ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

Adaptive Conformal Risk Control for Payment Fraud Detection under Concept Drift and Delayed Feedback

Elena Arora [1], Mia Powell [2], Julia Arora [2, *]

[1] Microsoft, Redmond, USA

[2] Manning College of Information and Computer Sciences, University of Massachusetts Amherst, USA

[*] Corresponding Author: julia.a5692@outlook.com

Abstract

Payment fraud engines turn risk scores into approve-or-review decisions, and the quantity an operator must keep within budget is the fraud value released to auto-approval. Two features of production systems undermine static guarantees on that quantity: the fraud mix drifts, and the chargebacks that reveal fraud arrive weeks after the decision, so a transaction without a chargeback is not yet known to be legitimate. We present an adaptive conformal risk controller that keeps an exposure-capped, amount-stratified and value-ordered conformal calibration layer as a monotone family of policies and steers one scalar, the nominal level of that layer, by integral feedback on chargebacks. The feedback is progressive: every approved transaction contributes inverse-probability-of-censoring-weighted increments at several label ages, which telescope to its true monetary loss when the dispute window closes. This yields a distribution-free certificate on the long-run released fraud value that holds under arbitrary drift and under a misspecified delay distribution, whereas naive and single-stage early feedback converge to biased levels. On five 24-month Sparkov streams with a camouflaged-fraud drift scenario and on the 183-day ULB/Worldline handbook stream, static calibration released 2.63 times the budget and monthly recalibration 1.55 times, while the proposed controller held 1.01 ± 0.02 times, close to an oracle with instantly revealed labels. When chargeback delays grew by 50% after deployment, it stayed at 1.08 times, against 1.48 times for single-stage correction.

Keywords

Payment fraud detection, conformal risk control, concept drift, delayed feedback, chargebacks, inverse probability of censoring weighting, online calibration.

1. Introduction

Card issuers and payment platforms score every authorization request, but the score is only half of a decision. The other half maps the score to an action, either automatic approval or a hold for an analyst or a step-up challenge, under a limited review capacity. Two properties of real payment streams make this mapping hard to keep correct over time. The ground truth arrives late: most fraud is confirmed through chargebacks that cardholders may file weeks or months after the purchase, so the most recent labels are both delayed and one-sidedly censored [1, 2]. The stream is also non-stationary. Customer behaviour follows seasonal cycles and fraudsters adapt to the controls they meet, which produces concept drift in the feature distribution and in the fraud mechanism itself [3, 4]. Evaluations that assume immediately available labels overstate what a deployed system achieves [5], and drift can stay undetected for as long as the labels needed to see it are missing [6].

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

The quantity an operator is accountable for is usually monetary: the value of fraud that passes through auto-approval. Cost-sensitive views of fraud detection have long argued for such measures over rank-based ones [7, 8]. Streaming architectures address throughput and model refresh [9], but they do not certify how much money a decision rule releases. Conformal methods supply such certificates in distribution-free form [10–12]. The LARC calibration layer recently made this practical for payments [13]. It ranks transactions by expected fraud value, sends amounts above an exposure cap to mandatory review so that the monetary loss is bounded, allocates a conformal risk budget across amount strata by Lagrangian water-filling, and removes the bias of immature chargeback labels by inverse probability of censoring weighting (IPCW). Its guarantee, however, refers to a calibration window that is exchangeable with the deployment window. Under drift this assumption fails, and certified thresholds can silently move away from the budget they were meant to enforce.

Online conformal methods are designed for this failure mode. Adaptive conformal inference and its descendants adjust a nominal level after each observed error and guarantee long-run validity for arbitrary, even adversarial, sequences [14–17]. Their feedback model does not match payments, though. They assume that the loss of each decision is revealed, immediately or after a fixed horizon, before it is used. A chargeback stream reveals losses at random times, and a transaction without a chargeback cannot be scored as legitimate until the dispute window has closed. Feeding observed labels back as they stand underestimates the loss by the share of chargebacks still outstanding; in our experiments this let the released fraud value exceed the budget by 59–84%. Waiting for the window to close removes the bias but delays every correction by months.

This paper closes that gap. Its contributions are as follows.

1) We formulate online monetary risk control for fraud decisioning with randomly delayed, one-sidedly censored chargeback feedback, and build an adaptive controller on top of the LARC layer. The layer becomes a monotone policy family whose nominal level is steered by integral feedback and periodically re-anchored on recent IPCW-weighted data.

2) We introduce progressive IPCW feedback with reconciliation, in which each approved transaction reports weighted loss increments at several label ages that telescope to its true loss. We prove a trajectory certificate on the long-run released fraud value that needs no assumption on the stream or its drift and remains valid when the delay distribution is misspecified, characterize the bias of naive and single-stage early feedback, and derive the gain from the dead-time stability of an integral loop.

3) We compare nine policies on five independent 24-month Sparkov streams, with and without a camouflaged-fraud drift scenario, and on the 183-day ULB/Worldline handbook stream, including stress tests on dispute delays, sensitivity studies and ablations.

2. Related Work

2.1. Fraud Detection with Delayed and Censored Labels

Dal Pozzolo et al. formalized verification latency together with class imbalance and drift, and separated the fast feedback produced by investigators from the delayed labels supplied by cardholders [1]; their practitioner study shows how both shape the operational pipeline [2]. SCARFF scales this setting to streaming infrastructure [9], and the reproducible handbook of the same group provides a public transaction simulator with explicit delay periods [18]. Work on drifting streams shows that delayed labels change both what can be detected and how methods must be evaluated [3–6]. Closer to decision certification, Deng studies when matured randomized audits can authorize automation under a declared temporal-stability allowance [19], and DISCO couples representation learning with conformal control of the false-negative rate under rolling-window calibration [20]. Both certify count-based risks from matured

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

evidence. We control a monetary risk online, with a guarantee that does not rest on a stability assumption.

