ANYTIME-VALID DETECTION OF SCALE-REGIME DRIFT FOR CONFORMAL TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Conformal prediction intervals for time series can silently lose coverage when the data-generating process switches between scale regimes. Fixed calibration windows then underestimate uncertainty, while continuously adaptive methods often recover coverage only by substantially widening the intervals. We propose CREST (Conformal Recalibration via an Anytime-valid Scale-drift detector), a method that detects scale changes online and uses the resulting alarms to trigger conformal recalibration. CREST applies a blockwise e-process to seasonally whitened and locally standardized residuals, which yields anytime-valid false-alarm control under a local-scale null. To preserve this guarantee across repeated detector restarts, we introduce a budgeted restart scheme that allocates the false-alarm budget harmonically across segments. Experiments on real financial and weather time series show that CREST restores coverage close to the nominal level while maintaining interval widths comparable to a rolling baseline and requiring no trained state-space model. On car-index and weather data, CREST improves empirical coverage from 64.9% to 88.7% and from 77.1% to 89.8%, respectively, bringing both close to the nominal 90% target.

1 INTRODUCTION

Reliable uncertainty quantification is essential for deploying time-series forecasting systems in high-stakes applications. Conformal prediction has emerged as a powerful framework for constructing prediction intervals with finite-sample guarantees and minimal assumptions on the underlying forecasting model. However, standard conformal prediction methods typically rely on the implicit assumption that past calibration errors remain representative of future errors. This assumption is often violated in practice. In particular, the scale of forecasting residuals can change abruptly over time, leading to systematic undercoverage during high-variability periods and unnecessarily wide intervals after the process returns to a calmer regime.

This paper focuses on scale-regime drift in conformal time-series forecasting: changes in the local scale of forecast residuals after predictable temporal structure has been removed. Such changes are common in financial, environmental, and operational time series, and may occur even when the conditional mean remains stable. Therefore, point forecasts can remain accurate while uncertainty estimates become stale. The key challenge is to determine when the current calibration set is no longer representative of the present regime, while controlling false alarms under continuous monitoring.

To address this problem, we propose CREST, an anytime-valid scale-drift detection and conformal recalibration framework. Figure 1 provides an illustrative example of this event-triggered mechanism. CREST first removes predictable temporal dependence and locally standardizes the forecasting residuals. It then aggregates squared standardized residuals into non-overlapping blocks and constructs an e-process from the resulting block-energy statistics. A geometric mixture over candidate change locations allows the detector to identify scale increases without knowing the change time in advance. Since the e-process can be monitored at arbitrary stopping times, the detector provides uniform-in-time false-alarm control within each monitoring segment.

Repeated recalibration introduces an additional multiple-testing challenge: after each alarm, the detector is restarted with a new baseline, and applying the same threshold repeatedly can invalidate global error control. CREST addresses this issue by allocating a summable false-alarm budget across

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.

successive segments. Combining within-segment anytime-validity with a union bound over restarts yields global false-alarm control across the entire monitoring sequence. When an alarm occurs, stale conformity scores are discarded and the calibration set is rebuilt using observations from the newly detected regime.

Our theoretical results establish false-alarm control under a local-scale null and characterize the validity of the restart mechanism. The method is modular and can be combined with different base forecasters without requiring model retraining after every alarm. Experiments on synthetic and real-world time series show that CREST improves coverage under abrupt scale changes while maintaining competitive interval widths compared with fixed, rolling, and continuously adaptive calibration methods. Additional analyses demonstrate the effects of block size, heavy-tailed innovations, and repeated detector restarts.

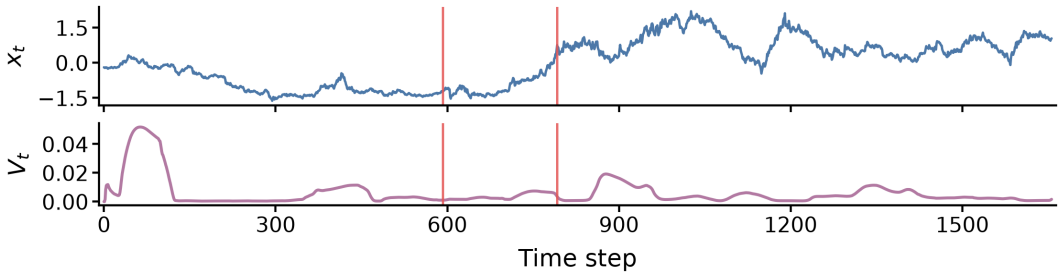


Figure 1: Top: the standardized CarIndex channel x_t; vertical lines mark the first two CREST alarms. Bottom: V_t, the 96-step rolling variance of the scale-energy ratio θ_t (Sec. 5).

In summary, our contributions are:

- **Anytime-valid scale-drift detection.** We develop an e-process-based detector for identifying changes in the local scale of time-series forecasting residuals, with uniform-in-time false-alarm control within each monitoring segment.
- **Global control under repeated recalibration.** We introduce a restart-aware false-alarm budget that preserves global control under repeated, data-dependent detector restarts and conformal recalibration.
- **Event-triggered conformal forecasting.** We combine scale-drift detection with event-triggered conformal recalibration and demonstrate improved coverage-efficiency trade-offs under nonstationary and heavy-tailed time series.

2 Related work

Conformal prediction for non-stationary time series. Adaptive conformal inference (ACI) (Gibbs & Candes, 2021) updates its quantile online and guarantees long-run average coverage. On drifting series it over-covers and widens intervals. Conformal PID control (Angelopoulos et al., 2023) reaches the same long-run guarantee with a proportional–integral update. It offers no explicit change-point signal. ECI (Wu et al., 2025) smooths the quantile-loss gradient and tracks miscoverage tightly, but under-covers under drift (Sec. 5). KOWCPI (Li et al., 2025a) gives conditional coverage under mixing through kernel weighting. CQACP (Yu et al., 2026) stabilizes sequential intervals with a Cornish–Fisher quantile model.

CPTC (Sun & Yu, 2025) is closest in spirit to our work. It combines a state-prediction model with online conformal prediction to handle potentially arbitrary changes in latent states. In contrast, CREST is specifically designed to detect scale-regime drift. It does not require a learned state model, and its alarms are supported by anytime-valid guarantees. ResCP (Neglia et al., 2026) replaces the calibration history with a reservoir-based notion of state similarity, but does not produce an explicit change-point signal. Liang et al. (Liang et al., 2026) establishes minimax-optimal training-conditional regret guarantees under both abrupt change points and smooth drift. Their objective is regret-optimal tracking, whereas ours is to construct an anytime-valid detection signal that triggers conformal recalibration.

Online change-point detection. E-process and e-detector methods (Ramdas et al., 2023; Shin et al., 2023) control type-I error at every stopping time. They have been applied to change-point detection (Shekhar & Ramdas, 2023) and prediction-powered inference (Csillag et al., 2025). Kalinke & Gavioli-Akilagun (2025) gives minimax-optimal nonparametric online detection via random Fourier features. Shaer et al. (2026) fixes the reference-set contamination of conformal test martingales. Conformal test martingales have also triggered model retraining (Vovk et al., 2021) and, in weighted form, monitored changepoint hypotheses (Prinster et al., 2025). These approaches primarily monitor violations of exchangeability; in contrast, CREST monitors scale-regime energy and uses it to trigger conformal recalibration. Our harmonic restart budget is related to group-sequential alpha spending (Lan & DeMets, 1983) and online alpha-investing (Foster & Stine, 2008; Wang & Ramdas, 2022). Finally, robust quickest-change-detection methods (Basseville & