posterior over each prompt's answer distribution, warm-started by an exponential moving average of past answer counts, and stops sampling once the posterior probability that some answer is the mode exceeds a certified threshold. We establish guarantees for adaptive stopping and gap-dependent rollout complexity, and characterize when the EMA-based memory remains accurate and reduces the rollout budget. Across Qwen2.5-Math-1.5B and Llama-3.1-8B-Instruct on MATH-500, AIME 2024, and LiveCodeBench, SCPO cuts rollouts per prompt by up to 71% and generated tokens by up to 66% relative to TTRL, while maintaining competitive accuracy.

1 INTRODUCTION

Large language models (LLMs) perform well across diverse tasks (Touvron et al., 2023; Grattafiori et al., 2024; Guo et al., 2025), yet they are typically trained once and deployed with fixed parameters. After deployment, they often face tasks or input distributions not covered during training. Test-time training (TTT) (Sun et al., 2020; Hardt & Sun, 2024; Feng et al., 2026; Zhang et al., 2026; Hübotter et al., 2026) addresses this gap by updating the model's parameters on the test data itself during inference. This allows the model to capture task-specific patterns that fixed-parameter adaptation often misses (Akyürek et al., 2025).

Within TTT, test-time reinforcement learning (TTRL) (Zuo et al., 2025) adapts models to unlabeled test inputs using reinforcement learning (RL). Unlike other RL frameworks that rely on human preferences or external rewards (Ouyang et al., 2022; Shao et al., 2024), TTRL (Zuo et al., 2025) samples multiple rollouts per prompt and uses majority voting to select the leading answer as the pseudo-label. The rewards of rollouts are then based on whether their answers are the same as this pseudo-label. TTRL and subsequent methods (Yan et al., 2026; Wang et al., 2026) substantially improve over the base model and also outperform majority-vote, suggesting that label-free RL can be a practical approach to adapt deployed LLMs at test time.

However, TTRL spends the same rollout budget on every prompt at every training step. For many prompts, the model returns one answer consistently so that the majority vote is settled long before this budget is used up. Since rollout generation is one of the major costs in on-policy training, this fixed allocation limits how cheaply test-time RL can be run. OptPO (Wang et al., 2025b) reduces this waste with a Bayesian sequential probability ratio test to stop generating rollouts when the leading answer becomes statistically dominant over the runner-up. Still, OptPO treats every visit to a prompt as new, assigning a uniform prior to all answers and considering only the current epoch's rollouts, discarding what earlier visits to the same prompt have already revealed.

That history turns out to be informative. TTRL revisits each prompt across many epochs, so we ask whether earlier visits predict a prompt's current leading answer. Tracking the leading answer of

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.

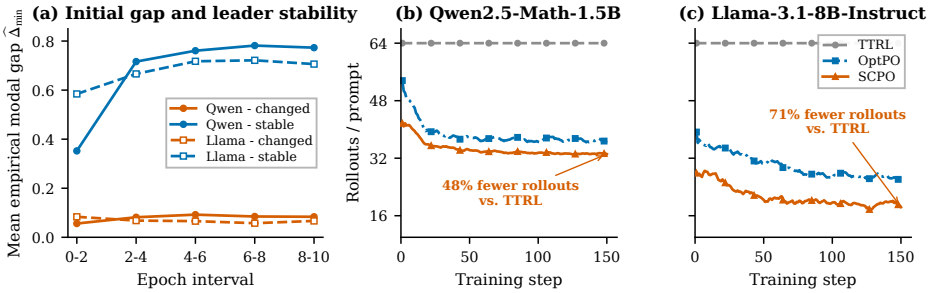


Figure 1: Leader stability and rollout efficiency of Qwen2.5-Math-1.5B and Llama-3.1-8B-Instruct on MATH-500. (a) Mean normalized leader vs. runner-up gap at the start of each epoch interval, grouped by whether the pseudo-label leader changes or remains stable during that interval. (b–c) The number of rollouts per prompt during training on MATH-500 of TTRL, OptPO, and our SCPO.

each prompt across two-epoch intervals on MATH-500 with Qwen2.5-Math-1.5B and Llama-3.1-8B-Instruct (Figure 1a), we find that prompts whose leader stays the same have a larger gap between the leader and runner-up at the *start* of the interval than prompts whose leader later changes. This suggests a simple principle: if a prompt's current leader also led at previous epochs by a wide gap, fewer new rollouts should suffice to confirm it, whereas a prompt with a contested history should retain its full budget.

Turning this principle into a reliable algorithm poses two challenges. First, the stopping decision must remain statistically valid when the confidence is estimated by finite simulation and the stopping time depends on the data. Second, past evidence was collected under earlier policies and can be stale, so an overconfident prior could lock in an outdated answer. We address both of these issues with *Sequential Consensus Policy Optimization* (SCPO). SCPO places a Dirichlet posterior over each prompt's answer probabilities, with a prior given by an exponential moving average (EMA) of answer counts from past epochs, and stops sampling once the posterior probability that an answer is the mode exceeds a threshold. Because the EMA prior carries a bounded pseudo-count in total mass, a consistent history lets the posterior reach the threshold with fewer fresh rollouts, while a contested or stale history adds little separation and can still be overturned by fresh evidence. As Figure 1b-c shows, this cuts the number of rollouts SCPO generates relative to TTRL by 48% on Qwen2.5-Math-1.5B and 71% on Llama-3.1-8B-Instruct, respectively. Meanwhile, compared to OptPO, SCPO also stops earlier on average, since its prior supplies evidence that a uniform prior cannot.

Our contributions are summarized as follows:

- We introduce SCPO, a test-time policy optimization framework that performs posterior-based early stopping of rollout generation, guided by a cross-epoch EMA prior that carries evidence from earlier visits to the same prompt.
- We prove finite-sample guarantees for the stopping rule: gap-dependent bounds on rollout complexity and mode-correct acceptance under a fixed within-visit answer distribution, and a characterization of when cross-epoch memory helps versus hurts under policy drift.
- We provide empirical results on different models and benchmarks to show that SCPO significantly reduces the number of generated rollouts and tokens compared to state-of-the-art frameworks, while achieving almost the same model performance.

2 Related Work

Reinforcement learning for LLM reasoning. Post-training is widely used to improve the reasoning capabilities of large language models. Some approaches combine supervised fine-tuning with reinforcement learning from human feedback (RLHF, Ouyang et al. 2022; Ackermann et al. 2026). Reinforcement learning with verifiable rewards (RLVR, Shao et al. 2024; Guo et al. 2025; Lee et al. 2026) instead uses factual correctness, such as math solutions or code tests, to derive rewards for training. Self-rewarding language models (Yuan et al., 2024; Wang et al., 2025c) use the same model to assign rewards for its own rollouts. Despite their different learning objectives, these methods rely on additional supervision, such as preferences or verifiable output. Test-time reinforcement learning (TTRL) (Zuo et al., 2025) removes this requirement: it samples a fixed number of rollouts

per unlabeled test prompt and uses the majority-voted answer as a pseudo-label for policy updates. Our work follows this setting, in which no ground-truth rewards or external verifiers are available, and targets the computational overhead of generating the same rollout budget for every prompt.

Adaptive rollout allocation and sequential mode estimation. A recurring lesson in both inference and training is that not every question deserves the same sampling budget. Self-consistency (Wang et al., 2023) improves reasoning by majority voting over sampled paths, and adaptive variants save compute by stopping as soon as the answers agree (Aggarwal et al., 2023; Huang et al., 2026). Compute-optimal test-time scaling pushes this further, spending more inference compute on questions estimated to be hard (Snell et al., 2024; Wang et al., 2025a; Liu et al., 2026). The same idea has since reached RL training; for example, rollout budgets are redistributed across prompts (Nguyen et al., 2026), or prompts that past rewards mark as uninformative are skipped altogether (Zheng et al., 2025). On the statistical side, sequential mode identification offers PAC guarantees and certified stopping rules for finding the most frequent outcome of a discrete distribution (Anand Jain et al., 2022; Cordero-Encinar & Duncan, 2025; Ota et al., 2026). None of these transfers directly to test-time policy optimization. Inference-time methods never update the model, adaptive RLVR relies on verifiable rewards, and mode identification assumes a fixed distribution and an unlimited budget. The closest work, OptPO (Wang et al., 2025b), brings adaptive stopping to TTRL through a Bayesian sequential test between the leading answer and the runner-up. However, it discards what it learned about a prompt once the visit ends. SCPO instead carries this evidence forward through a bounded cross-visit prior and characterizes when doing so remains reliable as the policy drifts.

Pseudo-label uncertainty and policy optimization. Majority-voted pseudo-labels in TTRL can be incorrect, motivating methods to increase their quality. SCOPE (Wang et al., 2026) uses step-wise confidence and subgroup consensus to construct pseudo-labels. SCRL (Yan et al., 2026) filters unreliable positive pseudo-labels and introduces entropy-gated negative supervision. CoRE (Mehrish & Vascon, 2026) builds a consensus graph over rollouts and uses its equilibrium solution to obtain a refined pseudo-label and graded rewards. TTRL-Guard (Lin et al., 2026) identifies unstable training periods and reduces their effect through reward scaling, minority-preserving sampling, and selective updates. These methods focus on improving the reliability of pseudo-labels or policy updates, rather than reducing the number of rollouts generated for each prompt.

3 Sequential Consensus Policy Optimization (SCPO)

We introduce *Sequential Consensus Policy Optimization* (SCPO), a test-time policy optimization framework that adaptively decides how many rollouts to generate for each prompt. SCPO maintains a Dirichlet posterior over the answer probabilities of each prompt. Its prior is a prompt-specific exponential moving average of answer counts from earlier visits. SCPO stops generation once the posterior probability that some answer is the policy's most probable answer exceeds a threshold, and the accepted answer then serves as the pseudo-label for the policy update. Figure 2 gives an overview of the framework.

A fundamental difference between SCPO and TTRL (Zuo et al., 2025) lies in how the pseudo-label is obtained. TTRL treats majority voting as a fixed-budget computation: every prompt receives the same number of rollouts at every visit, regardless of how quickly its vote settles. SCPO instead treats it as a sequential estimation problem, in which the rollout budget adapts to the observed evidence and each visit reuses what earlier visits revealed about the same prompt. OptPO (Wang et al., 2025b) also stops adaptively, but it relies only on the current visit and tests the leader against the runner-up. SCPO's stopping rule is multiway over the full answer set and is warm-started by cross-visit memory.

Problem Setup. Consider a model with parameters $\boldsymbol{\theta}$ and policy $\pi_{\boldsymbol{\theta}}(y \mid x)$. Each time a test prompt x is visited during training, we generate t rollouts $\{y_i\}_{i=1}^{t}$ and extract their answers $\mathcal{D}_t = \{\mathrm{ans}(y_i)\}_{i=1}^{t}$. Following Aggarwal et al. (2023), we model these answers as conditionally i.i.d. categorical draws over a finite set of values $\mathcal{V} = \{v^{(1)}, \ldots, v^{(m)}\}$, whose probabilities follow a Dirichlet prior, $\boldsymbol{p} \sim \mathrm{Dirichlet}(\boldsymbol{\alpha})$. Writing $n_j(t)$ for the number of occurrences of $v^{(j)}$ in $\mathcal{D}_t$, the likelihood of the observed answers is $\mathbb{P}(\mathcal{D}_t \mid \boldsymbol{p}) = \prod_{j=1}^{m} p_j^{n_j(t)}$. Since the Dirichlet prior is conjugate to this likelihood, Bayes' rule gives the posterior $\boldsymbol{p} \mid \mathcal{D}_t \sim \mathrm{Dirichlet}(\boldsymbol{\alpha} + \boldsymbol{n}(t))$, where $\boldsymbol{n}(t)$ collects the counts $n_j(t)$.

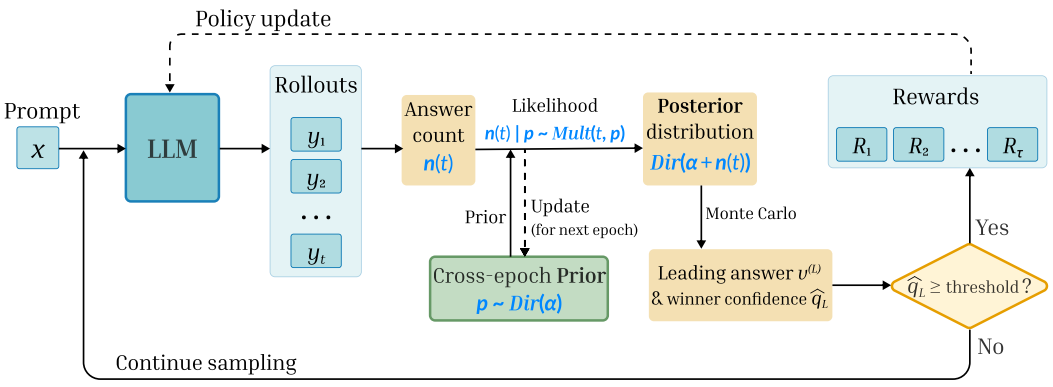


Figure 2: **Overview of SCPO.** SCPO combines the likelihood based on current answer counts with a prior parameterized by an EMA of past answer counts. Sampling continues until the winner confidence of an answer value exceeds the stopping threshold, after which the resulting pseudo-rewards are used to update the policy.

Estimating Winner Confidence. The stopping decision depends on which answer is most probable under the current policy. We therefore define the *winner confidence* $q_j(t)$ as the posterior probability that $v^{(j)}$ is the policy's most probable answer, i.e., that p_j exceeds all other answer probabilities:

$$q_j(t) = \mathbb{P}\left(p_j > \max_{k \neq j} p_k \,\middle|\, \mathcal{D}_t, \boldsymbol{\alpha}\right). \tag{1}$$

This probability has no closed form, so we approximate it by $\widehat{q}_j(t)$, the proportion of B Monte Carlo samples from the posterior in which p_j is the maximum.

Because the stopping condition is checked repeatedly within a visit, the Monte Carlo error must be controlled uniformly over all checks. Let $N_{\min}$ and $N_{\max}$ be the minimum and maximum numbers of rollouts per prompt in each visit, let $\mathcal{T} \subseteq \{N_{\min}, \ldots, N_{\max}\}$ be the set of times at which we check the stopping condition, and let $K = |\mathcal{T}|$. For any answer j and check time t, $\widehat{q}_j(t)$ is an average of B Bernoulli random variables with mean $q_j(t)$. Therefore, for a per-visit Monte Carlo failure probability δ_{MC}, Hoeffding's inequality and a union bound over at most mK pairs of answer values and check times give

$$\mathbb{P}_{\mathrm{MC}}\Big(\max_{t \text{ reached}} \max_{j} |\widehat{q}_j(t) - q_j(t)| \leq \varepsilon_{\mathrm{MC}}\Big) \geq 1 - \delta_{\mathrm{MC}}, \qquad \varepsilon_{\mathrm{MC}} = \sqrt{\frac{\log(2mK/\delta_{\mathrm{MC}})}{2B}}. \tag{2}$$

Here, the maximum ranges over the check times reached before stopping, so $\varepsilon_{\mathrm{MC}}$ is the margin that controls every Monte Carlo decision within a single visit.

Adaptive Stopping Rule. To reduce redundant rollout sampling, we stop sampling at the first step τ when the winner confidence for an answer reaches the certified threshold:

$$\tau = \inf\Big\{t \in \mathcal{T} : \max_{j \in [m]} \widehat{q}_j(t) \geq \gamma + \varepsilon_{\mathrm{MC}}\Big\}. \tag{3}$$

If no answer reaches this threshold, we continue sampling until the full budget of $N_{\max}$ rollouts is used. At stopping time, we choose the answer with the highest winner confidence as the pseudo-label $\widehat{L}$, and compute rewards $R_i = \mathbf{1}\{\mathrm{ans}(y_i) = v^{(\widehat{L})}\}$.

Policy Optimization. Using the rewards R_i defined above, we update model parameters $\boldsymbol{\theta}$ using GRPO (Shao et al., 2024):

$$\mathcal{L}_{\mathrm{SCPO}}(\boldsymbol{\theta}) = -\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|y_i|}\sum_{u=1}^{|y_i|}\min\Big\{r_{i,u}(\boldsymbol{\theta})\widehat{A}_i,\ \mathrm{clip}\big(r_{i,u}(\boldsymbol{\theta}), 1-\varepsilon, 1+\varepsilon\big)\widehat{A}_i\Big\} + \beta\,\mathbb{D}_{\mathrm{KL}}\big(\pi_{\boldsymbol{\theta}}(\cdot \mid x)\,\|\,\pi_{\mathrm{ref}}(\cdot \mid x)\big), \tag{4}$$

where G is the rollout group size, $r_{i,u}(\boldsymbol{\theta}) = \frac{\pi_{\boldsymbol{\theta}}(y_{i,u}|x, y_{i,<u})}{\pi_{\boldsymbol{\theta}_{\mathrm{old}}}(y_{i,u}|x, y_{i,<u})}$ is the per-token probability ratio, and $\widehat{A}_i = \frac{R_i - \mathrm{mean}(\boldsymbol{R})}{\mathrm{std}(\boldsymbol{R}) + \epsilon}$ is the normalized group advantage.

Cross-Epoch Prior. Since prompts are revisited across epochs, earlier answer counts can inform later stopping decisions. We maintain this historical evidence using a prompt-specific EMA prior. Let $\boldsymbol{\alpha}^{(r)}$ be the Dirichlet prior parameters for a prompt x at epoch r, initialized symmetrically as $\boldsymbol{\alpha}^{(1)} = \alpha_0 \mathbf{1}$. After each epoch, we update the prior for the next epoch by:

$$\boldsymbol{\alpha}^{(r+1)} = \mu \boldsymbol{\alpha}^{(r)} + (1-\mu)\boldsymbol{n}^{(r)}(N_{\min}). \tag{5}$$

Here, $\boldsymbol{n}^{(r)}(N_{\min})$ counts answers in the mandatory fixed prefix $N_{\min}$ rollouts. This avoids the optional-stopping bias and visit-dependent prior mass that arise from reusing the full completed rollout group. When the answer distribution remains stable, the resulting prior can reduce the fresh rollouts needed for stopping. Our complete algorithm is shown in Algorithm 1 of Appendix A.

4 Theoretical Analysis

In this section, we theoretically analyze the stopping gate in three steps. Section 4.1 shows that the threshold $\gamma + \varepsilon_{\mathrm{MC}}$ certifies an exact winner confidence of at least γ at the adaptive stopping time. This certifies the posterior score but not that the accepted answer is the mode. Section 4.2 closes this gap with a separation certificate, which yields gap-dependent bounds on stopping time, mode-correct acceptance, and expected rollouts. Section 4.3 bounds how far the EMA prior drifts from the current answer distribution and plugs the resulting prior lead into the same certificate, characterizing when memory speeds up certification.

4.1 Posterior certification with finite Monte Carlo simulation

At check $t \in \mathcal{T}$, the gate estimates the exact working-law winner probability $q_j(t)$ with B draws. The next result shows why thresholding at $\gamma + \varepsilon_{\mathrm{MC}}$ preserves level γ despite selecting the acceptance time and label from these estimates.

Proposition 4.1 (Uniform Monte Carlo certification at adaptive acceptance)**.** *Let $K = |\mathcal{T}|$ and $\varepsilon_{\mathrm{MC}} = \sqrt{\log(2mK/\delta_{\mathrm{MC}})/(2B)}$. For every fixed realization of the rollout stream, with conditional probability at least $1 - \delta_{\mathrm{MC}}$ over the auxiliary simulation seeds,*

$$|\widehat{q}_j(t) - q_j(t)| \le \varepsilon_{\mathrm{MC}} \quad \textit{for every reached } t \in \mathcal{T}, j \in [m]. \tag{6}$$

On this event, acceptance implies $q_{\widehat{L}}(\widehat{\tau}) \ge \gamma$.

Remark 4.2. The Monte Carlo radius is a valid finite-sample uniform correction, though the Hoeffding–union bound need not be tight. It certifies the working score even with an EMA prior, except on one per-visit simulation failure event. It does not by itself certify that the accepted label is the current mode or ground truth. Please see the full proofs in Appendix B.

4.2 Finite-sample rollout complexity and mode-correct acceptance

We now analyze frequentist stopping under an arbitrary fixed answer distribution: for each rival answer j, is the expected gap between the mode $j^\star$'s and j's Dirichlet parameters, accumulated from both the prior and the fresh rollouts collected so far, large enough to survive two sources of slack, namely fresh-count sampling noise and the margin the winner-score bound (Lemma C.1) requires given the total pairwise mass observed? The next block of notation sets up exactly these three quantities.

Let $j^\star = \arg\max_j p_j$ be the unique mode, $\Delta_j = p_{j^\star} - p_j > 0$, and $\boldsymbol{\beta}(t) = \boldsymbol{\alpha} + \boldsymbol{n}(t)$. We assume that the exact-score certification level satisfies $\gamma \in (1/2, 1)$, set $\overline{\gamma} = \gamma + 2\varepsilon_{\mathrm{MC}} < 1$ and, for $\delta \in (0,1)$, let $\ell_\delta = \log \frac{m-1}{\delta}$, $\ell_\gamma = \log \frac{m-1}{1-\overline{\gamma}}$. For competitor j, let $g_j(t) = \mathbb{E}[\beta_{j^\star}(t) - \beta_j(t) \mid \mathcal{H}]$ be its expected parameter separation, and let $s_j(t)$ upper-bound $\beta_{j^\star}(t) + \beta_j(t)$ almost surely. The *separation certificate* at check t requires, for every $j \ne j^\star$,

$$\underbrace{g_j(t)}_{\text{expected separation}} \quad \ge \quad \underbrace{\sqrt{2t\ell_\delta}}_{\text{fresh-sampling noise}} \quad + \quad \underbrace{\sqrt{2s_j(t)\ell_\gamma}}_{\text{winner-score requirement}} \quad . \tag{7}$$

Theorem 4.3 (Finite-sample stopping and mode-correct acceptance)**.** *Fix a pre-specified check $t \in \mathcal{T}$, let $\mathcal{W}_t$ be the event that an incorrect exact score crosses γ at some check through t, and let $\rho_{\mathrm{wrong}}^{(\le t)}(\gamma)$ be any upper bound on $\mathbb{P}(\mathcal{W}_t \mid \mathcal{H})$. If every $j \neq j^\star$ satisfies*

$$\alpha_{j^\star} - \alpha_j + t\Delta_j \geq \sqrt{2t\ell_\delta} + \sqrt{2(\alpha_{j^\star} + \alpha_j + t)\ell_\gamma}, \tag{8}$$

then, conditional on $\mathcal{H}$:

(i) Stopping Guarantee: $\mathbb{P}(\widehat{\tau} \leq t \mid \mathcal{H}) \geq 1 - \delta - \delta_{\mathrm{MC}}$.

(ii) Mode-Correct Acceptance: With $\rho_{\mathrm{wrong}}^{(\le t)}(\gamma)$ bounding earlier wrong crossings:

$$\mathbb{P}(\widehat{\tau} \leq t,\, \widehat{L} = j^\star \mid \mathcal{H}) \geq \left[1 - \delta - \delta_{\mathrm{MC}} - \rho_{\mathrm{wrong}}^{(\le t)}(\gamma)\right]_+, \tag{9}$$

where for $\boldsymbol{\alpha} = \alpha_0 \mathbf{1}$, $\rho_{\mathrm{wrong}}^{(\le t)}(\gamma) \leq \min\left\{1,\, (m-1)K_t e^{-N_{\min}\Delta_{\min}^2/2}\right\}$ with $K_t = |\{s \in \mathcal{T} : s \leq t\}|$.[1]

(iii) Expected Rollouts: For capped budget $\widehat{T} = \min\{\widehat{\tau}, N_{\max}\}$:

$$\mathbb{E}[\widehat{T} \mid \mathcal{H}] \leq t + \min\{1,\, \delta + \delta_{\mathrm{MC}}\}(N_{\max} - t). \tag{10}$$

Corollary 4.4 (Explicit sample complexity for symmetric prior)**.** *For $\boldsymbol{\alpha} = \alpha_0 \mathbf{1}$ and $\Delta_{\min} = \min_{j \neq j^\star} \Delta_j$, condition* (8) *holds at the earliest check:*

$$t_{\mathrm{sym}} \coloneqq \min\left\{ s \in \mathcal{T} : s \geq \max\left(\frac{16(\ell_\delta + \ell_\gamma)}{\Delta_{\min}^2},\ \frac{4\sqrt{\alpha_0 \ell_\gamma}}{\Delta_{\min}} \right) \right\}. \tag{11}$$

If $t_{\mathrm{sym}} < \infty$, conclusions (i)–(iii) of Theorem 4.3 apply at $t = t_{\mathrm{sym}}$.

Remark 4.5. Stopping by t and accepting the mode by t differ because the latter must exclude an earlier wrong crossing; ρ_{wrong} quantifies that extra risk. The third conclusion converts the stopping tail into completed-rollout cost. Please refer to Appendix C for full proofs.

4.3 Memory under policy drift

Memory reuses fixed-prefix evidence from earlier visits, when the policy may have been different. At visit $r \geq 2$, SCPO warm-starts the prior from fixed-prefix frequencies $\widehat{\boldsymbol{p}}^{(s)} = \boldsymbol{n}^{(s)}(N_{\min})/N_{\min}$ of earlier visits. Unrolling Equation (5) decomposes the prior into total mass M_r and historical consensus direction $\overline{\boldsymbol{p}}^{(r)}$: $\boldsymbol{\alpha}^{(r)} = \mu^{r-1}\alpha_0 \mathbf{1} + M_r \overline{\boldsymbol{p}}^{(r)}$ where $M_r = N_{\min}(1 - \mu^{r-1}) < N_{\min}$, $\overline{\boldsymbol{p}}^{(r)} = \sum_{s<r} w_{r,s} \widehat{\boldsymbol{p}}^{(s)}$ with normalized geometric weights $w_{r,s} = \frac{(1-\mu)\mu^{r-1-s}}{1-\mu^{r-1}}$.

