ALLOCATE BEYOND WHAT YOU SEE: CALIBRATED RESOURCE ALLOCATION UNDER ENDOGENOUS CENSORING

Anonymous authors
Paper under double-blind review

ABSTRACT

A deployed resource allocator censors its own training data: a job given too little memory or wall-clock is killed before finishing, so its true requirement is right-censored at the allocator's own decision, and calibrated prediction silently breaks under this feedback. We give three results. First, because the censoring level is a known decision, an exact finite-sample, distribution-free safe-allocation rule follows from a censored-implies-exceedance completion of split conformal, valid under exchangeability alone whenever the censored fraction is below the risk level. Second, deploying such a rule closes a loop that erases the upper tail. We prove that every support-confined estimator collapses to sub-nominal coverage, including a valid one-sided safe Kaplan–Meier band, whereas a completion that allocates beyond the observed support covers every round: conservatism inside the support cannot hold coverage, and what beyond-support reveals buy back is not coverage (the completion always covers) but non-vacuity. Third, a control arm restores identifiability and, via performative prediction, contracts retraining above f* = 1 $-$ 1/κ, which we measure on both a convex surrogate and the deployed gradient-boosted model. On a real 336K-job high-performance-computing (HPC) trace the method restores 95% per-partition coverage where naive conformal reaches 59–84% (a 16–41% out-of-memory rate, 3–8$\times$ the 5% budget; as low as 38% per partition), and an online adaptive-conformal-inference (ACI) margin reaches 95–97% on the deployed model. Deployed to general availability (80% treated, a permanent control arm above the required f^*), it holds treated out-of-memory at 0.06–0.09% across accrual windows (signature cluster-robust 95% CI far under the 5% guardrail ceiling; an absolute before/after safety result, the treated-versus-control gap unidentified on a quiet cluster), reclaiming terabytes of over-reserved memory, and reproduces a failure the theory predicts: a coverage-only margin recalibrates into the trivial allocate-the-request policy unless floored at an operating point.

1. INTRODUCTION

Systems that allocate a budget and then observe an outcome face a blind spot: when the budget is too small, the job does not merely under-perform, it is terminated, and the quantity the allocator wanted to predict, how much it actually needed, is never measured. A cluster scheduler that caps memory below a job's peak triggers an out-of-memory (OOM) kill; a wall-clock cap below the true runtime triggers a timeout kill. In both cases the log records only that the true requirement exceeded the allocation: the requirement is right-censored, and the censoring threshold is the allocator's own decision.

This structure is pervasive beyond high-performance computing (HPC): autoscalers OOM-kill pods at a memory limit, serverless platforms cap function memory and time, service-level-agreement (SLA) timeouts abort requests, and credit or budget limits truncate spend. In every case a decision sets the level at which the label becomes missing. We call this endogenous censoring (a function of the decision) to distinguish it from the exogenous censoring of survival analysis (dropout, end-of-study), where the censoring mechanism is independent of the decision-maker.

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.

Endogenous censoring is consequential for two reasons. First, the historical training data of any already-deployed allocator is censored by its own past policy, so a predictor fit naively on observed usage underestimates true requirements — a bias prior HPC and cloud predictors (3Sigma [12], PRIONN [13], Quasar [14], Pollux [15]) do not acknowledge. Second, deploying a predictor closes a feedback loop: its allocations censor the next round's labels, and retraining can spiral. Performative prediction [1] assumes labels are always observed; conformalized survival analysis [2] assumes censoring is exogenous. That forced exploration above the current allocation is needed to escape this feedback is long established, with regret/convergence proofs, in censored bandits [23] and Kaplan–Meier-based censored-demand inventory control [24]; Section 2 places us against the decision-aware-conformal and censored-newsvendor lineage. Our contribution is the distribution-free, coverage-flavored treatment for a deployed predictor: an exact finite-sample rule from the known threshold, a support-confinement impossibility that recasts forced exploration as a closed-loop coverage statement, and a control-arm performative stabilizer.

Contributions. (1) We formalize calibrated allocation under endogenous censoring, isolating the structural fact that the censoring level is the decision, hence known exactly at submit time (Lemma 1, Section 3). (2) A "censored-⇒-exceedance" completion of split-conformal gives an exact finite-sample, distribution-free safe-allocation bound under exchangeability alone, non-vacuous when the censored fraction is at most α (Propositions 1–2, Section 4.1), with a tighter Kaplan–Meier tier under an added condition (Theorem 1). (3) In the closed loop we prove a support-confinement impossibility (Theorem 2, Section 4.3): no within-support rule holds coverage, and not even a valid one-sided safe Kaplan–Meier confidence band escapes, because conservatism inside the observed support cannot buy back the truncated tail. This is the counterintuitive core, and it separates the online problem from the elementary static one. Escape holds iff one allocates beyond the observed support (Proposition 3; the completion covers unconditionally, so what these reveals buy back is non-vacuity, not coverage), unifying stochastic $+\infty$ completion, a control arm, and resubmission; a renewal-reward argument governs the collapse depth through $p_0 W$ (Proposition 6), and a controllable simulator shows the collapse is regime-dependent (routine resubmission re-injects the tail). (4) A control fraction f restores identifiability (Proposition 5) and, via performative prediction, contracts retraining for f > f* = $1 - 1/\kappa$ (Theorem 5); we measure κ on both the convex surrogate and the deployed boosted model (Section 6, Appendix D.1) and chart a cost/coverage frontier. Throughout we separate what is proved (the static guarantee; the convex-surrogate contraction) from what is demonstrated (the online collapse and its escape; κ; the frontier), reporting the latter with cross-shape robustness, seeds, and CIs. The contribution is machine-learning — distribution-free coverage and performative stability for a self-shaping predictor — with the deployment as impact evidence, not a scheduling result.

2. Related work

Conformal prediction. Split-conformal prediction gives distribution-free finite-sample coverage under exchangeability (Vovk et al. [5]; Lei et al. [6]). conformalized quantile regression (CQR [3]) adapts interval width to heteroscedasticity, adaptive conformal inference (ACI [4]) restores long-run coverage under shift via online feedback, and Mondrian conformal [7] targets conditional coverage. None address missing labels: conformity scores need the realized label, which censoring withholds.

Conformal prediction under censoring and shift. Conformalized survival analysis [2] gives calibrated bounds under exogenous censoring with an estimated distribution (e.g. Kaplan–Meier [8]); covariate-shift conformal [9] reweights under a shifted test covariate law. Our censoring is neither exogenous nor of unknown mechanism: it is the decision itself, known in closed form, which simplifies the static problem (Section 4.1) and creates the online pathology (Section 4.3) exogenous treatments do not exhibit. Both are closed-loop baselines (Appendix C.3): each collapses, and survival's inverse-probability-of-censoring-weighting (IPCW) correction backfires under endogenous censoring.

Performative prediction. Perdomo et al. [1] formalize decision-induced distribution shift and give contraction conditions (sensitivity ε, strong convexity γ, smoothness β) assuming fully-observed labels; a stateful-world variant [28] is closer to our online loop. We couple performativity with endogenous censoring and use a control arm to reduce the sensitivity and recover their contraction.

Decision-dependent censoring, censored exploration, and decision-aware conformal. An allocation censoring its own demand is the censored newsvendor problem [17], and observing the label only when the allocation is high enough is the apple-tasting / one-sided-feedback model of online learning [31]. That forced exploration above the current level is needed (while myopic within-support policies fail) is established with regret/convergence proofs in the dark-pool censored-bandit problem [23] and KM-based censored-demand inventory control [24], the ancestors of our beyond-support principle and KM tier, and the self-reinforcing collapse itself is the classical spiral-down effect of revenue management [32]. More broadly it is the selective-labels problem [25] and the runaway-feedback-loop pattern [26]. Concurrent work [18] names the same "low-provisioning trap" via an RL method, without a distribution-free guarantee or impossibility result; conformal work moves toward decisions and feedback (conformal decision theory [19]; performative/conformal risk control [20, 27]; feedback-based conformal prediction [21]). Our delta is the distribution-free, coverage-flavored treatment: an exact finite-sample bound from the known threshold and a support-confinement impossibility (Theorem 2), a coverage counterpart to the regret/convergence results of [23, 24]. Its cost face, Theorem 4, prices per-round coverage validity under rare tail information: [17, 23, 24] bound regret under censoring, conservative and stage-wise-constrained bandits [30, 33] price a reward budget under full feedback, the offline censored newsvendor has an analogous cost floor [34], whereas Theorem 4 shows per-round validity itself forces $\Theta(\Delta/\alpha)$ over-allocation. To our knowledge this is the first bound on a per-round coverage-validity constraint under decision-dependent censoring; being channel-independent (it holds under full observation, Appendix A), we claim no priority over the broader online valid-prediction literature.

Censored regression. Buckley–James [10] and censored quantile regression [11] de-bias point estimates under censoring but give no finite-sample coverage and ignore closed-loop dynamics; censored point-learning is the weakest of our levers.

HPC/cloud resource prediction. 3Sigma [12], PRIONN [13], Quasar [14], and Pollux [15] predict job resources from history; a recent uncertainty-aware HPC runtime predictor documents the same underestimation failure we target and flags conformal prediction as future work [22]. Production autoscalers such as Autopilot [29] right-size limits from usage history. All train on observed usage; none model that under-allocation censors the label or the feedback on a deployed predictor, and a benchmarked Autopilot-style right-sizer under-covers even at a 1.25× safety margin (Appendix C.3). Our method is complementary and could right-size any of them.

3. Problem setup

Data and decision. Jobs arrive with submit-time features $X \in \mathcal{X}$ and a latent true requirement $R \in \mathbb{R}$ (e.g. log peak memory or log runtime). A point model $\mu : \mathcal{X} \to \mathbb{R}$ and an allocation rule are fit on a disjoint split and held fixed; we work with residuals $r = R - \mu(X)$, taken exchangeable. The allocator picks an allocation $a = \mu(X) + Q$ before execution, so a is a function of X only.

Endogenous censoring. With allocation a we observe

$$Z = \min(R, a), \qquad \delta = \mathbf{1}\{R \le a\}.$$

$\delta = 1$: the job completes and R is seen exactly. $\delta = 0$: it is killed and we learn only $R > a$. In residual space the censoring level is $c = a - \mu(X)$, and we observe $\tilde{r} = \min(r, c)$ with $\delta = \mathbf{1}\{r \le c\}$.

Goal. A new rule $\hat{a}(\cdot) = \mu(\cdot) + \hat{Q}$ with a finite-sample safety guarantee

$$\Pr\left(R_{n+1} \le \hat{a}(X_{n+1})\right) \ \ge\ 1 - \alpha \qquad (\text{"OOM probability} \le \alpha\text{"}),$$

equivalently $\Pr(r_{n+1} \le \hat{Q}) \ge 1 - \alpha$: $\hat{Q}$ is a valid upper $(1-\alpha)$ prediction bound on the test residual. Among rules meeting the safety bound we prefer low over-allocation $\mathbb{E}[\,a - R \mid \delta = 1\,]$ (cost).

Lemma 1 (the censoring level is known and covariate-measurable).

Because $a = g(X)$ is chosen pre-execution with g, μ fixed, $c = g(X) - \mu(X)$ is a deterministic function of X; we know c_i for every calibration point and for the test point. Consequence: conditional on X, censoring is a fixed Type-I threshold carrying no information about r beyond X. This

is the structural advantage the endogenous setting hands us, and standard conformalized survival — which must estimate an unknown censoring law — does not have it.

The same known-threshold structure, iterated, is what makes the online loop pathological: when a is itself a quantile estimate, c tracks the estimand.

4. STATIC GUARANTEES AND THE ONLINE FAILURE

4.1 A DISTRIBUTION-FREE FINITE-SAMPLE SAFE ALLOCATION (TIER B)

Why does naive split-conformal fail? It would set $\hat{Q}$ to the $\lceil (1-\alpha)(n+1) \rceil$-th order statistic of the residuals, but under censoring the residual is unobserved when $\delta = 0$; substituting the truncated $\tilde{r} = \min(r, c)$ removes upper-tail mass, biasing $\hat{Q}$ low. Coverage drops below $1-\alpha$, monotonically in the censored fraction (Appendix C.1).

We instead complete censored points upward. Define the observable over-count of exceedances of a level t,

$$\bar{N}(t) = \sum_{i=1}^{n} \left[\mathbf{1}\{\delta_i = 0\} + \mathbf{1}\{\delta_i = 1\}\, \mathbf{1}\{r_i > t\} \right] = \sum_{i} \mathbf{1}\{r_i > \min(t, c_i)\},$$

i.e. count every censored calibration point as an exceedance (impute $r_i \leftarrow +\infty$ when $\delta_i = 0$). With $J = \lfloor \alpha(n+1) \rfloor - 1$, set

$$\hat{Q}^{\mathrm{cc}} = \inf\{t : \bar{N}(t) \le J\}, \qquad \hat{a}(x) = \mu(x) + \hat{Q}^{\mathrm{cc}}.$$

Proposition 1 (exact finite-sample, distribution-free).

Under exchangeability of $(X_i, R_i)_{i=1}^{n+1}$ and no independence/location assumption, $\Pr(R_{n+1} \le \hat{a}(X_{n+1})) \ge 1-\alpha$.

Proof (Appendix A): the $+\infty$-completion dominates the oracle exceedance count pointwise, so $\hat{Q}^{\mathrm{cc}} \ge Q^{\star}$ and the exchangeability rank bound gives $\Pr(r_{n+1} > \hat{Q}^{\mathrm{cc}}) \le \alpha$.

Proposition 2 (non-vacuousness, the price).

As $t \uparrow \infty$, $\bar{N}(t) \downarrow n_{\mathrm{cens}}$, so $\hat{Q}^{\mathrm{cc}} < \infty$ iff $n_{\mathrm{cens}} \le J$, i.e. roughly (censored fraction) $\lesssim \alpha$. When the deployed OOM rate is below the risk level (the operationally normal regime), the bound is finite and usable; otherwise it returns $+\infty$ (allocate maximally), which is safe but uninformative. On our cluster the measured pre-deployment OOM rate is 0.08–0.30% (memory) and 0.4–1.5% (runtime), all well below $\alpha = 0.05$, so Tier B is non-vacuous in precisely the regime that matters. The high-censoring regime in which it returns $+\infty$ is exactly where the beyond-support exploration of Section 5 is designed to act, so the static bound's vacuity there is the motivation for the online mechanism, not a competing failure: the two tiers are complementary.

4.2 A TIGHTER ASYMPTOTIC TIER (TIER A, KAPLAN–MEIER)

When the censored fraction is not small, a Kaplan–Meier (KM) [8] estimate of the residual survival S_r gives a sharper quantile, consistent under two added conditions: a marginal independence/location condition (A2) $r \perp c$ (Lemma 1 gives only conditional independence given X), and identifiability (A3) that uncensored mass lies above the target, $\Pr(c > q_{1-\alpha}) > 0$.

Theorem 1 (asymptotic KM coverage).

Under A2–A3, the KM quantile $\hat{Q}^{\mathrm{KM}} \xrightarrow{p} q_{1-\alpha}$ and $\Pr(R_{n+1} \le \mu(X_{n+1}) + \hat{Q}^{\mathrm{KM}}) \to 1-\alpha$. *(Proof: uniform KM consistency below the censoring horizon (cf. [16]) plus the conformal exchangeability argument on the consistent survival estimate.)*

Tier A is tight but assumption-laden; Tier B is robust but conservative. The static experiments (Appendix C.1) confirm both: KM is near-oracle at realistic censoring, Tier B is exact-and-finite below α.

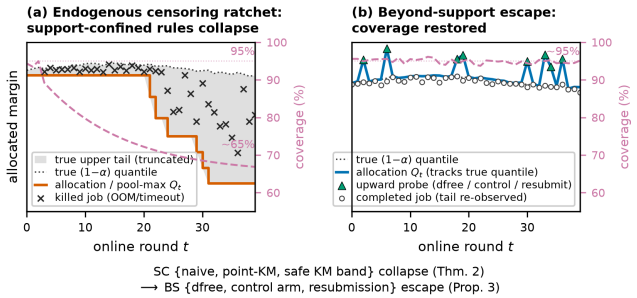


Figure 1: **The censoring ratchet (a) and its beyond-support escape (b)** (schematic). (a) A support-confined allocator (naive, point-KM, or a valid safe Kaplan–Meier band, all the descending step) sets its margin from its own pool; killed jobs feed back only as censored values, so the pool maximum ratchets down (Lemma 2) and coverage collapses (Theorem 2). (b) Any beyond-support lever (dfree, a control arm, or resubmission) occasionally allocates above the pool and re-reveals the tail, so the level tracks the true quantile and coverage holds (Proposition 3).

4.3 The online ratchet: a support-confinement impossibility

Figure 1 sketches the mechanism. Deploying either tier closes a loop. At round t the allocator sets a margin Q_t (an $\mathcal{F}_{t-1}$-measurable function of the recent window); a job with residual r_t is censored iff $r_t > Q_t$, so we observe $\tilde{r}_t = \min(r_t, Q_t)$ and no uncensored residual is ever recorded above the current allocation. Writing $M_t := \max\{\tilde{r}_i : i \in \text{window}_t\}$ for the pool maximum, the identifiability condition A3 self-destructs: $\Pr(c > q_{1-\alpha}) \to 0$ self-consistently, and the achievable proposal is pinned to the pool.

Call a rule support-confined (SC) if $\hat{Q}_t \le M_t$ almost surely (its proposal never exceeds the largest observed value), and beyond-support (BS) with strength p_0 if $\Pr(\hat{Q}_t > M_t \mid \mathcal{F}_{t-1}) \ge p_0$ infinitely often. Naive truncation, point KM, and the one-sided safe KM band are all SC. We now prove — not sketch — that SC rules cannot hold coverage. Assume residuals i.i.d. with continuous CDF F strictly increasing near $q := q_{1-\alpha}$, μ fixed, window W; write $\ell_\eta := F^{-1}(1-\alpha-\eta)$.

Lemma 2 (ratchet).

For any SC rule, $(M_t)_{t \ge W}$ is non-increasing; hence $M_t \downarrow M_\infty$ a.s. (Proof in Appendix A.)

Theorem 2 (a.s. failure of support-confined rules).

For any SC rule and any $\eta \in (0, 1-\alpha)$, almost surely there is a finite T with $\hat{Q}_t \le \ell_\eta$ and per-step coverage $F(\hat{Q}_t) \le 1-\alpha-\eta$ for all $t \ge T$. *(Proof, with the descent $M_\infty = \operatorname{essinf} r$ and a stationary/mixing extension: Appendix A.)*

The impossibility needs only support-confinement, a finite window, and exchangeable residuals (no independence, convexity, or tail bound). The safe KM band is a corollary — it is SC, so Theorem 2 covers it. Escape requires leaving the support:

Proposition 3 (escape; Appendix A).

For a BS rule with per-block strength $p_0 > 0$, reveals $r_t \in (M_t, \hat{Q}_t]$ recur infinitely often a.s., so Lemma 2 fails and identifiability is restored; ACI's long-run coverage identity then holds provided the rule can realize ACI's allocate-maximally boundary (dfree-cap does; a bare pool-max lift need not, Appendix A), needing only δ_t, and the reveals make the level $\hat{Q}_t$ track q.

Proposition 4 (Appendix A).

For a monotone base proposal (naive, point KM), long-run coverage $C(p_0)$ rises monotonically from $F(\operatorname{essinf} r)$ (at p_0=0) toward $1-\alpha$, its height set by the $p_0 W$ scaling of Proposition 6 (Appendix D.3).

Scope. Theorem 2 is the pure-SC ($p_0 = 0$) statement; a real stream is a BS rule with small p_0, so the observed 83%/59% plateau is residual leakage, not the theorem's zero prediction (Appendix A).

Theorem 3 (online validity of dfree; the converse of Theorem 2).

With i.i.d. residuals (continuous F) and exogenous arrivals, the fixed-level completion — $Q_t = \inf\{s : \bar{N}_t(s) \le J\}$ over the window (censored points counted as exceedances, Section 4.1), allocation $A_t = Q_t \wedge c_{\text{cap}}$ — satisfies $\Pr(r_t \le A_t) \ge 1 - \alpha - \rho_{\text{cap}}$ at every round $t \ge W+1$, against arbitrarily adaptive endogenous censoring: no independence of r and c, no reveal-rate condition, no identifiability ($\rho_{\text{cap}} := \Pr(r > c_{\text{cap}})$, zero for a dominating cap such as node memory). *(Proof and the conditional form: Appendix A.)*

So the loop splits cleanly: support-confined rules provably collapse (Theorem 2) while the completed rule provably covers every round (Theorem 3); the reveals and control fraction buy not coverage but non-vacuity (Proposition 2's price, online).

The cost face is a matching lower bound (**Theorem 4**, Appendix A): on a tail-ambiguity family, any rule holding per-round marginal coverage $\ge 1 - \alpha$ from round 1 — Theorem 3's included — pays $\Theta(\Delta/\alpha)$ expected over-allocation over the coverage-constrained oracle (tight; $\ge 2.2\Delta$ at $\alpha = 0.05$), so keeping coverage is not free. The driver is per-round validity meeting a rare tail (censoring is the operative setting, not the mechanism); time-average validity provably evades it.

5. Escaping the ratchet and stabilizing the loop

5.1 The beyond-support principle

The ratchet pins every within-support estimator to $\le \max(\text{pool})$: the only way to record a residual above the current allocation is to allocate above it and let the job complete. This dichotomy — support-confined rules collapse, the completed beyond-support rule holds — is Theorems 2 and 3 of Section 4.3, verified in Section 6.2. Two mechanisms realize beyond-support allocation:

- **Stochastic completion (dfree).** Proposition 1's $+\infty$ imputation makes $\hat{Q}^{\text{cc}} = +\infty$ whenever the windowed censored fraction exceeds $1 -$ level; the resulting over-allocation lets a job complete and re-reveals the tail. Deployed, $+\infty$ is replaced by a cap (a node/partition maximum), giving dfree_cap.
- **Deterministic exploration (control arm).** Allocate a fraction f of jobs at a generous, policy-independent level (e.g. the user's own request), so their true residuals, including tail values above $\max(\text{pool})$, are observed uncensored.

5.2 The control arm restores identifiability

Proposition 5 (control restores A3 unconditionally).

With control fraction $f > 0$ assigned independently of (X, R) at a policy-independent level, $\Pr(c > t) \ge f$ for all t. Hence A3 holds with margin $\ge f$ for any quantile and any policy: the loop cannot destroy identifiability. *(Proof: control jobs have $c = \infty$; an f-mass of $c = \infty$ enters the mixture.)*

The control level must be policy-independent: pegged to a quantile of the collapsing pool, control jobs are themselves under-allocated and reveal nothing. Honoring the user's own (over-)request satisfies this at near-zero marginal cost.

5.3 Retraining stability via performative prediction

Let μ_t also retrain on the loop's data. The observed-data law splits as a mixture $\mathcal{D}(\pi) = f\,\mathcal{D}_0 + (1 - f)\,\mathcal{D}_{\text{cens}}(\pi)$, where $\mathcal{D}_0$ (control) is policy-independent and only $\mathcal{D}_{\text{cens}}$ moves with the policy π. The control arm therefore linearly de-sensitizes the loop: its performative sensitivity is $\epsilon(f) = (1 - f)\,\epsilon_{\text{cens}}$.

Theorem 5 (contraction; convex surrogate).

If the per-round learner's risk is γ-strongly convex and β-smooth and the map is $\epsilon(f)$-sensitive, repeated retraining contracts with modulus $\rho(f) = (1-f)\,\beta\epsilon_{\text{cens}}/\gamma$ and converges to a unique performatively-stable policy iff

$$f > f^\star = 1 - \frac{\gamma}{\beta\,\epsilon_{\text{cens}}} = 1 - \frac{1}{\kappa}, \qquad \kappa := \frac{\beta\,\epsilon_{\text{cens}}}{\gamma}.$$

At the fixed point, combined with Prop. 5, coverage is maintained up to a slack $\varepsilon_{\text{cov}}(f) \downarrow 0$. *(Proof: plug $\epsilon(f) = (1-f)\,\epsilon_{\text{cens}}$ into the performative repeated-risk-minimization contraction of Perdomo et al. [1].)*

Strong convexity fails for the gradient-boosted learner we deploy, so Theorem 5 is rigorous only on a convex (linear-quantile) surrogate; for the boosted learner we instead measure the contraction directly, as the modulus of the iterated retraining map (Appendix D.1), and find it consistent with the surrogate's coarse dichotomy: the memory loop is contractive-or-marginal ($\kappa \approx 0.86$ 1), the runtime loop clearly diverges ($\kappa \approx 1.72$). The more sensitive the censoring, the larger the control fraction needed.

6. Experiments

Data. A real sacct (scheduler-accounting) trace from an HPC cluster serving a research center's diverse computational workloads, ~336K jobs over 18 months, split chronologically (train/calibration/test). Targets: peak memory (MaxRSS) and runtime, both in $\log$ space; per-partition (cpu, gpu-a, gpu-b, gpu-c). Point model μ: gradient-boosted trees on submit-time features (job name n-grams, partition, requested resources, cyclic time, user history). Unless noted, μ is fixed to isolate the conformal layer; risk level $\alpha = 0.05$ unless stated; online window $W = 1500$, ACI step $\gamma = 0.01$. Real-data tables report 95% bootstrap CIs with multi-split robustness (moving-block bootstrap, stratified-by-partition resampling, and K=5 rolling-origin folds; Appendix B). Coverage is evaluated on completed jobs, where the true requirement is observed (Appendix C.5 bounds the completed-only optimism at $\leq$0.4pp static and $\leq$0.2pp for the online loop).

6.1 Online collapse (Mondrian $\times$ ACI)

Statically, naive split-conformal under-covers monotonically as censoring rises, KM (Tier A) stays near-oracle, and Tier B (dfree) is exact and finite below α but vacuous above it, exactly as Proposition 2 predicts (Appendix C.1).

Running the same conformal layer in the closed loop, per-partition (Mondrian) with ACI, one-sided 95% target, naive and KM collapse to sub-nominal coverage (naive 83.5%/59.4% on memory/runtime, moving-block bootstrap $L = 1500$, B=300) and saturate ACI ($1 - \alpha_t \to 1.0$), while dfree (Tier B) holds 95% coverage on all 8 (partition $\times$ target) cells (95.6% memory, 95.4% runtime). These dfree figures are the dfree$\times$ACI composite (time-average guarantee, Appendix A); Theorem 3's exact per-round guarantee is the fixed-level dfree tier. Naive's implied kill rate is 16.5%/40.6% — 3–8$\times$ the 5% budget — versus dfree's $\approx$5%. The collapse is not a single-split artifact: naive misses target in all K=5 rolling-origin folds and dfree holds >94% in all (Appendix B); the sign is robust though runtime depth is not (the 59.4% point spans 32–75% across folds, sd 20pp, while dfree stays within $\leq$1pp). KM $\approx$ naive here: once the loop erases the tail, KM's event-weighting degenerates to $\max(\text{pool})$, refuting our hypothesis that KM would help on cost or conditional coverage. Figure 2 traces the rolling coverage; the full per-partition table is in Appendix C.2.

A controllable closed-loop simulator confirms the collapse across four residual tail shapes (including Gaussian, 8 seeds) and shows it is regime-dependent: a $\geq$25% resubmission rate restores coverage to 95%. A W-scaling sweep shows the plateau height is governed by $p_0 W$, and a newsvendor analysis prices the guarantee (Appendix D.2–D.3). The collapse-and-rescue is not specific to our cluster: reusing byte-identical estimator code on two public traces from the Parallel Workloads Archive (CEA-Curie runtime, 280K jobs, four queues; HPC2N memory, 202K jobs), naive and KM fall to 12–54% coverage where censoring bites while dfree holds 94.6–96.5% on every cell (Appendix D.4).