2.2. Distribution-Free Risk Control

Split conformal prediction gives finite-sample coverage under exchangeability [10]. Risk-controlling prediction sets [11] and conformal risk control [12] extend the idea to general monotone losses, which is the form LARC instantiates for monetary loss. Weighted variants restore validity under known covariate shift [21] or bound the cost of departures from exchangeability [22]. IPCW originates in missing-data estimation [23] and has been combined with conformal inference for censored survival outcomes [24]. These tools adjust a fixed calibration set; none of them is valid when the shift is unknown and the labels of the most recent data are still censored.

2.3. Online Calibration with Delayed Feedback

Adaptive conformal inference (ACI) updates the nominal miscoverage level online and attains long-run coverage for arbitrary sequences [14]. DtACI aggregates several step sizes [15], rolling risk control extends the update to general risks [16], and conformal PID control adds proportional and integral terms with a scorecaster [17]. Delays increase the regret of online learning additively [25], and a concurrent analysis of ACI with a fixed forecast horizon shows that its coverage guarantee survives deterministic delays [26]. Delayed conversions in online advertising produce the same one-sided censoring as chargebacks and are handled with delay models and importance weights during training [27, 28]. Our feedback differs from all of these settings at once: delays are random, unlabeled transactions are censored rather than missing, and the controlled quantity is a heavy-tailed monetary loss.

3. Problem Setup

3.1. Monetary Risk of a Decision Policy

Transactions arrive as a stream (x_t, a_t, d_t, y_t) with features x_t, amount $a_t>0$, day dt and eventual label $y_t\in\{0, 1\}$. A fixed scorer $s(x)\in[0, 1]$ is given. A policy π chooses between approval and review; analysts recover the exposure of reviewed fraud with probability q, and approved transactions are final. Following the monetary formulation of the calibration layer we build on, the loss of a transaction is the fraud value it releases to auto-approval,

$$L_t = a_t\, y_t\, 1\{\pi(x_t, a_t) = \text{approve}\} / \bar{a} \tag{1}$$

Where $\bar{a}$ is the mean training amount, so that $1000\bar{a}L$ is measured in dollars per thousand transactions. The operator sets a budget $\alpha=\alpha_{rel}B_0$ as a fraction of the uncontrolled loss B_0 of the training period and requires the realized average risk over the deployment period to stay at or below α. Net value is measured against approving everything, as $\Sigma_t 1\{\text{review}\}(qy_t(a_t+c_{cb})-c_r-(1-y_t)c_f)$ per thousand transactions, with dispute fee c_{cb}, review cost c_r and customer friction c_f.

3.2. Delayed and Censored Feedback

If transaction t is fraudulent, its chargeback arrives D_t days later. At label age u the observed label is $\tilde{y}_t(u)=y_t1\{D_t\le u\}$, so that $E[\tilde{y}_t(u)|y_t]=y_tG(u)$ when D_t has distribution function G independent of the transaction. Censoring is one-sided: a legitimate transaction and an unreported fraud look identical until the dispute window Δ_M closes, after which $\tilde{y}_t(\Delta_M)=y_t$. The operator only knows an estimate $\hat{G}$ obtained from the delays of historical chargebacks.

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

3.3. Online Objective under Drift

The joint law of (x_t, a_t, y_t) may change arbitrarily over time, including in response to the controls. We seek policies π_d, fixed within each day d and chosen from the information available at its start, such that the long-run released fraud value satisfies $\limsup_T N_T^{-1}\Sigma_{t\le T}L_t \le \alpha$ without distributional assumptions on the stream, while reviewing as little as possible.

4. Adaptive Conformal Risk Control

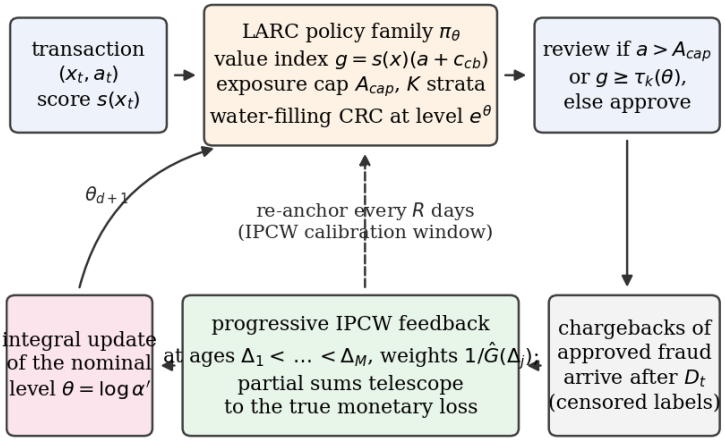


Figure 1. Overview. The LARC calibration layer defines a monotone policy family indexed by the nominal level e^θ. Chargebacks of approved transactions are converted into progressive IPCW feedback that drives an integral update of θ, and the layer is re-anchored on recent IPCW-weighted data every R days.

4.1. The LARC Layer as a Monotone Policy Family

LARC [13] contributes four components, all of which we adopt; the only addition, a residual fill that spends budget left unused by the water-filling, is described at the end of this subsection. First, transactions are ranked by the value index

$$g(x, a) = s(x)(a + c_{cb}) \tag{2}$$

Which LARC proves to be the risk-optimal triage order under a uniform review cost. Second, every transaction above an exposure cap A_{cap}, the 99th percentile of training amounts, is reviewed, which bounds the loss on the auto-approvable region by $B=A_{cap}/\bar{a}$. Third, amounts are split into K quantile strata with bounds $B_k=\min(e_{k+1}, A_{cap})/\bar{a}$. On a calibration sample with n_k cases in stratum k, approving the cases with $g<\tau_k$ has the conformal upper risk

$$R_k^+(\tau_k) = (\Sigma_{i\in I_k} \ell_i \, 1\{g_i < \tau_k\} + B_k) \,/\, (n_k + 1) \tag{3}$$

And the thresholds minimize the total review rate subject to $\Sigma_k p_k R_k^+(\tau_k) \le \alpha'$, where p_k is the share of stratum k; the allocation is found by Lagrangian water-filling over the convex frontiers of the strata. Fourth, when calibration labels are immature, case i of age u_i enters with the IPCW loss $\ell_i=a_i\tilde{y}_i(u_i)/(\bar{a}\hat{G}(u_i))$; cases with $\hat{G}(u_i)<G_{min}=0.5$ are dropped and B_k is inflated to B_k/G_{min}. We use the output of this layer as a map $\alpha' \mapsto \{\tau_k(\alpha')\}_k$ from a nominal level to thresholds. Two properties make it a suitable object to control. The thresholds are non-decreasing in α', so the released loss of every transaction is non-decreasing in α'. Below the feasibility floor $\Sigma_k p_k B_k/(n_k+1)$ no threshold can be certified, so the layer reviews every transaction. The Lagrangian solution only visits hull vertices of the frontiers, which makes the map coarse: in

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

our streams a single threshold set covered all nominal levels between $0.08\,B_0$ and $0.12B_0$. We therefore spend the budget left over by the water-filling by advancing the next stratum in the water-filling order to its last affordable candidate. This keeps $\Sigma_k p_k R_k + \le \alpha'$ and monotonicity, and yields 158 distinct threshold sets on $[0.01,0.2]\,B_0$. With $\alpha'=\alpha$ and a calibration window aged beyond G_{min}, the family reproduces the static LARC policy, which is the initial state of our controller.

4.2. Adaptive Nominal Level

Let $\theta_d=\log \alpha'_d$ be the state at the start of day d. All transactions of day d are decided with the thresholds $\tau_k(e^{\theta_d})$ of the current calibration snapshot, and at the end of the day the state is updated by

$$\theta_{d+1} = \min\{\theta_{max}, \theta_d + \eta(\alpha N_d - \Phi_d)/(\alpha \bar{n})\} \quad (4)$$

Where N_d counts the transactions that reach their first feedback stage on day d, Φ_d is the feedback received on day d, $\bar{n}$ is the mean daily volume and $\theta_{max}=\log(20\alpha)$ prevents wind-up. Each transaction credits α once and debits its estimated loss, so (4) is an integral controller on the gap between the budget and the realized risk. It is written on a log scale so that relative corrections are equally fast at any level.

4.3. Progressive IPCW Feedback with Reconciliation

Fix ages $0<\Delta_1<\cdots<\Delta_M$ with $\hat{G}(\Delta_M)=1$. At stage j the IPCW estimate of the loss released by transaction t is

$$\hat{S}_{t,j} = a_t\, 1\{\text{t approved}\}\, \tilde{y}_t(\Delta_j) \,/\, (\bar{a}\, \hat{G}(\Delta_j)) \quad (5)$$

And the feedback delivered on day $d_t+\Delta_j$ is the increment $\varphi_{t,j}=\hat{S}_{t,j}-\hat{S}_{t,j-1}$ with $\hat{S}_{t,0}=0$, so that Φ_d sums the increments due on day d. The first stage gives an unbiased estimate of the loss as soon as about half of the chargebacks have arrived. Later stages refine it, and an increment is negative when an early chargeback was up-weighted. The increments telescope,

$$\Sigma_{j=1}^{M} \varphi_{t,j} = \hat{S}_{t,M} = L_t \quad (6)$$

Because $\hat{G}(\Delta_M)=1$ and every chargeback arrives within the window. Whatever weighting error $\hat{G}$ introduces at early stages is therefore removed once the transaction is reconciled. We set Δ_1 to the smallest age with $\hat{G}\ge G_{min}$, matching the maturity floor of the calibration layer, and space Δ_2, …, Δ_M evenly up to the window.

4.4. Periodic Re-anchoring

The scalar state corrects the overall risk level, but drift also changes which strata and score ranges carry the loss. Every R=30 days the snapshot is therefore refitted on the transactions decided in the preceding C days, each weighted by $1/\hat{G}$ of its current age as in the fourth component above. The state θ_d is carried over unchanged, so that the controller corrects whatever bias remains in the refreshed snapshot.

4.5. Choice of the Gain

Near its operating point the released risk scales roughly in proportion to α', so the open-loop gain of (4) is close to η per day, and feedback arrives with a dead time of Δ_1 days. An integral loop with dead time Δ_1 is stable when $\eta\Delta_1<\pi/2$. We use $\eta=c/\Delta_1$ with $c=0.5$, a margin of three, and apply the same rule, with the age of the first feedback, to every controller we compare. Algorithm 1 summarizes the procedure. Serving costs one threshold comparison per

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

transaction, and a year of decisions on a stream of one million transactions takes about 0.25 s in our Python implementation.

Algorithm 1: Adaptive conformal risk control with progressive IPCW feedback

Input: scorer s, budget α, $\hat{G}$, stages $\Delta_1<\cdots<\Delta M$, gain η, period R, window C.

1: Fit A_{cap}, strata, $\bar{a}$, $\bar{n}$ on training data; $\theta \leftarrow \log \alpha$; fit the snapshot at deployment day d_0.

2: for each day d do

3: if $(d-d_0)$ mod R=0: refit the snapshot on days $[d-C, d-1]$ with weights $1/\hat{G}$(age).

4: $\tau \leftarrow$ thresholds of the snapshot at e^{θ} (water-filling + residual fill).

5: Review t if $a_t > A_{cap}$ or $g_t \geq \tau_{k(at)}$, else approve.

6: $\Phi \leftarrow$ sum of increments $\varphi_{t,j}$ with $d_t+\Delta_j=d$, by (5).

7: $N \leftarrow |\{t: d_t+\Delta_1=d\}|$; $\theta \leftarrow \min\{\theta_{max}, \theta+\eta(\alpha N-\Phi)/(\alpha\bar{n})\}$.

8: end for

5. Theoretical Properties

Let $F_T=\{t:d_t+\Delta_M \leq T\}$ be the transactions whose feedback is fully reconciled by day T, let $Q_T=\{t: d_t+\Delta_1 \leq T < d_t+\Delta_M\}$ be those with pending feedback, and let $\hat{S}_t^{(T)}$ be the latest estimate delivered for $t \in Q_T$.

Proposition 1 (trajectory certificate). Suppose every chargeback arrives within Δ_M days and $\hat{G}(\Delta_M)=1$. Then for every sequence of transactions, labels and delays, every estimate $\hat{G}$ and every T,

$$\Sigma_{t\in FT}(L_t - \alpha) \leq (\alpha\bar{n}/\eta)(\theta_1 - \min_{d\leq T+1} \theta_d) + \Sigma_{t\in QT}(\alpha - \hat{S}_t^{(T)}) \quad (7)$$

Proof. Write the clipping in (4) as a non-negative correction c_d, so that $\theta_{d+1}=\theta_d+\eta(\alpha N_d-\Phi_d)/(\alpha\bar{n})-c_d$. Summing over $d \leq T$ gives $\Sigma_d(\Phi_d-\alpha N_d) \leq (\alpha\bar{n}/\eta)(\theta_1-\theta_{T+1})$. By (6), $\Sigma_{d\leq T}\Phi_d=\Sigma_{t\in FT}L_t+\Sigma_{t\in QT}\hat{S}_t^{(T)}$, and $\Sigma_{d\leq T}N_d=|F_T|+|Q_T|$. Rearranging and bounding θ_{T+1} by the minimum of the trajectory gives (7).

The right-hand side of (7) can be computed by the operator at any time, so it serves as an auditable certificate. It also implies the long-run guarantee. Because the family reviews everything when e^{θ} is below the feasibility floor, decisions taken in that region release no loss, and θ can fall further only through feedback from earlier decisions. Hence $\min_d \theta_d$ is bounded below whenever the loss still reportable for already approved transactions is bounded. The pending term is at most $\alpha|Q_T|$, which covers $\Delta_M-\Delta_1$ days of volume. Dividing (7) by $|F_T|$ therefore gives $\limsup_T |F_T|^{-1}\Sigma_{t\in FT}L_t \leq \alpha$ under any drift. Proposition 1 never uses the accuracy of $\hat{G}$, which only affects how quickly information arrives. Without reconciliation the situation changes.

Proposition 2 (single-stage feedback). Let feedback be delivered once, at age Δ, with weight w (w=1 for naive feedback and $w=1/\hat{G}(\Delta)$ for IPCW), and let the delays be i.i.d. with distribution G, independent of the stream. The argument of Proposition 1 then bounds the long-run average of the reported losses $w a_t 1\{t \text{ approved}\}\tilde{y}_t(\Delta)/\bar{a}$. Their differences from $wG(\Delta)L_t$ form a bounded martingale difference sequence, so the true long-run risk is at most $\alpha/(wG(\Delta))$ almost surely, with equality whenever the controller is active.

Naive feedback at the age where half of the chargebacks have arrived therefore targets about 2α, and single-stage IPCW targets $\alpha\hat{G}(\Delta)/G(\Delta)$, which is correct only while the historical delay distribution still describes current disputes. Mature-only feedback ($\Delta=\Delta_M$) is unbiased but pays the full window as dead time. That forces a gain $\eta=c/\Delta_M$ four times smaller in our setting, so drift accumulates for months before it is corrected.

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

6. Experimental Setup

6.1. Transaction Streams

Sparkov. Long, timestamped public payment streams are rare, so we use the open-source Sparkov generator [29], as is common in this literature. Its inverse-CDF helper samples the time of day and the merchant category from the same uniform draw as the calendar date, which couples legitimate behaviour to the date: in unpatched output the share of morning transactions falls from 95% in the first eight months to 6% in the final year. We sample date, daypart and category from independent uniforms and otherwise keep the generator unchanged. Each stream has 450 cardholders over 2019–2020, and five streams are generated with different seeds. Sparkov-Std keeps the stock fraud process, in which each cardholder suffers one fraud burst on a random day. Sparkov-Drift shares the legitimate transactions but changes the fraud process in the second year. The episode rate rises linearly to three times its first-year level, and a growing share of episodes, 80% by December, uses a camouflaged profile with legitimate hours, a category mix tilted towards online merchants and amounts about twice the cardholder's usual level, mostly below the exposure cap. The scorer is a LightGBM model [30] trained on days 0–239 with 14 causal, label-free features: amount, category, hour, weekday, cardholder–merchant distance, amount ratio and z-score against the card's history, card velocity over 1 h, 24 h and 7 days, 24-h spend, time since the previous transaction and the card's night-time share. Static identity attributes such as age or city population are excluded, because each Sparkov cardholder is defrauded at most once and these attributes would let the model memorize customers that were already defrauded.

Handbook. The ULB/Worldline handbook stream [18] contains 1,754,155 transactions over 183 days, generated with compromised terminals and compromised customers as fraud mechanisms. We compute label-free customer and terminal windows of 1, 7 and 30 days, amount ratios and intra-day counts. We also compute terminal-risk windows of 7, 14 and 30 days that count only chargebacks that have already arrived under the delay model, so that delayed feedback also shapes the scorer's inputs. A LightGBM scorer is trained on days 7–59 and re-scores each replicate with that replicate's chargeback history. Table I summarizes the streams.

Table I. Streams and Operating Points

Quantity	Sparkov-Std	Sparkov-Drift	Handbook
Transactions	769,058	769,274	1,754,155
Fraud rate	0.55%	0.72%	0.84%
Horizon / deployment (days)	731 / 366	731 / 366	183 / 83
Scorer training days	[0, 240)	[0, 240)	[7, 60)
Scorer AUC (calib. / deploy.)	0.999 / 0.999	0.999 / 0.905	0.915 / 0.912
Deployment AP	0.942	0.702	0.754
Camouflaged share of deployment fraud	0%	30%	–
Exposure cap A_{cap}	\$609	\$609	\$172
Mean amount $\bar{a}$	\$71.57	\$71.59	\$53.64
Uncontrolled loss B_0 (\$/1k txn)	\$3,263	\$3,268	\$1,131
Budget α (\$/1k txn, αrel)	\$163 (5%)	\$163 (5%)	\$113 (10%)
Δ_1 / Δ_M (days)	32 / 120	31 / 120	10 / 36

Sparkov values are means over five generator seeds; AUC and AP refer to the fixed scorer.

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

6.2. Protocol

Chargeback delays are drawn as the ceiling of a Weibull variable with shape 1.5 and scale 40 days, truncated at a 120-day dispute window, for Sparkov. The handbook stream has a shorter horizon, so it uses scale 12 days and a 36-day window. $\hat{G}$ is the empirical distribution of training-period delays, which have all matured by deployment. Deployment covers days 365–730 (Sparkov) and 100–182 (handbook), with calibration windows of C=124 and C=39 days that start after the scorer's training period. The economics follow LARC: c_{cb}=\$15, c_r=\$2.50, c_f=\$0.50, q=0.9, K=4 strata and A_{cap} at the 99th training percentile. On Sparkov the budget is α_{rel}=0.05, the operating point used for this generator in the original LARC study. On the handbook stream we use α_{rel}=0.10, the smallest value in {0.05,0.1,0.2,0.3} at which static calibration reviews less than 2% of its own calibration window (0.05 would require 18%); it was fixed before any deployment outcome was observed. Δ_1 is the first age with $\hat{G}\geq 0.5$ (30–32 days on Sparkov, 10 days on the handbook stream), followed by evenly spaced stages up to the window. Re-anchoring uses R=30 days and the gain uses c=0.5. Sparkov results use 20 replicates per stream type (five generator seeds × four delay seeds), and handbook results use ten delay seeds. We report the long-run risk ratio (realized risk over the deployment period divided by α), the 90th percentile of the 30-day rolling risk ratio (P90), the review rate, dollar recall and net value (NV), and test differences with paired two-sided Wilcoxon signed-rank tests over replicates.

6.3. Compared Policies

Top-k reviews the 2% highest scores with a threshold set on the calibration window, and cost-sensitive Bayes reviews when the expected recovery exceeds the review cost [8]. LARC, static is the full calibration layer of [13], fitted once at deployment on the IPCW-weighted calibration window with G_{min}=0.5, following the variable-age protocol of that study; it is the strongest static baseline and the initial state of every adaptive controller. LARC, refresh refits the same layer every 30 days. The four controllers share the policy family, the re-anchoring and the gain rule, and differ only in their feedback. ACI-mature uses one stage at Δ_M, as rolling risk control would with complete labels [16]; ACI-naive uses one stage at Δ_1 without weighting; ACI-IPCW uses one stage at Δ_1 with weight $1/\hat{G}(\Delta_1)$; and Ours uses the progressive feedback of Section IV-C. An oracle receives every true loss at the end of the decision day. It is not deployable and shows what delay costs.

7. Results

7.1. Main Comparison

Table II reports all policies. On Sparkov-Drift the static layer was valid at deployment but released 2.63 times the budget over the year. The camouflaged episodes fall below the exposure cap and receive low scores, with a deployment AUC of 0.905 against 0.999 on the calibration window. Refitting every 30 days reduced the overshoot to 1.55 without removing it, because each refit learns from a window that is one to four months old and has no memory of the excess already released. Feedback control removed most of the remaining gap. Mature-only feedback reached 1.28, naive early feedback 1.84 (the censoring bias of Proposition 2), single-stage IPCW 1.03 ± 0.07 and progressive IPCW 1.01 ± 0.02. The proposed controller differed from every policy without IPCW feedback at $p<10^{-5}$. Against single-stage IPCW the mean risk did not differ significantly (p=0.22), but the tracking error |risk/α−1| fell from 0.057 to 0.021 ($p=4.8\times10^{-4}$), close to the oracle's 0.022. Top-k and cost-sensitive Bayes have no mechanism that links their thresholds to the budget, and they exceeded it by 53% and 88%.

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

Table II. Released Fraud Value, Tail Risk and Review Load

Policy	Risk/α	P90	Rev. %	NV/1k
Sparkov-Std				
Top-k (2%)	0.06 ± 0.04	0.15	2.0	2,628
Cost-sens. Bayes	0.16 ± 0.05	0.32	0.9	2,644
LARC, static	0.95 ± 0.14	1.55	1.1	2,505
LARC, 30-day refresh	0.99 ± 0.06	1.76	1.1	2,497
ACI-mature	0.94 ± 0.07	1.70	1.1	2,505
ACI-naive	1.61 ± 0.13	2.93	1.0	2,403
ACI-IPCW	0.93 ± 0.08	1.80	1.1	2,506
Ours (progressive)	0.95 ± 0.06	1.78	1.1	2,504
Oracle (instant)	0.93 ± 0.04	1.55	1.1	2,506
Sparkov-Drift				
Top-k (2%)	1.53 ± 0.22	2.99	2.1	3,133
Cost-sens. Bayes	1.88 ± 0.33	3.53	1.0	3,111
LARC, static	2.63 ± 0.30	4.40	1.1	2,973
LARC, 30-day refresh	1.55 ± 0.07	2.33	3.3	3,079
ACI-mature	1.28 ± 0.05	2.04	5.3	3,064
ACI-naive	1.84 ± 0.10	3.11	3.1	3,039
ACI-IPCW	1.03 ± 0.07	1.82	7.3	3,046
Ours (progressive)	1.01 ± 0.02	1.79	7.5	3,044
Oracle (instant)	1.02 ± 0.02	1.57	6.2	3,081
Handbook				
Top-k (2%)	0.86 ± 0.02	0.92	2.1	982
Cost-sens. Bayes	0.93 ± 0.02	0.99	1.5	989
LARC, static	1.00 ± 0.04	1.05	1.8	966
LARC, 30-day refresh	1.00 ± 0.02	1.06	1.8	967
ACI-mature	1.00 ± 0.02	1.06	1.8	967
ACI-naive	1.59 ± 0.05	1.95	1.4	896
ACI-IPCW	0.99 ± 0.03	1.07	1.8	966
Ours (progressive)	0.99 ± 0.01	1.07	1.8	966
Oracle (instant)	0.99 ± 0.01	1.05	1.8	967

Mean ± s.d. over replicates (20 per Sparkov stream, 10 for Handbook). Risk/α is the deployment-period released fraud value relative to the budget; P90 is the 90th percentile of the 30-day rolling ratio; NV in dollars per thousand transactions relative to approving everything.

Holding the budget under this drift is not free. The proposed controller reviewed 7.5% of the volume against 6.2% for the oracle and 1.1% for the static layer, with a dollar recall of 95.5%. Its NV was 1.2% below that of the oracle and within one standard deviation of every adaptive alternative. On this stream, the policies with higher NV, Top-k and Bayes, obtain it by exceeding the budget.

On the two streams without strong drift the controller costs nothing. On Sparkov-Std all calibrated policies stayed between 0.93 and 1.00 of the budget at the 1.1% review rate set by the exposure cap. On the handbook stream they held 0.99–1.00, and the proposed controller had the smallest dispersion among deployable policies (±0.015 against ±0.037 for the static layer; tracking error p=0.049). Naive early feedback overshot by 59–84% on every stream, which isolates censoring as a failure mode that does not depend on drift.

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

7.2. Dynamics under Drift

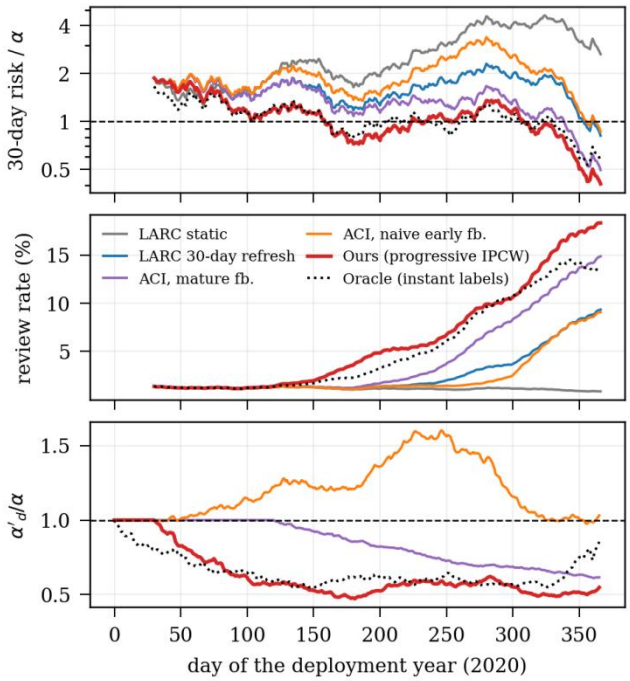


Figure 2. Second year of Sparkov-Drift, mean of 20 replicates. Top: 30-day rolling released fraud value relative to the budget. Middle: 30-day review rate. Bottom: nominal level α'_d relative to α for the feedback controllers. All policies start from the static LARC state.

Fig. 2 traces the deployment year of Sparkov-Drift. The static layer keeps its initial review rate while its 30-day risk climbs above four times the budget as camouflaged fraud grows. The proposed controller starts to tighten when its first feedback arrives on day 31, lowers the nominal level to about half of α by day 150, and then raises its review rate steadily, to 18% by December, as fraud intensity triples. After the first quarter its 30-day risk stays near or below the budget, and its trajectory follows the oracle closely. Naive feedback moves the nominal level in the wrong direction, with α'/α rising to 1.6, because censored losses look small. Mature-only feedback moves in the right direction about four months late.

7.3. Robustness to Slower Disputes

Table III. Dispute Delays 50% Longer after Deployment

	Sparkov-Drift		Sparkov-Std	
Policy	Risk/α	Rev. %	Risk/α	Rev. %
LARC, 30-day refresh	1.75 ± 0.07	2.7	1.10 ± 0.06	1.1
ACI-mature	1.39 ± 0.06	4.9	1.02 ± 0.05	1.1
ACI-naive	2.60 ± 0.17	1.8	2.21 ± 0.13	1.0
ACI-IPCW	1.48 ± 0.12	4.7	1.39 ± 0.11	1.0
Ours (progressive)	1.08 ± 0.03	7.4	1.02 ± 0.08	1.1
Oracle (instant)	1.04 ± 0.02	6.0	0.95 ± 0.04	1.1

$\hat{G}$ is estimated from training-period delays (scale 40 days); deployment delays use scale 60 days. Mean ± s.d. over 20 replicates.

Table III repeats the comparison when chargebacks slow down after deployment: the Weibull scale grows by 50% while $\hat{G}$ is still estimated from history, so $G(\Delta_1)$ falls from about 0.50 to 0.32. Single-stage IPCW then under-weights early chargebacks and reached 1.48 on Sparkov-Drift and 1.39 on Sparkov-Std, the inflation predicted by Proposition 2, and naive feedback reached

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

2.60. Progressive feedback held 1.08 ± 0.03 and 1.02 ± 0.08 and was significantly better than every deployable alternative on the drifting stream ($p<10^{-5}$). Its residual excess on Sparkov-Drift reflects slower information rather than bias: fewer chargebacks arrive by the first stage, so corrections lag more, and reconciliation removes the weighting error.

7.4. Sensitivity of the Feedback Loop

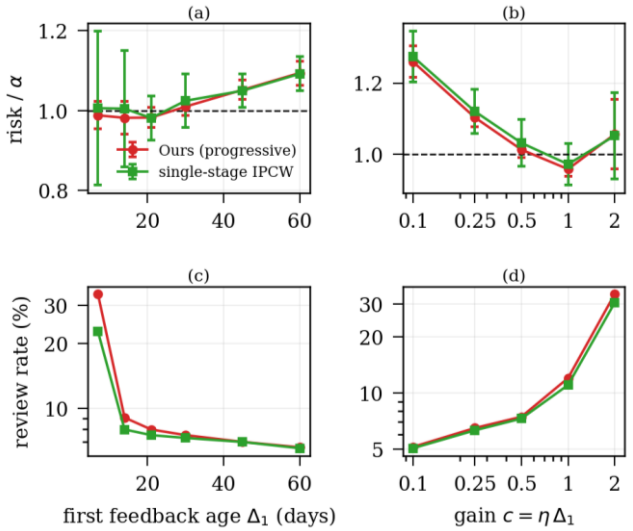


Figure 3. Sensitivity on Sparkov-Drift (mean ± s.d. over 20 replicates). (a), (c): first feedback age Δ_1, with later stages spread evenly up to 120 days and $\eta=0.5/\Delta_1$. (b), (d): gain $c=\eta\Delta_1$ at Δ_1 = 30–32 days.

Fig. 3 varies the two design choices of the loop on Sparkov-Drift. Moving the first stage earlier speeds up correction but raises the IPCW weights. At Δ_1=7 days the weight is about 14 and the review rate jumps to 34%, because single chargebacks trigger large corrections. Moving it later lowers review but lets risk drift upwards, to 1.09 at 60 days. The maturity floor G_{min}=0.5 inherited from the calibration layer lies at the knee of this trade-off. For the gain, $c\leq0.25$ reacts too slowly (1.10–1.26), c=1 meets the budget at 12% review, and c=2 starts to oscillate with 34% review. This agrees with the dead-time margin $\eta\Delta_1<\pi/2$. Progressive feedback had a lower dispersion than single-stage IPCW at every setting.

7.5. Ablations

Table IV. Ablations of the Proposed Controller

	Sparkov-Drift		Handbook	
Variant	Risk/α	Rev. %	Risk/α	Rev. %
Full controller	1.01 ± 0.02	7.5	0.99 ± 0.01	1.8
w/o re-anchoring	1.35 ± 0.08	23.2	1.00 ± 0.01	1.8
w/o IPCW in re-anchoring	1.07 ± 0.02	7.2	1.07 ± 0.02	1.7
score order (g=s)	1.01 ± 0.03	8.0	1.00 ± 0.02	1.8
single stratum (K=1)	1.01 ± 0.03	11.3	0.99 ± 0.02	1.8

Mean ± s.d. over 20 (Sparkov-Drift) and 10 (Handbook) replicates.

Table IV removes one component at a time. Without periodic re-anchoring the controller still corrects the overall level, but on Sparkov-Drift it must push a single stale snapshot far from its calibration point; risk rose to 1.35 and review tripled to 23.2%. Re-anchoring without IPCW biased the refreshed snapshots and added 5–8% risk on both streams. Replacing LARC's value order by the score order kept the risk at the budget but needed 8.0% instead of 7.5% review, and collapsing the strata into one needed 11.3%, so the water-filling allocation alone saves a

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

third of the review volume at equal risk. LARC's value ordering and amount stratification therefore remain load-bearing inside the adaptive controller: feedback corrects the risk level, while the LARC layer determines how little review is needed to reach it. On the handbook stream, where drift is mild, only the IPCW weighting of the snapshots matters.

7.6. Tightness of the Certificate

In all 50 runs of the proposed controller, the realized risk on the reconciled transactions stayed below the certificate (7). On average the realized ratios were 1.15 on Sparkov-Drift, 0.99 on Sparkov-Std and 0.99 on the handbook stream, against certificates of 1.27, 1.12 and 1.02. The reconciled set of Sparkov-Drift covers only the first eight months of deployment, during which the controller was still catching up with the drift. The full-year ratio of 1.01 shows how the integral action repays that early excess.

8. Discussion and Limitations

The guarantee is a long-run statement: it allows months above the budget if later months repay them, and 52% of the 30-day windows of the proposed controller exceeded α on Sparkov-Drift, against 89–97% for the static and refreshed layers. An operator who needs control within every window can run the controller at a reduced target or move to high-probability risk control [11] at the price of more review. Holding the budget under strong drift also raises review volume, and a hard capacity limit can make the budget infeasible; the certificate (7) then quantifies the shortfall instead of hiding it. Our experiments rely on synthetic streams because long, timestamped and labelled payment data are not public, and the camouflage scenario is one form of drift among many. The delay model assumes that chargebacks are independent of the transaction and that every fraud is disputed within the window. Proposition 1 needs only the second assumption, but the speed of correction depends on both. Finally, reviewed transactions reveal their labels sooner than approved ones in practice, and feeding those labels into the re-anchoring step is a natural extension.

9. Conclusion

Monetary risk control for payment decisions has to cope with a fraud mix that drifts and with evidence that arrives late and incomplete. We kept a conformal calibration layer as a policy family, steered its nominal level by integral feedback, and fed that loop with progressive IPCW estimates that telescope to the true loss once disputes close. The resulting controller carries an operator-computable certificate that depends neither on the stream nor on the accuracy of the delay model. On drifting streams it held the released fraud value at the budget where static and periodically refreshed calibration exceeded it by 55–163%, and it stayed robust when disputes slowed down. Per-window guarantees, explicit capacity constraints and label-dependent delays are the next steps.