Here $\overline{\boldsymbol{p}}^{(r)}$ is a weighted historical frequency vector and $M_r < N_{\min}$ is its pseudo-count mass. Let the current law have unique mode $j_r^\star$ and gaps $\Delta_{r,j} = p_{j_r^\star}^{(r)} - p_j^{(r)}$. Against competitor j, memory contributes $M_r(\overline{p}_{j_r^\star}^{(r)} - \overline{p}_j^{(r)})$ to the prior lead $\alpha_{j_r^\star}^{(r)} - \alpha_j^{(r)}$. Its usefulness therefore depends on how well $\overline{\boldsymbol{p}}^{(r)}$ tracks the current law. Adding and subtracting the historical laws gives

$$\overline{\boldsymbol{p}}^{(r)} - \boldsymbol{p}^{(r)} = \underbrace{\sum_{s<r} w_{r,s}(\widehat{\boldsymbol{p}}^{(s)} - \boldsymbol{p}^{(s)})}_{\text{sampling noise}} + \underbrace{\sum_{s<r} w_{r,s}(\boldsymbol{p}^{(s)} - \boldsymbol{p}^{(r)})}_{\text{policy drift}}. \tag{12}$$

Thus, whenever $\|\overline{\boldsymbol{p}}^{(r)} - \boldsymbol{p}^{(r)}\|_\infty \leq \varepsilon_r$, $\overline{p}_{j_r^\star}^{(r)} - \overline{p}_j^{(r)} \geq \Delta_{r,j} - 2\varepsilon_r$. For a drift bound $V \geq 0$ and confidence $\delta_{M,r}$, define

$$N_{\mathrm{eff},r} = N_{\min}\tfrac{1+\mu}{1-\mu}\tfrac{1-\mu^{r-1}}{1+\mu^{r-1}}, \quad \varepsilon_r = \sqrt{\tfrac{\log(2m/\delta_{M,r})}{2N_{\mathrm{eff},r}}} + V\left(\tfrac{1}{1-\mu} - \tfrac{(r-1)\mu^{r-1}}{1-\mu^{r-1}}\right). \tag{13}$$

where $N_{\mathrm{eff},r} \leq N_{\min}(r-1)$ is the historical evidence's effective sample size, reduced below the raw count by geometric downweighting of older visits, and ε_r combines the resulting sampling-noise term with a drift term weighted by the age of past evidence. Whenever the tracking event $\|\overline{\boldsymbol{p}}^{(r)} - \boldsymbol{p}^{(r)}\|_\infty \leq \varepsilon_r$ holds, every $j \neq j_r^\star$ satisfies

$$g_j(t) \geq (t + M_r)\Delta_{r,j} - 2M_r\varepsilon_r, \qquad s_j(t) = t + 2\mu^{r-1}\alpha_0 + M_r, \tag{14}$$

combining the fresh-sample lead $t\Delta_{r,j}$ with a historical contribution of at least $M_r(\Delta_{r,j} - 2\varepsilon_r)$. Substituting these into the general separation certificate yields the following.

[1] A tighter threshold-sensitive bound for general asymmetric priors is provided in Appendix C.3.

Theorem 4.6 (EMA tracking and stopping transfer)**.** *Fix $r \geq 2$, a pre-specified check $t \in \mathcal{T}$, $V \geq 0$, and $\delta_{M,r} \in (0,1)$. Assume $\|\boldsymbol{p}^{(s+1)} - \boldsymbol{p}^{(s)}\|_\infty \leq V$ almost surely for $s < r$. Let $\mathcal{A}_{r,t}$ be the pre-visit event that, for every $j \neq j_r^\star$,*

$$(t + M_r)\Delta_{r,j} \geq \sqrt{2t\ell_\delta} + \sqrt{2(t + 2\mu^{r-1}\alpha_0 + M_r)\ell_\gamma} + 2M_r\varepsilon_r. \tag{15}$$

Then, even when the policy adapts between visits:

(i) Tracking: $\mathbb{P}(\|\overline{\boldsymbol{p}}^{(r)} - \boldsymbol{p}^{(r)}\|_\infty \leq \varepsilon_r) \geq 1 - \delta_{M,r}$.
(ii) Stopping Transfer: jointly over history, rollouts, and simulation,

$$\mathbb{P}(\mathcal{A}_{r,t} \cap \{\widehat{\tau}_r > t\}) \leq \delta_{M,r} + \delta + \delta_{\mathrm{MC}}. \tag{16}$$

Remark 4.7. At the same check t, compare Theorem 4.6 with Theorem 4.3 specialized to the cold prior $\alpha_0\mathbf{1}$. Both certificates contain the current-rollout term $\sqrt{2t\ell_\delta}$. Rewriting the EMA lead as $t\Delta_{r,j} + M_r(\Delta_{r,j} - 2\varepsilon_r)$ cancels this common term, showing that the EMA certificate is no harder at t whenever

$$\underbrace{M_r(\Delta_{r,j} - 2\varepsilon_r)}_{\text{reliable historical lead}} \geq \underbrace{\sqrt{2\ell_\gamma}\left(\sqrt{t + 2\mu^{r-1}\alpha_0 + M_r} - \sqrt{t + 2\alpha_0}\right)}_{\text{change in concentration cost}}. \tag{17}$$

The right side has the sign of $N_{\min} - 2\alpha_0$: for $N_{\min} > 2\alpha_0$ this requires a positive robust gap $\Delta_{r,j} - 2\varepsilon_r$; at equality a nonnegative gap suffices; and for $N_{\min} < 2\alpha_0$ the change in concentration favors EMA outright, although stale memory can still hurt (see Proposition D.1). We set $N_{\min} > 2\alpha_0$ throughout all experiments, so EMA shortens the certified budget only by supplying a robustly positive tracked gap $\Delta_{r,j} - 2\varepsilon_r$. This is consistent with Figure 1, where prompts whose leader stays stable begin with much larger gaps and show modest but consistent savings from EMA (see Table 2). Please see the full proof in Appendix D.

5 Experiments

5.1 Experimental Setup

Baselines and Metrics. We compare SCPO with TTRL (Zuo et al., 2025) and OptPO (Wang et al., 2025b) on the total number of generated tokens and the average number of generated rollouts per prompt (denoted as ***R/P***) *during training*. To evaluate the final model performance, we sample 16 responses per question and report ***Mean@16***, the percentage of correct responses, and ***Pass@16***, the percentage of questions for which at least one response is correct.

Models and Benchmarks. We use Qwen2.5-Math-1.5B (Yang et al., 2024) and Llama-3.1-8B-Instruct (Grattafiori et al., 2024) as the backbone models. We evaluate methods on three main benchmarks: AIME 2024 (Art of Problem Solving, 2024), MATH-500 (Hendrycks et al., 2021b), and LiveCodeBench-v6 (Jain et al., 2024).

Implementation Details. Unless specified otherwise, we set the stopping threshold $\gamma + \varepsilon_{\mathrm{MC}} = 0.6$ and EMA coefficient $\mu = 0.5$ across all SCPO experiments. We set the initial Dirichlet prior α_0 to 1 for MATH-500 and 10^{-3} for the remaining benchmarks. Following TTRL (Zuo et al., 2025), we set the number of epochs to 5, 10, and 80 for LiveCodeBench-v6, MATH-500 and AIME 2024, respectively. For rollout generation, we generate up to $N_{\max} = 64$ responses per prompt for voting, using a temperature of 0.6 for Llama3.1-8B-Instruct and 1.0 for Qwen2.5-Math-1.5B. The minimum number of rollouts per prompt is model-dependent: 16 for Llama and 32 for Qwen. Further details are given in Appendix F.

5.2 Main Results

Table 1 shows that SCPO consistently improves rollout and token efficiency across models and benchmarks while maintaining comparable task performance. For example, with Qwen2.5-Math-1.5B on MATH-500, it reduces the number of rollouts per prompt to 34.2 (compared to 38.2 of OptPO and the full budget of 64 for TTRL), saving 44.1% of generated tokens during test-time training (Figure 3b-c), while Mean@16 and Pass@16 remain almost the same to those of the two baselines, driven by high pseudo-label accuracy throughout training (Figure 3a).

Table 1: Performance comparison of SCPO and baselines. R/P is the average number of training rollouts per prompt, and token savings are relative to TTRL. SCPO substantially reduces the number of rollouts and tokens compared to baselines while Mean@16 and Pass@16 remain comparable.

Benchmark	Method	Mean@16 ↑	Pass@16 ↑	R/P ↓	Num. tokens ↓	Saving ↑
Qwen2.5-Math-1.5B						
AIME 2024	*Base*	8.5	40.0	–	–	–
	TTRL	**17.8** ± 0.3	37.8 ± 3.9	64.0 ± 0.0	146.3 ± 2.8M	0.0%
	OptPO	16.2 ± 2.1	**40.0** ± 5.8	48.8 ± 1.3	117.6 ± 4.1M	19.6%
	SCPO	16.2 ± 2.1	36.7 ± 3.4	**41.2** ± 1.2	**98.0** ± 3.5M	**33.0%**
MATH-500	*Base*	32.5	83.6	–	–	–
	TTRL	72.5 ± 0.4	90.4 ± 0.3	64.0 ± 0.0	216.7 ± 2.2M	0.0%
	OptPO	**72.9** ± 0.2	**90.5** ± 0.1	38.2 ± 0.1	139.2 ± 1.0M	35.8%
	SCPO	72.8 ± 0.2	**90.5** ± 1.0	**34.2** ± 0.1	**121.2** ± 1.3M	**44.1%**
Llama-3.1-8B-Instruct						
AIME 2024	*Base*	7.1	33.3	–	–	–
	TTRL	8.7 ± 0.5	24.4 ± 5.1	64.0 ± 0.0	107.3 ± 1.0M	0.0%
	OptPO	8.1 ± 0.8	**27.8** ± 3.9	29.7 ± 5.0	54.6 ± 8.9M	49.2%
	SCPO	**9.3** ± 1.2	26.7 ± 6.7	**18.3** ± 0.8	**35.9** ± 0.1M	**66.5%**
MATH-500	*Base*	50.7	81.8	–	–	–
	TTRL	**60.7** ± 1.0	80.5 ± 0.5	64.0 ± 0.0	178.0 ± 3.6M	0.0%
	OptPO	59.7 ± 0.8	79.5 ± 0.3	29.6 ± 1.3	101.8 ± 7.1M	42.8%
	SCPO	57.2 ± 0.4	**81.1** ± 1.4	**21.0** ± 0.6	**61.8** ± 3.7M	**65.3%**
LiveCodeBench-v6	*Base*	15.1	26.9	–	–	–
	TTRL	17.6 ± 0.7	24.4 ± 0.9	64.0 ± 0.0	12.5 ± 1.6M	0.0%
	OptPO	**18.0** ± 0.3	24.2 ± 1.7	49.2 ± 0.9	10.7 ± 1.2M	14.1%
	SCPO	**18.0** ± 0.7	**26.1** ± 0.7	**34.4** ± 0.5	**6.5** ± 0.2M	**48.0%**

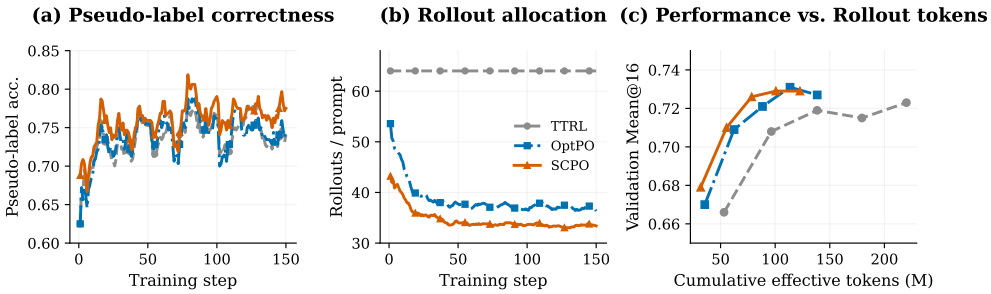


Figure 3: **(a)** Pseudo-label correctness (10-step mean); **(b)** Average number of rollouts per prompt (10-step mean); **(c)** Performance vs. training rollout tokens used during training on MATH-500 with Qwen2.5-Math-1.5B. SCPO achieves comparable model performance compared to TTRL and OptPO despite using fewer rollouts and tokens.

The efficiency gains are even larger with Llama-3.1-8B-Instruct. On AIME 2024, SCPO uses only 18.3 rollouts per prompt and 35.9M tokens, compared with 29.7 and 54.6M for OptPO and 64.0 and 107.3M for TTRL, while achieving the comparable Mean@16 and Pass@16. On LiveCodeBench-v6, SCPO reduces the number of tokens by 48.0%, more than three times the saving achieved by OptPO.