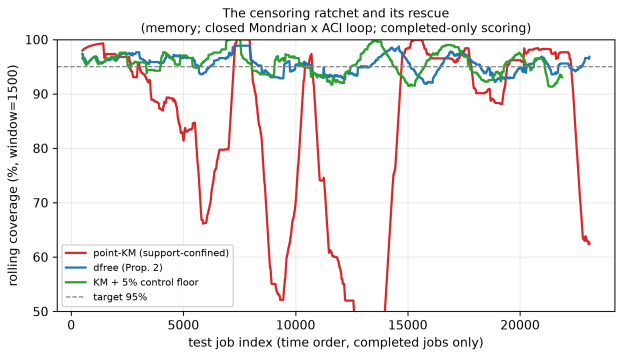


Figure 2: **The censoring ratchet and its rescue** (rolling coverage, memory, closed Mondrian×ACI loop). The support-confined point-KM rule (red) repeatedly collapses far below the 95% target (to 50–65%) as the loop erases the upper tail, whereas dfree (blue, $+\infty$ completion) and a 5% control floor (green), both of which allocate beyond the observed support, hold coverage throughout.

6.2 The beyond-support principle and the control sweep

The one-sided safe KM band, a strictly more conservative within-support estimator, collapses identically to naive/KM (83.0%/59.5%): conservatism within the observed pool cannot escape the ratchet. A control-fraction sweep shows f=0 reproduces the collapse and f $\approx$ 0.02 restores 95% on both targets. Cited baselines confirm the dichotomy: conformalized survival [2] collapses harder than naive (39.6%/17.7%, its IPCW correction backfiring on the allocator's own policy); censored quantile regression [11] reduces to the KM tier; and weighted conformal [9] only appears to hold 95% (a $+\infty$ point-mass artifact; disabling it collapses to 81.8%/56.6%). A parametric peaks-over-threshold GPD tail — naive or censoring-aware — also stays sub-nominal (83.7%/60.1% and 75.8%/61.2%): the loop censors the very tail such a fit needs, so extrapolation collapses or turns erratic, never reaching dfree's 95%. Full table and EVT analysis: Appendix C.3.

The retraining contraction modulus is $\kappa \approx 1.2$ (memory), ≈ 2.0 (runtime) on the convex surrogate where Thm 3 is rigorous, and the same coarse split on the deployed boosted learner (iterated map): $\kappa \approx 0.86$ (memory, contractive-or-marginal) vs. ≈ 1.72 (runtime, clearly diverging at the deployed 20% control). Runtime thus needs a larger retraining control fraction (f*$\approx$0.5) than the conformal-layer f*$\approx$0.02, so for heavy-tailed targets we freeze μ and adapt only the conformal margin (Appendix D.1).

6.3 The deployable cost/coverage frontier

Among strategies that hold 95% on both targets, with physical caps (node-max memory; 30-day runtime):

strategy	mem cov / med / p95	runtime cov / med / p95
dfree_cap (physical)	95.6% [94.6, 96.0] / **2.65**× / 35×	95.4% [94.5, 95.7] / **4.80**× / 1132×
control=user-request (physical)	96.2% [94.6, 97.1] / 4.94× / **23**×	95.5% [94.0, 96.5] / 6.75× / **217**×

Coverage as point [95% CI]; both bracket the target. Both rows are the ACI-adapted composite (time-average guarantee of Appendix A); Theorem 3's fixed-level per-round tier trades a little extra median cost for the stronger every-round bound. The split is clean: dfree_cap minimizes median cost, the control arm the p95 tail (its allocations are bounded by the user's request). A small control floor plus capped-dfree combines their strengths.

The frontier is plotted in Figure 3 (Appendix C.3).

6.4 LIVE DEPLOYMENT: SHADOW VALIDATION AND A CLOSED-LOOP RANDOMIZED A/B

On the deployed gradient-boosted learner an online ACI margin restores memory coverage 88.2%→97.1% (at lower cost) and holds runtime at 95.0% on the held-out split (Appendix C.4), so the mechanism is not limited to Theorem 5's convex surrogate; this ACI layer is the empirical, drift-tracking variant of Theorem 3's fixed-level per-round tier (time-average bound, Appendix A). Crucially this is drift-correction, not ratchet-rescue: pre-deployment censoring is <1.5% < α, so the ratchet stays dormant in production and the restoration is ACI tracking an exogenous validation→test shift. The collapse-and-rescue itself is evidenced by the simulator, the two public traces, and the impossibility/cost theorems (Sections 4.3, 6.1, Appendix D), and reproduced live in a self-submitted closed-loop demonstration with real memory kills — a support-confined arm ratchets to ≈80% while dfree holds ≈97% (faithful enforcement, Appendix E) — at milder depth than the trace, whose scale carries the powered severity.

We ran the hardened allocator on a production HPC cluster in two phases; Appendix E gives the full narrative and three deployment-only failure modes, each fixed (a non-slack-memory OOM, the self-inerting coverage margin, and an ITT monitor confound). A mutation-free shadow (11 days, N=431) reproduced the held-out safety target on live traffic (would-be OOM 2.3% memory / 0.0% runtime). After hardening, a pre-registered closed-loop A/B (staged ramp, permanent policy-independent control arm, frozen μ, automatic rollback) reached general availability (80% treated, $\geq$20% control): on the matched-cuttable stratum treated OOM is 0.06–0.09%, signature cluster-bootstrap 95% CI [0.00%, 0.40%] (≈12× under the 5% ceiling), no guardrail trip, reclaiming ≈5.6 TB at a 27% mean cut. The treatment-versus-control gap is not identified offline (its CI spans zero; one control signature carries 73% of control jobs; Appendix E), so a live switchback remains the gold standard.

7. DISCUSSION AND LIMITATIONS

- **Convex surrogate.** Theorem 5 is rigorous only on the linear-quantile surrogate; for the deployed boosted learner we measure the retraining contraction directly (memory $\kappa \approx 0.86$, runtime $\kappa \approx 1.72$ at the deployed 20% control; Appendix D.1). The combined contraction stays a conjecture for non-convex learners, though its qualitative prediction holds on the deployed model.
- **User-adaptation.** We fix the requirement law; requests changing in response to allocations (a Goodhart risk) go unmodeled, but an observational arm-contrast finds no inflation by treated users (bounded null; Appendix E), so a switchback would identify ϵ_{user} cleanly. On the deployed memory learner $\kappa \approx 0.86$ gives $f^* = 0$, so the 20% control arm's live role is identifiability, not retraining contraction.
- **Non-slack resources.** Memory right-sizing is safe only where the reservation carries slack: a live OOM of a full-node-GPU job forced passthrough rules and a never-below-peak floor (Appendix E), so the coverage rule must be composed with a slack model before it mutates a reserved resource.
- **Survivorship.** ~25% of trace jobs are non-completed, so completed-only evaluation is optimistic; an evidence-gated lower bound puts that optimism at $\leq$0.4pp with the dfree$\geq$naive ordering intact (Appendix C.5).

8. CONCLUSION

A decision that sets a budget censors the very quantity it wants to learn. We gave an exact distribution-free safe-allocation rule from the known threshold, proved the online loop erases the upper tail so calibration requires allocating beyond the observed support, and stabilized retraining with a control arm at f* = $1 - 1/\kappa$. The collapse is regime-dependent; we conjecture the mechanism transfers to autoscaling, serverless, and SLA-bounded systems where retries are costly (a directional Azure Functions probe, Appendix D.4, shows milder severity).

STATEMENTS

Data availability. Experiments use operational scheduler (sacct) logs from a research computing cluster (aggregate, opaque user identifiers); the raw logs are governed by cluster policy. Aggregated derived data, the full analysis code, and the controllable simulator are provided as an anonymized artifact.

Reproducibility. Proofs are in Appendix A; confidence-interval, seed, and rolling-fold configurations in Appendix B; simulator, κ, and master-curve configurations in Appendix D; the deployment design in Appendix E. The method code, the controllable simulator (Appendix D.2–D.3), and every synthetic-data table reproduce from the released anonymized artifact under fixed seeds and configurations. The trace-based and live-deployment tables depend on the non-public raw logs and so are reproducible only under cluster policy, but their aggregated inputs and the full analysis code are released. The Section 6.1 collapse-and-rescue is additionally reproduced end-to-end on two fully public HPC traces (Parallel Workloads Archive: CEA-Curie and HPC2N) using the released SWF adapter and the unmodified conformal harness (Appendix D.4), so the central empirical finding does not depend on the proprietary trace.

Ethics. No human-subjects data beyond opaque user identifiers. The live A/B includes per-user opt-out, a permanent policy-independent control arm, interactive/full-node-GPU passthrough guards, and an automatic safety rollback, and runs with administrator/PI sign-off.

AI use statement. In line with the ICLR 2027 AI Policy for Authors, we disclose the following. Generative AI tools were used to assist with implementing the analysis and evaluation code, designing, orchestrating, and running the experiments, drafting and editing the exposition, formatting and checking citations, and preparing figures, and with developing and checking the mathematical proofs and interpreting results. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

Author contributions / funding / conflicts. To be completed at submission.

REFERENCES

[1] Perdomo, J. C., Zrnic, T., Mendler-Dünner, C., Hardt, M. *Performative Prediction.* ICML 2020, PMLR 119:7599–7609.

[2] Candès, E., Lei, L., Ren, Z. *Conformalized Survival Analysis.* J. R. Stat. Soc. B 85(1):24–45, 2023. doi:10.1093/jrsssb/qkac004.

[3] Romano, Y., Patterson, E., Candès, E. *Conformalized Quantile Regression.* NeurIPS 2019. (confirmed)

[4] Gibbs, I., Candès, E. *Adaptive Conformal Inference Under Distribution Shift.* NeurIPS 2021. (confirmed)

[5] Vovk, V., Gammerman, A., Shafer, G. *Algorithmic Learning in a Random World.* 2nd ed., Springer, 2022. doi:10.1007/978-3-031-06649-8. (1st ed. 2005, doi:10.1007/b106715.)

[6] Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J., Wasserman, L. *Distribution-Free Predictive Inference for Regression.* JASA 113(523):1094–1111, 2018. doi:10.1080/01621459.2017.1307116.

[7] Vovk, V., Gammerman, A., Shafer, G. *Modifications of Conformal Predictors* (Ch. 4, pp. 107–142 — Mondrian / group-conditional conformal prediction), in *Algorithmic Learning in a Random World*, 2nd ed., Springer, 2022. doi:10.1007/978-3-031-06649-8_4.

[8] Kaplan, E. L., Meier, P. *Nonparametric Estimation from Incomplete Observations.* JASA 53(282):457–481, 1958. doi:10.1080/01621459.1958.10501452.

[9] Tibshirani, R. J., Barber, R. F., Candès, E. J., Ramdas, A. *Conformal Prediction Under Covariate Shift.* NeurIPS 2019. arXiv:1904.06019.

[10] Buckley, J., James, I. *Linear Regression with Censored Data.* Biometrika 66(3):429–436, 1979. doi:10.1093/biomet/66.3.429.

[11] Portnoy, S. *Censored Regression Quantiles.* JASA 98(464):1001–1012, 2003. doi:10.1198/016214503000000954.

[12] Park, J. W., Tumanov, A., Jiang, A., Kozuch, M. A., Ganger, G. R. *3Sigma: Distribution-Based Cluster Scheduling for Runtime Uncertainty.* EuroSys 2018. doi:10.1145/3190508.3190515.

[13] Wyatt, M. R., Herbein, S., Gamblin, T., Moody, A., Ahn, D. H., Taufer, M. *PRIONN: Predicting Runtime and IO using Neural Networks.* ICPP 2018. doi:10.1145/3225058.3225091.

[14] Delimitrou, C., Kozyrakis, C. *Quasar: Resource-Efficient and QoS-Aware Cluster Management.* ASPLOS 2014, pp. 127–144. doi:10.1145/2541940.2541941.

[15] Qiao, A., Choe, S. K., Subramanya, S. J., Neiswanger, W., Ho, Q., Zhang, H., Ganger, G. R., Xing, E. P. *Pollux: Co-adaptive Cluster Scheduling for Goodput-Optimized Deep Learning.* OSDI 2021.

[16] Stute, W., Wang, J.-L. *The Strong Law under Random Censorship.* Ann. Statist. 21(3), 1993. doi:10.1214/aos/1176349273.

[17] Besbes, O., Muharremoglu, A. *On Implications of Demand Censoring in the Newsvendor Problem.* Management Science 59(6), 2013. doi:10.1287/mnsc.1120.1654. (Alternative canonical censored-newsvendor reference: Huh, W. T., Rusmevichientong, P., *Math. of OR* 34(1), 2009, doi:10.1287/moor.1080.0355.)

[18] Chen, Y., Huang, R., Liu, C. *Decision Support under Prediction-Induced Censoring.* arXiv:2602.18031, Feb 2026. **Closest concurrent work**: same "low-provisioning trap" in resource allocation, but an RL decision-support method with a selection-bias-correction guarantee — no distribution-free coverage bound and no impossibility result.

[19] Lekeufack, J., Angelopoulos, A. N., Bajcsy, A., Jordan, M. I., Malik, J. *Conformal Decision Theory: Safe Autonomous Decisions from Imperfect Predictions.* ICRA 2024. doi:10.1109/ICRA57147.2024.10610041. arXiv:2310.05921.

[20] Li, V., Chen, B., Mao, Y., Lei, Q., Deng, Z. *Performative Risk Control: Calibrating Models for Reliable Deployment under Performativity.* NeurIPS 2025. arXiv:2505.24097.

[21] Wang, H., Ning, C. *Conformal Prediction in The Loop: A Feedback-Based Uncertainty Model for Trajectory Optimization.* NeurIPS 2025 (Main Track). arXiv:2510.16376.

[22] Choi, J., Oh, S. *UARP: Uncertainty-Aware Runtime Prediction for Preventing Scheduler Termination under Wallclock Constraints in HPC.* J. Supercomputing 82(5), 2026. doi:10.1007/s11227-026-08422-8. (Documents the HPC wall-clock underestimation failure and flags conformal prediction as future work.)

[23] Ganchev, K., Kearns, M., Nevmyvaka, Y., Wortman Vaughan, J. *Censored Exploration and the Dark Pool Problem.* UAI 2009, pp. 185–194. (Journal version: *Communications of the ACM* 53(5):99–107, 2010. doi:10.1145/1735223.1735247. arXiv:1205.2646.)

[24] Huh, W. T., Levi, R., Rusmevichientong, P., Orlin, J. B. *Adaptive Data-Driven Inventory Control with Censored Demand Based on Kaplan–Meier Estimator.* Operations Research 59(4):929–941, 2011. doi:10.1287/opre.1100.0906.

[25] Lakkaraju, H., Kleinberg, J., Leskovec, J., Ludwig, J., Mullainathan, S. *The Selective Labels Problem: Evaluating Algorithmic Predictions in the Presence of Unobservables.* KDD 2017, pp. 275–284. doi:10.1145/3097983.3098066.

[26] Ensign, D., Friedler, S. A., Neville, S., Scheidegger, C., Venkatasubramanian, S. *Runaway Feedback Loops in Predictive Policing.* Conference on Fairness, Accountability and Transparency (FAT 2018), PMLR 81:160–171. arXiv:1706.09847.

[27] Angelopoulos, A. N., Bates, S., Fisch, A., Lei, L., Schuster, T. *Conformal Risk Control.* ICLR 2024. arXiv:2208.02814.

[28] Brown, G., Hod, S., Kalemaj, I. *Performative Prediction in a Stateful World.* AISTATS 2022, PMLR 151:6045–6061. arXiv:2011.03885.

[29] Rzadca, K., Findeisen, P., widerski, J., Zych, P., Broniek, P., Kumierek, J., Nowak, P., Strack, B., Witusowski, P., Hand, S., Wilkes, J. *Autopilot: Workload Autoscaling at Google.* EuroSys 2020. doi:10.1145/3342195.3387524.

[30] Wu, Y., Shariff, R., Lattimore, T., Szepesvári, C. *Conservative Bandits.* ICML 2016, PMLR 48:1254–1262. (T-free additive price of a hard budget constraint in a fully-observed bandit; the shape-precedent for Theorem 4's cost bound.)

[31] Helmbold, D. P., Littlestone, N., Long, P. M. *Apple Tasting.* Information and Computation 161(2):85–139, 2000. doi:10.1006/inco.2000.2870. (The canonical one-sided-feedback online-learning model: the label is observed only when the instance is accepted — here, only when the allocation is high enough to let the job complete.)

[32] Cooper, W. L., Homem-de-Mello, T., Kleywegt, A. J. *Models of the Spiral-Down Effect in Revenue Management.* Operations Research 54(5):968–987, 2006. doi:10.1287/opre.1060.0304. (The classical self-reinforcing forecast-collapse whose coverage analogue is our censoring ratchet.)

[33] Pacchiano, A., Ghavamzadeh, M., Bartlett, P. *Contextual Bandits with Stage-wise Constraints.* Journal of Machine Learning Research 26, 2025. (Stage-wise / per-round constrained bandits with a minimax lower bound; an expected-reward baseline constraint under full feedback, without censoring or a coverage constraint.)

[34] Hssaine, C., Sinclair, S. R. *The Data-Driven Censored Newsvendor Problem.* arXiv:2412.01763, 2024. (Offline lower bound / cost floor from demand censoring — the closest cost-framed cousin to Theorem 4, without a per-round validity constraint.)

Appendix A. Proofs for Section 4

Proof of Proposition 1 (Section 4.1; exact finite-sample, distribution-free). Let $N(t) = \#\{i \le n : r_i > t\}$ be the unobserved oracle count and $Q^\star = \inf\{t : N(t) \le J\}$ the standard split-conformal residual quantile; by the exchangeability rank argument $\Pr(r_{n+1} > Q^\star) \le \alpha$. Pointwise $\bar{N}(t) \ge N(t)$ (a censored point contributes $1 \ge \mathbf{1}\{r_i > t\}$; a completed point contributes equally), so $\{t : \bar{N}(t) \le J\} \subseteq \{t : N(t) \le J\}$ and taking infima over a subset can only raise the value: $\hat{Q}^{\mathrm{cc}} \ge Q^\star$ as a sample-path inequality. Hence $\{r_{n+1} > \hat{Q}^{\mathrm{cc}}\} \subseteq \{r_{n+1} > Q^\star\}$ and $\Pr(r_{n+1} > \hat{Q}^{\mathrm{cc}}) \le \alpha$. $\hat{Q}^{\mathrm{cc}}$ uses calibration data only, so the bound holds unconditionally.

Proof of Theorem 3 (Section 4.3; online per-round validity of dfree). Fix $t \ge W+1$ and condition on $\mathcal{G}_t := \sigma\big(r_1, \dots, r_{t-W-1}, (\xi_s)_{s\ge 1}\big)$, the pre-window past together with all internal randomness (independent of the residual stream). Domination. Let $N_t(s) = \#\{i \in [t-W, t-1] : r_i > s\}$ be the oracle exceedance count of the true window residuals and $Q_t^\star = \inf\{s : N_t(s) \le J\}$ its split-conformal quantile. A censored calibration point ($\delta_i = 0$) contributes $1 \ge \mathbf{1}\{r_i > s\}$ to $\bar{N}_t(s)$ and a completed point contributes $\mathbf{1}\{\tilde{r}_i > s\} = \mathbf{1}\{r_i > s\}$, so $\bar{N}_t(s) \ge N_t(s)$ on every sample path, whatever the adaptive levels $c_i = Q_i$ were; hence $Q_t \ge Q_t^\star$ pathwise, and $Q_t^\star$ is a symmetric function of $(r_{t-W}, \dots, r_{t-1})$ that never sees the c_i. Rank bound. Given $\mathcal{G}_t$ the vector $(r_{t-W}, \dots, r_t)$ is i.i.d. F, so the rank of r_t is uniform and $\Pr(r_t > Q_t^\star \mid \mathcal{G}_t) \le \lfloor \alpha(W+1) \rfloor / (W+1) \le \alpha$. With $A_t = Q_t \wedge c_{\mathrm{cap}}$ we have $\{r_t > A_t\} \subseteq \{r_t > Q_t^\star\} \cup \{r_t > c_{\mathrm{cap}}\}$, and $r_t \perp \mathcal{G}_t$ gives $\Pr(r_t > A_t \mid \mathcal{G}_t) \le \alpha + \rho_{\mathrm{cap}}$ a.s.; the marginal bound follows by taking expectations, and exchangeability alone suffices for it. The argument uses no independence of c_i from r and no reveal-rate condition; it does need exogenous arrivals, since decision-dependent re-arrival (resubmission) breaks the exchangeability of the stream, and it is stated for the fixed level α — the shared-window ACI composite of Section 6 is not $\mathcal{G}_t$-measurable and is covered instead by the deterministic time-average bound (below).

ACI composition (deployed layer, time-average). Running dfree-cap at the ACI-adapted level α_t ($\alpha_{t+1} = \alpha_t + \gamma(\alpha - \mathrm{err}_t)$, $\mathrm{err}_t = \mathbf{1}\{r_t > A_t\}$ observed every round by Lemma 1) telescopes to $\big|\frac{1}{T}\sum_{t\le T} \mathrm{err}_t - \alpha\big| \le (\max\{\alpha_1, 1-\alpha_1\} + \gamma)/(\gamma T)$, hence time-average coverage $\ge 1 - \alpha - (1 +$

$\gamma)/(\gamma T)$, provided $r_t \le c_{\mathrm{cap}}$ a.s. so that dfree-cap realizes ACI's allocate-maximally boundary (this is where a finite runtime cap leaks: with $\rho_{\mathrm{cap}} > 0$ the level is not confined and the bound degrades). This is the Gibbs–Candès identity [4]; the only endogenous-censoring content is that err_t stays observable and that a support-confined rule cannot realize the boundary (its maximal allocation is $M_t < \operatorname{esssup} r$), which is the precise sense in which Theorem 2 and this identity cannot coexist for SC rules — correcting the "holds regardless" phrasing of Proposition 3.

Theorem 4 (the price of validity; cost face of Theorem 2).

Fix $\alpha \le \frac{1}{20}$, $\Delta, u > 0$. On the family $F_\theta = (1-\alpha')G + \alpha' \,\mathrm{Unif}[u{+}\theta, u{+}\theta{+}h]$ ($\theta \in [0, \Delta]$, $\alpha' \approx 3.51\alpha$, agreeing below u), any rule with per-round marginal coverage $\ge 1-\alpha$ from round 1 incurs expected cumulative over-allocation excess over the coverage-constrained oracle $\ge (\Delta{-}h)/(12\alpha)$ (exactly $\ge 2.2\Delta$ at $\alpha = 0.05$), under $\theta = 0$, within its first $O(1/\alpha)$ rounds; a two-phase rule matches it (minimax $\Theta(\Delta/\alpha)$, sharpening to $(\Delta{-}h)/\alpha$ under conditional coverage). The bound uses only per-round validity and tail-rarity — it holds under full observation, with decision-dependent censoring the operative setting — and time-average validity evades it.

Proof of Theorem 4 (Section 4.3; the price of validity). Coupling. On the event $\{\tau \ge t\}$ that no tail has arrived by round t (arrivals are exogenous, so $\Pr(\tau \ge t) = (1-\alpha')^{t-1}$ under every θ), all observations are bulk draws from G, identical in law across the family; hence a_t has the same conditional law under $\theta = 0$ and $\theta = \Delta$: $\Pr_\Delta(a_t < u{+}\Delta, \tau \ge t) = \Pr_0(a_t < u{+}\Delta, \tau \ge t)$. Forcing. Marginal coverage at $\theta = \Delta$ gives $\alpha \ge \Pr_\Delta(r_t > a_t) \ge \alpha' \Pr_\Delta(a_t < u{+}\Delta)$, so $\Pr_\Delta(a_t < u{+}\Delta) \le \beta := \alpha/\alpha'$; with the coupling, $\Pr_0(a_t \ge u{+}\Delta, \tau \ge t) \ge (1-\alpha')^{t-1} - \beta$. Aggregation. Under $\theta = 0$ the oracle is $q_0 = u$, so $(a_t - u)^+ \ge \Delta \mathbf{1}\{a_t \ge u{+}\Delta\}$ and $\mathrm{Exc}_T(\pi; 0) \ge \Delta \sum_t [(1-\alpha')^{t-1} - \beta]^+$. Taking $\alpha' = \kappa^*\alpha$ with $\kappa^* \approx 3.513$ the root of $\kappa = 1 + 2\ln\kappa$ maximizes the sum at $\ge c^*(1 - \kappa^*\alpha)\Delta/\alpha$, $c^* = (\kappa^* - 1)/(2\kappa^{*2}) \approx 0.102$, i.e. $\ge \Delta/(12\alpha)$ for $\alpha \le 0.05$; a sequence-form LP over all adaptive randomized rules confirms the bound is attained (tight). The argument uses only per-round validity and tail-rarity, so it holds under full observation as well; the "from round 1" hypothesis is load-bearing (a warm-up costed above $u{+}\Delta$ is the only escape, which is why Theorem 3's $t \ge W{+}1$ rule must be read as costing its warm-up). The conditional-coverage sharpening to $(\Delta{-}h)/\alpha'$ and the matching two-phase rule follow, and a probe rule shows time-average (limsup) validity attains $o(1)$ excess, evading the bound — pricing the guarantee class, not any particular algorithm. The indistinguishability device is standard [17, 23].

Setup and notation (Section 4.3). Residuals $r_1, r_2, \ldots$ are i.i.d. with continuous CDF F, strictly increasing on a neighborhood of $q := F^{-1}(1-\alpha)$, $\alpha \in (0, \frac{1}{2})$; $S = 1 - F$ and $\ell_\eta := F^{-1}(1 - \alpha - \eta)$ for $\eta \in (0, 1-\alpha)$, so $F(\ell_\eta) = 1 - \alpha - \eta$. The margin Q_t is $\mathcal{F}_{t-1}$-measurable, $\mathcal{F}_{t-1} := \sigma(r_1, \ldots, r_{t-1}, \text{internal randomness through } t{-}1)$, with $r_t \perp \mathcal{F}_{t-1}$. We observe $\tilde{r}_t = \min(r_t, Q_t)$ and $\delta_t = \mathbf{1}\{r_t \le Q_t\}$; per-step coverage $c_t = F(Q_t)$. Pool $\mathcal{W}_t = \{\tilde{r}_{t-W}, \ldots, \tilde{r}_{t-1}\}$, $M_t = \max \mathcal{W}_t$. A rule is support-confined (SC) if $Q_t \le M_t$ a.s.; beyond-support (BS) with per-block strength p_0 if $\Pr(Q_t > M_t \mid \mathcal{F}_{t-1}) \ge p_0$ infinitely often. The one fact used throughout: $\tilde{r}_t = \min(r_t, Q_t) \le r_t$ unconditionally.

Lemma 2 (ratchet).

For any SC rule, $(M_t)_{t \ge W}$ is non-increasing; hence $M_t \downarrow M_\infty \in [\operatorname{essinf} r, \infty)$ a.s.

Proof. For $t \ge W$ the window is full; $\mathcal{W}_{t+1}$ removes $\tilde{r}_{t-W}$ and inserts $\tilde{r}_t \le Q_t \le M_t$ (structural fact then SC). Every retained element is $\le M_t$ by definition of the maximum, and the inserted element is $\le M_t$, so $M_{t+1} \le M_t$. Monotone and bounded below by $\operatorname{essinf} r$, hence convergent. (During warm-up $t < W$ the pool is still filling and M_t may rise; the ratchet is a full-window statement.)

Theorem 2 (a.s. failure of support-confined rules).

For any SC rule and any $\eta \in (0, 1-\alpha)$,

$$\Pr\left(\exists\, T < \infty :\ Q_t \le \ell_\eta \text{ and } c_t \le 1 - \alpha - \eta \;\forall\, t \ge T\right) = 1,$$

so the target is missed a.s.; moreover $M_\infty = \operatorname{essinf} r$ a.s. (coverage $\to 0$ if r is unbounded below).

Proof. Fix η and let $A_t := \{r_t, \dots, r_{t+W-1} \le \ell_\eta\}$. On A_t, each of the W observed values inserted over rounds $t, \dots, t+W-1$ satisfies $\tilde{r} = \min(r, Q) \le r \le \ell_\eta$, so $M_{t+W} \le \ell_\eta$ regardless of the intervening Q-path. By i.i.d.-ness $\Pr(A_t \mid \mathcal{F}_{t-1}) = F(\ell_\eta)^W = (1-\alpha-\eta)^W =: p > 0$. Over the disjoint blocks $t \in \{1, W{+}1, 2W{+}1, \dots\}$ the A_t are independent with $\sum_t \Pr(A_t) = \infty$, so by Borel–Cantelli some A_t occurs a.s. Then Lemma 2 gives $M_s \le \ell_\eta$ for all $s \ge t+W$, and SC gives $Q_s \le M_s \le \ell_\eta$, whence $c_s = F(Q_s) \le 1-\alpha-\eta$; take $T = t+W$. For the descent, intersect over rational $\eta \uparrow (1-\alpha)$: a.s. $M_\infty \le \ell_\eta$ for every such η, so $M_\infty \le \inf_\eta \ell_\eta = \operatorname{essinf} r$; the reverse is trivial.

Proposition 3 (escape under beyond-support).

For a BS rule with per-block strength p_0: reveals $\{r_t \in (M_t, Q_t]\}$ occur infinitely often a.s.; ACI's long-run coverage identity $\frac{1}{T}\sum_t \mathbf{1}\{r_t > Q_t\} \to \alpha$ holds; and the level Q_t identifies q.

Proof. Conditional on the BS event $\{Q_t > M_t\}$ (probability $\ge p_0$), $r_t \in (M_t, Q_t]$ with probability $F(Q_t) - F(M_t) > 0$; by Borel–Cantelli such reveals occur i.o., so Lemma 2's monotonicity fails and $\{M \le q\}$ is not absorbing. The coverage identity is Gibbs–Candès [4] and uses only the observed miss indicator δ_t — it requires neither identifiability nor a reveal every step. Identifiability, restored by the i.o. reveals, is what makes the level track q (the score law above M_t becomes estimable). A per-block $p_0 > 0$ suffices.

Proposition 4 (monotone collapse in the reveal rate).

Let $C(p_0) := \lim_T \frac{1}{T}\sum_{t \le T} c_t$ be the long-run per-step coverage of a BS rule of strength p_0. Suppose the base proposal is monotone under element-wise domination of the pool $\mathcal{W}_t$ (dominating every recorded value dominates Q_t) — true for naive truncation $Q_t = \mathrm{Quant}_{1-\alpha}(\mathcal{W}_t)$ and for point KM. Then (i) $C(p_0)$ is non-decreasing in p_0; (ii) $F(\operatorname{essinf} r) \le C(p_0) \le 1-\alpha$; and (iii) $C(p_0) \to F(\operatorname{essinf} r)$ as $p_0 \to 0$.

Proof. (i) Couple two streams with strengths $p_0 \le p_0'$ on the same residuals and internal randomness, thinning the strength-p_0' reveals to strength p_0. Induct: each recorded value satisfies $\tilde{r}_t \le \tilde{r}_t'$ — retained window elements are dominated by hypothesis, $\tilde{r}_t = \min(r_t, Q_t) \le \min(r_t, Q_t') \le \tilde{r}_t'$ since $Q_t \le Q_t'$ by pool domination, and at a primed-only reveal the primed value is larger still. Hence $M_t \le M_t'$, $Q_t \le Q_t'$, and $c_t = F(Q_t) \le F(Q_t')$ pathwise, so $C(p_0) \le C(p_0')$. (For point KM: turning a censored point into a larger uncensored one and enlarging any observation each shift the KM survival curve right, so its quantile is domination-monotone under the joint $(\tilde{r}, \delta)$ coupling.) (ii) The lower rail is $C(0) = F(\operatorname{essinf} r)$ (Theorem 2) with (i); the upper rail is the ACI target — the online level aims at $1-\alpha$ and does not overshoot in the long run. (iii) Fix η. Between consecutive reveals Lemma 2 is in force, so a clean W-run (event A_t, probability $(1-\alpha-\eta)^W$) forces $M \le \ell_\eta$ and, absent a reveal, holds there; the expected number of steps before the first clean run is finite and independent of the inter-reveal gap, so as $p_0 \to 0$ (gaps $\to \infty$) the fraction of steps with $c_t \le 1-\alpha-\eta$ tends to 1, giving $\limsup_{p_0 \to 0} C(p_0) \le 1-\alpha-\eta$. Intersecting over rational $\eta \uparrow (1-\alpha)$ yields $C(p_0) \to F(\operatorname{essinf} r)$. Remark (scope of the proved neighbourhood). The clean-run engine yields collapse only for reveal rates below $(1-\alpha-\eta)^W$ — astronomically small at $W = 1500$ — so what is proved is the monotone approach and the $p_0 \to 0$ limit, not a physically wide collapse interval. The relevant crossover $p_0 W = \Theta(1)$ and the plateau height are pinned by the renewal-reward scaling of Proposition 6 (governing variable $p_0 W$) together with the simulator's master curve (Appendix D.3), not by this proposition.

Remarks. (R1, minimality) Only SC, a finite window, and i.i.d. continuous residuals with μ fixed are used — no independence of r and the censoring level (the A2 crutch of Tier A), no convexity, no tail bound. (R2, mixing extension) i.i.d. enters only through the constant lower bound on $\Pr(A_t \mid \mathcal{F}_{t-1})$; the correct general condition is joint W-run positivity $\Pr(r_1, \ldots, r_W \le \ell_\eta) > 0$ (marginal positivity alone fails — a deterministic rotation is a counterexample), which holds under ϕ-mixing with marginal positivity. Since A_t is measurable w.r.t. the raw-residual σ-field and the run forces the ceiling for any Q-path, the adaptivity of Q_t does not enter the bound, and the conditional (Lévy) Borel–Cantelli applies. (R3) Point KM and the one-sided safe KM band are SC, so Theorem 2 covers them: conservatism within the support cannot escape. (R4) dfree's $+\infty$ completion, the control arm, and user resubmission are all BS, so Prop. 3 treats them as one condition; a per-job resubmission probability plays the role of p_0, explaining the "$\ge$25% resubmission rescues" finding folded into Section 6.1 (Appendix D.2). (R5) Theorem 2 is the dynamic companion to Prop. 2: the loop drives the achievable margin down until censoring at that margin exceeds α and stays there — the static bound is exact below α, and beyond-support allocation is the only escape the loop leaves.

Proposition 6 (plateau scaling: the shortfall is governed by p_0W).

Take the naive support-confined rule perturbed by an exogenous beyond-support reveal of per-step probability p_0 (each step, independently, the proposal is lifted above M_t, revealing r_t). Let $C(p_0, W)$ be the long-run time-average coverage and $C_0 := \lim_{p_0 \downarrow 0} C(p_0, W)$ the pure-SC floor of Theorem 2 / Proposition 4. In the regenerative model where each reveal injects a value that pins the pool for the W steps it stays in the window and the pool otherwise decays under Lemma 2, the coverage shortfall $\Delta(p_0, W) := (1-\alpha) - C(p_0, W)$ obeys

$$\Delta(p_0, W) = \big[(1-\alpha) - C_0\big]\, h(p_0 W)\, \big(1 + o(1)\big),$$

a function of (p_0, W) only through the dimensionless product $x := p_0 W$ (expected reveals per window), with h non-increasing, $h(0) = 1$, $h(\infty) = 0$, and crossover at $x = \Theta(1)$.

Proof sketch. Reveals form a Bernoulli(p_0) process, so inter-reveal gaps G are geometric with mean $1/p_0$. A reveal injects a near-q value that stays the pool maximum for the W steps it remains in the window (barring an earlier reveal); over those steps the level is identifiable and coverage $\approx 1-\alpha$, and once it slides out with no fresh reveal the pure-SC ratchet (Lemma 2, Theorem 2) drives coverage toward C_0. The fraction of steps in the decayed regime is $\mathbb{E}[(G-W)^+]/\mathbb{E}[G] = (1-p_0)^W = e^{-p_0 W}(1 + o(1))$ — a function of $x = p_0 W$ alone — and the within-gap decay timescale is itself $\Theta(W)$ (window turnover), so the cycle-average shortfall rescales in x. Monotonicity in x follows from Proposition 4(i); the two limits are Theorem 2 ($x \to 0$) and Proposition 3 with the ACI level at $1-\alpha$ ($x \to \infty$). The exact shape of h is not pinned by this argument (the within-gap decay is not a step function); Appendix D.3 measures it.

Remark. The governing variable $x = p_0 W$ is beyond-support reveals per window. The simulator (Appendix D.3) confirms the scaling — $C(p_0, W)$ collapses onto one master curve $g(x)$ across a $12\times$ range of W (isotonic $R^2 \approx 0.85$–0.91, empirical crossover $x \approx 0.2$–0.4) — upgrading the Section 7 renewal-balance heuristic to a derived-and-validated scaling law. The floor $C_0 = F(\operatorname{essinf} r)$ is strictly positive for the bounded-below (log-space) residuals of the trace, a competing and here dominant contributor to the observed 83.5%/59.4% plateau alongside the small-x leakage — which answers the concern that Theorem 2's "coverage $\to 0$" (an unbounded-support statement) sits far from the observed height.

Appendix B. Confidence intervals and multi-split robustness (Section 6)

B=300, seeds fixed (partition cpu). The harness imports the published conformal/eval functions, so every bootstrap center reproduces the corresponding point estimate; full CSVs and per-cell tables are in the released analysis code.

Section 6.1 online collapse — K=5 rolling-origin (expanding-window) folds; coverage mean $\pm$ sd (min–max). The collapse appears in every fold and dfree holds target on every fold; the runtime-baseline fold spread widens under completed-only scoring (uneven completed-row counts across the chronological folds).

target	method	mean ± sd	min–max
memory	naive	83.1 ± 9.0	71.9–91.2
memory	KM	82.5 ± 9.1	71.6–91.2
memory	dfree	95.5 ± 0.9	94.2–96.5
runtime	naive	54.9 ± 20.1	32.4–75.4
runtime	KM	54.9 ± 20.1	32.4–75.4
runtime	dfree	95.3 ± 0.6	94.5–95.9

Section 6.1 block-length insensitivity — overall coverage 95% CI at $L \in \{500, 1500, 3000\}$. The L=1500 result is not an artifact of the block choice.

target	method	L=500	L=1500	L=3000
memory	naive	[76.2, 90.0]	[72.2, 91.3]	[72.8, 88.4]
memory	dfree	[94.8, 96.4]	[94.6, 96.0]	[94.8, 95.8]
runtime	naive	[51.9, 67.0]	[48.3, 69.6]	[47.3, 66.1]
runtime	dfree	[94.7, 96.0]	[94.6, 95.7]	[94.6, 95.5]

Section 6.1 per-partition coverage [95% CI] ($L = 1500$). The within-support worst cells (gpu-b-runtime, gpu-a-runtime) have CIs far below 95% — the collapse is severe and significant; dfree's per-partition CIs all bracket 95%. (The gpu-c partition is omitted: its per-partition test stream is a single short block, giving a degenerate near-point interval.)

target	partition	naive	KM	dfree
memory	gpu-a	84.9 [80.0, 95.4]	84.9 [79.9, 95.4]	95.6 [94.0, 96.3]
memory	gpu-b	84.4 [63.6, 94.7]	83.5 [64.4, 94.6]	95.2 [93.8, 96.5]
memory	cpu	77.2 [41.2, 92.5]	77.2 [42.9, 94.6]	96.0 [94.2, 96.8]
runtime	gpu-a	63.3 [38.0, 84.4]	63.3 [39.4, 87.2]	94.7 [92.7, 95.9]
runtime	gpu-b	37.5 [23.6, 52.5]	37.5 [23.1, 51.6]	95.6 [94.4, 96.3]
runtime	cpu	81.4 [74.4, 87.9]	81.4 [73.8, 89.2]	95.6 [94.6, 96.3]

APPENDIX C. RELOCATED RESULT TABLES (STATIC CENSORING, SECTION 6.1, SECTION 6.2, AND THE DEPLOYED BOOSTED MODEL)

C.1 Static censoring — full table (folded into Section 6.1). Holding a fixed model and varying the calibration censoring rate (nominal 90% target):

censoring	naive	KM (Tier A)	dfree (Tier B)
4.5%	89.5% [89.1, 90.0] / 1.9×	91.0% [90.5, 91.4] / 2.1×	92.3% [91.8, 92.7] / 2.2×
9.4%	89.1% [88.7, 89.6] / 1.8×	92.3% [91.8, 92.7] / 2.2×	97.5% [96.5, 98.4] / 3.7×
14.4%	88.5% [88.1, 89.0] / 1.7×	93.4% [92.9, 93.8] / 2.4×	vacuous (+∞)
29.5%	85.5% [84.9, 86.1] / 1.5×	98.8% [97.9, 99.4] / 5.3×	vacuous (+∞)

(coverage [95% CI] / median over-allocation; oracle 89.9% / 1.90×; stratified bootstrap, B=300.) Naive under-covers at every rate and worsens monotonically with censoring — its coverage CI lies entirely below 90% at 9.4–29.5%, so the bias is significant, not noise. KM is safe everywhere (CIs ≥90%) and near-oracle at realistic (5–10%) censoring. Tier B is exact and finite while censored-fraction $\leq \alpha$ (Prop. 2), then vacuous (+∞ in 100% of resamples at 14.4/29.5%), exactly as proved.

