CERTIFIED EARLY REJECTION: PROMPT OPTIMIZERS CAN SKIP A THIRD OF THEIR VALIDATION ROWS

Anonymous authors
Paper under double-blind review

ABSTRACT

Prompt optimizers such as GEPA, MIPROv2 and SIMBA spend most of their task-model calls scoring candidate prompts on every row of a validation set: in our GEPA runs these full evaluations take 97.9% of the calls, and four in five of the candidates scored end below the incumbent and are never returned. Stopping such an evaluation early is only safe if it is *certified*: under every possible outcome of the unseen rows, the chance of discarding a candidate that would have won must stay below a stated level. For a candidate evaluated on n rows in a random order, the largest expected saving of *any* admissible adaptive randomised score-blind rule is the value of an explicit linear programme, which we bracket in rational arithmetic; the same programme with a different error event governs screening a pool of discards. For each fixed recorded candidate stream, fresh level-α audits against the retained incumbent preserve the best presented recorded count with probability at least $1 - \alpha$, without a multiplicity penalty. On 162 optimizer runs we executed (54 each of GEPA, SIMBA and MIPROv2; three task models, three benchmarks, $1{,}187$ fully evaluated candidates), certified rejection at a 1% error budget, computed exactly on the runs' per-row evaluation records, skips 35.9%, 24.5% and 12.5% of the audited validation rows. A calibrated finite-population likelihood-ratio rule skips 37.2% of GEPA's rows, 97% of the certified per-candidate ceiling on the candidates below their incumbent. Run live inside the optimizers against paired controls, the certificate skips 28.7%, 11.9% and 10.5% of GEPA's, MIPROv2's and SIMBA's audited rows; on GEPA this is about two points below the exact replay expectation on the paired control streams, and it cuts task-model calls by 25.5%. The same theory explains the opposite regime: on a pre-registered cohort of generated programs, at the evaluated nomination budget and measured cost model, no admissible certificate screening the nominated pool could break even in expectation, even with perfect pool selection, and we compute how large, and how clean, the pool would have had to be.

1 INTRODUCTION

Developers who run a prompt optimizer pay mostly for evaluation. A reflective optimizer proposes a candidate prompt, checks it on a small minibatch, and scores the survivors on every row of a fixed validation set; the prompt it returns is the one with the best count. In our runs of GEPA (Agrawal et al., 2025) these full evaluations are 97.9% of all task-model calls, and 431 of the 540 fully evaluated candidates end below the incumbent — the best fully evaluated prompt so far — and are never returned. Stopping such an evaluation after a weak prefix saves those calls, but it can also discard a candidate that would have won. Cutting such an evaluation short is an old idea — racing, successive halving, best-arm identification (Shi et al., 2024; Zhou et al., 2025; Zehle et al., 2025) — and these methods use different sampling models, error targets or empirical trade-offs. We ask the question a practitioner acts on when the discarded evaluation is never revisited: how much can be skipped while *guaranteeing*, under every possible outcome of the unseen rows, that a candidate which would have won is discarded with probability at most α?

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.

One theory, two regimes. The answer is a ceiling, and it is exact. A candidate's scores on the n validation rows are a fixed $0/1$ vector; the certificate sees them in a uniformly random order. At a candidate with T successes against an incumbent with t_0, the largest expected number of skipped rows over *all* admissible adaptive randomised rules is the value of a finite linear programme (Theorem 1), enclosed by rational primal and dual bounds with the reported gap. The same programme, with a different error event, governs a different task: screening a pool of generated candidates before verification, where the tolerance is counted in positives lost. The two regimes behave in opposite ways, and the contrast is the paper's practical message. Against an incumbent, the certificate is fed: we measure large savings on three optimizers. In a sparse screening pool, the certificate starves: on a pre-registered cohort, at the evaluated nomination budget $M = 383$, under the stated mean-continuation-cost model, no admissible certificate can break even in expectation, even with perfect pool selection (Theorem 4); this does not exclude a profitable individual audit realization.

Contributions.

- **Exact ceilings.** A linear programme whose value is the best expected saving of any admissible score-blind certificate, with rational dual certificates, for both the candidate-versus-incumbent and the screening error events (Section 3.2).
- **One budget per run.** On a fixed candidate stream, if each candidate is audited against the *retained* incumbent with fresh randomness, the returned count falls more than d below the best candidate presented with probability at most α — no union over candidates (Theorem 3), and at most α^k when k candidates attain the best count. Our replays nearly attain the bound.
- **Measured savings on three optimizers.** Exact computation on 162 optimizer runs we executed, 54 each of GEPA (Agrawal et al., 2025), SIMBA (Khattab et al., 2023) and MIPROv2 (Opsahl-Ong et al., 2024): 35.9%, 24.5% and 12.5% of audited rows at $\alpha = 1\%$, with a certified oracle ceiling to compare against (Section 5).
- **Live runs with paired controls.** Inside GEPA, MIPROv2 and SIMBA (33 control/certified pairs, protocol and addenda fixed before their runs) the certificate skips 28.7%, 11.9% and 10.5% of audited rows; on GEPA the realised saving is about two points below the exact replay expectation on the paired control streams (Section 5.2).
- **A rule comparison against the ceiling.** A design-point LP rule, curtailment, and six without-replacement likelihood-ratio rules; the best tested rule reaches 97% of the ceiling on GEPA, so little is left for any rule to gain there.
- **A negative result that is a theorem about the instance.** In the sparse regime, a predictor-free ceiling shows that, at the evaluated nomination budget $M = 383$, under the stated mean-continuation-cost model, no admissible certificate can break even in expectation, even with perfect pool selection, so cleaning the pool at that budget cannot help; this does not exclude a profitable individual audit realization, a larger nominated pool with sufficiently few positives, or a cheaper screening apparatus (Sections 3.4 and 5.3).

Practical takeaway. Put a certified rule in front of the full-evaluation stage: audit each candidate in a fresh random row order against the retained incumbent, discard it when the rule fires, and, if later proposals depend on earlier rejections, split one error budget across the audits (Section 3.3). On GEPA's recorded streams this skips about a third of the validation rows, within 2.6 points of what any admissible score-blind rule could skip; where the screen upstream is already strict, as in MIPROv2, there is less to skip.

2 Related work

Sequential inference without replacement. Exact finite-population sequential auditing is not new. Confidence sequences for sampling without replacement (Waudby-Smith & Ramdas, 2020) and anytime-valid inference from predicted test decisions (Holmes & Walker, 2026) provide valid stopping rules of the kind we compare; Kato & Nakagawa (2026) construct and exactly evaluate finite-population sequential boundaries by hypergeometric recursion, with scheduled decisions and a one-sided variant, and use monotonicity to control composite error probabilities, and we use

the same recursion and coupling principles. Our likelihood-ratio rules are standard members of this family, and calibrating their thresholds addresses the conservatism of Ville's inequality at a finite horizon. Anthony & Salehzadeh Nobari (2026) give exact finite-corpus coverage certificates, zero-miss minimax label-complexity results and fixed-sequence certification without a multiplicity penalty; our retained-incumbent result applies a related single-witness principle to a fixed, not necessarily nested, optimizer stream with history-dependent thresholds, and our LP contribution is an instancewise expected-savings envelope over admissible stopping rules. Hu et al. (2026), concurrently, formulate model-proposed selective release as a risk-constrained finite-population audit-design problem and evaluate its verification demand; we share the frozen-candidate setting and exact finite-population policy evaluation. Our claimed contributions are the score-blind envelope over *all* admissible rules at a named instance, optimization over the stated rule class, the retained-incumbent guarantee for optimizer streams, and the measurement on prompt optimizers against that envelope.

Cheaper prompt optimization. CAPO (Zehle et al., 2025) races candidates inside an evolutionary prompt optimizer with statistical tests; TRIPLE (Shi et al., 2024) casts prompt selection as best-arm identification; Zhou et al. (2025) evaluate adaptively to find good configurations with fewer calls; cost-aware transfer uses cheap task models during search (Oved et al., 2026). These methods use different sampling models, error targets or empirical trade-offs; ours bounds the loss of the returned count on a fixed candidate stream, for every complete score vector, and is measured against an exact envelope; in adaptive live search its effect is measured, not guaranteed. The optimizers we audit are GEPA (Agrawal et al., 2025), MIPROv2 (Opsahl-Ong et al., 2024) and DSPy's SIMBA (Khattab et al., 2023).

3 METHOD

3.1 SETTING

Candidate versus incumbent. A candidate has a fixed $0/1$ vector on n rows with T ones (its count). The incumbent's count b is known exactly because it was fully evaluated. A certificate audits the candidate's rows in a uniformly random order without replacement, and may *reject* before auditing any row or after any audited prefix; its saving is the number of unaudited rows. It is *admissible at level* α for the null $T \geq \tau$ if under every vector with at least τ ones it rejects with probability at most α. With $\tau = b$ ties are protected; $\tau = b + 1$ protects only candidates that would raise the returned count; $\tau = b + d + 1$ allows a loss of at most d rows (a tolerance of d/n in accuracy). Probability is over the audit permutation and any internal policy randomization, conditional on the complete fixed score vector.

Screening a pool. A predictor nominates a pool of M candidates for discard, holding T positives, with K positives retained. A certificate audits the pool and may discard the unaudited remainder; it is admissible if for every labelling the probability of losing more than $\varepsilon(K + T)$ positives is at most α. Unless stated otherwise $\alpha = 1/100$, and screening uses $\varepsilon = 1/20$.

Both are instances of one object: a stopping rule on the prefix counts of a uniformly permuted $0/1$ vector — m rows audited, s ones among them — with an objective under the true count and constraints under every count in a *bad set*. Such a rule is *score-blind*: it uses only the audited outcomes and its own randomness, not row identities or a predictor's scores (Section 3.4 shows that scores cannot raise the guaranteed saving). A rejected candidate is discarded and the incumbent is unchanged; only a fully evaluated candidate can replace it (Figure 1).

3.2 THE EXACT CEILING

Theorem 1 (Exact ceiling)**.** *Fix n, the true count T, a bad set $\mathcal{B}$ of counts, and α. The supremum of the expected number of skipped rows under T, over all adaptive randomised score-blind rules whose probability of a bad stop is at most α under every count in $\mathcal{B}$, is the value of an explicit finite linear programme in the per-prefix stopping masses. For rational inputs and rational nonnegative multipliers a backward recursion gives a rational upper bound, and every admissible rationally randomised threshold rule gives a rational lower bound; at every instance we report the two agree to within 10^{-6} rows.*

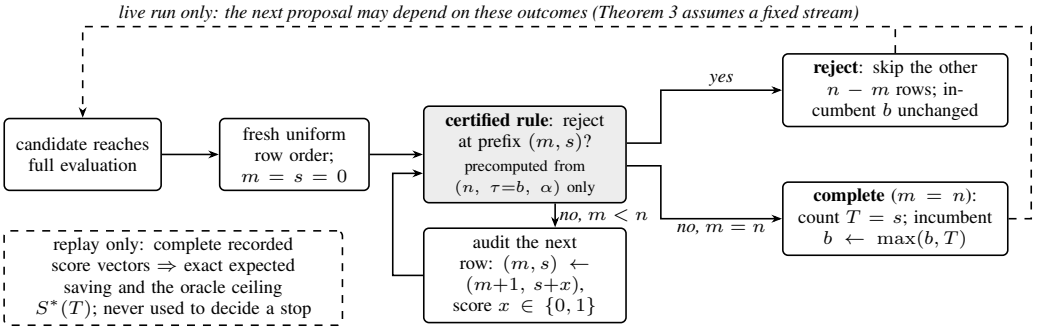

Figure 1: Where certified rejection sits in a prompt optimizer. Each candidate that reaches full evaluation is audited in a fresh uniformly random row order; after each row, a rule precomputed from the pool size n, the null boundary $\tau = b$ (ties protected) and α decides from the prefix (m, s) alone whether to reject. A rejection skips the remaining rows and leaves the incumbent unchanged; only a completed evaluation can replace it. The candidate's true count T is never used online: it enters only the replay calculations and the oracle ceiling S^*. Dashed: in a live run later proposals may depend on these outcomes, which is why Theorem 3 is stated for a fixed stream and the live runs are measured separately (Section 5.2).

For the candidate-versus-incumbent problem the bad set is $\{T' \geq \tau\}$ and every stop under it is bad. We call the value S^* the *score-blind ceiling*. It is an oracle envelope, not a deployable rule: it is computed at the candidate's true count T, which no rule knows during an audit, and it says how much *any* admissible rule could save on that candidate. The deployable rules below use only (n, τ, α) and the audited prefix. For threshold rules the boundary count $T' = \tau$ is least favourable, by a coupling of the two hypergeometric laws, so checking its constraint suffices; a single boundary-null multiplier in the dual recursion then gives an upper bound, which certifies the composite-null optimum when matched by a feasible rule (Appendix J gives the programme, in per-prefix averages that keep every variable in $[0, 1]$, and the recursion). Without any programme, for $0 \leq T < \tau \leq n$, $\max_k (n-k)\beta_k \leq S^* \leq \sum_k \beta_k$, where β_k is the power at T of the uniformly most powerful level-α test on the first k rows; for $T \geq \tau$, $S^* = \alpha n$. The bracket also covers pools too large for the programme.

Certified rules. Three families are admissible at every vector. *Curtailment* rejects when $s + (n - m) < \tau$, which no vector in the null can produce; it never errs, and for $0 \leq T < \tau \leq n$ its expected saving is $n - (n-\tau+1)(n+1)/(n-T+1)$ (all n rows when $\tau > n$). The *design-point LP rule* $R2$ is the programme's optimal rule, within its certified primal–dual gap, for a candidate at the design count $\max(0, \tau - \lfloor 0.05n \rfloor)$, built from (n, τ, α) alone and checked exactly: its rejection probability at $T' = \tau$ is a rational number at most α, and since it randomises state by state we also computed its rejection probability at every null count $T' \geq \tau$ for every table used; the maximum is always at the boundary (Appendix J). The *mixture likelihood-ratio* rules reject at the first t at which

$$L_t = \sum_{a<\tau} w(a)\, \frac{a^{\underline{k}}\,(n-a)^{\underline{t-k}}}{\tau^{\underline{k}}\,(n-\tau)^{\underline{t-k}}} \ \geq\ 1/\alpha,$$

with $t = m$ rows audited, $k = s$ ones seen, w a probability mass function on counts $a < \tau$ and $x^{\underline{j}} = x(x-1)\cdots(x-j+1)$, or when curtailment applies. We use w uniform on $\{0, \ldots, \tau-1\}$ (*uniform*), uniform on the $\max\{1, \lfloor 0.2n \rfloor\}$ counts below τ, clipped at 0 (*local20*), and a point mass at the design count of $R2$ clipped above at $\tau - 1$ (*point*), each also with an *attained-risk calibrated* constant threshold whose exact size at $T' = \tau$ is at most α.

Proposition 2 (Validity of the likelihood-ratio rules)**.** *L_t is a nonnegative supermartingale under $T' = \tau$ with $L_0 = 1$, so the $1/\alpha$ rules have size at most α there by Ville's inequality; each mixture term is decreasing in k, so the rejection region is a lower set $\{k \leq c_t\}$; and exposing a population with τ ones and one with $T' > \tau$ ones (containing them) through a common permutation gives $\mathbb{P}_{T'}(\textit{reject}) \leq \mathbb{P}_{\tau}(\textit{reject})$. The calibrated rules are valid by their computed size and the same coupling.*

The proof, with the support conventions at zero denominators (reject by curtailment, never reject when $k > \tau$ or $\tau = 0$), is in Appendix A.

3.3 One error budget per run

A run proposes candidates in some order and returns the one with the best count. Certified rejection changes which candidates are fully evaluated, so it can change the incumbent; the guarantee has to be about the incumbent the procedure actually retains. It protects the returned count on a fixed candidate stream; prompt identity, test accuracy and the later proposals of an adaptive search are outside it and are measured in Section 5.2.

Theorem 3 (Retained-incumbent guarantee)**.** *Fix candidate score vectors $x_1, \dots, x_m \in \{0,1\}^n$ with counts T_j and a fully evaluated seed count B_0, independently of all audit randomness, and fix an integer tolerance $d \geq 0$. Conditional on each preceding audit history, audit candidate j with a rule admissible at level α for $T \geq B_{j-1} + d + 1$, using a fresh independent audit permutation and fresh policy randomization. Fully evaluate every candidate not rejected, and update the retained count B_j by taking the maximum. With $T^* = \max(B_0, T_1, \dots, T_m)$, $\mathbb{P}(T^* - B_m > d) \leq \alpha$.*

Fix an index j^* attaining T^*. If the final count falls more than d short, then B_{j^*-1} was more than d below T^* when j^* arrived, so j^* satisfied its null, and it was rejected; the audit of j^* is fresh given the history, so this has probability at most α (Appendix B). The proof uses a single-witness principle related to the fixed-sequence certification of Anthony & Salehzadeh Nobari (2026); the specialization here is to history-dependent retained-incumbent thresholds on a fixed optimizer stream. The bounded event is a loss in the returned count, not the rejection of some good candidate, so no union over candidates is needed. The theorem needs fresh audit permutations, a common score scale, and a fixed candidate stream; if later proposals depend on earlier rejections, as in a live run, the single-witness argument no longer applies and the bound becomes α times the expected number of opportunities for a wrong rejection (Appendix B; a chain of ten such opportunities gives $1-0.99^{10} = 9.6\%$ at $\alpha = 1\%$). Splitting a total budget δ over the audits, $\sum_j \alpha_j \leq \delta$, restores a guarantee of δ for adaptive streams (Appendix B). The expected saving and the probability of loss under the retained incumbent are computed exactly by an outer dynamic programme over the retained count: with $r_j(b)$ the rejection probability of candidate j at incumbent b, $\pi_j(b) \mathrel{+}= \pi_{j-1}(b)\, r_j(b)$ and $\pi_j(\max(b, T_j)) \mathrel{+}= \pi_{j-1}(b)(1 - r_j(b))$.

3.4 Screening a sparse pool: a fixed-budget expected-cost limit

The same programme with the screening error event behaves in the opposite way. A certificate may exploit negative audit outcomes, but it may not assume that the unaudited pool is clean; it must exclude a hidden count among rows it has not audited. What limits it is the number of positives the error event can hide, not the quality of the ranking that produced the pool. Write $S^*(M, K, T)$ for the ceiling of Theorem 1 with the screening event and $S_0^*(M, h)$ for its explicit clean-pool value.

Theorem 4 (Predictor-free outer ceiling)**.** *Fix a cohort of N rows containing V positives, a feasible nominated pool of size M containing T positives, and $h(V) = \lfloor \varepsilon V/(1-\varepsilon) \rfloor + 1$. Every admissible score-blind certificate satisfies $\mathbb{E}[\textit{skip}] \leq S^*(M, V-T, T) \leq S_0^*\big(M, h(V)\big) < M\,\alpha^{1/h(V)} + h(V)$. For an arbitrary admissible procedure, including one choosing its audit order from fixed side information, the pointwise bound $\mathbb{E}[\textit{skip}] \leq S_0^*(M, h(V))$ still holds, provided admissibility is required under every labelling with that side information held fixed. When $M \leq N - V$, the outer envelope is attained by nominating a clean pool and using the optimal clean-pool rule, whose single audit look may be randomized between two adjacent sizes.*

The proof plants T fictitious positives at random rows (Appendix E). A score-aware audit order cannot raise the *guaranteed* saving either: relabelling by a random permutation turns it into a score-blind rule whose saving is the average over labellings (Theorem 6, stated in Appendix M.6 and proved in Appendix F); informative scores can help only on favourable realised labellings.

An event criterion. Protocols often require an event, $\mathbb{P}[\text{skip} \geq r] \geq p$, rather than an expectation. Any admissible rule can then be replaced by one that decides once, at row $M - r$, and on a clean pool, with $h = \lfloor \varepsilon K/(1-\varepsilon) \rfloor + 1 \leq M$ and $0 < p \leq 1$, possible and impossible are separated by $\binom{r}{h}/\binom{M}{h} \leq \alpha/p$, a boundary a practitioner can evaluate before auditing anything (Appendix M).

4 EXPERIMENTAL SETUP

Streams. We executed 54 runs of each optimizer and recorded every per-row evaluation outcome: three task models of 3–4B parameters (Granite-4.1-3B, Phi-4-mini, Qwen2.5-3B), three intent and question-type benchmarks (Banking77, MASSIVE-en, TREC-6), two data regimes (LOW: 10 training examples per class, 25 on TREC; FULL: the whole training pool) and three seeds, DSPy 3.3.1, task model at temperature 0. For GEPA the stream is the 540 candidates fully evaluated on $n = 300$ validation rows. For MIPROv2 it is the 324 programs fully evaluated on the same 300 rows. For SIMBA it is the 323 candidates of its final validation, on the training pool ($n = 150$–11,472). A candidate's vector is its recorded per-row outcome. Every replay uses the retained incumbent that Theorem 3 covers, for all 162 runs. Savings are exact expectations over the audit order, pooled over candidates, as a share of audited rows at batch size one; they carry no Monte Carlo error but are conditional on the recorded streams. The replay target is each candidate's count on the audited pool, the quantity the optimizer itself maximises, not held-out accuracy; the returned programs' test accuracy is measured in the live study, after the programs are frozen. A rule depends only on (n, τ) at fixed α and is cached; building one in exact arithmetic takes up to a minute at $n = 300$ and ten to fifteen minutes at $n = 600$–800. The LP rule $R2$ was solved for pools of at most 800 rows; above that SIMBA uses the uniform mixture rule (for supports above 400 counts, "uniform" means equal weights on the deterministic grid of Appendix A).

Live runs inside the optimizers. Replays fix the stream. In a live run a rejection changes what the optimizer does next — a rejected GEPA candidate never becomes a parent — so Theorem 3 does not cover the returned program, and the question is empirical. Under a protocol fixed before the runs, each optimizer ran twice per seed on the same machine: once with the certificate and once as a control, with the same blocked evaluator that never stops. There were three cells, one per model and dataset (LOW regime), seeds 1–5 for GEPA and 1–3 for MIPROv2 and SIMBA. Rows are dispatched in blocks of 16 in an audit order drawn independently of the optimizer. The certified arm runs $R2$ at $\alpha = 1\%$ per candidate against the retained incumbent (the error-spending variant of Section 3.3 was not run). All 300 incumbent tables are precertified in exact arithmetic (largest size 0.0099999999999956); pools of other sizes get a rule built and size-checked on demand. Each entrant is charged its nominal rows, so the optimizer's own budget is unchanged. A re-check from the raw ledgers confirms that each of the 126 rejections was permitted by a certified rule and that no control arm ever stopped.

5 RESULTS

5.1 SAVINGS ON RECORDED OPTIMIZER RUNS

Headline. Table 1 gives the savings. On GEPA and MIPROv2 the retained incumbent reproduces the savings of a fixed-incumbent replay to within 0.01 points, and $R2$ recovers 0.93 and 0.91 of the full-workload ceiling. In the seeded audit orders no candidate at least as good as its incumbent was rejected; the expected numbers of such rejections under the primary policies are 0.17 (GEPA), 0.39 (SIMBA) and 0.25 (MIPROv2). On SIMBA's small pools the retained incumbent gives 11.4%, the same as the recorded one, with all 131 rules it needs certified in exact arithmetic. The largest computed per-run probability of losing the best count is 0.64%–0.69%, below the 1% of Theorem 3. SIMBA pools $R2$ on its small pools (11.4%) with the uniform mixture rule on its large ones (25.2%); its runs differ, and the equal-run mean is 16.9%. GEPA's saving is large because its three-example screen lets through survivors a median 5.3 points below their incumbent; MIPROv2's minibatch screen already removes most of what a certificate could skip. The savings measure what each optimizer's stream offers to a certificate, not a ranking of optimizers.

How close to the ceiling. Table 2 compares the rules at the same error budget. The design-point rule is not the most efficient on these workloads: the calibrated local-mixture rule improves on $R2$ by 1.3, 0.4 and 0.4 points on GEPA, the $n \leq 800$ SIMBA subset and MIPROv2 (calibrated large-pool rules were not evaluated), and the best tested rule depends on the workload. On the 431 GEPA candidates below their fixed incumbent, the calibrated local-mixture rule attains 97.1% of the sum of the per-candidate ceilings; over the whole workload, including the αn any admissible rule may spend on the other candidates, the best admissible rule saves between 37.2% and 38.5%.

Table 1: Exact replay of the 54 recorded runs of each optimizer at $\alpha = 1\%$, ties protected, with the retained incumbent (exact outer dynamic programme) and batch size one. Rows audited: the nominal full-evaluation rows of all candidates. Savings: expected skipped rows pooled over candidates, as a share of those rows. Certified: $R2$ on pools of at most 800 rows, the uniform mixture rule above. Ceiling: the sum of per-candidate score-blind oracle ceilings at the original fixed incumbents, including the share αn any admissible rule may spend on candidates at or above the incumbent; it is not the value of one jointly optimal retained policy (–: SIMBA's large pools exceed the programme). $\mathbb{P}$(loss): per run, the exact probability that the returned count is below the best presented (mean / maximum over runs).

optimizer	candidates	rows audited	curtailment	certified	ceiling	$\mathbb{P}$(loss) mean / max
GEPA	540	162,000	14.0%	**35.9%**	38.5%	0.11% / 0.64%
SIMBA	323	1,499,026	2.8%	**24.5%**	–	0.25% / 0.69%
MIPROv2	324	97,200	4.9%	**12.5%**	13.8%	0.15% / 0.64%

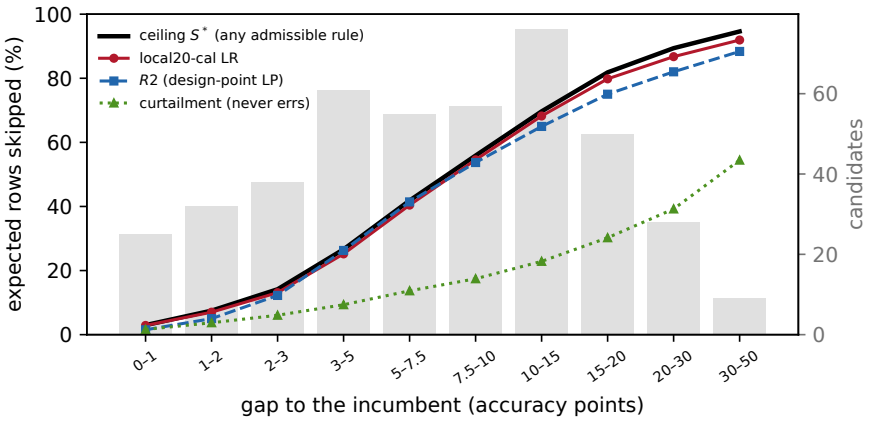

Figure 2: Expected share of a candidate's 300 validation rows skipped, against its gap $100(b - T)/n$ to the incumbent in accuracy points, for the 431 GEPA candidates below their incumbent in the 54 recorded runs ($\alpha = 1\%$, fixed incumbent). Lines: equal-candidate means within bins of unequal width; grey bars: candidates per bin. The ceiling is the score-blind oracle envelope over all admissible rules. All values are exact expectations over the audit order, so there are no Monte Carlo error bars; variation across runs is not shown.

Replayed under its own retained incumbent the same rule still skips 37.2% of GEPA's rows, with a per-run probability of loss of at most 0.68%. The fixed-incumbent envelope also bounds the retained savings: a retained incumbent never exceeds the original fully evaluated one, and a lower null boundary only tightens admissibility. The calibrated threshold is set by exact null crossing probabilities, not by savings; the decision to evaluate and emphasise it was made after the earlier comparisons, on the same streams, so the efficiency ranking is exploratory on these streams and not a held-out estimate for new runs, while each frozen rule's validity is a separate property. $R2$ keeps its distinct role: it is certified within the reported primal–dual gap of the programme's optimum at its design count, a statement about one count, not about a workload's mixture of counts.

Where the saving comes from. Figure 2 resolves the saving by the candidate's gap to its incumbent. In the observed 0–1-point bin the mean ceiling is about 3%, leaving little saving on average. Across the plotted 3–10-point bins $R2$ is within about 2.2 points of the ceiling; beyond ten points the likelihood-ratio rule, which adapts to the observed gap, gains on it. Among the tested rules the best one varies with the gap, and the ceiling tells a practitioner how much any further rule engineering could buy: on GEPA, at most 1.3 points.

Table 2: Rules at the same per-candidate error budget $\alpha = 1\%$ on the recorded runs of Table 1 (SIMBA split by pool size), fixed incumbent, batch size one. Entries: expected share (%) of audited rows skipped, pooled over candidates. *cal*: threshold calibrated to exact size $\leq \alpha$. Ceiling: sum of per-candidate score-blind ceilings, including αn on candidates at or above the incumbent. Bold: the largest computed saving in the row, an exploratory ranking on these streams, not a significance statement; –: not computed.

workload	$R2$	uniform	local20	point	uniform-cal	local20-cal	point-cal	ceiling
GEPA	35.9	34.8	36.2	32.9	35.8	**37.2**	33.7	38.5
SIMBA, $n > 800$	–	25.2	**26.7**	20.3	–	–	–	–
MIPROv2	12.5	11.3	12.3	11.8	11.9	**12.9**	12.3	13.8
SIMBA, $n \leq 800$	11.4	10.1	11.3	11.7	10.6	11.8	**12.0**	13.1

Table 3: Live runs, certified against control on the same seed and machine, in three LOW-regime cells (Qwen–Banking77, Phi–MASSIVE, Granite–TREC); GEPA five seeds per cell, MIPROv2 and SIMBA three. The certified arm runs $R2$ at $\alpha = 1\%$ per candidate in blocks of 16 rows. Rejected: rejected / eligible candidates. Saved: skipped / nominal audited rows of the certified arms, pooled. ΔS, Δtest: accuracy of the returned program on the selection and test splits, certified minus control, in points; equal-weight mean over cells of seed means, with 95% percentile intervals from 10,000 stratified bootstrap resamples of seeds within cells. Every pair is listed in Appendix C.

optimizer	pairs	rejected	saved	ΔS [95% CI]	Δtest [95% CI]
GEPA	15	92/150	**28.7%**	−1.6 [−4.8, +1.2]	−1.2 [−4.8, +2.0]
MIPROv2	9	18/45	**11.9%**	−2.7 [−5.3, +1.2]	−2.5 [−7.3, +2.1]
SIMBA	9	16/63	**10.5%**	−3.1 [−8.6, +2.8]	−4.6 [−11.0, +4.5]

Allowing a small loss. With the tolerance of Theorem 3 the saving grows quickly (Table 6, Appendix D): on GEPA the programme's rule designed at the incumbent count, with null $T \geq b+d+1$, skips 28.1, 39.4, 41.2 and 43.4% at $d = 0, 1, 3, 6$ rows and $\alpha = 1\%$ ($d = 6$ rows is two accuracy points; at $d = 0$ it may reject ties, unlike Table 1). The largest computed per-run probabilities of a loss above d nearly attain the bound of Theorem 3: they range from 0.9994α to α at 1% and from 0.992α to α at 5%. For GEPA and MIPROv2 a controlled ablation holds the LP design count fixed and varies tie protection and the error budget separately (Table 7, Appendix D): protecting only count improvements ($H_0 : T \geq b+1$ instead of $T \geq b$) adds 1.1–1.5 points, and raising α from 1% to 5% adds 4.8–7.5; both are distinct from the tolerance d.

Robustness. Evaluating rows in synchronous batches of q and acting on a rule's first crossing at the end of its batch costs at most $q-1$ rows per rejected candidate relative to the same rule at batch size one, and the batched ceiling is within $q-1$ rows of the row-level one (Appendix D, Proposition on batched audits): $R2$ on GEPA skips 35.9, 35.1, 34.2 and 32.5% at $q = 1, 8, 16, 32$ (the evaluator used 16 threads), SIMBA's large pools are unaffected (25.2 to 25.1%), and MIPROv2 goes from 12.5 to 10.7%. Across the 18 model–dataset–regime cells GEPA's saving ranges from 17 to 58% (median 36%) and never falls below 32% when a model or dataset is left out; SIMBA ranges from 2 to 34%, MIPROv2 from 3 to 28% (Appendix D).

5.2 Live runs inside the optimizers

On GEPA the certificate skipped 28.7% of the audited rows. The exact replay expectation of the same rule, with blocks of 16, on the paired control arms' streams — what it would have saved without intervening — is 30.7%; on seeds 1–3 it is 30.6%, against 31.9% realised. On the recorded runs of the same cells from the replay study it is 38.8%. A certified arm's own stream differs from both, since rejections change later proposals, so these are comparisons across streams rather than a prediction for the stream it faced: the saving is a property of the stream, which varies between runs of one configuration, and the live saving lies close to the replay expectation on comparable streams. On GEPA, total task-model calls fell by 25.5%, full-validation time by 32% and total optimisation time by 11%; reflection calls are unchanged. Across the three optimizers the returned programs'

mean test accuracy moved by -1.2 to -4.6 points: all three point estimates are negative, and every 95% interval contains zero but also losses of several points. We make no non-inferiority claim; the live runs are not powered for one, and the row, call and time reductions describe the executed runs, not a guarantee for future searches.

5.3 THE SPARSE REGIME, MEASURED

The measured cohorts. On a frozen 256-candidate program-generation study the deployed predictor nominated 122 candidates holding $T = 7$ of 40 valid programs. Its four-look rule could not fire for any labelling: firing needs $K \geq 76$ retained positives, a count that is constant for pools from 128 to 640 rows and does not involve T (Proposition 5, Appendix M). Refusing was correct: discarding the pool uncertified would have lost 17.5% of the valid programs against a 5% tolerance. A pre-registered 896-candidate cohort then made firing possible ($K = 99$) and its predictor separated (8.9% of nominated candidates valid against 19.3% retained); nothing certified, and rejecting the pool uncertified would have lost 25.6% of the valid programs. At the evaluated nomination budget of 383 rows and under the measured marginal cost model, no admissible audit could offset the screening overhead in expectation, even with perfect pool selection: the exact ceiling at $(383, 99, 34)$ is 26.7 rows (3.0% of the 896-root cohort) against a measured 32.3% break-even, and a perfect predictor at that budget caps at 24.2% (the comparison prices each avoided continuation at the measured mean continuation cost). This is a fixed-budget impossibility: cleaning the pool at this budget cannot help, while a larger nominated pool with sufficiently few positives or a cheaper screening apparatus could. By instancewise linear programmes at $V = 133$ valid programs (feasible pool sizes $M \leq N - V + T$), score-blind audits under the stated cost model require pools of at least 512, 593 and 684 rows at $T = 0, 1, 2$; no feasible pool with $T = 3$ reaches expected break-even. These dirty-pool thresholds are not pointwise bounds for arbitrary score-aware audit orders. The executed controller used the four-look rule inherited from the parent protocol instead of the pre-registered every-row rule, a deviation we disclose; replayed exactly on the recorded order, the registered rule certifies correctly and would have avoided 72.9 GPU-seconds, 1.8% of the recorded cost, before screening overhead.

6 DISCUSSION

What it saves and what it does not. The saving is in audited rows, the unit of each optimizer's own budget and of the theorem. The saving in time is smaller. In our setup the reflection model is a remote API and takes 76% of GEPA's wall clock, against 21.6% for full evaluations; the live runs measured an 11% reduction in total optimisation time. With an expensive task model the share grows. The guarantee concerns the *best candidate presented* to the certified stage. In off-trajectory rescoring, MIPROv2 never fully evaluated 56.3–59.4% of programs exceeding their first-appearance incumbent by margins of at least 3–15 rows; including programs that tie the incumbent ($k = 0$), the fraction is 62.6%. Its returned program is a median of one row below the best generated program in this rescoring diagnostic. These are diagnostic comparisons, not wrong-rejection rates of the certified stage.

6.1 LIMITATIONS

The replay savings are computed exactly on the executed runs' evaluation records; the live runs measure the certificate inside the search on three cells. A certificate running inside the optimizer changes which candidates are proposed later: in 21–27 of the 54 GEPA runs the returned prompt descends from a candidate the certified rules would have removed, and the guarantee covers the returned count on a given candidate stream, not the search. Scores are binary outcomes of three classification benchmarks with 3–4B task models; non-binary metrics and larger models are not measured. Re-scoring a prompt at temperature 0 flips a median 3 of 300 rows, so, like the optimizers' own selection, the guarantee is about the recorded outcomes. The LP rule was solved only up to 800 rows within our budget. The sparse-regime impossibility fixes the nomination budget and the screening-cost model, and its evidence is two cohorts of one task source.

AI USE STATEMENT

Large-language-model assistants were used in this project as productivity tools, in the same way as other research software, with every step directed and verified by the authors. They helped develop and refine the theoretical models, the conceptual framing and the hypotheses; assisted in exploring and refining the formulation of mathematical statements and in writing and checking proofs; discussing the experimental design and the analysis; writing and testing the replay, linear-programming and likelihood-ratio code under the authors' supervision; summarising and interpreting results; and drafting, editing and translating text, which the authors reviewed. They were also used for literature search, figure preparation and reference formatting. Human verification was applied throughout, and no claim of the paper rests on the assistants alone: all proofs were verified by the authors; the ceilings are enclosed by rational primal and dual bounds and the size of every implemented integer boundary on pools of at most 800 rows is certified in exact rational arithmetic by a separate path-counting program; every reported number is regenerated from frozen records by the released scripts; and every reference was checked against its source. The assistants supplied no experimental measurements or reference labels; the candidate prompts, programs and task-model predictions under study are, by design, generated by the language models being studied, and the pipeline scores them against the benchmarks' labels. The authors take full responsibility for the content of this paper, including any text, claims or code produced with the aid of these tools.

ETHICS STATEMENT

This work uses public classification benchmarks and recorded runs of openly available prompt optimizers and task models; it involves no human subjects and no personal data. Its aim is to reduce the evaluation compute of prompt optimization. On a fixed candidate stream the procedure preserves, up to the stated error budget, the best recorded count among the candidates presented for full evaluation; in adaptive live search its effect on the returned program is measured, not guaranteed.

REPRODUCIBILITY STATEMENT

The anonymous supplementary material opens with `README.md`; `HEADLINE-MANIFEST.json` maps every headline number to its generating script and output, and `code/`, `live/` and `journals/` hold the scripts, the live study and the recorded streams. It contains the recorded optimizer journals, the replay and dynamic-programme scripts, the frozen integer boundaries and rule-specific certification records, distinguishing exact rational size certificates from Ville-based analytic guarantees and numerical implementation checks, the rational primal and dual bounds, and a machine-readable manifest giving, for every headline number, its candidate universe, rule, null, α, incumbent convention, batching model, denominator and certification status, with per-file SHA-256. For the live runs it contains the protocol and its addenda (hashed before the runs), the certified evaluators, every run's ledger, returned program and test record, the incident log, and the analysis and raw-ledger re-check scripts. Exact ceilings carry rational primal and dual bounds; the admissibility of $R2$ is checked in exact arithmetic at every null count; the implemented integer boundaries of the likelihood-ratio rules on pools of at most 800 rows are certified in exact rational arithmetic at every state used (Appendix A). The GEPA replay and the MIPROv2/SIMBA replay plan were stated before the journals were read. Three later changes are disclosed: an earlier implementation of the likelihood-ratio baselines omitted the curtailment region, which made them weaker and led us to a claim, now withdrawn, that $R2$ matched or exceeded them on every workload; the large-pool SIMBA rule was changed from a union-bound Hoeffding–Serfling race (18.8% pooled) to the uniform mixture rule (24.5%), both reported; and the calibrated variants were added after these comparisons.

REFERENCES

L. A. Agrawal, S. Tan, D. Soylu, N. Ziems, R. Khare, K. Opsahl-Ong, A. Singhvi, H. Shandilya, M. J. Ryan, M. Jiang, C. Potts, K. Sen, A. G. Dimakis, I. Stoica, D. Klein, M. Zaharia, and O. Khattab. GEPA: Reflective prompt evolution can outperform reinforcement learning. arXiv:2507.19457, 2025.

M. Anthony and K. Salehzadeh Nobari. Finite-sample coverage audits for high-recall candidate generation: Certification and learning-theoretic design. arXiv:2607.21480, 2026.

C. Holmes and S. G. Walker. Predicting fixed-sample test decisions enables anytime-valid inference. arXiv:2602.13872, 2026.

Y. Hu, Z. Zhou, Y. Zhu, and K. Liang. Exact finite-population sequential auditing for selective release from model-proposed candidate sets. *Mathematics*, 2026. doi:10.3390/math14183427.

M. Kato and K. Nakagawa. Sequential audit sampling for finite populations with exact and simulation-based guarantee. arXiv:2604.06116, 2026.

O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, et al. DSPy: Compiling declarative language model calls into self-improving pipelines. arXiv:2310.03714, 2023.

K. Opsahl-Ong, M. J. Ryan, J. Purtell, D. Broman, C. Potts, M. Zaharia, and O. Khattab. Optimizing instructions and demonstrations for multi-stage language model programs. arXiv:2406.11695, 2024.

T. Oved, R. Pony, O. Naparstek, and U. Barzelay. Optimize cheap, deploy strong: Cost-aware cross-tier transfer for evolutionary optimization. arXiv:2608.10694, 2026.

C. Shi, K. Yang, Z. Chen, J. Li, J. Yang, and C. Shen. Efficient prompt optimization through the lens of best arm identification. arXiv:2402.09723, 2024.

I. Waudby-Smith and A. Ramdas. Confidence sequences for sampling without replacement. arXiv:2006.04347, 2020.

T. Zehle, M. Schlager, T. Heiß, and M. Feurer. CAPO: Cost-aware prompt optimization. arXiv:2504.16005, 2025.

J. P. Zhou, C. K. Belardi, R. Wu, T. Zhang, C. P. Gomes, W. Sun, and K. Q. Weinberger. On speeding up language model evaluation. In *International Conference on Learning Representations (ICLR)*, 2025.

A VALIDITY OF THE LIKELIHOOD-RATIO RULES

Fix n and the null boundary $\tau \geq 1$. Under a population with T ones, the probability of a particular audited prefix with k ones in t rows is $P_T(x_{1:t}) = T^{\underline{k}}(n-T)^{\underline{t-k}}/n^{\underline{t}}$. Let $Q = \sum_{a<\tau} w(a)P_a$ with nonnegative weights summing to one, fixed before the audit (they may depend on n and on the retained incumbent, never on the audited prefix), and $L_t = Q(x_{1:t})/P_\tau(x_{1:t})$ on histories of positive P_τ-probability.

Supermartingale. For such a history h, $\mathbb{E}_\tau[L_{t+1} \mid h] = P_\tau(h)^{-1}\sum_{x:\,P_\tau(hx)>0} Q(hx) \leq Q(h)/P_\tau(h) = L_t$. The inequality, not equality, matters: alternatives put mass on prefixes the boundary null cannot produce, and indeed $L_n = 0$ almost surely under $T = \tau$ since every complete sequence has τ ones. Ville's inequality for the nonnegative supermartingale L gives $\mathbb{P}_\tau(\max_{t\leq n} L_t \geq 1/\alpha) \leq \alpha$.

Support conventions. When $k + (n-t) < \tau$ the null cannot be reached and the rule rejects (curtailment; impossible under every $T' \geq \tau$). When $k > \tau$ both numerator and denominator vanish and the rule never rejects. For $\tau = 0$ the alternative set is empty and the rule never rejects. These conventions are part of the rule; the reported savings use them.

Lower region. For a component $a < \tau$, write $r_{a,t}(k) = a^{\underline{k}}(n-a)^{\underline{t-k}}/(\tau^{\underline{k}}(n-\tau)^{\underline{t-k}})$. Where the adjacent terms are positive, $r_{a,t}(k+1)/r_{a,t}(k) = \frac{a-k}{\tau-k}\cdot\frac{n-\tau-t+k+1}{n-a-t+k+1} \leq 1$, and zero terms preserve the order. Mixtures preserve it, so rejection at time t is a set $\{k \leq c_t\}$.

Composite null. Take two labelled populations whose sets of ones satisfy $S_\tau \subseteq S_{T'}$, $|S_\tau| = \tau$, $|S_{T'}| = T'$, and expose both through the same uniformly random permutation. Each marginal is a sample without replacement and $k_t^{(\tau)} \le k_t^{(T')}$ for every t. If the higher path crosses a lower boundary, the lower path has crossed it by the same time. Hence $\mathbb{P}_{T'}(\text{reject}) \le \mathbb{P}_\tau(\text{reject}) \le \alpha$ for all $T' \ge \tau$.

Calibration and thinning. A calibrated rule replaces $1/\alpha$ by the smallest constant h on a bisection grid whose exact size $\mathbb{P}_\tau(\text{reject})$, computed by the forward recursion over the audit order, is at most α; validity then rests on that computed size and the coupling. The grid mixtures use the deterministic support $\text{unique}(\text{round}(\text{linspace}(lo, \tau - 1, m)))$, $m = \min(\tau - lo, 400)$, with $lo = 0$ (uniform) or $lo = \max\{0, \tau - \max\{1, \lfloor 0.2n \rfloor\}\}$ (local20), rounding to the nearest integer, endpoints included, duplicates removed and equal weights; the point rule puts its mass on $a_\star = \min\{\tau - 1, \max(0, \tau - \lfloor 0.05n \rfloor)\}$. Every pool we replay has $n \ge 150$, so both clips are inactive there. Thinning changes efficiency, not validity.

Exact certification of the implemented boundaries. Thresholds are computed in double precision, so we certify the implemented integer boundaries $\{c_t\}$ themselves. For pools of at most 800 rows, at all 221 (n, τ) states used — the fixed incumbents of every candidate and every count reachable as a retained incumbent in the GEPA and MIPROv2 runs — the null crossing probability $\mathbb{P}_\tau(\text{reject}) = \sum_{(m,s)} N(m,s)\, \tau^{\underline{s}} (n-\tau)^{\underline{m-s}} / n^{\underline{m}}$, with $N(m,s)$ the number of non-rejected paths reaching the first-rejection state, is evaluated in exact rational arithmetic. Every one of the six rules is at most α at every state (maxima 0.00848 uniform, 0.00853 local20, 0.00924 point; for the three calibrated rules the exact rational checks leave a minimum slack of approximately $2.5 \cdot 10^{-8}$ below α); no boundary needed tightening, and the lower-set property and the curtailment region were verified. On the SIMBA pools above 800 rows Ville's inequality proves admissibility of the ideal, exact-arithmetic uniform rule; the implemented integer boundaries were computed and their sizes checked numerically in double precision (maximum 0.0089), which is a numerical check, not a rigorous size certificate for the implemented boundaries. The admissibility of $R2$ is certified exactly at every null count (Appendix J).

B THE RETAINED-INCUMBENT GUARANTEE

Proof of Theorem 3. Let the vectors $T_1, \dots, T_m$ and the seed count B_0 be fixed, and let each audit use a fresh permutation independent of everything before it. Put $T^* = \max\{B_0, T_1, \dots, T_m\}$ and fix an index j^* attaining it, or stop if B_0 does. The retained counts are nondecreasing and $B_{\text{final}} \ge T_{j^*}$ whenever j^* is accepted. If $B_{\text{final}} < T^* - d$, then $B_{j^*-1} \le B_{\text{final}} < T_{j^*} - d$, so $T_{j^*} \ge B_{j^*-1} + d + 1$: candidate j^* satisfied the null of its own test, and it was rejected. Conditional on the history before j^*, which fixes B_{j^*-1}, the audit of j^* is fresh and admissible for that null, so the conditional probability of rejection is at most α. Averaging over histories, $\mathbb{P}(T^* - B_{\text{final}} > d) \le \alpha$. □

Corollary (several best candidates). Under the assumptions of Theorem 3, if $k \ge 1$ fixed indices $j_1 < \cdots < j_k$ attain T^* and $B_0 < T^* - d$, then $\mathbb{P}(T^* - B_m > d) \le \alpha^k$. *Proof.* Let A_i be the event that j_i satisfied its null $T_{j_i} \ge B_{j_i-1} + d + 1$ and was rejected. As in the proof above, the loss event is contained in $A_1 \cap \cdots \cap A_k$. Given the history before j_i, the audit of j_i is fresh and admissible, so $\mathbb{P}(A_i \mid \mathcal{F}_{j_i-1}) \le \alpha$, and $A_1, \dots, A_{i-1}$ are $\mathcal{F}_{j_i-1}$-measurable. Hence $\mathbb{P}(A_1 \cap \cdots \cap A_i) = \mathbb{E}[\mathbf{1}_{A_1 \cap \cdots \cap A_{i-1}} \mathbb{P}(A_i \mid \mathcal{F}_{j_i-1})] \le \alpha\, \mathbb{P}(A_1 \cap \cdots \cap A_{i-1})$, and induction gives α^k. No independence of the rejection events is assumed. □ With $k = 1$ this is Theorem 3, which the replays nearly attain, so the bound is not uniformly improvable; the corollary quantifies how much safer streams with repeated best counts are.

The event bounded is the final-quality loss, not the event that some good candidate is rejected; the expected number of wrong rejections (reported in the tables) can exceed α. The rules may differ across candidates (for example by pool size) as long as each is admissible given the history.

Adaptive proposals. If T_j depends on earlier audit decisions, j^* is no longer fixed independently of those decisions. Per-candidate validity still gives $\mathbb{P}(T^* - B_{\text{final}} > d) \le \mathbb{E}\big[\sum_j \alpha\, \mathbf{1}\{T_j \ge$

$B_{j-1}+d+1\}\big]$, α times the expected number of opportunities for a wrong rejection; a chain of ten such opportunities at $\alpha = 1\%$ reaches $1-0.99^{10} = 9.56\%$.

Corollary (adaptive error spending). Let the candidates be proposed adaptively, x_j being $\mathcal{F}_{j-1}$-measurable (it may depend on every earlier audit decision and on the optimizer's own randomness), let the levels α_j be $\mathcal{F}_{j-1}$-measurable with $\sum_j \alpha_j \le \delta$ almost surely, and audit candidate j with a rule admissible at level α_j for $T \ge B_{j-1}+d+1$, with fresh audit randomness. Then $\mathbb{P}(T^* - B_m > d) \le \delta$ for $T^* = \max(B_0, T_1, \dots, T_m)$, the best count presented. *Proof.* Let A_j be the event that j satisfied its null and was rejected. If $T^* - B_m > d$, any index attaining T^* is such a j, as in the proof of Theorem 3, so the loss event lies in $\bigcup_j A_j$. Since x_j, B_{j-1} and α_j are fixed by $\mathcal{F}_{j-1}$ and the audit is fresh, $\mathbb{P}(A_j \mid \mathcal{F}_{j-1}) \le \alpha_j$, hence $\mathbb{P}(\bigcup_j A_j) \le \mathbb{E}\sum_j \alpha_j \le \delta$. □ A fixed split $\alpha_j = \delta/m$ or a summable schedule such as $\alpha_j = \delta 2^{-j}$ gives a guarantee that no longer requires a fixed stream, at the price of a smaller per-candidate level.

Exact computation. For a fixed stream, the law of the retained count after candidate j is supported on $\{B_0, T_1, \dots, T_j\}$. With $r_j(b)$ and $a_j(b)$ the rejection probability and expected skipped rows of candidate j's rule at incumbent b (each from the forward recursion over the audit order), $\pi_j(b)$ += $\pi_{j-1}(b)\, r_j(b)$ and $\pi_j(\max(b, T_j))$ += $\pi_{j-1}(b)(1 - r_j(b))$; the expected saving is $\sum_j \sum_b \pi_{j-1}(b)\, a_j(b)$ and $\mathbb{P}(\text{loss} > d) = \sum_{b<T^*-d} \pi_m(b)$. Rules are rebuilt for every reachable b. The per-run maximum of $\mathbb{P}(\text{loss} > d)$ lies between 0.9994α and α at $\alpha = 1\%$, and between 0.992α and α at $\alpha = 5\%$, in every R_{tie} setting of Table 6: the bound is attained up to rounding.

C LIVE RUNS: PROTOCOL AND EVERY PAIR

The protocol (`LIVE-PROTOCOL-A.md` in the supplement) was written and hashed before the first live run. Addendum 1 extended GEPA uniformly to seeds 4–5 in every cell before any of those runs. Addendum 2 added MIPROv2 and SIMBA with the same cells, seeds 1–3, arms and evaluator. Every optimizer used its configuration from the replay study (Section 4): DSPy 3.3.1, the same reflector at temperature 1.0, the task model served at temperature 0, and the same pools and selection split.

GEPA. The certificate runs in the full-validation stage ($n = 300$). A rejected candidate enters neither the Pareto nor the merge set.

MIPROv2. The certificate runs in the full-evaluation stage ($n = 300$). A rejected program cannot become the best and is not given a score.

SIMBA. The certificate runs in the final validation on the training pool ($n = 150$, 594 or 770). Candidates are audited in SIMBA's order against the retained incumbent, and a rule is built and checked for exact size before its first use.

In every arm the audit order comes from a random generator separate from the optimizer's. Test accuracy is scored natively, after all programs of a block were frozen and hashed. Runs that failed before producing a program (host faults, or an exception at the end of MIPROv2's certified arm) were repeated once from scratch under the identical configuration; no run was repeated because of its outcome, and the incident log is in the supplement.

D FULL TABLES

Result specification. Every number in Section 5 carries: candidate universe (GEPA: the 540 fully evaluated candidates; a secondary stream adds 33 proposals GEPA discarded on its minibatch, scored off-trajectory, where $R2$ skips 36.7% of the enlarged audit), rule, null boundary, α, incumbent convention (retained unless stated), batching model (synchronous batches of b rows, a crossing at an internal prefix is kept; not measured wall-clock), and denominator (rows the full evaluation would audit). For SIMBA the retained incumbent is computed exactly on every pool: with the uniform mixture rule above 800 rows, and with $R2$ rebuilt and size-certified in exact arithmetic at each of the 131 reachable incumbent states on the pools of at most 800 rows (5.4% of SIMBA's rows; 11.36% saved, against 11.4% with the recorded incumbent).

Table 4: All 33 live pairs (15 GEPA, 9 MIPROv2, 9 SIMBA); each row is one observed pair, not an average. Rejected: rejected / eligible candidates in the certified arm. Rows: audited validation rows executed by the certified arm out of its nominal rows (the control arm executes all of its own). S and test: accuracy (%) of the returned program on the selection and test splits, control / certified. B77, MAS, TREC: Banking77, MASSIVE-en, TREC-6.

optimizer	cell	seed	rejected	rows executed / nominal	S control	S certified	test control	test certified
GEPA	Granite TREC	1	4/10	2,248/3,000	72.7	78.0	78.0	86.6
GEPA	Granite TREC	2	7/10	1,412/3,000	72.7	76.0	81.0	84.2
GEPA	Granite TREC	3	7/10	1,972/3,000	75.3	58.0	86.4	66.6
GEPA	Granite TREC	4	5/10	2,556/3,000	75.0	76.7	84.4	82.6
GEPA	Granite TREC	5	6/10	1,872/3,000	73.0	66.7	78.4	75.4
GEPA	Phi MAS	1	8/10	1,736/3,000	60.7	57.0	58.6	53.0
GEPA	Phi MAS	2	8/10	2,072/3,000	62.3	47.3	54.7	41.4
GEPA	Phi MAS	3	5/10	1,948/3,000	58.7	59.0	53.8	52.9
GEPA	Phi MAS	4	8/10	1,912/3,000	56.3	56.0	50.4	49.1
GEPA	Phi MAS	5	7/10	2,100/3,000	57.0	57.7	50.2	56.3
GEPA	Qwen B77	1	6/10	2,448/3,000	51.0	53.3	49.1	51.1
GEPA	Qwen B77	2	6/10	2,160/3,000	48.3	48.7	49.0	48.8
GEPA	Qwen B77	3	6/10	2,384/3,000	52.3	55.3	49.9	53.5
GEPA	Qwen B77	4	5/10	2,476/3,000	50.3	51.0	47.4	51.7
GEPA	Qwen B77	5	4/10	2,792/3,000	46.7	47.7	49.9	50.0
MIPROv2	Granite TREC	1	2/5	1,444/1,500	72.0	59.3	84.2	59.8
MIPROv2	Granite TREC	2	1/5	1,280/1,500	79.0	66.7	84.0	75.6
MIPROv2	Granite TREC	3	3/5	1,272/1,500	67.0	73.0	76.2	81.2
MIPROv2	Phi MAS	1	1/5	1,392/1,500	65.0	64.0	57.2	58.7
MIPROv2	Phi MAS	2	3/5	1,336/1,500	63.3	62.0	56.6	58.8
MIPROv2	Phi MAS	3	3/5	1,176/1,500	61.3	60.3	56.1	56.8
MIPROv2	Qwen B77	1	2/5	1,412/1,500	60.3	57.3	55.7	56.0
MIPROv2	Qwen B77	2	2/5	1,204/1,500	56.7	58.0	52.4	55.0
MIPROv2	Qwen B77	3	1/5	1,376/1,500	61.3	61.0	59.7	58.0
SIMBA	Granite TREC	1	0/7	1,050/1,050	62.0	74.7	65.6	82.6
SIMBA	Granite TREC	2	1/7	1,044/1,050	73.7	51.0	82.6	52.4
SIMBA	Granite TREC	3	3/7	1,000/1,050	65.0	57.3	77.4	59.2
SIMBA	Phi MAS	1	2/7	3,514/4,158	58.0	60.3	55.3	56.2
SIMBA	Phi MAS	2	2/7	3,674/4,158	61.0	58.0	57.6	55.0
SIMBA	Phi MAS	3	2/7	4,026/4,158	58.7	53.7	56.0	52.4
SIMBA	Qwen B77	1	1/7	4,876/5,390	59.3	55.3	57.9	53.5
SIMBA	Qwen B77	2	4/7	4,102/5,390	60.0	60.3	56.4	56.7
SIMBA	Qwen B77	3	1/7	5,164/5,390	59.3	58.0	54.8	54.5

Proposition (batched audits). Let S_q^* be the ceiling of Theorem 1 when rows are evaluated in synchronous batches of q and a certificate may reject only at time zero or at a batch boundary (rows $0, q, 2q, \dots$). Then $\max\{0, S_1^* - (q-1)\} \le S_q^* \le S_1^*$. *Proof.* Batch-restricted certificates are row-level certificates, which gives the upper bound. For the lower bound, take any admissible row-level certificate π, run it on the rows as they arrive inside each batch, latch its first rejection at prefix m, and reject at the end of the current batch, row $q\lceil m/q \rceil$; if that row is n or beyond, complete the evaluation without rejecting. For incumbent rejection, every rejection of the delayed rule is a rejection of π. For screening, the delayed discarded set is contained in the original one, so the delayed rule's bad event is contained in that of π under every labelling. The delayed rule is therefore admissible; its pathwise loss of saving is at most $(q-1)\mathbf{1}\{\pi \text{ rejects}\}$, so its expected saving is at least that of π minus $(q-1)\mathbb{P}(\pi \text{ rejects}) \ge$ that of π minus $(q-1)$, and taking the supremum over π proves the claim. □ The batch columns of Table 5 use exactly this latch construction. At $n = 300$ and $q = 16$ the bound allows a loss of at most 15 rows per rejected candidate; the measured loss of $R2$ on GEPA is 1.7 points of the audited rows (35.9% at $q = 1$, 34.2% at $q = 16$).

Per configuration. Pooling the three seeds of each of the 18 model–dataset–regime cells, $R2$ skips 17–58% of GEPA's audited rows (median 36%; leaving out one model or one dataset, 32–38%); SIMBA, with the uniform mixture rule on large pools, ranges from 2 to 34% (median 15.4%; leave-one-out 20–27%), and MIPROv2 from 3 to 28% (median 9.5%; leave-one-out 8–15.5%). Leave-one-out values are influence analyses, not held-out validation.

SIMBA large-pool rule. Pooled over all SIMBA candidates, with $R2$ on pools of at most 800 rows and, above, the local20 mixture, the uniform mixture, the point rule or the Hoeffding–Serfling race

Table 5: Batching model: expected share (%) of audited rows skipped when rows are evaluated in synchronous batches of q and a rule's first crossing is acted on at the end of its batch ($\alpha = 1\%$, ties protected; the recorded runs of Table 1, SIMBA split by pool size). Exact expected row savings, not measured wall-clock time.

rule and workload	$q = 1$	$q = 8$	$q = 16$	$q = 32$
$R2$, GEPA	35.9	35.1	34.2	32.5
local20-cal, GEPA	37.2	36.3	35.3	33.6
uniform (retained), SIMBA $n > 800$	25.22	25.20	25.18	25.13
local20-cal, MIPROv2	12.9	12.3	11.7	10.8
$R2$, MIPROv2	12.5	12.0	11.6	10.7
local20-cal, SIMBA $n \leq 800$	11.8	11.5	11.2	10.6

Table 6: Tolerance d rows on the 54 recorded runs each of GEPA and MIPROv2 ($n = 300$), retained incumbent, exact outer dynamic programme, null $T \geq b + d + 1$. Each entry: pooled saving (%) / mean per-run $\mathbb{P}(\text{loss} > d)$ (%). R_{tie} is designed at the incumbent count b; at $d = 0$ it may reject ties. $R0d$ is curtailment at $b + d + 1$ and never errs.

	$d = 0$	$d = 1$	$d = 3$	$d = 6$
GEPA, R_{tie}, $\alpha = 1\%$	28.1 / 0.50	39.4 / 0.27	41.2 / 0.10	43.4 / 0.07
GEPA, R_{tie}, $\alpha = 5\%$	33.3 / 2.9	45.5 / 1.9	49.9 / 0.9	52.4 / 0.5
GEPA, $R0d$	15.0 / 0	15.6 / 0	16.7 / 0	18.3 / 0
MIPROv2, R_{tie}, $\alpha = 5\%$	12.8 / 3.4	18.8 / 2.3	23.5 / 1.3	27.4 / 0.5
MIPROv2, $R0d$	5.4 / 0	5.9 / 0	6.8 / 0	8.1 / 0

first reported: 25.9, 24.5, 19.8 and 18.8%. All four are recomputed exactly on the same candidates and denominator. The uniform mixture remains the primary policy, chosen because it has no width parameter.

Matched ceiling comparison. For a candidate at or above its incumbent the ceiling is exactly αn only for candidates eligible for rejection (already fully evaluated seed programs are not in the candidate universes and contribute neither saving nor ceiling; reject before auditing with probability α; no admissible rule can do better since $\mathbb{E}_T[\text{skip}] \leq n\,\mathbb{P}_T(\text{reject}) \leq \alpha n$), which adds 0.20, 0.46 and 0.46 points to the GEPA, MIPROv2 and small-pool SIMBA envelopes (38.34 → 38.54, 13.31 → 13.78, 12.63 → 13.09). On GEPA's 431 candidates below the incumbent, the ratio of each rule's saving to the sum of their ceilings is 0.971 (local20-cal), 0.944 (local20), 0.936 ($R2$), 0.933 (uniform-cal), 0.907 (uniform), 0.878 (point-cal) and 0.859 (point). Rows these rules skip by rejecting candidates at or above the incumbent add 0.01–0.06 points to the pooled savings and are excluded from these ratios. No rule exceeds the ceiling at any candidate.

Screening diagnostic. For MIPROv2, every generated program was re-scored off-trajectory on all 300 rows at temperature 0 (these calls are outside every denominator). A program is *useful at margin* k if that count exceeds the best full-evaluation count at its first appearance by at least k rows; of the useful programs, 62.6, 59.4, 57.4, 56.9 and 56.3% never received a full evaluation at $k = 0, 3, 6, 9, 15$. Margins $k \geq 6$ exceed the observed median of three changed rows on re-scoring; this is not a uniform or high-probability noise bound. At $k = 0$ the diagnostic includes programs that tie the incumbent. This measures MIPROv2's own screen, not the certified stage.

Expected and realised. Savings are exact expectations over the audit order; a single seeded order per candidate gives realised values (36.2% for $R2$ on GEPA, 12.0% on MIPROv2). They are different quantities, not competing estimates.

Certification numerics. The ceiling is bracketed by exact rational primal and dual values; over the 431 GEPA candidates below their incumbent the widest bracket is $5.6 \cdot 10^{-7}$ rows. Building one LP rule in exact arithmetic takes about half a minute to a minute at $n = 300$ and ten to fifteen minutes at $n = 600$–800; rules depend only on (n, τ) and are cached.

Table 7: Controlled ablation: LP rules with the design count held at $b-\lfloor 0.05n \rfloor$, rebuilt for each null ($T \geq b$ protects ties, $T \geq b+1$ protects count improvements) and each level α. Entries: expected share (%) of audited rows skipped, pooled; fixed incumbent, batch size one; GEPA and MIPROv2 on their 54 recorded runs, SIMBA on its TREC LOW pools.

	$T \geq b$, 1%	$T \geq b+1$, 1%	$T \geq b$, 5%	$T \geq b+1$, 5%
GEPA	35.9	37.3	43.3	44.8
MIPROv2	12.5	13.6	17.3	18.7
SIMBA, TREC LOW pools	5.5	6.7	8.3	9.9

E Proof of Theorem 4

Throughout, a certificate audits rows in a uniformly random order, stops at a random time, and is *admissible at* K if for every number T' of pool positives the probability that it stops with more than $\lfloor \varepsilon(K+T') \rfloor$ unseen positives is at most α.

Lemma (transfer). For all M, K, T, $S^*(M,K,T) \leq S^*(M,K+T,0)$, and the same holds for the event ceiling at every target.

Proof. Let π be admissible at K. Build π' on the same pool: draw a uniformly random T-subset F of the M audit positions, independent of everything, and run π on the combined labels $x_i'' = \max(x_i, \mathbf{1}[i \in F])$. On a clean pool the combined process is exactly the label process of a uniform order with T positives, so π' has, under $T'=0$, the law of skip that π has under T. For admissibility at $K+T$, fix a real labelling with T' positives at a uniform set R of positions. The combined set $F \cup R$ has size $T'' \in [\max(T,T'), T+T']$ and, conditional on T'', is a uniform T''-subset by exchangeability. A stop of π' with real count s is bad at $K+T$ iff $T'-s > \lfloor \varepsilon(K+T+T') \rfloor$; at that stop the unseen combined positives number at least $T'-s$, while $\lfloor \varepsilon(K+T'') \rfloor \leq \lfloor \varepsilon(K+T+T') \rfloor$. So a bad stop of π' at $K+T$ is a bad stop of π at K under T'' combined positives, and $\Pr_{T'}(\pi' \text{ bad}) \leq \sum_{t''} \Pr(T''=t'' \mid T') \Pr_{t''}(\pi \text{ bad}) \leq \alpha$. □

Proof of Theorem 4. The first inequality is the transfer lemma with $K+T=V$. For the second, at $T'=h=h(V)$ a stop is bad iff no positive has been seen, because $\lfloor \varepsilon(V+h) \rfloor = h-1$. So the admissibility constraint at $T'=h$ involves only clean states, as does the objective at $T=0$. Dropping every other constraint relaxes the linear programme to a two-constraint clean-path programme, whose optimum is the two-row randomisation $M - m_{\min} + q$ with $q = (\alpha - F(m_{\min}))/(F(m_{\min}-1) - F(m_{\min}))$. The rule attaining it, "stop at row $m_{\min}-1$ with probability q if clean, else at row $m_{\min}$ if clean", is admissible at every T': for $T' < h$ no stop is bad, and for $T' \geq h$ its error probability $qF_{T'}(m_{\min}-1) + (1-q)F_{T'}(m_{\min})$ is nonincreasing in T' by a coupling and equals α at $T'=h$. So the relaxation is exact. The closed form follows from $m_{\min}/M \geq (1-(h-1)/M)(1-\alpha^{1/h})$ and $q<1$. The cohort-1 numbers in Table 9 are exact linear-programme values certified by rational duals; the transfer lemma was additionally checked on 228 small instances with no violation. □

The outer bound for every procedure. Let π be any admissible procedure, adaptive, randomised and free to use side information and any audit order, let y be the real pool labelling with $|y|=T$, $V=K+T$, $h=h(V)$, and let U be the skipped set, $R=|U|$. *Lemma (inclusion).* For every set H of h rows, $\Pr_y(H \subseteq U) \leq \alpha$. *Proof.* Let y' equal y off H and 1 on H, so $|y'| \leq T+h$. Run π on y and on y' with the same internal randomness; on the event $H \subseteq U$ no row of H is ever audited, so the two transcripts coincide and so do the skipped sets. Under y' that event loses at least h positives, and $h > \varepsilon(V+h) \geq \varepsilon(K+|y'|)$ because $h(1-\varepsilon) > \varepsilon V$ by the definition of h. So the event is a failure under y', whose probability is at most α. □Summing over all h-subsets gives $\mathbb{E}_y\binom{R}{h} \leq \alpha\binom{M}{h}$. Since $r \mapsto \binom{r}{h}$ is convex on the integers, the largest $\mathbb{E}R$ compatible with this moment constraint is attained by a law on two consecutive integers, and equals $G(M,h,\alpha) = r + \left(\alpha\binom{M}{h} - \binom{r}{h}\right)/\binom{r}{h-1}$ with $r = \max\{j : \binom{j}{h} \leq \alpha\binom{M}{h}\}$. The clean one-look rule that skips a uniformly random set of that law attains it at $T=0$, so $G(M,h,\alpha) = S_0^*(M,h)$ (checked against the linear-programme values of Table 9: 11.3498 and 24.4862 are reproduced exactly). Hence $\mathbb{E}_y[\text{skip}] \leq S_0^*(M,h(V))$

	positives needed	N for $\Pr = 0.80$	N for $\Pr = 0.95$
both point estimates	80	556	604
validity rate at its exact lower bound	80	724	788
both inputs at their lower bounds	90	812	876

Table 8: Cohort size needed for certified rejection to be *structurally possible*, as an event rather than in expectation. Exact binomial arithmetic; the Clopper–Pearson bounds are found by bisection on exact binomial tails (one-sided 95% lower bounds, 0.1201 for the validity rate and 0.851 for the retained share). Model: the number of valid roots is $\mathrm{Bin}(N, p)$ and the retained share q is treated as fixed, so structural possibility is $\{V \geq \lceil 76/q \rceil\}$; sizes are searched in steps of four. Under the alternative model $K \sim \mathrm{Bin}(N, pq)$, the first-row minima are 557 and 605 over unrestricted integer sizes, or 560 and 608 on the four-root grid. The expectation-based answer of 512 roots delivers $\Pr[K \geq 76] = 0.519$ in the first row and 0.009 in the second. We pre-register 896.

for every admissible procedure and every labelling, which is the outer bound of Theorem 4 without the random-order assumption.

F Proof of Theorem 6

Let g be a uniformly random permutation of the rows, independent of everything. Define $\bar{\pi}$: run the score-aware procedure π on a virtual pool in which virtual row i carries score σ_i and label $y_{g(i)}$; when π asks for virtual row i, audit real row $g(i)$; stop when π stops. Under a real labelling y the virtual labelling is $y \circ g$, so the loss of $\bar{\pi}$ has the law of the loss of π averaged over $y \circ g$, and $\Pr_{\bar{\pi}}(\text{bad}; y) = \mathbb{E}_g \Pr_{\pi}(\text{bad}; y \circ g) \leq \alpha$: $\bar{\pi}$ is admissible. Its saving is $\mathbb{E}_g \mathbb{E}_{\pi}[\text{skip}; y \circ g]$, the average of π's saving over labellings with $|y| = T$, which is at least the minimum. Finally, given the real history, the next real row $g(i)$ is uniform over the unaudited real rows, because the restriction of g to unaudited virtual rows is a uniform bijection independent of i. So $\bar{\pi}$ is score-blind and $\mathbb{E}_{\bar{\pi}}[\text{skip}] \leq S^*(M, K, T)$ by Theorem 1. □

G Sizing a cohort as an event, not an expectation

The expectation thresholds of Appendix M (973, 590 and 512 roots) should not be used for design. They are the sizes at which $K \geq K_{\min}$ holds *in expectation*, and structural possibility is not an expectation but an event. Computing the event exactly at 512 roots gives $\Pr[K \geq 76] = 0.519$ under the point estimates, and 0.009 once the validity rate is taken at its exact Clopper–Pearson lower bound of 0.1201 rather than its point value of 0.1562. The retained share is the weaker input of the two: $38/40$ rests on forty positives, and a point estimate of 0.95 is compatible with 0.851.

The first cohort spent its whole budget to discover $K = 33 < 76$; sizing the replacement to the expectation would have bought a coin flip on repeating that. We pre-registered 896 roots, which exceeds the stated feasibility threshold under the most conservative input scenario of Table 8. This is a conditional planning calculation, not a distribution-free guarantee under cohort shift.

H A non-monotone saving, on a projected instance

The paragraph below is computed exactly, but on a projected cleanliness profile belonging to a predictor that was not run on the pre-registered cohort; the measured cohort used a different predictor and had $T = 34$ at $\varphi = 0.45$. It is reported as a computed phenomenon, not a measurement.

On the specified projected cleanliness profile, certified saving has an interior optimum. Because widening the nominated pool moves positives into the region the certificate is blind to while removing them from the count it depends on, the certified saving is *not monotone* in how aggressively one screens. Under the boundary we recommend, rebuilt and revalidated at each pool size, the saving on our projected instance rises to 27.0 percent of the cohort at $\varphi = 0.45$ and falls to 9.7 percent at $\varphi = 0.50$ — a 2.8-fold loss for a five-point widening, against an exact ceiling that falls

the same way (27.09 to 9.76 percent), so this is a property of the problem and not of our rule. The uncertified saving keeps rising throughout, so a practitioner tuning a threshold on the saving curve sees nothing wrong. We report this as a phenomenon computed exactly on a named instance, framed by two theorems rather than proved in general: the non-monotonicity is conditional on the pool's cleanliness profile, and no property of the certificate class implies that profile. Under the four-look rule the same widening takes the saving to zero, but that rule is not the one we recommend, and quoting its collapse would overstate the effect.

I ANYTIME-VALID CERTIFICATES, COMPARED EXACTLY

Anytime-valid rules on the projected instance. A confidence sequence, e-process or betting boundary is a natural way to decide after every row, and it is valid in this setting — Waudby-Smith & Ramdas (2020) give confidence sequences for exactly this sampling scheme, and Holmes & Walker (2026) the curtailment form — because under each hypothesis the next label is Bernoulli$((t_0 - s_m)/(M - m))$ and the likelihood-ratio supermartingale is explicit. On the specified projected instances, the tested Ville-threshold mixtures underspend the permitted error budget (0.4 to 38 percent of it) and lose 12 to 42 rows against the greedy boundary; calibrating their first-crossing probabilities recovers most of that and lands 0.3 to 7.6 rows below it, tolerating $T \le 3$ against its $T \le 4$. This is a comparison of particular rule families and calibrations, not a separation between anytime-valid and non-anytime-valid certification: every admissible rule is bounded by the same ceiling, and Ville's inequality can be tight at a finite horizon.

At $M = 122$, $K = 38$, the frozen four-look Bonferroni rule cannot fire. This is a restriction of that schedule and allocation, not an impossibility for all admissible boundaries: auditing 95 rows and discarding the remaining 27 only if no positive was seen has size $\binom{27}{3}/\binom{122}{3} \approx 0.0099 < 0.01$ at the least unsafe total $h = 3$ (this $K = 38$ is the stronger predictor's instance, not the deployed $K = 33$ one).

J PROOFS FOR SECTION 3.2

In this appendix t_0 denotes the null boundary τ of the main text.

Programme. A score-blind certificate is a stopping probability per label prefix. Under a labelling with T ones every prefix of length m with s ones is equally likely, so the objective and the constraints depend on the certificate only through the per-state averages $u_{m,s}$ (stop mass) and $v_{m,s}$ (reach mass), which satisfy $v_{0,0} = 1$, $0 \le u \le v$ and $v_{m,s} = \frac{m-s}{m}(v-u)_{m-1,s} + \frac{s}{m}(v-u)_{m-1,s-1}$, with the counting weights $\binom{m-1}{s}/\binom{m}{s}$ and $\binom{m-1}{s-1}/\binom{m}{s}$. The objective is $\sum_{m,s} p_T(m,s)(n-m)\,u_{m,s}$ and there is one constraint $\sum_{m<n,s} p_{T'}(m,s)\,e_{T'}(m,s)\,u_{m,s} \le \alpha$ for every count T', with error indicator $e_{T'}(m,s) = \mathbf{1}\{T' \ge t_0\}$ for incumbent rejection (every stop under such a labelling is a wrong rejection) and $e_{T'}(m,s) = \mathbf{1}\{T' - s > \lfloor \varepsilon(K+T') \rfloor\}$ for screening. Stopping decisions are indexed by $0 \le m < n$; at $m = n$ the evaluation is completed without rejection (equivalently $u_{n,s} = 0$), and out-of-range predecessor terms in the flow recursion are zero. Conversely, any feasible (u, v) is implemented by stopping at state (m, s) with probability $u_{m,s}/v_{m,s}$ when $v_{m,s} > 0$; the flow recursion gives the prescribed masses under every count, so the programme is exact, not a relaxation; $p_T(m,s)$ is the hypergeometric probability of s ones in the first m rows. This is the programme of Theorem 1 with the bad set changed, and its value is the ceiling.

Proposition (a closed-form bracket). For the candidate-versus-incumbent problem, assume $0 \le T < t_0 \le n$ and $0 < \alpha < 1$. Let s_k be the number of ones in the first k audited rows and β_k the power at T of the lower-tail uniformly most powerful level-α test of $\{T' \ge t_0\}$ based on s_k, randomised to have size exactly α at $T' = t_0$. Then $\max_{0 \le k < n}(n-k)\beta_k \le S^* \le \sum_{k=0}^{n-1} \beta_k$. For $t_0 \le T \le n$, instead $S^* = \alpha n$: the upper bound follows from $\mathbb{E}_T[\text{skip}] \le n\mathbb{P}_T(\text{reject})$, and rejecting at time zero with probability α attains it. *Proof.* Given s_k, the order of the first k rows is uniform whatever T', so s_k is sufficient for T' on those rows and every test based on them has the power function of a test based on s_k. The law of s_k is hypergeometric, which has a monotone likelihood ratio in T', so the test exists, is uniformly most powerful by the Karlin–Rubin theorem, and is valid for every $T' \ge t_0$. For any admissible rule with stopping prefix τ ($\tau = n$ if it never

rejects), $n - \tau = \sum_{k<n} \mathbf{1}\{\tau \le k\}$, and $\{\tau \le k\}$ is a level-α rejection region based on the first k rows, so $\mathbb{P}_T(\tau \le k) \le \beta_k$ and $\mathbb{E}_T[n - \tau] \le \sum_k \beta_k$. Conversely, the rule that looks once, at prefix k, and applies that test is admissible and saves $(n - k)\beta_k$. □ The bracket needs no programme, so it is available beyond the pool sizes at which we solved one. On GEPA's 431 candidates below their incumbent it places the pooled exact ceiling of 48.0% of their rows between 36.0% and 54.6%, and it contains the exact ceiling of every candidate.

Dual bound. Fix multipliers $\lambda_{T'} \ge 0$, put $g(m, s) = p_T(m, s)(n - m) - \sum_{T'} \lambda_{T'} p_{T'}(m, s)\, e_{T'}(m, s)$ and define, with $Y \equiv 0$ beyond row $n - 1$,

$$Y(m, s) = \max\Big\{ g(m, s),\ \tfrac{m+1-s}{m+1}\, Y(m+1, s) + \tfrac{s+1}{m+1}\, Y(m+1, s+1) \Big\}.$$

For incumbent rejection with one boundary multiplier λ, $g(m, s) = p_T(m, s)(n-m) - \lambda\, p_{t_0}(m, s)$ and the bound below is $Y(0, 0) + \lambda\alpha$. With $r = v - u \ge 0$, the flow identity gives $\sum_s Y(m, s)\, v_{m,s} = \sum_{s'} r_{m-1,s'} \big[\frac{m-s'}{m} Y(m, s') + \frac{s'+1}{m} Y(m, s'+1) \big] \le \sum_{s'} Y(m-1, s')\, (v - u)_{m-1,s'}$, the bracket being the continuation term of the recursion. Telescoping from $v_{0,0} = 1$ yields $\sum Yu \le Y(0, 0)$; since $Y \ge g$ and $u \ge 0$, $\sum gu \le Y(0, 0)$, that is, the objective is at most $Y(0, 0) + \alpha \sum_{T'} \lambda_{T'}$. For incumbent rejection, multipliers outside $T' \ge t_0$ may be set to zero. We evaluate this bound in exact rational arithmetic at a rationalised multiplier. A single boundary-null multiplier gives an upper bound; it certifies the composite-null optimum when matched by a feasible lower-set primal rule. This single-boundary-null reduction applies to incumbent rejection under the lower-set condition; it does not eliminate the state-dependent screening constraints, which the screening verifier keeps in full. Where primal and dual bounds differ, the optimum is enclosed by them and displayed decimals are rounded. Boundary conventions: a negative design count is replaced by 0; when $\tau > n$ the null is empty, so curtailment and an optimal LP policy reject at time zero and save all n rows (no such state occurs in our replays: the largest incumbent count is 235 of 300); when $\tau = 0$ the reported LR policies never reject; these cases are handled before an LR mixture is constructed; for the clean-pool formula $S_0^*(M, h) = M$ when $h > M$.

The boundary constraint suffices for threshold rules. Let a rule reject at the first row m with $S_m \le c_m$, or be a mixture of two such rules. Realise T' and $T' + 1$ ones on one uniform order by adding a uniformly placed extra one; then $S_m^{T'+1} \ge S_m^{T'}$ for every m, so the rejection event can only shrink and $\mathbb{P}_{T'}(\text{reject})$ is nonincreasing in T'. Such a rule is therefore admissible if and only if it is admissible at $T' = t_0$, and when the single-constraint programme is attained by such a rule the two programmes have the same value. For incumbent-rejection threshold policies, composite-null validity follows from the boundary-null size and the coupling; other null constraints may also be tight. Arithmetic certification is specified by policy in Appendix A. Screening policies retain their state-dependent error event and are checked over all relevant counts.

The composite null for $R2$. $R2$ stops at state (m, s) with probability $u_{m,s}/v_{m,s}$, so it randomises state by state rather than mixing deterministic thresholds before the audit, and a down-set of states with positive stopping probability does not by itself make the coupling above apply. We therefore computed its rejection probability directly at every null count. For the 300 tables at $n = 300$ (every incumbent state of the GEPA and MIPROv2 replays and of the live runs), each table's rejection probability at every $T' \in [t_0, 300]$ was scanned in double precision and recomputed in exact rational arithmetic at the worst count: the maximum is attained at the boundary $T' = t_0$ in all 300 tables, never exceeds α (largest 0.0099999999999956), and is at most 0.0073 strictly inside the null. The same computation for the 131 rules of SIMBA's small pools ($n = 150$, 594, 770; rebuilt with the same constructor) gives the maximum at the boundary in all 131, never above α, and at most 0.0083 strictly inside the null. Scripts and outputs: `r2_composite_size_check.py` and `simba_composite_size_check.py` in the supplement.

Curtailment. It rejects when $s + (n - m) < t_0$, which no labelling with t_0 ones can produce, so it never errs. For $0 \le T < t_0 \le n$ it stops at the $(n - t_0 + 1)$-th failure; the $n - T$ failures are uniformly placed, the j-th sits at expected row $j(n + 1)/(n - T + 1)$, and the expected saving is $n - (n - t_0 + 1)(n + 1)/(n - T + 1)$. It is n when $t_0 > n$ and 0 when $t_0 \le T$.

The run-level guarantee, with the retained incumbent and a tolerance d, is proved in Appendix B.

K THE ALLOWANCE HEURISTIC AND THE RATE, IN FULL

Before any sequential analysis, an allowance argument suggests what should be available. The guarantee permits losing $\varepsilon(K+T)$ positives; the pool holds T of them; so one expects a score-blind discard to cover at most a $\varepsilon(K+T)/T$ fraction of the pool. As a fraction of the nominated pool that reads $\varepsilon(K/T+1)$, and K/T is the only quantity a predictor can move. At the measured $K/T = 33/7 = 4.714$ it gives 0.286.

It is a heuristic and not a bound. At $(M, K, T) = (60, 0, 60)$ the randomised rule "stop at row 0 with probability $1/100$; otherwise stop at row 57 if all 57 audited rows were positive" has worst-case error exactly α, so it is admissible, and its expected saving is 3.57 rows against the heuristic's 3.00. The gap is the α-mass: a certificate may spend its whole error budget on a single early stop that skips the entire pool, and the allowance argument does not account for that term. A sweep of 760 small instances finds 30 violations, every one of size at most αM.

What survives is a bound of the same order with the α-mass carried explicitly. For every admissible certificate,

$$\mathbb{E}_T[\text{skip}] \;\le\; \alpha M \;+\; \frac{M(A+1)-(T-A)}{T+1}, \qquad A = \lfloor \varepsilon(K+T) \rfloor,$$

and it is what the paper uses from here on. At the measured instance $(122, 33, 7)$, where $A = 2$, it gives $1.22 + (366-5)/8 = 46.345$ rows, 0.380 of the pool, against the exact ceiling of 6.9433 rows: this analytic bound is loose at the measured instance by a factor of about 6.7. It is an upper bound on a saving, not a saving. Justification: when $A < T$, let P_{T-A} be the position of the $(T-A)$-th positive in the random audit order. On every non-error outcome, stopping occurs no earlier than P_{T-A}; error outcomes contribute at most αM. Using $\mathbb{E}[P_{T-A}] = (T-A)(M+1)/(T+1)$ gives the upper bound. When $A \ge T$, the displayed bound is at least M and is trivial.

The following guarded single-look construction gives an explicit finite-sample lower bound; no uniform constant-factor approximation is claimed. Consider the guarded single look "audit the first $M-r$ rows, skip the remaining r if at most t positives were seen". Its admissibility is a finite hypergeometric condition that can be checked exactly. Take $t = T$, $h_0 = h(K)$ and $T_{\mathrm{sat}} = T + h(K+T)$, and $r = \lfloor M h_0\, \alpha^{1/h_0} / (e\, T_{\mathrm{sat}}) \rfloor$. Under the target count the guard always fires. For $T' \le T_{\mathrm{sat}}$ a bad discard requires at least h_0 positives in the skipped block, whose probability is at most $\binom{r}{h_0}(T'/M)^{h_0} \le (erT_{\mathrm{sat}}/(Mh_0))^{h_0} \le \alpha$; for larger T' the coupling bounds the guard probability by its value at T_{sat}. Hence

$$\frac{M h_0\, \alpha^{1/h_0}}{e\, T_{\mathrm{sat}}} - 1 \;\le\; S^*(M,K,T) \;\le\; \alpha M + \frac{M(A+1)-(T-A)}{T+1},$$

where $h_0 = h(K)$, $T_{\mathrm{sat}} = T + h(K+T)$ and $A = \lfloor \varepsilon(K+T) \rfloor$: a closed-form bracket at every sample size; we do not claim a rate beyond it.

L RECOVERING THE BONFERRONI LOSS, IN FULL

The frozen rule spends $\alpha/4$ at each of four looks by a union bound. Measured on the projected instance, its worst-case error over every labelling is 0.001795 against a budget of 0.01: it uses under a fifth of the error it is allowed. The waste has a cause. Stopping is a single first-crossing event over the audit sequence, and its probability is computable exactly by the same hypergeometric recursion the rule already uses; the union bound is a bound on that probability, and the recursion is the probability.

Kato & Nakagawa (2026) formalise this for finite-population audit sampling in financial statement auditing: a fixed population sampled without replacement, stopping boundaries computed by hypergeometric recursion, decisions at prespecified times, one-sided error control. Their state at time t is the cumulative deviation count, which is our (m, s). We cite them for the recursion and for their coupling; their Theorem 3.1 does *not* transfer as stated, because their error event is fixed while ours grows with the labelling, so validity in our setting is proved separately (Appendix M, paragraph "Validity of the frozen rule").

Two cautions belong here rather than in a footnote, because both cost us a wrong result first. A boundary tuned at one pool size can be a parity accident: the vector that is valid at $M = 409$ is

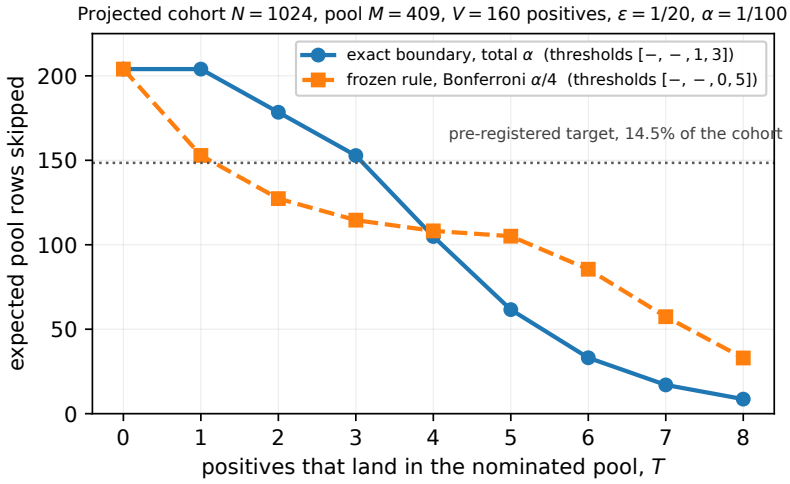

Figure 3: Expected number of pool rows a certified rejection skips, against the number of positives that land in the nominated pool, on the projected cohort. Both curves are exact: the expectation is the hypergeometric forward recursion over audit prefixes evaluated in rational arithmetic, not a simulation, so there are no error bars and no runs to report. Both rules are valid — worst-case error over every labelling is 0.001795 for Bonferroni and 0.009797 for the exact boundary, against a budget of 0.01: the union bound spends under a fifth of what it is allowed. In expectation the frozen rule falls below the target at $T = 2$ and the exact boundary at $T = 4$. **Under the protocol's own probability requirement** $\Pr[\text{skip} \geq \text{target}] \geq 0.8$ the two tolerate $T = 0$ and $T \leq 1$ respectively, so the improvement is from none to one, not a tripling. Beyond $T = 4$ the exact boundary is *worse*, because its thresholds stop earlier; we plot the crossing rather than truncate at it. Both are superseded by the every-row boundary below.

invalid at $M = 410, 412, 416, 420, 440, 480$ and 512, since $F(M, 10, \lceil M/2 \rceil)(1)$ crosses α between 409 and 410. And "tolerates $T \leq 3$" holds in expectation; under the protocol's own requirement that $\Pr[\text{skip} \geq \text{target}] \geq 0.8$, the four-look exact boundary tolerates $T \leq 1$ and the frozen rule $T = 0$.

In expectation this moves the design's tolerance from $T \leq 1$ to $T \leq 3$ (Figure 3); under the protocol's probability requirement, from $T = 0$ to $T \leq 1$. Beyond $T = 4$ the exact boundary is worse than Bonferroni.

Neither is what we would deploy. A deterministic boundary that may decide after *every* audited row — every-row decisions remove the dependence on the four prescribed look fractions, and the boundary is rebuilt and verified at each instance — is better than both, and its construction has a subtlety.

Two parameters of that construction are jointly load-bearing, and each alone is misleading. Capping the per-row threshold at 12, an implementation convenience, makes the boundary collapse once the pool is dirty: it saves 1.97 rows at $T = 20$ and nothing at all by $T = 30$. Raising the cap alone produces an *invalid* rule, worst-case error 0.1223 against a budget of 0.01, because the construction then spends large late thresholds without being obliged to defend the T at which the error actually binds, which is $T = 67$ on this instance. Widening the guarantee range alone changes nothing whatever: while it defends only $T \leq 40$ the construction never wants a threshold above 12, and the two capped vectors agree in every entry. Only the pair is both valid and better. Comparing the two every-row boundaries exhaustively over all 409 values $T = 1, \ldots, 409$ in rational arithmetic, the uncapped one is strictly better at 81, exactly equal at 328, and worse at none, with an unchanged worst-case error of 0.009939: it saves 63.83 rows at $T = 20$ and 41.43 at $T = 40$ where the capped one saves nothing. The cap was never the defect; it was concealing one.

With both parameters set correctly the boundary tolerates $T \leq 4$ under the protocol's probability requirement, against $T = 0$ for the frozen rule, and tracks the exact linear-programming ceiling

over all admissible adaptive randomised certificates to within 0.689 rows for $T \le 8$, 0.89 rows to $T \le 19$, and at most 2.25 rows through $T = 40$. We had previously reported that an every-row boundary stops tracking the ceiling beyond $T \approx 10$; that was an artefact of the cap and we withdraw it.

M THE SPARSE SCREENING REGIME IN FULL

This appendix gives the sparse-regime material summarised in Sections 3.4 and 5.3: the setting, the exact ceilings at the measured instances, the binding threshold, the two cohorts, full cost accounting and the event criterion. Theorem 1 and Theorem 4 are stated in the main text.

M.1 SETTING

The frozen sequential rule of the program-generation study fires at look m if and only if

$$F\big(M,\ s + h(K+s),\ m\big)(s)\ \le\ \beta, \qquad h(K') = \left\lfloor \tfrac{\varepsilon K'}{1-\varepsilon} \right\rfloor + 1, \tag{1}$$

where s is the number of positives seen in the first m audited rows, M the pool size, K the retained positive count, β the per-look error budget and F the hypergeometric CDF. The number T of positives inside the pool does not appear.

A batch of N candidates is partitioned by a cheap predictor into a *nominated pool* of size M, proposed for discard, and a retained remainder. Each candidate is latently positive (a valid output) or negative. Write T for the positives inside the pool and K for the positives outside it. Verification is expensive and is what a certificate tries to skip; the predictor's scores are cheap and already paid for.

A certificate audits rows of the pool in a uniformly random order and may stop early and discard the unaudited remainder. It is *admissible at level α with tolerance ε* if, for every labelling, the probability that it discards more than $\varepsilon(K+T)$ positives is at most α. Probability is over the audit permutation and any internal policy randomization, conditional on the fixed labelling, which is adversarial.

Our frozen instantiation is the four-look rule of (1): looks at $\{\lceil M/8\rceil, \lceil M/4\rceil, \lceil M/2\rceil, \lceil 3M/4\rceil\}$, each spending $\beta = \alpha/4$, with $\varepsilon = 1/20$ and $\alpha = 1/100$, and F the hypergeometric CDF.

Validity of the frozen rule. Fix a true pool count T' and write $t(s) = s + h(K+s)$. For integers, $T' - s > \lfloor \varepsilon(K+T') \rfloor$ if and only if $T' \ge t(s)$, so a discard at observed count s is bad exactly when $T' \ge t(s)$. On a bad firing at look m, the hypergeometric CDF being nonincreasing in its positive-count argument gives $F(M, T', m)(s) \le F(M, t(s), m)(s) \le \beta$; and for $S_m \sim \mathrm{Hypergeom}(M, T', m)$ the discrete-CDF inequality gives $\mathbb{P}_{T'}[F(M, T', m)(S_m) \le \beta] \le \beta$. The probability of a bad firing at each look is therefore at most β, and the union over four looks is at most $4\beta = \alpha$. If $t(s) > M$ no compatible total is unsafe; in (1) the comparison value is then defined to be zero (a safe discard).

An allowance heuristic, and why it is not a bound The guarantee permits losing $\varepsilon(K+T)$ positives and the pool holds T, which suggests that a score-blind discard covers at most an $\varepsilon(K+T)/T$ fraction of the pool. This is not a bound: a certificate may spend its whole error budget on one early stop that skips the entire pool. In the 760-instance enumeration reported in Appendix K, all 30 observed violations were at most αM; that enumeration does not establish a universal maximum violation. With that α-mass carried explicitly, every admissible certificate satisfies $\mathbb{E}_T[\text{skip}] \le \alpha M + \big(M(A+1) - (T-A)\big)/(T+1)$ with $A = \lfloor \varepsilon(K+T) \rfloor$, and a guarded single look gives a closed-form lower bound (Appendix K).

Why the asymptotic condition misleads at small counts A certified discard at audit fraction f requires, asymptotically, $K/T \ge (1-\varepsilon-f)/\varepsilon$, which is scale-free: enlarging the batch moves both sides together. This form is common, but at the sample sizes that occur it is the wrong object. At $T = 2$ the neglected confidence width is about four positives against a T of two, so the comparison is uninformative in both directions. Section M.2 replaces it.

Table 9: Exact ceilings over all admissible adaptive randomised score-blind certificates ($\varepsilon = 0.05$, $\alpha = 0.01$), enclosed by rational lower and upper certificates; displayed decimals are rounded. Last column: ceiling rows over the 256 roots.

instance		ceiling (rows)	of the pool	of the 256-root work
$M = 49, K = 40, T = 0$	nomination 0.20	11.3498	23.16%	4.4%
$M = 110, K = 40, T = 0$	nomination 0.45	24.4862	22.26%	9.6%
$M = 122, K = 33, T = 7$	the deployed run	6.9433	5.69%	2.7%

Parametrising the programme by prefix *counts* is numerically fragile without rescaling: at $(M, K, T) = (122, 33, 7)$ the magnitudes reach $\binom{122}{7} \approx 6.70 \times 10^{10}$ and a floating-point solver returns an "optimum" of 29.82 whose extracted rule has exact violation 0.974, i.e. it is not admissible at all. Parametrising instead by per-prefix averages, with all variables in $[0, 1]$ and flow weights $(m + 1 - s)/(m + 1)$ and $s/(m + 1)$, gives 6.943309 with exact certificates on both sides.

Table 9 gives the ceilings at the instances this paper measures.

At $T = 0$ the optimum audits about 77% of the pool at random and skips the rest only if no positive appeared; at the deployed instance it audits 94.3% on average and usually stops after seeing positives. Deterministically discarding the whole pool without audit is inadmissible at every $\alpha < 1$ when $h(K) \le M$, since the labelling with h positives in the pool is violated with certainty; randomized whole-pool rejection is allowed, and $h(K) > M$ is a safe-discard case.

M.2 The threshold that binds, exactly

Equation (1) is equivalent to the source form of the rule and is the form that makes the situation transparent. Writing $V(s) = \lfloor (s+\varepsilon K)/(1-\varepsilon) \rfloor$ for the largest safe total, the identity $(s+\varepsilon K)/(1-\varepsilon) = s+\varepsilon(K+s)/(1-\varepsilon)$ gives $V(s) = s+h(K+s)-1$, and monotonicity of F in its positive-count argument together with integrality turns the source comparison into (1).

Proposition 5 (The requirement is a count, not a ratio)**.** *At $M = 122$, $\varepsilon = 1/20$, $\alpha = 1/100$ and the four frozen looks, the minimum K at which the rule can fire on any permutation is $K_{\min} = 76$, attained first at $(m, s) = (92, 0)$, and this value is the same for $T = 0$, $T = 2$ and $T = 7$. Moreover $K_{\min} = 76$ for every pool size from $M = 128$ to $M = 640$.*

Two consequences matter more than the constant. First, the requirement is not a ratio requirement: it is the allowance h reaching 5, which happens at $K = 76$ whatever T is. Second, the bar does not rise with the batch, which is what makes enlarging the cohort a coherent strategy rather than a treadmill. At $M = 49$ finite-population effects make the threshold the easier $K \ge 57$.

M.3 Experiment

Protocol. A frozen 256-candidate program-generation study with a 32-token early-stopping predictor, under a protocol fixed before any label was read. Of the 256 roots, 245 carry every one of the ten frozen features and 11 do not; an unscorable root cannot be nominated, so it is retained by construction and its label counts towards K. The cohort holds 40 valid roots.

The deployed predictor. Nominated pool 122, $T = 7$, $K = 33$, so $K/T = 4.714$ against an asymptotic requirement of 4.00 — cleared by 18%. Nothing fired.

Refusing was correct. Discarding the nominated pool uncertified would have destroyed 7 of 40 valid outputs, 17.5% of the yield, against an allowance of $\varepsilon(K + T) = 2$ roots: an overshoot of 3.50×. The guarantee had something to bind and it bound. This is what makes the negative result evidence rather than an absence of it.

Table 10: The frozen predictor on the 256-root test cohort (40 valid roots), $\varepsilon = 0.05$: counts from this single cohort, not means over runs. Pool: nominated candidates; T, K: valid roots inside and outside the pool; $h = \lfloor \varepsilon K/(1-\varepsilon) \rfloor + 1$. At the nomination fraction the deployed predictor used, 122 of 256, it lifts K/T from 4.71 to 19.0.

nomination	pool	T	K	h	K/T
0.20–0.45	49–110	0	40	3	undefined
0.50	122	2	38	3	19.0

A much stronger predictor, frozen and applied once. We fitted gradient-boosted depth-one stumps on a disjoint 224-root cohort, fixed every hyperparameter before the test cohort was scored, serialised the model to thresholds and leaf values, and applied it exactly once.

And it buys nothing here. At $s = 0$ the firing condition is a single tail probability, and it fails everywhere on these instances, by factors of about 4.78, 5.42 and 5.50: $F(49, 3, 37)(0) = 55/4606 = 0.011941$; $F(110, 3, 83)(0) = 65/4796 = 0.013553$; $F(122, 3, 92)(0) = 203/14762 = 0.013752$; all against $\beta = 0.0025$. Checked over every look and every reachable s, the firing set is empty, so the firing probability is exactly zero for every labelling and every audit permutation, and the audited fraction at $T = 0$ is 100%.

M.4 Full cost accounting

Token savings do not count; occupied resources do. The measured ledger over 21 events totals 2378.8 GPU-seconds, decomposed as prior study charges 1135.6 (47.7%), continuation acquisition 929.6 (39.1%), resource-prefix acquisition 313.5 (13.2%), and orchestration and exact evaluation at 0.0 GPU-seconds. **Measured saving:** 0.0 **GPU-seconds.**

One consequence needs no modelling assumption. The apparatus that makes a certified skip *possible* — prior charges plus computing the 32-token prefixes the predictor scores — is 1449.1 GPU-seconds, 60.9% of the total, while the entire body of work a rejection could have avoided is 929.6 seconds, 39.1%. The screening apparatus costs more than the work it screens, before any question of how much of that work a certificate may skip. On this cohort the method is upside down as arithmetic, independently of the audit rule, the predictor and the ceiling.

The negative result survives the amortisation objection. Of the 60.9 percent, 47.7 points are prior-study charges: sunk, one-off, and not repeated on a second cohort where the predictor is already fitted; the headline is nevertheless not an artefact of amortisation. Let $p = 1.2247$ GPU-seconds be the screening cost per cohort root and $c = 3.7944$ GPU-seconds the mean cost of an avoided continuation. Under this mean-cost model, expected net saving is $c\,\mathbb{E}[\text{skip}] - pN$, so break-even requires $\mathbb{E}[\text{skip}]/N > p/c \approx 0.3228$, where the denominator is all cohort roots. What must clear that threshold is the ceiling as a fraction of the *cohort*, which is the ceiling as a fraction of the pool times the nomination fraction φ: 0.0443 and 0.0957 at the actual pool fractions $49/256$ and $110/256$ (nominal 0.20 and 0.45). The best case is short by a factor of 3.37. Discarding every sunk cost and granting the certificate its exact information-theoretic maximum — not our rule, but the ceiling over all admissible adaptive randomised certificates — screening still costs about three times what can be saved.

That product is also what fixes the nomination fraction. The ceiling as a fraction of the pool is nearly flat across the range we priced, so the cohort-level saving *increases* with φ, and a larger nomination is better economics rather than worse. We pre-register $\varphi = 0.45$, where the expectation-criterion evidence requirement is also already met at $203 \geq 198$. The two arguments are independent and point the same way.

The protocol's criterion is an event, and the event has its own exact ceiling. The linear programme bounds the *expected* saving, and the protocol does not ask for an expectation: it requires $\Pr[\text{skip} \geq r] \geq 0.8$, so we bound the event itself.

The reduction is exact. For a threshold on the skipped count, any admissible adaptive randomised score-blind certificate may be replaced by one that makes a *single* decision, at row $M - r$, without changing its objective or its validity: with $c = M - r$, replace the stopping time σ by $\sigma' = c$ if $\sigma \le c$ and $\sigma' = M$ otherwise (continue auditing to row c after any earlier simulated rejection; otherwise finish the pool). This preserves $\Pr[\text{skip} \ge r]$, and the new discarded set is contained in the original one whenever rejection occurs, so the new rule's bad event is contained in the original's under every labelling. The event ceiling is therefore the value of a linear programme in the $T + 1$ acceptance probabilities, and on a clean pool it is a closed form,

$$\Pr{}^{*}[\text{skip} \ge r] \;=\; \begin{cases} 1, & r < h, \\ \min\{1,\ \alpha\binom{M}{h}/\binom{r}{h}\}, & h \le r \le M, \end{cases} \qquad h \;=\; \left\lfloor \tfrac{\varepsilon K}{1-\varepsilon} \right\rfloor + 1,$$

for integer $0 \le r \le M$; the first case includes $h > M$, when discarding the whole pool is safe.

For the target $r = \lceil 0.145 \times 256 \rceil = 38$, the event ceiling is approximately 0.2558 at the clean instance $(M, K, T) = (110, 40, 0)$, and 0.01809 at the measured instance $(122, 33, 7)$, against the required probability 0.8. So the pre-registered criterion **cannot be met by any admissible adaptive randomised score-blind certificate on these instances**, whatever rule is used. At the measured instance, Markov's inequality applied to the expected-savings ceiling gives only $\mathbb{P}(\text{skip} \ge 38) \le 6.943309/38 \approx 0.1827$; the direct event calculation therefore establishes a shortfall factor of approximately 44.2, compared with 4.38 from Markov's inequality. These conclusions fix the instance and target.

The same closed form draws the boundary in general. For $0 < p \le 1$ and $h \le M$, a certificate meeting the criterion on a clean pool exists if and only if $\binom{r}{h}/\binom{M}{h} \le \alpha/p$ (for $h > M$ discarding the whole pool is safe and the criterion is met), which is monotone in each of M, K and r: a phase boundary a practitioner can evaluate before auditing anything. The threshold $K \ge 76$ of Section M.2 is the special case $r = \lfloor M/4 \rfloor$ at level $\alpha/4$. For $(M, r, \alpha, p) = (122, 38, 0.01, 0.8)$, the clean-pool criterion requires $K \ge 57$; for $(110, 38, 0.01, 0.8)$ it requires $K \ge 76$. These thresholds depend on the pool size and target as well as the error budget.

M.5 Recovering the Bonferroni loss

The frozen rule spends $\alpha/4$ at each of four looks by a union bound, and on the projected instance its worst-case error over every labelling is 0.001795 against a budget of 0.01: it uses under a fifth of the error it is allowed. Stopping is one first-crossing event whose probability the hypergeometric recursion computes exactly (Kato & Nakagawa, 2026), so the union bound can be replaced by the probability itself. A deterministic boundary that may decide after *every* audited row, built greedily subject to the α constraint (on the stated projected instance its worst-case error is 0.009939), tolerates $T \le 4$ under the protocol's probability requirement against $T = 0$ for the frozen rule, and tracks the exact ceiling over all admissible certificates to within 0.689 rows for $T \le 8$ and 2.25 rows through $T = 40$. Two parameters of its construction are jointly load-bearing, and a look schedule tuned at one pool size can be a parity accident; both cost us a wrong result first and are documented, with the exact curves, in Appendix L. The anytime-valid comparison is in Appendix I.

M.6 The ceiling does not depend on the predictor

Everything so far is a statement about an instance (M, K, T), and an instance is what a predictor produces. The objection it invites is that a better predictor would have produced a better instance. Theorem 4 (Section 3.4) removes the predictor from the question. Write $S^{*}(M, K, T)$ for the exact ceiling of Theorem 1 and $S_0^{*}(M, h)$ for its clean-pool value, which is explicit: with $F(m) = \binom{M-h}{m}/\binom{M}{m}$ and $m_{\min} = \min\{m : F(m) \le \alpha\}$, $S_0^{*}(M, h) = M - m_{\min} + q$ for a randomisation weight $q \in [0, 1)$.

The proof plants T fictitious positives on random rows fixed before the audit: a certificate admissible at K becomes one admissible at $K + T$ on a clean pool, with the same law of skip (Appendix E). The quantifier is over the pools a predictor can produce, not over predictors as functions, and that is the sense in which the ceiling is predictor-free: the cap is set by the tolerance, the confidence and the number of valid programs in the cohort, and a predictor can only fall short of it.

The break-even window, exactly. A certified skip pays only if it covers at least ρN rows, where ρ is the marginal break-even fraction of the cohort ($\rho = 0.323$ on the pre-registered cohort, $N = 896$). With nomination fraction $\varphi = M/N$ and $0 < \rho < \varphi \leq 1$, Theorem 4 makes break-even require, to leading order, $\alpha^{1/h(V)} \geq \rho/\varphi$, that is

$$V \;\geq\; \tfrac{1-\varepsilon}{\varepsilon}\Big(\Big\lceil \tfrac{\ln(1/\alpha)}{\ln(\varphi/\rho)} \Big\rceil - 1\Big),$$

not a finite-sample necessary condition (exactly: $S_0^*(M, h(V)) \geq \rho N$). If $\varphi < \rho$ break-even is impossible, since $\mathbb{E}[\text{skip}] \leq M < \rho N$; if $\varphi = \rho$ it requires $\mathbb{E}[\text{skip}] = M$. The nominal nomination parameter was 0.45, but the executed cohort fraction was $\varphi = 383/896 \approx 0.427455$; substitution into the displayed leading-order expression gives $V \geq 304$, while the illustrative value 247 corresponds to a cohort fraction of 0.45. Neither is an exact finite-sample necessary condition; the finite-sample comparison uses $S_0^*(M, h(V))$. The cohort had 133. Computed exactly at $V = 133$ (linear programmes with rational duals), score-blind audits under the stated cost model require pools of at least 512, 593 and 684 rows at $T = 0, 1, 2$ of the 133 valid programs; no feasible pool with $T = 3$ reaches expected break-even. These dirty-pool thresholds are not pointwise bounds for arbitrary score-aware audit orders. At the pool size used, even a perfect predictor caps at 216.9 rows, 24.2% of the cohort against 32.3%: at the evaluated nomination budget $M = 383$, under the stated mean-continuation-cost model, no admissible certificate can break even in expectation, even with perfect pool selection. This does not exclude a profitable individual audit realization, a larger nominated pool with sufficiently few positives, or a cheaper screening apparatus; making the pool cleaner without increasing $M = 383$ cannot overcome the outer bound under the same cost model.

Theorem 6 (Scores cannot buy a guarantee)**.** *Let σ be any side information fixed before the audit, such as the predictor's scores, and let a procedure choose which row to audit next from σ, the labels seen so far and internal randomness, subject to* $\Pr(\text{loss} > \varepsilon(K + |y|); y) \leq \alpha$ *for every labelling y. Then* $\min_{|y|=T} \mathbb{E}[\text{skip}; y] \leq S^*(M, K, T)$.

Relabelling the pool by a uniformly random permutation turns any such procedure into an admissible score-blind one whose saving is the average over labellings, so the ceiling of Theorem 1 holds for every procedure, score-aware or not, for the saving it can *guarantee* (Appendix F). A score-aware procedure can beat S^* only on a favourable realised labelling, and every admissible procedure, score-aware or not, is capped by the fixed-budget outer bound $S_0^*(M, h(V))$ of Theorem 4: 216.9 rows, 24.2% of the pre-registered cohort, against the guaranteed ceiling of 26.7 rows at the measured labelling.

M.7 What a better predictor buys, and the trap in saying so

Over the stated pool-size range, the frozen four-look rule's retained-positive threshold is constant at $K_{\min} = 76$; the cohort size needed to reach it depends on the validity rate and retained-positive share. Under the planning approximation $\mathbb{E}[K] = Npq$, with $p = 40/256$ and retained shares $q = 1/2$ (chance at nomination one half), $33/40$ (deployed) and $38/40$ (frozen predictor of Table 10), the expectation thresholds are 972.8, 589.6 and 512 roots; the corresponding integer sizes are 973, 590 and 512, or 976, 592 and 512 when restricted to multiples of four. These are expectation thresholds, not guarantees of firing. The predictor that cannot make the rule fire here halves the cohort on which it could fire.

Sizing to that expectation is the trap: structural possibility is an event, and at 512 roots it holds with probability 0.519 on point estimates and 0.009 at the validity rate's exact lower bound (Appendix G). We therefore pre-registered 896.

The pre-registered cohort, measured. The 896-root cohort (disjoint from the first and from every protected allocation, frozen predictor, nominal nomination setting 0.45 and executed fraction $M/N = 383/896$, same pinned evaluator) reached $K = 99 \geq 76$; one retained root whose test suite the evaluator refuses by design is excluded before treatment, so $K \in [99, 100]$. Certification was structurally possible. Of 383 candidates $T = 34$ were valid (8.9%, against 19.3% among the retained), so the predictor separates, yet no look certified and nothing was rejected. The rule could fire only at its terminal look with at most one positive among 288 audited rows; at the measured labelling a certificate has probability $2.2 \cdot 10^{-21}$, and a firing would discard at least 33 valid programs, exceeding the allowance of 6 by at least 27. Firing with probability one half needed $T \leq 1$,

a pool 34 times cleaner than delivered. Rejecting the pool *uncertified* would have lost 34 of 133 valid programs (25.6%, five times the tolerance): the case the certificate exists to refuse. At the fixed nomination budget $M = 383$ no admissible certificate can break even in expectation under the stated mean-continuation-cost model: even the universal outer ceiling is 216.9 rows, 24.2% of the cohort, below the 32.3% break-even requirement, and for score-blind audits at the measured instance $(383, 99, 34)$ the ceiling is approximately 26.7 rows (LP with an exact rational dual), 2.98% of the cohort. These expected-saving bounds do not exclude a profitable individual audit realization. Full cost: 4152.1 occupied GPU-seconds.

A disclosed deviation. The pre-registration fixed an every-row rule (the greedy boundary, hash-bound); the executed controller inherited the four-look rule from the parent protocol, and no review caught it. Since every candidate was then labelled in the committed order, the registered rule's decision follows exactly from recorded data: its boundary is valid at $(383, 99)$ (worst error $0.009987 \leq \alpha$), it certifies at row 357 ($s = 31$), skipping 26 candidates that hold 3 positives against an allowance of 6 — *correct*, and worth 72.9 GPU-seconds, 1.8% of the cost. It fires, is right, and does not pay. This is a counterfactual about a rule not executed, with no selection involved.