5.3 Ablation Study

Impacts of Cross-Epoch Prior. Table 2 shows that our proposed cross-epoch prior update improves sampling efficiency on both benchmarks. On AIME 2024, it reduces the average number of rollouts per prompt from 45.4 to 41.2 and token usage from 111.7M to 98.0M, increasing the token saving relative to TTRL from 23.7% to 33.0%. The same trend holds on MATH-500, where prior update reduces token usage from 126.2M to 121.2M.

Impacts of Stopping Threshold. Table 3 illustrates the trade-off between stopping threshold $\gamma + \varepsilon_{\text{MC}}$ and generation cost. As the threshold increases from 0.5 to 0.9, the stopping rule becomes

Table 2: Ablation study on cross-epoch prior update for Qwen2.5-Math-1.5B. R/P and token saving are relative to TTRL. The EMA prior consistently reduces rollouts and tokens on both benchmarks without sacrificing accuracy.

Benchmark	Method	Mean@16 ↑	Pass@16 ↑	R/P ↓	Num. tokens ↓	Saving ↑
AIME 2024	w/o EMA	16.0	**36.7**	45.4	111.7M	23.7%
	SCPO	**16.2**	**36.7**	**41.2**	**98.0M**	**33.0%**
MATH-500	w/o EMA	72.5	90.0	35.1	126.2M	41.8%
	SCPO	**72.8**	**90.5**	**34.2**	**121.2M**	**44.1%**

stricter: R/P increases monotonically from 33.6 to 37.1, while token usage rises from 118.8M to 137.7M. In contrast, task performance remains stable, with Mean@16 ranging from 72.5 to 73.1, showing that our method works well with a wide range of stopping threshold.

Table 3: Stopping-threshold sensitivity on MATH-500 with Qwen2.5-Math-1.5B. Task performance stays stable across thresholds, while lower threshold enables more rollout savings and token savings

$\gamma + \varepsilon_{\mathrm{MC}}$	Mean@16 ↑	Pass@16 ↑	R/P ↓	Tokens ↓	Saving ↑
0.5	72.6	90.8	33.6	118.8M	45.2%
0.6	72.8	90.5	34.2	121.2M	44.1%
0.7	72.5	91.0	35.1	126.7M	41.5%
0.8	72.8	90.2	36.0	131.7M	39.2%
0.9	73.1	89.0	37.1	137.7M	36.5%

Token vs. Runtime Trade-off. To evaluate SCPO in a practical runtime setting, we set the extra rollout batch size (EBS), i.e. the number of additional rollouts generated between two consecutive stopping checks after $N_{\min}$, to 8 for both OptPO and SCPO. Although $\mathrm{EBS} = 1$ allows earlier stopping, it does not take advantage of batched generation in inference engines such as vLLM (Kwon et al., 2023). As shown in Table 4, SCPO reduces 9.5% runtime compared with OptPO, but is 9.3% slower than TTRL, which can generate its rollouts with better parallelism. On the other hand, compared with the $\mathrm{EBS} = 1$ results in Table 1, increasing EBS raises OptPO's average rollouts per prompt from 38.2 to 44.0 and its token usage from 139.2M to 158.0M. In contrast, rollout and token efficiency of SCPO remains stable, from 34.2 to 34.6 rollouts per prompt and from 121.2M to 122.6M tokens, which is still 43.4% lower than TTRL. These results show that SCPO provides a better practical trade-off between runtime and token efficiency than the two baselines.

Table 4: Efficiency on MATH-500 with Qwen2.5-Math-1.5B. Using an extra rollout batch size of 8, SCPO reduces both runtime and tokens compared with OptPO, and achieves a 43.4% token reduction with minimal wall-clock overhead compared with TTRL.

Method	R/P ↓	Tokens ↓	Runtime ↓	Mean@16 ↑
TTRL	64.0	216.7M	4h30m	72.5
OptPO	44.0	158.0M	5h26m	72.5
SCPO	34.6	122.6M	4h55m	72.8

6 Limitations and Conclusion

In this paper, we present SCPO, an adaptive stopping method that reduces test-time RL rollout costs by maintaining a bounded cross-epoch EMA Dirichlet prior to stop sampling once confidence in the leading answer exceeds a threshold. Empirically, SCPO substantially reduces rollout and token usage compared to standard TTRL while preserving final model performance.

SCPO does not address a fundamental weakness of majority-vote training: its stopping rule identifies the policy's consensus answer rather than ground-truth. It may therefore accept an incorrect pseudo-label with high confidence. Furthermore, the cross-epoch prior can become misleading after large policy updates. Finally, our theoretical analysis assumes a fixed finite answer set over epochs, which may not hold in open-ended generation tasks.

AI USE STATEMENT

We used generative AI to assist us with manuscript editing, related work retrieval, coding implementation, and proof checking and editing. All AI-assisted content was reviewed by the authors. AI tools were not used to generate synthetic data, or to generate empirical results.

REPRODUCIBILITY STATEMENT

Section 4 and Appendices B- D detail all theoretical assumptions, finite-sample guarantees, and complete proofs for SCPO. Algorithm 1 in Appendix A provides the full pseudocode, while Section 5.1 and Appendix F specify all experimental setups, model hyperparameters, evaluation benchmarks, and hardware details.

REFERENCES

Johannes Ackermann, Michael Noukhovitch, Takashi Ishida, and Masashi Sugiyama. Gradient regularization mitigates reward hacking in reinforcement learning from human feedback and verifiable rewards. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=T67db38qhr.

Pranjal Aggarwal, Aman Madaan, Yiming Yang, and Mausam. Let's sample step by step: Adaptive-consistency for efficient reasoning and coding with LLMs. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12375–12396, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.761. URL https://aclanthology.org/2023.emnlp-main.761/.

Ekin Akyürek, Mehul Damani, Adam Zweiger, Linlu Qiu, Han Guo, Jyothish Pari, Yoon Kim, and Jacob Andreas. The surprising effectiveness of test-time training for few-shot learning. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=asgBo3FNdg.

Shubham Anand Jain, Rohan Shah, Sanit Gupta, Denil Mehta, Inderjeet J. Nair, Jian Vora, Sushil Khyalia, Sourav Das, Vinay J. Ribeiro, and Shivaram Kalyanakrishnan. Pac mode estimation using ppr martingale confidence sequences. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 5815–5852. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/anand-jain22a.html.

Art of Problem Solving. AIME problems and solutions, 2024. URL https://artofproblemsolving.com/wiki/index.php/AIME_Problems_and_Solutions.

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

Guhao Feng, Shengjie Luo, Kai Hua, Ge Zhang, Wenhao Huang, Di He, and Tianle Cai. In-place test-time training. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=dTWfCLSoyl.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Moritz Hardt and Yu Sun. Test-time training on nearest neighbors for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=CNL2bku4ra.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021b. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Jingkai Huang, Will Ma, and Zhengyuan Zhou. Optimal bayesian stopping for efficient inference of consistent LLM answers. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=JXN3npSRRD.

Jonas Hübotter, Patrik Wolf, Aleksandr Shevchenko, Dennis Jüni, Andreas Krause, and Gil Kur. Specialization after generalization: Towards understanding test-time training in foundation models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=1c6Ao3CpKt.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024.

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

Chanuk Lee, Minki Kang, and Sung Ju Hwang. SAGE: Shaping anchors for guided exploration in RLVR of LLMs. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=VzRUI5ZVkp.

Hongxiang Lin, Zhirui Kuai, Erpeng Xue, and Lei Wang. Detecting and mitigating the correct-answer extinction window in test-time reinforcement learning with majority voting. *arXiv preprint arXiv:2605.19444*, 2026.

Zhangyi Liu, Huaizhi Qu, Xiaowei Yin, He Sun, Yanjun Han, Tianlong Chen, and Zhun Deng. PETS: A principled framework towards optimal trajectory allocation for efficient test-time self-consistency. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=L9jcGvJP2N.

Ambuj Mehrish and Sebastiano Vascon. Core: Consensus rewards via equilibrium for test-time reinforcement learning. *arXiv preprint arXiv:2608.09324*, 2026.

Hieu Trung Nguyen, Bao Nguyen, Wenao Ma, Yuzhi Zhao, Ruifeng She, and Viet Anh Nguyen. Adaptive rollout allocation for online reinforcement learning with verifiable rewards. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Z5sWYACAop.

Hirofumi Ota, Naoto Iwase, Yuki Ichihara, Junpei Komiyama, and Masaaki Imaizumi. Cite: Anytime-valid statistical inference in llm self-consistency. *arXiv preprint arXiv:2605.05873*, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters, 2024. URL https://arxiv.org/abs/2408.03314.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9229–9248. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/sun20b.html.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Weiqin Wang, Yile Wang, Kehao Chen, and Hui Huang. Beyond majority voting: Towards fine-grained and more reliable reward signal for test-time reinforcement learning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 37251–37265, 2026.

Xinglin Wang, Shaoxiong Feng, Yiwei Li, Peiwen Yuan, Yueqi Zhang, Chuyi Tan, Boyuan Pan, Yao Hu, and Kan Li. Make every penny count: Difficulty-adaptive self-consistency for cost-efficient reasoning. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 6919–6932, 2025a.

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.

Youkang Wang, Jian Wang, Rubing Chen, Tianyi Zeng, Xiao-Yong Wei, and Qing Li. OptPO: Optimal rollout allocation for test-time policy optimization. *arXiv preprint arXiv:2512.02882*, 2025b. URL https://arxiv.org/abs/2512.02882.

Zhaoyang Wang, Weilei He, Zhiyuan Liang, Xuchao Zhang, Chetan Bansal, Ying Wei, Weitong Zhang, and Huaxiu Yao. CREAM: Consistency regularized self-rewarding language models. In *The Thirteenth International Conference on Learning Representations*, 2025c. URL https://openreview.net/forum?id=Vf6RDObyEF.

Dong Yan, Jian Liang, Yanbo Wang, Shuo Lu, Ran He, and Tieniu Tan. What if consensus lies? selective-complementary reinforcement learning at test time. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28957–28970, 2026.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E Weston. Self-rewarding language models. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=0NphYCmgua.

Tianyuan Zhang, Sai Bi, Yicong Hong, Kai Zhang, Fujun Luan, Songlin Yang, Kalyan Sunkavalli, William T. Freeman, and Hao Tan. Test-time training done right. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Tb9qAxT3xv.

Haizhong Zheng, Yang Zhou, Brian R. Bartoldson, Bhavya Kailkhura, Fan Lai, Jiawei Zhao, and Beidi Chen. Act only when it pays: Efficient reinforcement learning for LLM reasoning via selective rollouts. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=x5lITYXmW2.

Yuxin Zuo, Kaiyan Zhang, Li Sheng, Shang Qu, Ganqu Cui, Xuekai Zhu, Haozhan Li, Yuchen Zhang, Xinwei Long, Ermo Hua, Biqing Qi, Youbang Sun, Zhiyuan Ma, Lifan Yuan, Ning Ding, and Bowen Zhou. TTRL: Test-time reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=VuVhgEiu20.

The appendix is organized as follows:

- Appendix A presents the complete SCPO algorithm.
- Appendix B provides Posterior and Monte Carlo certification.
- Appendix C provides the stopping guarantees, rollout cost, and earlier wrong crossings.
- Appendix D analyzes EMA tracking under policy drift, transfer to stopping, when memory helps.
- Appendix E adds additional experiment results.
- Appendix F provides more information on implementation details.

A SCPO ALGORITHM

Algorithm 1 presents the complete SCPO framework.

Algorithm 1 Sequential Consensus Policy Optimization (SCPO)

Require: Policy $\pi_{\boldsymbol{\theta}}$, prompt x with possible answers $\mathcal{V} = \{v^{(1)}, \ldots, v^{(m)}\}$, rollout budgets $N_{\min}, N_{\max}$, confidence threshold γ, number of Monte Carlo samples B, per-visit Monte Carlo failure probability δ_{MC}, initial Dirichlet prior parameter α_0, EMA decay factor μ

for each prompt x **do**
  Initialize prior parameter $\boldsymbol{\alpha}_x \leftarrow \alpha_0 \mathbf{1}_m$
end for
for each epoch r **do**
  for each prompt x **do**
    $K = N_{\max} - N_{\min} + 1$
    $\varepsilon_{\mathrm{MC}} \leftarrow \sqrt{\frac{\log(2mK/\delta_{\mathrm{MC}})}{2B}}$
    Initialize current counts $\boldsymbol{n} \leftarrow \mathbf{0}$
    for $t = 1, 2, \ldots, N_{\max}$ **do**
      Sample $y_t \sim \pi_{\boldsymbol{\theta}}(\cdot \mid x)$; $a_t \leftarrow \mathrm{ans}(y_t)$
      for $j = 1, 2, \ldots, m$ **do**
        Update counts $n_j(t) \leftarrow n_j(t-1) + \mathbb{I}(v_j = a_t)$
      end for
      if $t \geq N_{\min}$ **then**
        $\widehat{\boldsymbol{q}} \leftarrow \mathbf{0}$
        for $b = 1, 2, \ldots, B$ **do**
          Draw: $\widehat{\boldsymbol{p}} \sim \mathrm{Dirichlet}(\boldsymbol{\alpha}_x + \boldsymbol{n}(t))$
          $l = \arg\max_j \widehat{p_j}$
          $\widehat{q_l} \leftarrow \widehat{q_l} + 1/B$
        end for
        $\widehat{L} \leftarrow \arg\max_j \widehat{q}_j$
        if $\widehat{q}_{\widehat{L}} \geq \gamma + \varepsilon_{\mathrm{MC}}$ **then**
          break
        end if
      end if
    end for
    Update prior parameter $\boldsymbol{\alpha}_x \leftarrow \mu \boldsymbol{\alpha}_x + (1-\mu)\boldsymbol{n}(N_{\min})$
    for each training rollout y_i **do**
      $R_i \leftarrow \mathbb{I}[\mathrm{Extract}(y_i) = v^{(\widehat{L})}]$
    end for
    Update $\boldsymbol{\theta}$ using Equation 4
  end for
end for

B FINITE MONTE CARLO CERTIFICATION

Simultaneous accuracy. Conditional on the potential answer stream and $\mathcal{H}$, each coordinate's winner count at check t is a sum of B independent Bernoulli variables with mean $q_j(t)$. Therefore

$$\mathbb{P}_{\mathrm{MC}}(|\widehat{q}_j(t) - q_j(t)| > \varepsilon \mid \text{potential stream}, \mathcal{H}) \le 2e^{-2B\varepsilon^2}. \tag{18}$$

A union bound over mK coordinate–check pairs yields

$$\mathbb{P}_{\mathrm{MC}}\left(\max_{t\in\mathcal{T}} \max_{j} |\widehat{q}_j(t) - q_j(t)| > \varepsilon \,\middle|\, \text{potential stream}, \mathcal{H}\right) \le 2mKe^{-2B\varepsilon^2}. \tag{19}$$

Substituting Equation 2 proves the uniform event used in the main results. It also covers the subset of checks actually reached, so data-dependent stopping creates no additional simulation failure term. Unused posterior draws are only a proof device. The fresh independent seeds assumed in the method justify the conditional construction; a particle-reuse scheme that changes the conditional sampling law requires separate analysis.

Certification versus stopping feasibility. The gate requires $\gamma + \varepsilon_{\mathrm{MC}} \le 1$, whereas the sufficient stopping theorem uses $\gamma + 2\varepsilon_{\mathrm{MC}} < 1$. Solving these inequalities gives, respectively,

$$B \ge \frac{\log(2mK/\delta_{\mathrm{MC}})}{2(1-\gamma)^2}, \qquad B > \frac{2\log(2mK/\delta_{\mathrm{MC}})}{(1-\gamma)^2}. \tag{20}$$

Per-visit scope. The Monte Carlo, fresh-rollout, and memory probabilities are per-visit budgets. For finitely many visits, a union bound adds the allocated failures. For an unbounded sequence, summable allocations are needed; a fixed nonzero budget at every visit does not supply a run-wide guarantee. Equation 19 does not account for biased numerical sampling or floating-point failures.

B.1 PROBABILITY MODEL AND ADAPTIVE-ACCEPTANCE CERTIFICATION

Let $\mathcal{H}$ contain the full pre-visit history. The policy, positive prior $\boldsymbol{\alpha}$, check set $\mathcal{T}$, and gate parameters are fixed conditional on $\mathcal{H}$. In the frequentist analysis, the answer vector $\boldsymbol{p}$ is also fixed. Imagine a potential stream of $N_{\max}$ conditionally i.i.d. answers; the algorithm reveals only the generated prefix. A check t in a sufficient-budget claim is chosen before the current data are seen, possibly using $\mathcal{H}$. It cannot be selected retrospectively from that visit's observations. Independent auxiliary seeds generate the posterior draws. For analysis, unused draws may be defined at unreached checks as well.

Proof of Proposition 4.1. Above, we construct an event $\mathcal{E}_{\mathrm{MC}}$ with probability at least $1 - \delta_{\mathrm{MC}}$, even conditional on the potential answer stream, on which every checked score is accurate to $\varepsilon_{\mathrm{MC}}$. On an accepted history in this event,

$$q_{\widehat{L}}(\widehat{\tau}) \ge \widehat{q}_{\widehat{L}}(\widehat{\tau}) - \varepsilon_{\mathrm{MC}} \ge \gamma. \tag{21}$$

This argument requires no calibration of $\boldsymbol{\alpha}$ for $\boldsymbol{p}$. □

Optional Bayesian interpretation. If $\boldsymbol{p} \mid \mathcal{H} \sim \mathrm{Dirichlet}(\boldsymbol{\alpha})$ and the answers are conditionally i.i.d. categorical draws given $\boldsymbol{p}$, stopped-data conjugacy remains valid because the gate uses only observed data and independent auxiliary seeds. Hence, for $M = \arg\max_j p_j$ and $A = \{\widehat{\tau} < \infty\}$, Proposition 4.1 also yields

$$\mathbb{P}(M \ne \widehat{L}, A \mid \mathcal{H}) \le (1-\gamma)\mathbb{P}(A \mid \mathcal{H}) + \delta_{\mathrm{MC}}, \tag{22}$$

Dividing by $\mathbb{P}(A \mid \mathcal{H}) > 0$ gives the corresponding conditional bound, capped at one. This optional reading is unused in the stopping and cost results, and EMA pseudo-counts alone do not supply the required generative model.

C Posterior Stopping: Formal Statements and Proofs

C.1 Pairwise Dirichlet separation

Lemma C.1 (Dirichlet tails and parameter ordering). *Let $\boldsymbol{U} \sim \mathrm{Dirichlet}(\boldsymbol{\beta})$ with positive shapes. If $a = \beta_i \geq b = \beta_j$, $d = a - b$, and $s = a + b$, then*

$$\mathbb{P}(U_j \geq U_i) \leq \exp\left\{-a \log \frac{2a}{s} - b \log \frac{2b}{s}\right\} \leq \exp\left(-\frac{d^2}{2s}\right). \tag{23}$$

Also, $\beta_j < \beta_i$ implies $\mathbb{P}(U_j > U_i) < 1/2$. Consequently, a coordinate with global winner probability at least $1/2$ must have a shape at least as large as every competitor's.

Remark C.2 (Separation relative to mass). A parameter lead becomes persuasive relative to the total pairwise mass. The result applies to fractional EMA pseudo-counts, not only integer counts.

Proof. Write $U_k = G_k / \sum_\ell G_\ell$ for independent $G_k \sim \mathrm{Gamma}(\beta_k, 1)$. Ordering is unchanged by the common denominator. For $0 \leq z < 1$, Markov's inequality yields

$$\mathbb{P}(G_j \geq G_i) \leq \mathbb{E}e^{z(G_j - G_i)} = (1-z)^{-b}(1+z)^{-a}. \tag{24}$$

Choose $z = (a-b)/(a+b)$ (zero if $a = b$) to obtain the first bound. For $v = d/s \in [0, 1)$,

$$a \log \frac{2a}{s} + b \log \frac{2b}{s} = \frac{s}{2}\big[(1+v)\log(1+v) + (1-v)\log(1-v)\big] \geq \frac{sv^2}{2}. \tag{25}$$

The bracket is zero at $v = 0$ and its derivative is $\log((1+v)/(1-v)) \geq 2v$, proving the second bound.

The ratio $U_j/(U_i + U_j)$ has law $\mathrm{Beta}(\beta_j, \beta_i)$. For its density f and $u > 1/2$,

$$\frac{f(u)}{f(1-u)} = \left(\frac{u}{1-u}\right)^{\beta_j - \beta_i} < 1 \qquad \text{if } \beta_j < \beta_i. \tag{26}$$

Integrating over $(1/2, 1)$ gives probability less than $1/2$. A global winner must beat coordinate i, so its winner probability is no larger than this pairwise probability. Equality of shapes gives pairwise probability $1/2$; reversing the shapes reverses the strict inequality. □

C.2 Crossing by a specified check and capped cost

Proof of Theorem 4.3. Condition on $\mathcal{H}$, so $\boldsymbol{p}$, $\boldsymbol{\alpha}$, and the bounds $s_j(t)$ are fixed. We prove the conclusions under the common condition in Equation 7. The theorem follows by setting $s_j(t) = \alpha_{j^\star} + \alpha_j + t$, and the same argument permits the EMA mass bound. Let $X_{i,j} = \mathbf{1}\{a_i = v^{(j^\star)}\} - \mathbf{1}\{a_i = v^{(j)}\}$. Then $X_{i,j} \in [-1, 1]$, $\mathbb{E}[X_{i,j} \mid \mathcal{H}] = \Delta_j$, and $\sum_{i=1}^t X_{i,j} = n_{j^\star}(t) - n_j(t)$. In particular, $g_j(t) = \alpha_{j^\star} - \alpha_j + t\Delta_j$.

Concentrate the current evidence. Hoeffding's inequality for range width two gives, for $u \geq 0$,

$$\mathbb{P}\big(n_{j^\star}(t) - n_j(t) - t\Delta_j \leq -u \mid \mathcal{H}\big) \leq e^{-u^2/(2t)}. \tag{27}$$

Taking $u = \sqrt{2t\ell_\delta}$ and union-bounding over $m - 1$ competitors produces an event $\mathcal{E}_t$ of probability at least $1 - \delta$ on which

$$n_{j^\star}(t) - n_j(t) \geq t\Delta_j - \sqrt{2t\ell_\delta} \quad \text{for all } j \neq j^\star. \tag{28}$$

Turn separation into a crossing. Put $\beta_k(t) = \alpha_k + n_k(t)$. The fresh-count bound and the definition of the conditional mean gap give, on $\mathcal{E}_t$,

$$\begin{aligned} \beta_{j^\star}(t) - \beta_j(t) &\geq g_j(t) - \sqrt{2t\ell_\delta} \geq \sqrt{2s_j(t)\ell_\gamma} > 0, \\ \beta_{j^\star}(t) + \beta_j(t) &\leq s_j(t). \end{aligned} \tag{29}$$

Lemma C.1 and a posterior union bound imply

$$1 - q_{j^\star}(t) \leq \sum_{j \neq j^\star} \Pi_t\{u_j \geq u_{j^\star}\} \leq (m-1)e^{-\ell_\gamma} = 1 - \overline{\gamma}. \tag{30}$$

On $\mathcal{E}_t \cap \mathcal{E}_{\mathrm{MC}}$, the algorithm has either already accepted or reaches t with $\widehat{q}_{j^\star}(t) \geq \overline{\gamma} - \varepsilon_{\mathrm{MC}} = \gamma + \varepsilon_{\mathrm{MC}}$. Thus it accepts by t with probability at least $1 - \delta - \delta_{\mathrm{MC}}$.

Exclude an earlier wrong crossing. Let $\mathcal{W}_t$ be the event that some wrong exact score crosses γ at a checked time through t. Appendix C.3 proves

$$\mathbb{P}(\mathcal{W}_t \mid \mathcal{H}) \le \rho_{\mathrm{wrong}}^{(\le t)}(\gamma). \tag{31}$$

On $\mathcal{E}_{\mathrm{MC}}$, a wrong implemented acceptance would imply $q_{\widehat{L}}(\widehat{\tau}) \ge \gamma$, hence $\mathcal{W}_t$. Therefore

$$\mathcal{E}_t \cap \mathcal{E}_{\mathrm{MC}} \cap \mathcal{W}_t^c \subseteq \{\widehat{\tau} \le t, \widehat{L} = j^\star\}. \tag{32}$$

A union bound gives the claimed correct-acceptance probability. The same Monte Carlo event serves both steps, so its failure is counted once.

Account for the cap. Pathwise,

$$\widehat{T} \le t + (N_{\max} - t)\mathbf{1}\{\widehat{T} > t\}, \qquad \{\widehat{T} > t\} \subseteq \{\widehat{\tau} > t\}. \tag{33}$$

Taking expectations gives Equation 10, including the endpoint $t = N_{\max}$. At fixed failure probabilities, its cap-dependent residual does not imply an uncapped expected hitting-time rate. □

Explicit sufficient budget. Equation 8 specializes the general certificate to $g_j(t) = \alpha_{j^\star} - \alpha_j + t\Delta_j$ and $s_j(t) = \alpha_{j^\star} + \alpha_j + t$. Define

$$a = \sqrt{2\ell_\delta} + \sqrt{2\ell_\gamma}, \qquad c_j = [\sqrt{2(\alpha_{j^\star} + \alpha_j)\ell_\gamma} - (\alpha_{j^\star} - \alpha_j)]_+.$$

The inequality $\sqrt{x+t} \le \sqrt{x} + \sqrt{t}$ shows that the certificate holds whenever $t\Delta_j \ge a\sqrt{t} + c_j$. Solving the quadratic in $\sqrt{t}$ gives

$$\widetilde{t}_{\mathrm{hit}} = \max\left\{N_{\min}, \left\lceil \max_{j \ne j^\star} \left(\frac{a + \sqrt{a^2 + 4\Delta_j c_j}}{2\Delta_j} \right)^2 \right\rceil \right\}, \tag{34}$$

$$t_{\mathrm{hit}} = \min\{t \in \mathcal{T} : t \ge \widetilde{t}_{\mathrm{hit}}\}, \tag{35}$$

with $t_{\mathrm{hit}} = \infty$ if no such check exists within the cap. For $\boldsymbol{\alpha} = a_0\mathbf{1}$, write $\Delta_{\min} = \min_{j \ne j^\star} \Delta_j$. A simpler sufficient threshold is

$$t \ge \max\left\{ \frac{16(\ell_\delta + \ell_\gamma)}{\Delta_{\min}^2}, \frac{4\sqrt{a_0 \ell_\gamma}}{\Delta_{\min}} \right\}. \tag{36}$$

Indeed, the right side of the crossing certificate is at most $2\sqrt{t(\ell_\delta + \ell_\gamma)} + 2\sqrt{a_0\ell_\gamma}$; each term is at most $t\Delta_{\min}/2$ under the displayed conditions. This proves Equation 11 with $a_0 = \alpha_0$. The first allowed check at or above the real-valued threshold is certified if it lies at or below the cap. The permitted check set already enforces $t \ge N_{\min}$. If no such check exists, this sufficient bound is inconclusive; it does not show that actual acceptance is impossible. For fixed prior and confidence parameters, the inverse-square term dominates as $\Delta_{\min} \downarrow 0$, while the second term captures the prior's concentration cost. A smaller sufficient budget need not order realized stopping times, and certifying a crossing still leaves the separate risk of an earlier wrong crossing. The expectation statement remains capped as in Equation 10.

C.3 Threshold-sensitive control of earlier wrong crossings

Fix a terminal check $u \in \mathcal{T}$ and a competitor $j \ne j^\star$. For fresh counts $k, \ell \ge 0$, define the pairwise confidence and its smallest attainable count lead at time s:

$$\Psi_j(k,\ell) = \mathbb{P}\{Z > 1/2\}, \qquad Z \sim \mathrm{Beta}(\alpha_j + k, \alpha_{j^\star} + \ell), \tag{37}$$

$$u_{j,\eta}(s) = \min_{\substack{k,\ell \in \mathbb{N}_0,\ k+\ell \le s \\ \Psi_j(k,\ell) \ge \eta}} (k - \ell), \qquad \eta \in (1/2, 1). \tag{38}$$

An empty minimum is $+\infty$; for $m = 2$ restrict to $k + \ell = s$. For $z > 0$, let

$$\varphi_j(z) = 1 - p_j - p_{j^\star} + p_j e^z + p_{j^\star} e^{-z}, \tag{39}$$

$$G_{j,\eta}^{(\le u)}(z) = \min_{s \in \mathcal{T}:\, s \le u} \{z u_{j,\eta}(s) - s \log \varphi_j(z)\}, \tag{40}$$

$$\rho_{\mathrm{wrong}}^{\mathrm{M},(\le u)}(\eta) = \min\left\{1, \sum_{j \ne j^\star} \inf_{z>0} \min\{1, e^{-G_{j,\eta}^{(\le u)}(z)}\} \right\}. \tag{41}$$

Proposition C.3 (Threshold-sensitive wrong crossing)**.** *For fixed $\boldsymbol{p}$ with unique mode and any realized positive $\boldsymbol{\alpha}$,*

$$\mathbb{P}\big(\exists s \in \mathcal{T},\ s \le u,\ \exists j \neq j^\star : q_j(s) \ge \eta \mid \mathcal{H}\big) \le \rho^{\mathrm{M},(\le u)}_{\mathrm{wrong}}(\eta). \tag{42}$$

The bound is nonincreasing in $\eta \in (1/2, 1)$.

Remark C.4 (Why retain the threshold). The result controls wrong crossings at all earlier checks, including paths that later favor the true mode. Its threshold dependence explains how stricter confidence can improve reliability. It is an explanatory fixed-$\boldsymbol{p}$ bound, since the current distribution is unknown online.

Proof. Dirichlet aggregation gives $U_j/(U_j + U_{j^\star}) \sim \mathrm{Beta}(\alpha_j + n_j(s), \alpha_{j^\star} + n_{j^\star}(s))$ under Π_s. A global win implies a pairwise win, hence

$$q_j(s) \le \Psi_j(n_j(s), n_{j^\star}(s)), \qquad q_j(s) \ge \eta \implies n_j(s) - n_{j^\star}(s) \ge u_{j,\eta}(s). \tag{43}$$

For fixed $z > 0$,

$$\mathcal{M}^{(j)}_s(z) = \frac{\exp\{z[n_j(s) - n_{j^\star}(s)]\}}{\varphi_j(z)^s} \tag{44}$$

is a nonnegative mean-one martingale conditional on $\mathcal{H}$: each count increment is $1, -1, 0$ with probabilities $p_j, p_{j^\star}, 1 - p_j - p_{j^\star}$, respectively. A crossing at a checked $s \le u$ forces

$$\log \mathcal{M}^{(j)}_s(z) \ge z u_{j,\eta}(s) - s \log \varphi_j(z) \ge G^{(\le u)}_{j,\eta}(z). \tag{45}$$

Ville's inequality bounds its probability by $\min\{1, e^{-G^{(\le u)}_{j,\eta}(z)}\}$. Optimize over $z > 0$, then union-bound over competitors. Increasing η shrinks each pairwise danger set, so $u_{j,\eta}(s)$ and $G^{(\le u)}_{j,\eta}(z)$ are nondecreasing. The resulting probability bound is therefore nonincreasing. □

Finite-state refinement. The pairwise crossing probability can also be evaluated exactly. Let $s_{j,0}(0,0) = 1$, with other initial states and negative-index states zero. Propagate the mass that has not previously crossed:

$$\begin{aligned}\widetilde{s}_{j,s}(k,\ell) &= p_j s_{j,s-1}(k-1,\ell) + p_{j^\star} s_{j,s-1}(k,\ell-1) \\ &\quad + (1 - p_j - p_{j^\star}) s_{j,s-1}(k,\ell), \end{aligned} \tag{46}$$

$$s_{j,s}(k,\ell) = \widetilde{s}_{j,s}(k,\ell)\mathbf{1}\{s \notin \mathcal{T} \text{ or } \Psi_j(k,\ell) < \eta\}. \tag{47}$$

Induction on s identifies this as the joint probability of the two counts and survival through s: the first equation enumerates the three next-answer outcomes, and the second removes newly crossed states. Therefore

$$R^{(\le u)}_{j,\eta} = 1 - \sum_{k,\ell \ge 0:\, k+\ell \le u} s_{j,u}(k,\ell), \tag{48}$$

$$\rho^{\mathrm{DP},(\le u)}_{\mathrm{wrong}}(\eta) = \min\left\{1, \sum_{j \neq j^\star} R^{(\le u)}_{j,\eta}\right\}, \tag{49}$$

$$\rho^{(\le u)}_{\mathrm{wrong}}(\eta) = \min\{\rho^{\mathrm{M},(\le u)}_{\mathrm{wrong}}(\eta), \rho^{\mathrm{DP},(\le u)}_{\mathrm{wrong}}(\eta)\}. \tag{50}$$

For $m = 2$, the recursion is exact for any wrong exact-score crossing in the potential stream; for $m > 2$, it is a pairwise relaxation followed by a union bound. It need not equal the probability of wrong *first acceptance*, because a correct crossing may have terminated the actual visit earlier. Nested danger events make the DP bound, and hence the minimum, nonincreasing in η. Setting $\eta = \gamma$ gives the bound used in the main theorem.

Corollary C.5 (A simple threshold-uniform bound)**.** *For $\eta \ge 1/2$, the wrong exact-score crossing probability through u is at most*

$$R_{\boldsymbol{\alpha}}(u) = \min\left\{1, \sum_{s \in \mathcal{T}:\, s \le u} \sum_{j \neq j^\star} \exp\left(-\frac{[\alpha_{j^\star} - \alpha_j + s\Delta_j]^2_+}{2s}\right)\right\}. \tag{51}$$

For $\eta > 1/2$, $\rho^{(\le u)}_{\mathrm{wrong}}(\eta) \le R_{\boldsymbol{\alpha}}(u)$. For $\boldsymbol{\alpha} = \alpha_0 \mathbf{1}$ and $K_u = |\{s \in \mathcal{T} : s \le u\}|$, this also gives

$$R_{\alpha_0 \mathbf{1}}(u) \le \min\{1, (m-1) K_u e^{-N_{\min}\Delta^2_{\min}/2}\}. \tag{52}$$

Remark C.6 (Simplicity versus information). This closed-form bound discards confidence beyond $1/2$ and can be vacuous when the threshold-sensitive bound is useful. It is a readable relaxation, not the error term used in the main theorem.

Proof. Lemma C.1 implies that $q_j(s) \geq 1/2$ requires $n_{j^\star}(s) - n_j(s) \leq -(\alpha_{j^\star} - \alpha_j)$. Apply Equation 27 with $u = [\alpha_{j^\star} - \alpha_j + s\Delta_j]_+$, using the trivial bound one when the untruncated expression is nonpositive, and union-bound over checks and competitors. The same parameter reversal is necessary for each pairwise danger event at $\eta > 1/2$. Its exact probability is thus bounded by the corresponding sum of Hoeffding terms. Summing over competitors and clipping at one proves $\rho^{\mathrm{DP}} \leq R_{\boldsymbol{\alpha}}$ and hence the comparison with ρ_{wrong}. For a symmetric prior the parameter gap vanishes. Every summand is then at most $e^{-N_{\min}\Delta_{\min}^2/2}$, since $s \geq N_{\min}$ and $\Delta_j \geq \Delta_{\min}$, and there are $(m-1)K_u$ summands. This proves Equation 52. □

Finite Monte Carlo wrong acceptance. On $\mathcal{E}_{\mathrm{MC}}$, wrong implemented acceptance by u implies an exact wrong crossing of γ. Hence its probability is at most $\min\{1, \rho_{\mathrm{wrong}}^{(\leq u)}(\gamma) + \delta_{\mathrm{MC}}\}$. A sharper numerical alternative uses $h = \gamma + \varepsilon_{\mathrm{MC}}$, $c_B = \lceil Bh \rceil$, $K_u = |\{s \in \mathcal{T} : s \leq u\}|$, and

$$\mathrm{BT}_B(h, \eta) = \sum_{v=c_B}^{B} \binom{B}{v} \eta^v (1-\eta)^{B-v}. \tag{53}$$

For $\eta \in (1/2, c_B/B)$, split on whether any wrong exact score crosses η. At each reached check where $q_j(s) < \eta$, its simulated winner count has law $\mathrm{Binomial}(B, q_j(s))$, whose upper tail is bounded by Equation 53. Union-bounding the remaining events gives

$$\mathbb{P}(\widehat{\tau} \leq u, \widehat{L} \neq j^\star \mid \mathcal{H}) \leq \min\{1, \rho_{\mathrm{wrong}}^{(\leq u)}(\eta) + (m-1)K_u \,\mathrm{BT}_B(h, \eta)\}. \tag{54}$$

At $\eta = \gamma$, one-sided Hoeffding and the definition of $\varepsilon_{\mathrm{MC}}$ bound the additive term by $(m-1)\delta_{\mathrm{MC}}/(2m)$. The main correct-acceptance theorem instead uses the shared two-sided event to count its failure only once.

D Fixed-Prefix EMA Memory Under Policy Drift

Memory can help only if historical evidence remains relevant to the current policy. We first bound its tracking error, then transfer that bound to the stopping certificate, and finally compare it with resetting the original symmetric prior. Throughout, r indexes visits to the same prompt, the answer coordinates are persistent, and $N_{\min}$, $\mu \in (0,1)$, and $\alpha_0 > 0$ are fixed across visits. Before visit s, the history-measurable policy induces $\boldsymbol{p}^{(s)}$; conditional on that history its first $N_{\min}$ answers are i.i.d. The prior is frozen during the visit and updated afterward by Equation 5.

D.1 Tracking accuracy with adaptive policies

The main argument needs a high-probability bound on the discrepancy between the historical and current answer distributions. For $\delta_{M,r} \in (0,1)$, the effective sample size and tracking radius are

$$N_{\mathrm{eff},r} = N_{\min} \frac{1+\mu}{1-\mu} \frac{1-\mu^{r-1}}{1+\mu^{r-1}}, \tag{55}$$

$$\varepsilon_r = \sqrt{\frac{\log(2m/\delta_{M,r})}{2N_{\mathrm{eff},r}}} + V\left(\frac{1}{1-\mu} - \frac{(r-1)\mu^{r-1}}{1-\mu^{r-1}}\right). \tag{56}$$

The first term is a sampling radius for the weighted historical frequencies; the second bounds their bias relative to the current policy. The proof below derives the effective sample size from the squared weights and the drift term from the weighted age, while allowing adaptive policy updates.

Tracking part of Theorem 4.6. **Identify the normalized historical average.** Unrolling the recurrence gives

$$\boldsymbol{\alpha}^{(r)} = \mu^{r-1}\alpha_0\mathbf{1} + (1-\mu)\sum_{s=1}^{r-1}\mu^{r-1-s}\boldsymbol{n}^{(s)}(N_{\min}), \tag{57}$$

$$\sum_{j=1}^{m}\alpha_j^{(r)} = m\mu^{r-1}\alpha_0 + N_{\min}(1-\mu^{r-1}). \tag{58}$$

Every prefix contains exactly $N_{\min}$ answers. Subtract the decayed initialization and divide by $M_r = N_{\min}(1-\mu^{r-1})$ to obtain

$$\overline{\boldsymbol{p}}^{(r)} = \sum_{s<r} w_{r,s}\widehat{\boldsymbol{p}}^{(s)}, \qquad \widehat{\boldsymbol{p}}^{(s)} = \frac{\boldsymbol{n}^{(s)}(N_{\min})}{N_{\min}}, \qquad w_{r,s} = \frac{(1-\mu)\mu^{r-1-s}}{1-\mu^{r-1}}. \tag{59}$$

The weights are deterministic for fixed r, μ, nonnegative, and sum to one. Thus $\overline{\boldsymbol{p}}^{(r)}$ is a probability vector derived from the single stored state $\boldsymbol{\alpha}^{(r)}$. Adding and subtracting the historical policies,

$$\overline{\boldsymbol{p}}^{(r)} - \boldsymbol{p}^{(r)} = \underbrace{\sum_{s<r} w_{r,s}(\widehat{\boldsymbol{p}}^{(s)} - \boldsymbol{p}^{(s)})}_{\text{historical sampling noise}} + \underbrace{\sum_{s<r} w_{r,s}(\boldsymbol{p}^{(s)} - \boldsymbol{p}^{(r)})}_{\text{policy-drift bias}}. \tag{60}$$

Control drift pathwise. The assumption $\|\boldsymbol{p}^{(s+1)} - \boldsymbol{p}^{(s)}\|_\infty \le V$ gives $\|\boldsymbol{p}^{(s)} - \boldsymbol{p}^{(r)}\|_\infty \le (r-s)V$ by telescoping. The second term in Equation 60 is therefore at most $V\sum_{s<r} w_{r,s}(r-s)$ in max norm. Put $n = r-1$. The finite geometric sum and its derivative give

$$\begin{aligned}\sum_{s<r} w_{r,s}(r-s) &= \frac{1-\mu}{1-\mu^n}\sum_{k=1}^{n} k\mu^{k-1} = \frac{1}{1-\mu} - \frac{n\mu^n}{1-\mu^n} \\ &= \frac{1 - r\mu^{r-1} + (r-1)\mu^r}{(1-\mu)(1-\mu^{r-1})}.\end{aligned} \tag{61}$$

This is the weighted mean age of the reused evidence.

Control noise without conditioning on future policies. Reveal all training randomness chronologically, including non-prefix rollouts and policy updates. Immediately before prefix draw i in visit s, $\boldsymbol{p}^{(s)}$ is measurable. For a fixed coordinate j,

$$Z_{s,i,j} = \frac{w_{r,s}}{N_{\min}}(\mathbf{1}\{a_i^{(s)} = v^{(j)}\} - p_j^{(s)}) \tag{62}$$

has conditional mean zero and conditional range width $w_{r,s}/N_{\min}$. Other revealed observations contribute zero to this sum. The conditional Hoeffding lemma and iterated expectations imply

$$\mathbb{E}\exp\left(\lambda\sum_{s<r}\sum_{i=1}^{N_{\min}} Z_{s,i,j}\right) \le \exp\left(\frac{\lambda^2}{8N_{\min}}\sum_{s<r} w_{r,s}^2\right). \tag{63}$$

Optimizing the Chernoff bound over $\lambda > 0$ and applying the same argument to the negative tail yields

$$\mathbb{P}\left(\left|\sum_{s<r} w_{r,s}(\widehat{p}_j^{(s)} - p_j^{(s)})\right| \ge z\right) \le 2\exp\left(-\frac{2N_{\min}z^2}{\sum_{s<r} w_{r,s}^2}\right). \tag{64}$$

The squared weights satisfy

$$\sum_{s<r} w_{r,s}^2 = \frac{(1-\mu)^2}{(1-\mu^{r-1})^2}\frac{1-\mu^{2(r-1)}}{1-\mu^2} = \frac{1-\mu}{1+\mu}\frac{1+\mu^{r-1}}{1-\mu^{r-1}}, \tag{65}$$

so $N_{\mathrm{eff},r} = N_{\min}/\sum_{s<r} w_{r,s}^2$ is precisely Equation 55. A union bound over the m coordinates bounds sampling noise by $\sqrt{\log(2m/\delta_{M,r})/(2N_{\mathrm{eff},r})}$ with probability at least $1-\delta_{M,r}$. Adding the pathwise drift bound proves

$$\mathbb{P}\Big(\|\overline{\boldsymbol{p}}^{(r)} - \boldsymbol{p}^{(r)}\|_\infty \le \varepsilon_r\Big) \ge 1 - \delta_{M,r}. \tag{66}$$

No independence across visits was used. Conditioning on the entire realized policy sequence would generally reveal information about past sampling noise, and is not the argument made here. □

A path-dependent refinement. The same decomposition and sampling event also give

$$D_r = \sum_{s<r} w_{r,s} \|\boldsymbol{p}^{(s)} - \boldsymbol{p}^{(r)}\|_\infty, \quad \mathbb{P}\left(\|\overline{\boldsymbol{p}}^{(r)} - \boldsymbol{p}^{(r)}\|_\infty \le \sqrt{\frac{\log(2m/\delta_{M,r})}{2N_{\mathrm{eff},r}}} + D_r \right) \ge 1 - \delta_{M,r}. \tag{67}$$

This may improve on the worst-case drift radius along a particular policy path. Both D_r and the current gaps depend on unknown distributions; neither is an automatically observable certificate. The bound is unconditional over history, even though its radius contains the history-dependent D_r.

Effective sample size and confidence across visits. At $r = 2$, $\overline{\boldsymbol{p}}^{(2)} = \boldsymbol{n}^{(1)}(N_{\min})/N_{\min}$ and $N_{\mathrm{eff},2} = N_{\min}$, although only $M_2 = (1-\mu)N_{\min}$ pseudo-counts are reused. Effective sample size describes averaging precision; historical mass describes influence on the posterior. As $r \to \infty$ at fixed μ and pointwise confidence,

$$M_r \longrightarrow N_{\min}, \qquad N_{\mathrm{eff},r} \longrightarrow N_{\min}\frac{1+\mu}{1-\mu},$$
$$\varepsilon_r \longrightarrow \sqrt{\frac{1-\mu}{2N_{\min}(1+\mu)} \log \frac{2m}{\delta_{M,r}}} + \frac{V}{1-\mu}. \tag{68}$$

Here the last limit holds with $\delta_{M,r}$ held constant. It describes the deterministic radius, not convergence of the random policies or tracking errors. For a deterministic horizon R, choose $\sum_{r=2}^{R} \delta_{M,r} \le \delta_M$; then a union bound gives simultaneous tracking probability at least $1 - \delta_M$. For infinitely many visits, $\delta_{M,r} = \delta_M/[r(r-1)]$ is summable, but its logarithm grows with r and the displayed fixed-confidence limit no longer applies.

The drift assumption concerns induced answer distributions. Ratio clipping and a soft KL penalty do not enforce it. If an actual consecutive-rollout-law trust region $D_{\mathrm{KL}}(P^{(s)} \| P^{(s+1)}) \le \kappa$ were enforced, data processing and Pinsker's inequality would imply $\|\boldsymbol{p}^{(s+1)} - \boldsymbol{p}^{(s)}\|_\infty \le \sqrt{\kappa/2}$. This is a sufficient mechanism, not a property asserted for the current optimizer. Here $P^{(s)}$ denotes a probability law, not a vector coordinate.

D.2 Transfer to the stopping certificate

Stopping part of Theorem 4.6. Define the tracking event

$$\mathcal{E}_{M,r} := \left\{ \|\overline{\boldsymbol{p}}^{(r)} - \boldsymbol{p}^{(r)}\|_\infty \le \varepsilon_r \right\}. \tag{69}$$

The tracking part of the theorem gives $\mathbb{P}(\mathcal{E}_{M,r}) \ge 1 - \delta_{M,r}$. We now work on this event and show explicitly how its coordinatewise control enters the general stopping condition in Equation 7.

Convert tracking accuracy into a remembered gap. For every coordinate k, the definition of $\mathcal{E}_{M,r}$ gives

$$-\varepsilon_r \le \overline{p}_k^{(r)} - p_k^{(r)} \le \varepsilon_r. \tag{70}$$

Fix a competitor $j \ne j_r^\star$. Adding and subtracting the current probabilities in the remembered gap yields

$$\begin{aligned} \overline{p}_{j_r^\star}^{(r)} - \overline{p}_j^{(r)} &= \underbrace{p_{j_r^\star}^{(r)} - p_j^{(r)}}_{\Delta_{r,j}} + \left(\overline{p}_{j_r^\star}^{(r)} - p_{j_r^\star}^{(r)}\right) - \left(\overline{p}_j^{(r)} - p_j^{(r)}\right) \\ &\ge \Delta_{r,j} - 2\varepsilon_r. \end{aligned} \tag{71}$$

The factor two is necessary because normalized memory may simultaneously underestimate the current mode by ε_r and overestimate the competitor by ε_r.

Recall from the unrolling of Equation 5 that, coordinatewise,

$$\alpha_k^{(r)} = \mu^{r-1}\alpha_0 + M_r \overline{p}_k^{(r)}. \tag{72}$$

The decayed symmetric initialization therefore cancels in a pairwise difference. Combining Equations 71 and 72 gives

$$\alpha_{j_r^\star}^{(r)} - \alpha_j^{(r)} = M_r(\overline{p}_{j_r^\star}^{(r)} - \overline{p}_j^{(r)}) \geq M_r(\Delta_{r,j} - 2\varepsilon_r). \tag{73}$$

For the pairwise prior mass, the same decomposition gives

$$\alpha_{j_r^\star}^{(r)} + \alpha_j^{(r)} = 2\mu^{r-1}\alpha_0 + M_r(\overline{p}_{j_r^\star}^{(r)} + \overline{p}_j^{(r)}) \leq 2\mu^{r-1}\alpha_0 + M_r. \tag{74}$$

The last inequality holds because $\overline{\boldsymbol{p}}^{(r)}$ is a probability vector, so the sum of any two of its coordinates is at most one.

Compute the conditional expected posterior gap. At time t in visit r, write $\beta_k^{(r)}(t) = \alpha_k^{(r)} + n_k^{(r)}(t)$. Conditional on the full pre-visit history $\mathcal{H}_r$, the stored prior $\boldsymbol{\alpha}^{(r)}$ and the current answer probabilities $\boldsymbol{p}^{(r)}$ are fixed. Only the current counts are random, and conditional i.i.d. sampling gives

$$\mathbb{E}[n_k^{(r)}(t) \mid \mathcal{H}_r] = tp_k^{(r)}, \qquad \mathbb{E}[n_{j_r^\star}^{(r)}(t) - n_j^{(r)}(t) \mid \mathcal{H}_r] = t\Delta_{r,j}. \tag{75}$$

Consequently, the expected parameter separation appearing in Equation 7 is

$$\begin{aligned} g_j(t) &:= \mathbb{E}[\beta_{j_r^\star}^{(r)}(t) - \beta_j^{(r)}(t) \mid \mathcal{H}_r] \\ &= \alpha_{j_r^\star}^{(r)} - \alpha_j^{(r)} + t\Delta_{r,j} \\ &= t\Delta_{r,j} + M_r(\overline{p}_{j_r^\star}^{(r)} - \overline{p}_j^{(r)}) \\ &\geq (t + M_r)\Delta_{r,j} - 2M_r\varepsilon_r. \end{aligned} \tag{76}$$

This calculation averages only the fresh counts. In particular, it does not treat the stored historical frequencies as unbiased observations from the current policy; their possible mismatch is retained explicitly through $2M_r\varepsilon_r$.

To control the pairwise posterior mass, observe pathwise that $n_{j_r^\star}^{(r)}(t) + n_j^{(r)}(t) \leq t$. Combining this fact with Equation 74 gives

$$\beta_{j_r^\star}^{(r)}(t) + \beta_j^{(r)}(t) \leq t + 2\mu^{r-1}\alpha_0 + M_r. \tag{77}$$

Thus a valid almost-sure mass bound is

$$s_j(t) := t + 2\mu^{r-1}\alpha_0 + M_r. \tag{78}$$

Equations 76 and 78 establish precisely the two substitutions in Equation 14.

Verify the shared stopping certificate. Suppose the pre-visit event $\mathcal{A}_{r,t}$ in Equation 15 holds. On $\mathcal{E}_{M,r} \cap \mathcal{A}_{r,t}$, subtracting $2M_r\varepsilon_r$ from both sides of that condition and using Equation 76 yields, for every $j \neq j_r^\star$,

$$\begin{aligned} g_j(t) &\geq \sqrt{2t\ell_\delta} + \sqrt{2(t + 2\mu^{r-1}\alpha_0 + M_r)\ell_\gamma} \\ &= \sqrt{2t\ell_\delta} + \sqrt{2s_j(t)\ell_\gamma}. \end{aligned} \tag{79}$$

This is exactly the general sufficient condition in Equation 7. Conditional on any pre-visit history in $\mathcal{E}_{M,r} \cap \mathcal{A}_{r,t}$, Theorem 4.3 therefore gives

$$\mathbb{P}(\widehat{\tau}_r > t \mid \mathcal{H}_r) \leq \delta + \delta_{\mathrm{MC}}. \tag{80}$$

Combine memory, fresh-sampling, and simulation failures. Both $\mathcal{E}_{M,r}$ and $\mathcal{A}_{r,t}$ are determined before the current visit. On their intersection, condition on the full history and apply Equation 80. Splitting according to the tracking event and integrating over the pre-visit history gives

$$\begin{aligned} \mathbb{P}(\mathcal{A}_{r,t}, \widehat{\tau}_r > t) &\leq \mathbb{P}(\mathcal{E}_{M,r}^c) + \mathbb{E}\big[\mathbf{1}_{\mathcal{E}_{M,r} \cap \mathcal{A}_{r,t}} \mathbb{P}(\widehat{\tau}_r > t \mid \mathcal{H}_r)\big] \\ &\leq \delta_{M,r} + \delta + \delta_{\mathrm{MC}}. \end{aligned} \tag{81}$$

If $\mathcal{A}_{r,t}$ holds almost surely, its intersection may be removed. If it holds with probability at least $1 - \delta_A$, add δ_A to the failure budget. For a general event of positive probability, the same bound conditionally on $\mathcal{A}_{r,t}$ is only

$$\mathbb{P}(\widehat{\tau}_r > t \mid \mathcal{A}_{r,t}) \leq \min\left\{1, \frac{\delta_{M,r} + \delta + \delta_{\mathrm{MC}}}{\mathbb{P}(\mathcal{A}_{r,t})}\right\}. \tag{82}$$

Mode-correct acceptance additionally needs the wrong-crossing term in Theorem 4.3; tracking alone does not remove that term. □

D.3 Comparison with the original cold-prior certificate

Proposition D.1 (Cold-prior certificate comparison)**.** *At a fixed check t, the robust EMA certificate is no harder than the certificate for the original cold prior $\alpha_0\mathbf{1}$ if, for every $j \neq j_r^\star$,*

$$\Delta_{r,j} - 2\varepsilon_r \geq \frac{\sqrt{2\ell_\gamma}(1 - 2\alpha_0/N_{\min})}{\sqrt{t + 2\mu^{r-1}\alpha_0 + M_r} + \sqrt{t + 2\alpha_0}}. \tag{83}$$

Let $t_{\mathrm{cert}}^{(0)} < \infty$ be the first permitted check satisfying the cold-prior certificate. If Equation 83 holds there, the first check satisfying the robust EMA certificate obeys

$$t_{\mathrm{cert}}^{(\mathrm{EMA})} \leq t_{\mathrm{cert}}^{(0)}. \tag{84}$$

The comparison uses the same current distribution, check set, and failure and confidence parameters.

Remark D.2 (Separation versus concentration). For $N_{\min} > 2\alpha_0$, the pairwise concentration bound increases and memory needs sufficient robust separation. For equality, a nonnegative robust gap suffices. For $N_{\min} < 2\alpha_0$, the concentration change itself is favorable, though sufficiently stale memory can still hurt. These are comparisons of sufficient certificates, not a stochastic ordering of realized stopping times.

Proof. Write $U_t = \sqrt{2t\ell_\delta}$. The cold and robust-memory certificates against competitor j are

$$t\Delta_{r,j} \geq U_t + \sqrt{2(t + 2\alpha_0)\ell_\gamma}, \tag{85}$$

$$t\Delta_{r,j} + M_r(\Delta_{r,j} - 2\varepsilon_r) \geq U_t + \sqrt{2(t + 2\mu^{r-1}\alpha_0 + M_r)\ell_\gamma}. \tag{86}$$

The EMA slack is at least the cold slack when its added separation offsets the difference of the square roots.

The difference rationalizes exactly as

$$\begin{aligned}&\sqrt{t + 2\mu^{r-1}\alpha_0 + M_r} - \sqrt{t + 2\alpha_0} \\ &\quad = \frac{M_r - 2\alpha_0(1 - \mu^{r-1})}{\sqrt{t + 2\mu^{r-1}\alpha_0 + M_r} + \sqrt{t + 2\alpha_0}} = \frac{M_r(1 - 2\alpha_0/N_{\min})}{\sqrt{t + 2\mu^{r-1}\alpha_0 + M_r} + \sqrt{t + 2\alpha_0}}.\end{aligned} \tag{87}$$

Multiply by $\sqrt{2\ell_\gamma}$ and divide by $M_r > 0$ to obtain Equation 83. If it holds at the first cold-certified check, the EMA condition also holds there, proving the budget comparison. Retaining the initialization decay is why the factor $1 - 2\alpha_0/N_{\min}$ appears. Comparing only with the decayed symmetric prior would be more conservative than this direct comparison with $\alpha_0\mathbf{1}$. □

E Additional Experimental Results

E.1 Additional Pass@k Results

The main results already report Mean@16 and Pass@16. To provide a more complete view of downstream performance, Table 5 additionally reports Pass@4 and Pass@8 for the same experiments.

Overall, these results are consistent with the main findings. On Qwen2.5-Math-1.5B, SCPO remains close to TTRL and OptPO on both AIME 2024 and MATH-500: for example, on MATH-500, Pass@8 is 87.5% for SCPO compared with 87.5% for TTRL and 88.0% for OptPO. On Llama-3.1-8B-Instruct, SCPO is also competitive and in several cases achieves the highest results. On AIME 2024, its Pass@4/8 values are 20.0/25.6, compared with 17.8/21.1 for TTRL and 16.7/24.4 for OptPO; on LiveCodeBench-v6, SCPO is also slightly better in Pass@4 and Pass@8 compared with the two baselines. These results suggest that reducing the number of training rollouts with SCPO does not degrade final model performance.

E.2 Results on Other Benchmarks

To further evaluate the efficiency of SCPO, we conduct experiments with Llama3.1-8B-Instruct on the *college_physics* subset of MMLU (Hendrycks et al., 2021a), a multiple-choice benchmark with four answer options per question. As shown in Table 6, SCPO reduces the number of generated tokens by 72.3% and 59.7%, relative to TTRL and OptPO, respectively.

Table 5: Pass@4 and Pass@8 performance of SCPO and baselines across different LLMs and benchmarks. R/P is the average number of rollouts per prompt.

Qwen2.5-Math-1.5B

Method	R/P ↓	Pass@4 ↑	Pass@8 ↑
AIME 2024			
Base	–	23.3	33.3
TTRL	64.0 ± 0.0	**28.9** ± 1.9	**34.4** ± 3.9
OptPO	48.8 ± 1.3	**28.9** ± 7.0	34.4 ± 2.0
SCPO	**41.2** ± 1.2	27.8 ± 5.1	33.3 ± 3.4
MATH-500			
Base	–	63.8	75.4
TTRL	64.0 ± 0.0	83.9 ± 1.8	87.5 ± 0.6
OptPO	38.2 ± 0.1	**84.5** ± 0.7	**88.0** ± 0.8
SCPO	**34.2** ± 0.1	83.7 ± 0.3	87.5 ± 0.5

Llama-3.1-8B-Instruct

Method	R/P ↓	Pass@4 ↑	Pass@8 ↑
AIME 2024			
Base	–	23.3	26.7
TTRL	64.0 ± 0.0	17.8 ± 1.9	21.1 ± 1.9
OptPO	29.7 ± 5.0	16.7 ± 3.4	24.4 ± 2.0
SCPO	**18.3** ± 0.8	**20.0** ± 5.8	**25.6** ± 8.4
MATH-500			
Base	–	70.6	77.4
TTRL	64.0 ± 0.0	71.8 ± 1.1	75.6 ± 1.6
OptPO	29.6 ± 1.3	**72.9** ± 0.2	76.1 ± 0.5
SCPO	**21.0** ± 0.6	**72.9** ± 1.3	**77.1** ± 1.2
LiveCodeBench-v6			
Base	–	22.3	24.6
TTRL	64.0 ± 0.0	**22.3** ± 1.2	22.9 ± 1.2
OptPO	49.2 ± 0.9	20.9 ± 0.9	22.7 ± 0.9
SCPO	**34.4** ± 0.5	**22.3** ± 1.2	**24.2** ± 0.3

Table 6: Performance comparison on MMLU College Physics with Llama3.1-8B-Instruct. R/P is the average number of training rollouts per prompt, and token savings are relative to TTRL.

Benchmark	Method	Mean@16 ↑	Pass@16 ↑	R/P ↓	Num. tokens ↓	Saving ↑
MMLU College Physics	*Base*	61.3	97.1	–	–	–
	TTRL	67.2	94.1	64.0	9.9M	0.0%
	OptPO	**68.5**	95.1	24.1	6.7M	32.3%
	SCPO	66.0	**96.1**	**16.4**	**2.7M**	**72.3%**

E.3 Additional Ablation Study

Impacts of Minimum Rollouts per Prompt. Table 7 studies the sensitivity of SCPO to the minimum rollout budget $N_{\min}$. A smaller $N_{\min}$ allows SCPO to stop earlier and therefore training gets fewer rollouts. Raising $N_{\min}$ from 8 to 32 increases the average number of rollouts per prompt from 13.4 to 34.3 and token usage from 59.8M to 123.5M. Mean@16 also gains from 67.4 to 72.8, whereas Pass@16 remains nearly unchanged. These results show a trade-off between token efficiency and Mean@16 performance: smaller $N_{\min}$ provide greater rollout savings and token savings, while larger values collect more evidence before stopping and achieve higher Mean@16.

Table 7: Minimum-rollout sensitivity on full MATH-500 with Qwen2.5-Math-1.5B.

$N_{\min}$	Mean@16 ↑	Pass@16 ↑	R/P ↓	Tokens ↓	Saving ↑
8	67.4	**90.8**	**13.4**	**59.8M**	**72.4%**
16	69.8	89.6	20.2	79.4M	63.4%
32	**72.8**	90.5	34.2	121.2M	44.1%

F Implementation Details

Our detailed experimental settings are given in Tables 8 and 9. In most of our experiments, except the ablation study on runtime, after first $N_{\min}$ rollouts, we sequentially generate one rollout then check the stopping condition after each time an additional rollout is generated: $\mathcal{T} = \{N_{\min}, N_{\min} + 1, \ldots, N_{\max}\}$. For majority voting on LiveCodeBench, we execute the code generated by models on the official test cases set and comparing outputs for the test cases to identify the consensus. For other configurations, we follow the defaults of TTRL (Zuo et al., 2025). All experiments are conducted on 2 NVIDIA H100 GPUs.

Hyperparameter	**Qwen2.5-Math-1.5B**	**Llama-3.1-8B-Instruct**
Policy optimization	GRPO with AdamW	GRPO with AdamW
Training epochs	5 (LiveCodeBench, MMLU), 10 (MATH-500), 80 (AIME24)	
Prompt batch size	32 (MATH-500), 8 (AIME24, LiveCodeBench, MMLU)	
Max voting rollouts per prompt	64	64
Max training rollouts per prompt	32	32
Rollout dtype	bfloat16	fp16
Optimizer state dtype	fp32	bfloat16
Training decoding	temperature 1.0, top-p 1.0	temperature 0.6, top-p 1.0
Maximum prompt length	1,024	1,024
Maximum response length	3,072	2,048
Actor learning rate	5×10^{-7}	1×10^{-6} (AIME24, MMLU), 2×10^{-6} (MATH-500), 1.5×10^{-6} (LiveCodeBench)
Learning-rate warmup	3% of training steps	3% of training steps
Learning-rate scheduler	cosine schedule	cosine schedule
Weight decay	0.01	0.01
KL-loss coefficient	0.001	0.001
Evaluation decoding	temperature 0.6, top-p 0.95	temperature 0.6, top-p 0.95
Evaluation samples	16	16

Table 8: Test-time policy optimization common hyperparameters

Hyperparameter	**Value**
Initial Dirichlet parameter α_0	1 (MATH-500, MMLU) 1×10^{-3} (AIME24, LiveCodeBench)
EMA coefficient μ	0.5
Minimum rollout budget $N_{\min}$	16 (Llama), 32 (Qwen)
Maximum rollout budget $N_{\max}$	64
Posterior confidence threshold $\gamma + \varepsilon_{\mathrm{MC}}$	0.6
Per-visit Monte Carlo failure probability δ_{MC}	0.05
Number of Monte Carlo posterior samples B	4096

Table 9: SCPO-specific hyperparameters