C.2 Section 6.1 online collapse (Mondrian × ACI) — full table and rescue trajectory. Running the same conformal layer in the closed loop, per-partition (Mondrian) with ACI, one-sided 95% target:

method	mem cov	mem cost	runtime cov	runtime cost
naive	83.5% [72, 91]	2.6×	59.4% [48, 70]	2.4×
KM (Tier A)	83.0% [72, 91]	2.6×	59.4% [48, 70]	2.4×
dfree (Tier B)	**95.6%** [94.6, 96.0]	2.6×	**95.4%** [94.6, 95.7]	4.3×

Coverage as point [95% CI] (moving-block bootstrap, $L = 1500 \approx W$, B=300). Naive and KM collapse to sub-nominal coverage and saturate ACI ($1-\alpha_t \to 1.0$); per-partition the collapse is worst where it hurts most (gpu-b runtime 37.5% [24, 53], gpu-a runtime 63.3% [38, 84], cpu memory 77.2% [41, 93]), while dfree holds 95% on all 8 (partition × target) cells. This is not a single-split artifact: across K=5 rolling-origin folds naive/KM hold ≈83±9 (mem) and ≈55±20 (runtime, the wider spread from uneven completed-row counts per fold) while dfree holds ≥95.3 on every fold, and the CIs are insensitive to block length $L \in \{500, 1500, 3000\}$ (the two tables above). KM ≈ naive here because, once the loop erases the uncensored tail, KM's event-weighting has no tail mass to exploit and degenerates to $\max(\text{pool})$.

The rolling-coverage trajectory for this collapse and its rescue is shown in Figure 2 (main text).

C.3 Section 6.2 beyond-support principle and control sweep — full table and baseline analysis. Adding the one-sided safe KM band and a control-fraction sweep:

strategy	mem cov	runtime cov	escapes?
naive	83.5%	59.4%	**[no]**
KM (point)	83.0%	59.4%	**[no]**
KM band (safe)	**83.0%**	**59.5%**	**[no]**
conf. survival [2]	39.6%	17.7%	**[no]**
censored-QR [11]	83.0%	59.4%	**[no]** (≡ KM)
weighted-conformal [9]	95.6% / 81.8*	95.4% / 56.6*	**[!]***
Autopilot p95×1.25 [29]	94.5%	87.1%	**[no]**
EVT-GPD naive [POT]	83.7%	60.1%	**[no]**
EVT-GPD cens-MLE	75.8%	61.2%	**[no]**
dfree / dfree_cap	95.6%	95.4%	**[yes]**
control (user-request)	96.2%	95.5%	**[yes]**

The safe KM band — a strictly more conservative within-support estimator — collapses identically, confirming the beyond-support principle: conservatism within the observed pool cannot help. Control-fraction sweep (point-KM treatment): f=0 reproduces the collapse (≈82%/64%); f ≈ 0.02 restores 95% on both targets and un-saturates ACI. Empirical f* (≈0.02) is below the variance heuristic (≈0.10) because a generous control margin reveals much tail per control job. An adaptive control fraction that raises f where reveals are scarce (ACI level saturating, pool-max stale) does not improve control-draw efficiency over fixed $f \approx 0.02$, but at matched coverage it lowers over-allocation cost (memory median $3.9\times$ vs. $4.5\times$; runtime $6.4\times$ vs. $9.3\times$) for a higher, near-free control-draw rate — so fixed $f \approx 0.02$ is the simplest near-optimal choice, adaptive-f when tail cost dominates.

The three cited-but-unbenchmarked baselines (rows 4–6) all fall on the support-confined side of Theorem 2, confirming its reach beyond our own estimators. Conformalized survival [2] collapses harder than naive (39.6%/17.7% overall; gpu-b-runtime to ≈0%): its inverse-probability-of-censoring weighting treats the censoring law as an external nuisance to correct, but here that "nuisance" is the allocator's own policy, so the correction piles mass onto a shrinking uncensored

subsample and overshoots downward — the very machinery that makes it principled under exogenous censoring backfires under endogenous censoring (its Lemma 1 known-threshold structure is exactly what CLR cannot exploit). Portnoy's censored quantile regression [11], reduced to residual space, is numerically the KM tier (verified to 10^{-4}) and collapses cell-for-cell identically — it occupies the same point in method-space as our Tier A, offering no new lever. Weighted conformal under covariate shift [9] appears to hold 95% — but the escape is accidental. Our shift is label-censoring, not covariate shift, so its reweighting is inert (weights $\approx$ uniform on genuine covariates; static coverage $\approx$ naive); disabling the $+\infty$ point-mass branch its finite-sample formula happens to carry (the starred variant) collapses it to 81.8%/56.6%, indistinguishable from naive/KM. The as-published number is an incidental beyond-support tail-probe, not covariate-shift correction — the exception that proves the rule.

Production-autoscaler baseline (Autopilot). We benchmark an Autopilot-style [29] usage-percentile right-sizer: the allocation is the 95th percentile of a sliding window of observed usage times a safety margin, run in the same closed loop, so its own allocations censor the usage it later reads (no ACI feedback, matching Autopilot, which has no coverage-control loop). It under-covers on both targets even at a generous 1.25× margin (94.5% memory, 87.1% runtime); on memory it also costs more than dfree (4.76× vs. 2.56×), while on runtime its cost merely ties dfree's (4.31× vs. 4.28×) without reaching target coverage. At margin 1.0 the recommendation saturates at the pool maximum on 87–89% of steps — direct evidence that a percentile-of-history rule with no exploration cannot see the tail its own decisions truncated. A safety margin is a costly and incomplete patch, whereas the censoring-aware completion (dfree) holds 95% at equal-or-lower cost. This is the empirical counterpart to the Section 2 observation that production autoscalers train on observed usage without modeling that under-allocation censors the label. (Static offline calibration is different: at the low censoring of a single snapshot, Autopilot with a margin tracks naive, exactly the regime where the online ratchet has not yet acted.)

Parametric tail-extrapolation baseline (EVT/GPD). Why not fit a parametric tail above the observed max instead of the $+\infty$ completion? We benchmark a peaks-over-threshold generalized-Pareto right-sizer in the same closed loop. A naive GPD (method-of-moments fit, censored jobs treated as observed) collapses like naive (83.7%/60.1%): in the loop the pool is truncated at the current allocation, so on 89–95% of steps no exceedance survives above the threshold to fit and the rule falls back to the empirical quantile. A censoring-aware GPD (right-censored MLE using each killed job as a "true residual $> Q_t$" exceedance, with the threshold drawn from the uncensored values so the censored point-mass is usable) fires far more often but is erratic and still sub-nominal (75.8%/61.2%): with the tail censored to the single point Q_t, the GPD shape is barely identified, so the rule extrapolates on the tail assumption alone and, in the closed loop, neither reaches 95% nor controls cost. Both stay on the support-confined side of Theorem 2; extrapolating beyond support requires either seeing the tail (which the loop destroys) or assuming its shape (unidentified here), whereas the distribution-free completion needs neither. Loop and estimators are the published harness verbatim (naive/dfree reproduce 83.5/59.4 and 95.6/95.4 as a faithfulness check).

C.4 Online evidence on the deployed boosted model — full table (folded into Section 6.4). We evaluate the hardened production model (log-domain LightGBM μ, per-partition distribution-free margin, ACI online layer; killed jobs enter calibration as right-censored, never as usage labels) on the held-out test split (coverage on completed jobs, target 95%):

target	static per-partition margin	+ ACI online (production layer)	user-request baseline
Memory	88.2% [87.1, 89.9] @ 4.0×	**97.1% [96.7, 97.5] @ 3.18×**	100% @ 18.5×
Runtime	97.4% [97.2, 97.7] @ 5698×	**95.0% [94.7, 95.3] @ 4.14×**	99.4% @ 14400×

Two findings shape deployment. First, a static per-partition margin is not enough: memory coverage plateaus at ~88–90% even at calibration level 0.99 because the validation→test shift is large and heterogeneous; the ACI online layer restores 97.1% (non-overlapping CIs vs the static 88.2%, so the restoration is significant) and is cheaper (3.18× vs 4.0×), adapting the level rather than over-buffering. Second, runtime over-allocation is intrinsically large (a heavy tail at a conservative static

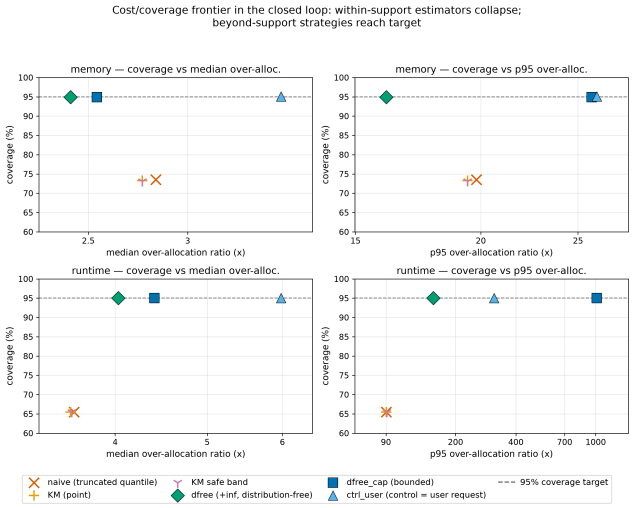


Figure 3: **Deployable cost/coverage frontier** across censored-allocation strategies in the closed loop (memory and runtime; the plot relocated from Section 6.3). Strategies confined to the observed support (naive, KM, safe KM band) cluster at sub-nominal coverage; the three beyond-support strategies (dfree, dfree_cap, control=user-request) reach the 95% target, tracing the deployable frontier.

level overshoots to 5698×), but ACI lands at 95.0% @ 4.14× — a ~1000× cost reduction by finding the right operating point; memory is the real prize, since memory is reserved whereas wall-time mostly affects backfill. This is open-loop ACI on historically user-censored data (pre-deployment censoring <1.5%, so no ratchet); the closed-loop ratchet is the future concern handled by freezing μ and the control arm.

C.5 Survivorship-adjusted coverage (static cross-check, folded into Section 7). Completed-only evaluation is optimistic because ~25% of trace jobs are non-completed and right-censored. As a complementary static cross-check (a single-shot split-conformal margin, not the online loop), we bracket coverage on the full test split ($N = 104{,}942$) using each job's original scheduler grant as a_{orig}. Only OOM/timeout jobs give real evidence that $R > a_{\text{orig}}$; FAILED (mostly zero-elapsed crashes), CANCELLED, and RUNNING jobs do not, so we report an evidence-gated lower bound (counting only OOM/timeout non-completions as exceedances) beside the completed-only rate:

target	method	completed-only	evidence-gated LB
memory	naive	97.45%	97.37%
memory	dfree	97.45%	97.37%
runtime	naive	99.34%	98.94%
runtime	dfree	99.43%	99.03%

The evidence-gated bound sits only 0.08pp (memory) and 0.40pp (runtime) below completed-only, and dfree $\geq$ naive throughout. Two honest caveats. First, this is the static single-shot setting, where real resource-driven censoring is rare (0.08–0.44% of the calibration pool), so it does not and should not reproduce the online ratchet's naive collapse (dfree $\geq$95% vs. naive 59–84%), which is a property of the self-reinforcing loop rather than of completed-versus-population accounting. Second, a task-literal worst case that treats every non-completed job (including ~15K zero-elapsed FAILED crashes carrying no resource evidence) as a non-cover drops the floor to ~82%, but that gap is dominated by non-censoring failures, not by the censoring mechanism this paper studies.

Online survivorship bound (the closed-loop headline). The cross-check above deliberately fixes one margin; the sharper question is whether the online Section 6.1 numbers, scored on completed jobs, are optimistic where the loop truncates the tail. They are not. The loop's realized per-slot allocation $\hat{a}_t = \exp(\mu + Q_t)$ is a fixed number, so each resource-censored test job can be scored

against its own grant a_{orig} with no re-derivation of the ACI dynamics: it is a guaranteed online miss iff $\hat{a}_t < a_{\text{orig}}$ (since $R > a_{\text{orig}}$). Running the published Mondrian $\times$ ACI loop unchanged (completed-only reproduces the headline 95.61%/95.37% exactly — a faithfulness check), the online completed-only optimism for dfree is **0.11pp (memory) / 0.19pp (runtime)** counting only proven misses, and 0.30pp/1.23pp counting every resource-censored job as a miss — comparable to the static bound, so the loop does not materially amplify survivorship optimism, and dfree $\geq$ naive under every accounting (memory 95.5 vs. 83.2; runtime 95.2 vs. 58.7). The task-literal floor that also counts the roughly 10–11k zero-evidence FAILED crashes as misses is 65.6%/66.9% (dfree) vs. 57.3%/41.7% (naive), the gap again crash-dominated.

APPENDIX D. NUMERIC κ (FOLDED INTO SECTION 6.2) AND THE CONTROLLABLE SIMULATOR (FOLDED INTO SECTION 6.1): FULL DETAILS

D.1 Numeric κ and retraining stability — full table (folded into Section 6.2). Measuring the contraction modulus $\rho(f)$ of the one-step retraining map directly (empirical Lipschitz constant, common random numbers) on the convex linear-quantile surrogate where Thm 3 is rigorous:

target	$\kappa = \rho(0)$	f* = $1 - 1/\kappa$	reading
memory	0.9 – 1.2	0 – 0.18	marginally stable, barely performative
runtime	1.8 – 2.2	0.43 – 0.54	clearly unstable, needs a large control fraction

Three findings. (a) κ is O(1–2) and target-dependent: memory sits at the stability boundary (matching the mild spiral observed in feedback simulations), runtime is genuinely unstable because heavy tails propagate censoring bias harder. (b) $\rho(f)/(1-f) \approx$ const validates the linear de-coupling $\epsilon(f) = (1-f)\epsilon_{\text{cens}}$. (c) The retraining f* and the conformal-layer f* differ: for memory both are small (a small floor suffices), but for runtime the retraining f*$\approx$0.5 $\gg$ conformal f*$\approx$0.02 — a small control floor keeps the quantile identified but does not stabilize aggressive μ-retraining. Practical upshot: for heavy-tailed targets, freeze μ and adapt only the conformal margin (where f*$\approx$0.02 suffices), or use a censored-aware learner to cut κ.

Boosted learner (iterated-map contraction). The parameter-space Lipschitz probe above does not transfer to the deployed gradient-boosted learner: a tree ensemble is piecewise-constant, so a small random input perturbation only flips a few censoring decisions and the refit jumps by a fixed discretization floor — the single-step modulus is then δ-dependent ($\rho \cdot \delta \approx$ const, the signature of a non-Lipschitz map) and inflates to an implausible $\kappa \approx$ 40–140 (which would demand f*$\approx$0.99, contradicting the stably-running GA at f=0.2; we retain this as a documented negative result). We therefore measure the contraction of the actual iterated retraining map on the deployed LightGBM (production tree config, 0.95-quantile objective): from a base policy a^* (fit on true labels) and a coherently perturbed copy $a^* + 0.3$ (a realizable uniform log-space margin shift), we iterate the full censor-then-refit map under common random numbers and track the RMS policy gap g_t; the one-step modulus $\rho_1 = g_1/g_0$ at the fixed point is the faithful analog of $\kappa = \rho(0)$ and the least discretization-contaminated point ($g_1 \approx 2\times$ the tree-jitter floor). Over 3 base seeds:

target	boosted $\kappa = \rho_1(0)$	f*	ρ_1 at deployed f=0.20
memory (MaxRSS)	0.86 ± 0.03	0 (already contracts)	0.78 $\rightarrow$ contracts
runtime	1.72 ± 0.16	≈ 0.42	1.50 $\rightarrow$ diverges

The deployed learner is consistent with the surrogate's coarse dichotomy — memory near/below the stability boundary ($\kappa\approx$0.86; a single-direction difference quotient lower-bounds the operator norm, so this reads as contractive-or-marginal, matching the surrogate's marginally-stable memory), runtime clearly unstable ($\kappa\approx$1.72$\gg$1, robust to the jitter floor) — and the ordering runtime $\gg$

memory. It sharpens the deployment reading: at the deployed 20% control the memory retrain loop contracts but the runtime loop does not (f*_runtime $\approx 0.42 > 0.20$), which is exactly why we freeze μ for the heavy-tailed (runtime) target. Absolute magnitudes differ from the parameter-space surrogate because the spaces differ; the invariants that transfer are the contract-vs-diverge classification and the target ordering.

D.2 Sensitivity and external validity (controllable simulator) — full tables (folded into Section 6.1). A controllable closed-loop residual simulator (tail shape, drift, resubmission, and control all tunable; N=20000, 8 seeds, drift on) tests whether the Sections 6.1–6.2 collapse is an artifact of the trace or of heavy tails, and whether the headline is carried by a single pessimistic, no-resubmission loop.

Cross-shape robustness (E-sens-1). With no resubmission, first-attempt coverage at a 95% target:

residual tail	within-support (naive = KM = safe KM band)	beyond-support (dfree / control)
light (Gaussian)	79.3 ± 0.3	$\mathbf{95.0 \pm 0.0}$
medium (t_5)	76.6 ± 0.2	$\mathbf{95.0 \pm 0.0}$
heavy (t_3)	85.0 ± 0.2	$\mathbf{95.0 \pm 0.0}$
cloud (spike-mixture)	80.4 ± 0.2	$\mathbf{95.0 \pm 0.0}$

The support-confined collapse appears on every shape, including a plain Gaussian, with tight CIs over 8 seeds; the safe KM band collapses identically to naive/KM on each, and beyond-support holds 95.0% exactly throughout. The mechanism is therefore not specific to the HPC trace or to heavy tails.

Resubmission scopes the collapse (E-sens-2). Real users often resubmit an OOM-killed job at a larger allocation. We add a probability p_{resub} that a killed job retries at a generous, truth-revealing level (heavy tail):

p_{resub}	naive	KM	dfree
0.00	85.0%	85.0%	**95.0%**
0.25	**95.0%**	**95.0%**	95.0%
0.50	95.0%	95.0%	95.0%
1.00	95.0%	95.0%	95.0%

Even a 25% resubmission rate fully restores within-support coverage: resubmission feeds the true requirement back into the pool, re-injecting the tail the ratchet erased. This sharpens rather than weakens the contribution: a user who resubmits at higher memory is manually allocating beyond the failed allocation, so resubmission, dfree's $+\infty$ probe, and the control arm are instances of the same beyond-support principle — within-support estimators never self-rescue; they are rescued only by an external (resubmission) or internal (dfree/control) beyond-support signal. We modeled the most favorable resubmission (a single, truth-revealing retry); incremental real retries (bump $2\times$, possibly OOM again) rescue no faster, so the reported effect is optimistic for the within-support methods.

When is the guarantee worth its cost? (newsvendor). Coverage is not a resource saving by itself, so we price it. Total cost per job = capped over-reservation + an OOM tax (resubmit: wasted compute + latency; abandon: wasted compute + a per-kill failure cost C_u for lost work / SLA / trust). For big jobs with incremental resubmission:

C_u (per-kill failure cost)	no resubmit: naive / dfree (winner)	50% resubmit: winner
0	**12.3** / 35.5 (naive)	naive
100	47.0 / 47.1 ($\approx$ tie)	naive
500	186 / **94** (dfree, $1.99\times$)	naive
2000	708 / **268** (dfree, $2.64\times$)	dfree

The data refuted our going-in guess that beyond-support saves resource cost: at $C_u = 0$ tight allocation with abandon/resubmit is cheaper (it over-reserves less). Beyond-support wins only once a kill is expensive — crossover $C_u^\star \approx 100\times$ a job's compute with no resubmission, rising to $\approx 2000\times$ with 50% resubmission (resubmission converts a failure into mere wasted compute + latency, lowering the effective C_u); the asymptotic advantage approaches the OOM-rate ratio (15%/5% = 3×). The contribution is a calibrated coverage guarantee, and its operational worth equals the operator's C_u: high for expensive, long, or hard-to-resubmit jobs and for hard SLAs; low for cheap, freely-resubmittable batch work, where tight allocation plus resubmission already suffices. This is the newsvendor reading of the safety level α.

D.3 The plateau is governed by p_0W (master-curve W-scaling). To test Proposition 6 we sweep the exogenous reveal rate $p_0 \in \{0, 5\times10^{-4}, \ldots, 0.1\}$ against the window $W \in \{250, 500, 1000, 1500, 3000\}$ (a 12× range), $N = 20000$, 8 seeds, drift on, for two residual laws. Time-average coverage $C(p_0, W)$ (heavy tail $0.8\,t_3$, target 95%):

W \ p_0	0	0.001	0.002	0.005	0.01	0.05
250	81.2%	86.7%	92.5%	94.2%	94.7%	95.0%
1000	82.9%	87.3%	92.4%	94.3%	94.9%	95.0%
3000	82.9%	89.1%	93.2%	94.4%	95.0%	95.0%

Plotted against $x = p_0W$, the five W-curves collapse onto one: an isotonic fit versus x has residual RMSE 1.7–2.0 pp (heavy/light), only 1.2–1.4× the per-point seed-sampling std (≈ the noise floor), $R^2 \approx 0.85$–0.91; at every shared $x \geq 5$ the cross-W spread is ≤ 0.6 pp. The floor→target crossover sits at $x \approx 0.2$–0.4 (one reveal per few windows), of the order-one scale Proposition 6 predicts. Two honest points. (i) The shortfall is not well fit by a one-parameter closed form — both $1/(x{+}1)$ and e^{-x} give $R^2 < 0$ — so the operative object is the empirical master curve $g(x)$, not a formula. (ii) The pure-SC ($p_0 = 0$) floor is 82.9% (heavy) / 75.9% (light), not zero and only mildly W-dependent, because these residual laws are bounded below and a finite window does not reach $\operatorname{essinf} r$ in N steps; the trace's 83.5%/59.4% plateau is therefore not reproduced by the simulator's t_3 law (a different residual distribution) but the scaling mechanism is confirmed — the trace plateau is the point on its own (steeper-tailed) master curve at the trace's operating p_0W.

D.4 Replication on public HPC traces (external validity). To test whether the online collapse-and-rescue is specific to our cluster, we replicate Section 6.1 on two independent public traces from the Parallel Workloads Archive (standard workload format, SWF), reusing the published harness with its estimator functions and every tuning constant ($W = 1500$, ACI step $\gamma = 0.01$, $\alpha = 0.05$) byte-identical; only the data loader and a NaN-safety filter differ (released adapter pub_trace_adapter.py). No single public trace carries both rich partition structure and a usable memory field, so we use one of each: CEA-Curie (CEA/CNRS, 2011; 280K completed jobs; runtime target; four queues with ≥ 300 test jobs) and HPC2N (Umeå, 2002; 202K jobs; the memory target, its one usable field). Coverage at the 95% target in the closed Mondrian × ACI loop:

trace	target	partition (n_test)	naive	KM	dfree
CEA-Curie	runtime	OVERALL	49.7	49.4	**95.0**
CEA-Curie	runtime	standard (26,166)	12.5	12.1	**95.0**
CEA-Curie	runtime	large (19,342)	94.5	94.5	**95.0**
CEA-Curie	runtime	hybrid (1,564)	94.6	94.6	**94.6**
CEA-Curie	runtime	BCS (792)	93.8	93.9	**95.1**
HPC2N	memory	single stream (40,370)	32.1	32.1	**96.5**
HPC2N	runtime	single stream (38,007)	53.7	53.7	**95.0**

The pattern reproduces on both traces, which span different countries, schedulers (Maui vs. Slurm/sacct), decades, and resource axes: the support-confined estimators under-cover wherever censoring bites — down to 12.5% on CEA-Curie's high-throughput standard queue and 32.1% on HPC2N memory — while dfree holds 94.6–96.5% on every partition × target cell. Consistent with the regime-dependence of Section 6.1 and Proposition 4, the collapse is partition-specific: CEA-Curie's large, hybrid, and BCS queues censor little and naive already covers, whereas the standard

queue collapses hard. Two honest limits. First, over-allocation cost is trace-specific but never large here: on CEA-Curie runtime dfree is in fact cheaper than the collapsed naive rule (median $8.2\times$ vs. $11.3\times$ over-allocation) while covering 95% against 50%; on HPC2N memory it lifts coverage from 32% to 96% at essentially the same cost ($1.01\times$ vs. $1.0\times$, the residuals being near-degenerate); and only HPC2N runtime shows the expected coverage-for-cost trade ($2.9\times$ vs. $1.1\times$). The replication is primarily of the coverage mechanism, since the cost multiple depends on each trace's residual law. Second, a third trace (ANL-Intrepid-2009) was excluded because its status and partition fields are absent (-1 for all records). The traces and the SWF-to-parquet adapter are public, so this replication is fully reproducible, unlike the proprietary sacct trace of Section 6.

Serverless probe (Azure Functions), directional transfer. To probe the Section 8 transfer conjecture beyond HPC we run the same harness on the public Azure Functions 2019 trace (a serverless domain), reconstructing per-invocation execution duration from the published per-function percentile aggregates by inverse-transform sampling (84K test invocations across seven trigger-type partitions). The qualitative signature reproduces on every partition — naive and KM under-cover the 95% target (93.6%/93.8% overall, 92.3–94.8% per partition) while dfree restores 95.0% (95.0–95.2%) — but the collapse is far milder than on HPC (where naive falls to 38–84%). This is consistent with the regime-dependence of Section 6.1: a serverless partition pools thousands of distinct functions rather than repeat job-signatures, so population heterogeneity continually re-injects tail information (a passive analog of the resubmission rescue), softening the ratchet. Two caveats keep this directional rather than definitive: the reconstruction truncates the tail beyond the p99 knot (likely understating the censoring), and the trace carries no per-invocation memory field (duration only). The transfer of the mechanism is thus supported; its severity is regime-dependent and, for heterogeneous serverless pools, small.

Appendix E. Live deployment: full narrative (Section 6.4)

We took the hardened allocator (Appendix C.4) to production in two phases on a research-center HPC cluster. The live deployment is not the paper's primary quantitative evidence; the safety target and the savings are established offline, on the 336K-job trace and the held-out split (Section 6.3, Appendix C.4). Its role is to (i) corroborate those results on real traffic and (ii) surface three failure modes no offline experiment could: an OOM on a non-slack resource, a confounded safety monitor, and an intervention that silently goes inert. Each is a concrete deployment lesson, and the third is a live instance of a pathology this paper predicts. What the deployment validates is therefore the safe composition — the coverage rule together with a slack model, an operating-point floor, and a rollback guardrail — not the distribution-free bound in isolation, which in the closed loop we find becomes operationally vacuous (below).

Phase 1 — shadow (open loop). A scheduler submit-time hook queried the served conformal allocator and logged its would-be allocation while altering no job (zero mutations), with the ACI margin refreshed from realized outcomes every 30 min and hot-reloaded into the live service. Over eleven days it logged ~6.4K would-be decisions across 28 users at ~1,880 terminating jobs/day. Scoring each recommendation against the job's realized requirement at the deployed operating point ($N = 431$ completed jobs matched to a recommendation): would-be OOM 2.3% (memory) / 0.0% (runtime), both within the $\alpha = 0.05$ target — a production-traffic replication of the held-out open-loop result (Appendix C.4). In shadow the allocator never censors, so the ratchet of Section 4.3 cannot arise and the user-adaptation channel is not exercised; the shadow number validates the safe rule but not the closed loop.

Phase 2 — closed-loop randomized A/B. We then enabled mutation under the pre-registered design: per-job deterministic-hash assignment, a staged ramp (5%→25%→50% treated) advancing only while guardrails hold, a permanent policy-independent control arm ($\geq 50\%$ at the current stage, never below the 20% pre-registered floor) that preserves a clean user-request baseline, a frozen μ during the trial (only the conformal margin adapts), and an automatic rollback guardrail that reverts to shadow if the treated OOM rate exceeds $2\times$ control or a 5% ceiling (with a ≥ 5 absolute-kill floor to avoid tripping on noise). The deployment is memory-only; runtime is passed through (Appendix C.4), so the guardrail counts only OUT_OF_MEMORY terminations.

A safety failure the offline analysis could not surface. The first mutating ramp exposed a failure mode absent from every offline experiment: memory-only right-sizing is not "safe by construction." An opaque interactive job (no usage signature) that held all GPUs on its node was cut 81% on host memory; because the GPUs already gated the node, the cut reclaimed nothing schedulable and instead OOM-killed a live, hours-long training session. We halted to shadow the same day and hardened the cutting side with four policy-level guards, each a concrete instance of the Section 7 limitation that memory-only allocation is safe only where the reservation carries slack: (i) pass through interactive/shell-command and full-node-GPU jobs (no signature, or memory is not the binding constraint); (ii) a never-below-observed-peak floor (per-(user, job, partition) $p_{95} \times 1.5$); (iii) a cap on no-history jobs ($\leq 30\%$ reduction); and (iv) per-user opt-out. The lesson is that the safe coverage rule must be composed with a slack model of the reserved resource before it can mutate live.

Result after hardening. We re-ramped 5%→25%→50% under the hardened policy. Over the 25% window (the first stage to reach the guardrail's 100-job-per-arm quorum), the causal safety contrast is (intention-to-treat over jobs hashed to each arm):

arm	jobs	OOM / timeout
treatment (25% window)	295	**0.0% (0 / 295)**
control	576	0.35% (2 / 576)

Treated OOM in the 25% window is not elevated over control and is far under the 5% ceiling, and the guardrail did not trip in that window. Of the treated jobs the hook applied a recommendation to (35 in the hardened window), 22 were actual reductions: on a never-inflate service an applied recommendation can equal the request, so "mutated" and "cut" are not the same count. Those 22 cuts had a median of 41.5% (range 19.6–65.1%, versus a pre-hardening maximum of 97%: the peak floor is binding), avoiding ≈0.75 TB of reserved-but-unneeded memory. Advancing to 50% surfaced the residual risk: the treated kills that appeared were all a single low-history job that the no-history cap cut 30%, whose sibling signatures also OOM at the user's own full request in the untouched control arm (a chronically under-provisioned job family, ~55% OOM at its own 16 GB request). This is not a coverage failure of the safe rule but a weakness of the two guards that operate outside the calibrated bound: the no-history cap (a job with no signature but a memory-hungry family) and never-inflate (which cannot rescue a user's own under-request). The next paragraph gives the reconciled treated-versus-control accounting for this 50% window, and the monitoring confound it exposed.

A confounded safety monitor, and the causal fix. As treated volume accrued at 50%, the automatic guardrail rolled the deployment back to shadow when its intention-to-treat (ITT) trip rule crossed threshold. The rollback was safe but miscaused. The two ways to count OOM over the full 50% window reconcile as follows:

50%-window OOM rate	value
treatment, ITT (every job hashed to treatment)	6.2% (42/679)
treatment, as-treated (only jobs the allocator actually cut)	2.1% (5/235)
control arm (never mutated), same window	7.0% (35/501)

Both treatment rates sit at or below the 7.0% control arm, and even the ITT treated rate (6.2%) is below it. The trip fired only because the ratio test crossed its threshold on a transient count imbalance earlier in the window, before the control signature's own kills had accrued. Of the 42 ITT "treated" kills, 37 were a single signature of an under-provisioning user's metric family that the allocator never cut (never-inflate returned the request unchanged) and that merely hashed into the treatment arm; the five as-treated kills were the no-history job of the previous paragraph, none caused by the calibrated cut. This is a general hazard of monitoring a performative deployment randomized at the unit level: the treatment arm is full of un-mutated jobs (passthrough, no-history, or never-inflate held at the request), so an ITT OOM rate conflates the intervention with the users' own under-provisioning,

and a single high-volume self-OOMing signature can trip a reversal the policy did not cause. The causally correct trigger counts only jobs the allocator actually cut; we moved the guardrail to this as-treated rule (a treatment-arm job is scored only if it was mutated), which reproduces the 2.1% rate and would not have tripped. Independently, the genuine no-history kills were removed at the source by a hierarchical floor that borrows a size-matched sibling's peak when a job has no exact history (offline replay: no-history would-be OOM 0.8%→0.0% at 56% of the savings retained). The re-enabled 50% run carries both corrections. The lesson transfers to any unit-randomized performative deployment: attribute harm as-treated, not by assignment, or the safety valve itself becomes a source of spurious reversals.

A self-inerting margin. Left to recalibrate through the 50% ramp, the online memory margin drifted into its pool-max regime, the distribution-free rule's "allocate the user's request" fallback, so within two days the treated arm cut essentially nothing: maximally safe, and entirely inert (zero treated cuts accruing). This is the operational face of Proposition 2 and the Section 4.3 ratchet. As realized exceedances accumulate in the calibration window, the safe margin returns allocate maximally, which on the never-inflate service means serve the request. The deployment thus reproduced, unprompted, the support-confinement pathology the paper analyzes: the guarantee did not fail, it became vacuous. We restored an active intervention by capping the memory margin at the Appendix C.4 certified operating point (per-partition seed values), a deliberate trade of a slice of the distribution-free worst-case for a non-trivial cut, with the peak-floor, no-history, and guardrail protections unchanged. This does not contradict Appendix C.4's finding that production must run ACI rather than a fixed margin: ACI still adapts the level to track drift, but it must be floored at an operating point so it cannot converge to the trivial fixed point. Adapt for cost; cap to stay active. Capping restored active cutting, and the operating-point ceiling has since sustained multi-TB reclamation at scale (below). The lesson generalizes. A coverage-only margin can converge to the trivial policy, so a self-recalibrating safe allocator needs an explicit operating-point floor (or the resubmission/control signal of Section 5) to stay active.

General availability and the absolute safety contrast. After a certified go/no-go cleared its gates — treated OOM under the ceiling and under $2\times$ control at $\geq$100 jobs per arm, $\geq$50 distinct cut signatures exercised, zero guardrail trips, zero open incidents, plus PI sign-off — the allocator was promoted from the 50% trial to general availability: treated ramped 50%→80% with a permanent policy-independent control arm held at $\geq$20% (never 100% — the control is standing safety infrastructure, not a trial artifact). On the matched-cuttable-stratum estimand the safe allocator holds its absolute safety target: treated OOM 4/4375 = 0.091% on the full window to date (0.060% at the earlier pinned snapshot; the count accrues and the rate drifts up modestly), an order of magnitude under the 5% guardrail ceiling, at $\geq$100 jobs per arm and with no guardrail trip, while reclaiming $\approx$5.6 TB of reserved-but-unneeded memory at a 27% mean cut. The clustered reanalysis (next paragraph) establishes this absolute rate under signature-robust inference (cluster-bootstrap 95% CI [0.00%, 0.40%]); it does not establish a treatment-versus-control gap. The residual treated OOMs are the class flagged above: a no-history / self-bump family the guards outside the calibrated bound (the no-history cap and never-inflate) cannot rescue. We added a fifth, submit-time guard — a request-aware floor that abstains (allocates the request) when a job's requested memory exceeds its own recent request median by a factor τ, a submit-time-legal covariate with no realized-usage leakage — which targets exactly that self-bump miss without touching the calibrated cut. Finally, the GA control fraction (20%) exceeds both the identifiability margin of Proposition 5 and the surrogate contraction threshold $f^* = 1 - 1/\kappa \approx 16.7\%$ (memory, surrogate $\kappa \approx 1.2$); for the deployed memory learner the measured $\kappa \approx 0.86$ gives $f^* = 0$, so the control arm's live role is identifiability and de-biasing rather than retraining contraction. (Numbers pinned 2026-08-14; the live estimand accrues monotonically.)

Scope and power. The matched cuttable-stratum sample now reaches the $\geq$100-jobs-per-arm quorum at GA (treated 2/3327 = 0.060% OOM), so the live trial has graduated from its 2026-07 corroboration framing — when $0/295$ treated OOMs gave only a rule-of-three $\approx 1.0\%$ upper bound — to an absolute treated-safety statement whose signature cluster-robust CI sits an order of magnitude under the ceiling (not a treatment-versus-control gap). Three honest caveats still bound what it does not establish. First, the treated–control gap remains a high-variance estimand under per-$(\text{user}, \text{job})$ assignment: the control OOM rate is signature-driven (0.35% [2/576] in the 25% window but 7.0% [35/501] at 50%, the latter from a single prolific control-arm signature exceeding its own never-inflate request — memory the allocator never touched), and the residual treated kills concentrate

in another under-provisioned signature rather than spreading across the cut population. The defensible headline is therefore the absolute one (as-treated treated OOM far under the 5% ceiling and at/under the control rate), not a precise gap; a signature-clustered reanalysis (below) confirms the absolute rate but leaves the gap unidentified; a time-based switchback (alternating policy in fixed epochs) remains the gold standard we recommend. Second, the realized savings (≈5.6 TB to date at a 27% mean cut) characterize the operating point and under-estimate the full-deployment benefit — per-job randomization lets a treated job's freed memory shorten the control arm's own queue — so reserved GB-hour, queue-wait, and job-completion-time (JCT) effects need the same switchback and remain future work. Third, the user-adaptation sensitivity ϵ_{user}: probing the frozen-μ control contrast, among repeat-submitting signatures the treated (cut) users inflate their memory requests no more than control (difference in log-request slope ≈ -0.01/day, 95% CI $[-0.13, +0.08]$, permutation $p \approx 0.7$; the cut-magnitude-to-request-growth dose-response is weakly negative) — a bounded null consistent with cuts being largely invisible at the deployed OOM rate. This probe is underpowered (tens of repeat signatures) and confounded by the staged ramp, so a request-visibility switchback remains the clean estimator. (Deployment artifacts, the pre-registered design, and A/B scoring are in the project repository.)

Signature-clustered reanalysis (the offline alternative to a switchback). Because arm is a permanent per-signature hash, the treatment and control strata are disjoint signature populations, so the treated–control gap is irreducibly a between-cluster contrast. We reanalyze the live A/B with signatures as the clusters (on the full window to date; treated-cut $n = 4375$, would-cut-control $n = 1254$, reproducing the deployed stratum estimator exactly). Resampling signatures (not jobs) with replacement (2000 replicates): the absolute treated OOM rate is $4/4375 = 0.091\%$ with a signature cluster-bootstrap 95% CI $[0.00\%, 0.40\%]$ — its upper bound is still $\approx 12\times$ below the 5% ceiling, so the absolute-safety claim is robust to clustering. The treated–control gap ($+0.091\%$) instead has a cluster-robust CI spanning zero and is jackknife-fragile: leaving out the single most-OOM treated signature (which carries 2 of the 4 treated OOMs) nearly halves it, and one control signature accounts for 73% of all control jobs — the exact concentration that let the control rate swing $0.35\% \to 7.0\%$ across earlier windows. The clustered reanalysis therefore confirms the paper's absolute headline while showing the gap is not identified offline; a within-signature time-based switchback (alternating policy in fixed epochs on the same signatures) is the one instrument that would estimate the gap without the between-signature confound, and remains future work. (Numbers accrue monotonically; the $4/4375$ here is the same estimand as the pinned $2/3327$ snapshot above, at a later, larger window.)

In-vivo closed-loop demonstration (real memory kills). To witness the ratchet on real infrastructure rather than in replay or the safe-censoring deployment, we ran a self-contained, self-submitted experiment (no production allocator, no other users' jobs): a controller drives the closed loop over a stream of real memory-allocating processes, each executed under an address-space budget equal to the arm's current allocation, so a process whose true footprint exceeds its budget is really killed — a genuine censoring event fed back into the arm's pool. Two arms — a support-confined 95th-percentile rule and the dfree $+\infty$ completion, both fixed-level — run the same stream across light, medium, and heavy tail regimes. The real kill outcome matched the intended $A_t \geq R_t$ on 99–100% of rounds, so the loop is a faithful realization rather than a simulation. The dichotomy reproduces live in every regime: the support-confined arm ratchets its allocation down and its steady-state coverage falls to 79–85% (12–21pp under the 95% target) while dfree holds 97–98% — a 12–18pp gap. This is a small fixed-level demonstration, so its depth is milder and flatter across regimes than the trace's 59–84% collapse (a tight bulk floors the ratchet above essinf; the powered severity rests on Section 6.1, the public traces, and the simulator). Its contribution is to establish that the ratchet and its beyond-support escape are a real closed-loop phenomenon under actual memory enforcement, exactly as Theorems 2 and 3 predict — the live counterpart the deployment cannot provide at its safe (sub-α) censoring.

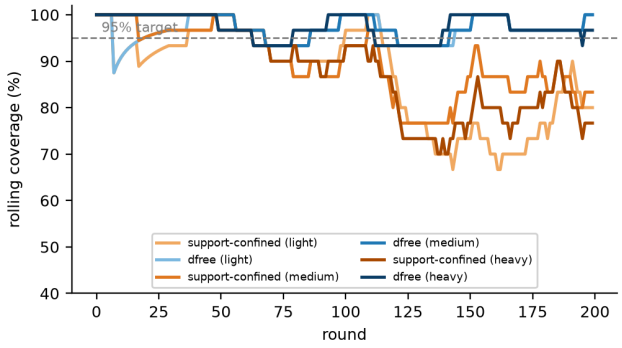


Figure 4: **In-vivo censoring ratchet under real memory kills.** Rolling coverage (window 30) of a support-confined 95th-percentile rule (orange) versus the dfree $+\infty$ completion (blue) in a self-submitted closed loop where each round's allocation is enforced by a real address-space limit, across light/medium/heavy tail regimes. As the loop runs and the window fills with censored feedback, the support-confined arm ratchets its allocation down and its coverage falls to 79–85% (below the 95% target, dashed), while dfree holds 97–98%; the real kill outcome matched the intended budget on 99–100% of rounds.