References

[1] Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., & Bontempi, G. (2018). Credit card fraud detection: A realistic modeling and a novel learning strategy. IEEE Transactions on Neural Networks and Learning Systems, 29(8), 3784–3797. https://doi.org/10.1109/TNNLS.2017.2736643

[2] Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., & Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. Expert Systems with Applications, 41(10), 4915–4928. https://doi.org/10.1016/j.eswa.2014.02.026

[3] Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. ACM Computing Surveys, 46(4), Article 44. https://doi.org/10.1145/2523813

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

[4] Lu, J., Liu, A., Dong, F., Gu, F., Gama, J., & Zhang, G. (2019). Learning under concept drift: A review. IEEE Transactions on Knowledge and Data Engineering, 31(12), 2346–2363. https://doi.org/10.1109/TKDE.2018.2876857

[5] Grzenda, M., Gomes, H. M., & Bifet, A. (2020). Delayed labelling evaluation for data streams. Data Mining and Knowledge Discovery, 34, 1237–1266. https://doi.org/10.1007/s10618-019-00654-y

[6] Žliobaitė, I. (2010). Change with delayed labeling: When is it detectable? In Proceedings of the IEEE International Conference on Data Mining Workshops (ICDMW) (pp. 843–850). https://doi.org/10.1109/ICDMW.2010.49

[7] Correa Bahnsen, A., Aouada, D., Stojanovic, A., & Ottersten, B. (2016). Feature engineering strategies for credit card fraud detection. Expert Systems with Applications, 51, 134–142. https://doi.org/10.1016/j.eswa.2015.12.030

[8] Elkan, C. (2001). The foundations of cost-sensitive learning. In Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI) (pp. 973–978).

[9] Carcillo, F., Dal Pozzolo, A., Le Borgne, Y.-A., Caelen, O., Mazzer, Y., & Bontempi, G. (2018). SCARFF: A scalable framework for streaming credit card fraud detection with Spark. Information Fusion, 41, 182–194. https://doi.org/10.1016/j.inffus.2017.09.001

[10] Vovk, V., Gammerman, A., & Shafer, G. (2005). Algorithmic learning in a random world. Springer.

[11] Bates, S., Angelopoulos, A., Lei, L., Malik, J., & Jordan, M. I. (2021). Distribution-free, risk-controlling prediction sets. Journal of the ACM, 68(6), Article 43. https://doi.org/10.1145/3478535

[12] Angelopoulos, A. N., Bates, S., Fisch, A., Lei, L., & Schuster, T. (2024). Conformal risk control. In Proceedings of the 12th International Conference on Learning Representations (ICLR).

[13] Zi, B., Wang, J., & Zhang, Y. (2026). Latency-aware risk control for budget-constrained payment fraud decisioning. In Proceedings of ICCIA 2026 (pp. 257–269).

[14] Gibbs, I., & Candès, E. (2021). Adaptive conformal inference under distribution shift. Advances in Neural Information Processing Systems, 34, 1660–1672.

[15] Gibbs, I., & Candès, E. J. (2024). Conformal inference for online prediction with arbitrary distribution shifts. Journal of Machine Learning Research, 25(162), 1–36.

[16] Feldman, S., Ringel, L., Bates, S., & Romano, Y. (2023). Achieving risk control in online learning settings. Transactions on Machine Learning Research.

[17] Angelopoulos, A. N., Candès, E. J., & Tibshirani, R. J. (2023). Conformal PID control for time series prediction. Advances in Neural Information Processing Systems, 36.

[18] Le Borgne, Y.-A., Siblini, W., Lebichot, B., & Bontempi, G. (2022). Reproducible machine learning for credit card fraud detection—Practical handbook. Université Libre de Bruxelles. https://github.com/Fraud-Detection-Handbook/fraud-detection-handbook

[19] Deng, J. (2026). When can fraud operations authorize automation? A decision-support framework for fresh audit evidence and review workload. arXiv:2608.08577.

[20] Zhu, R., Zhang, X., Wang, T., Liao, J., Chung, S. H., & Zhang, X. (2026). DISCO: Decoupling representation learning and risk control for reliable credit card fraud detection. Decision Support Systems, 208, Article 114717. https://doi.org/10.1016/j.dss.2026.114717

[21] Tibshirani, R. J., Foygel Barber, R., Candès, E. J., & Ramdas, A. (2019). Conformal prediction under covariate shift. Advances in Neural Information Processing Systems, 32, 2530–2540.

[22] Foygel Barber, R., Candès, E. J., Ramdas, A., & Tibshirani, R. J. (2023). Conformal prediction beyond exchangeability. The Annals of Statistics, 51(2), 816–845. https://doi.org/10.1214/23-AOS2276

[23] Robins, J. M., Rotnitzky, A., & Zhao, L. P. (1994). Estimation of regression coefficients when some regressors are not always observed. Journal of the American Statistical Association, 89(427), 846–866. https://doi.org/10.1080/01621459.1994.10476818

[24] Candès, E. J., Lei, L., & Ren, Z. (2023). Conformalized survival analysis. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 85(1), 24–45. https://doi.org/10.1093/jrsssb/qkac004

ISSN: 3080-9843 (Print), ISSN: 3080-9851 (Online)

[25] Joulani, P., György, A., & Szepesvári, C. (2013). Online learning under delayed feedback. In Proceedings of the 30th International Conference on Machine Learning (ICML), PMLR (Vol. 28, No. 3, pp. 1453–1461).

[26] El Halabi, L., & Brandt, A. (2026). Adaptive conformal inference under delayed feedback: Coverage guarantees and a delay-to-memory diagnostic. arXiv:2609.07251.

[27] Chapelle, O. (2014). Modeling delayed feedback in display advertising. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD) (pp. 1097–1105). https://doi.org/10.1145/2623330.2623634

[28] Yasui, S., Morishita, G., Fujita, K., & Shibata, M. (2020). A feedback shift correction in predicting conversion rates under delayed feedback. In Proceedings of the Web Conference (WWW) (pp. 2740–2746). https://doi.org/10.1145/3366423.3380032

[29] Sparkov data generation. (2022). GitHub repository (Commit b5eb45c). https://github.com/namebrandon/Sparkov_Data_Generation

[30] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems, 30, 3146–3154.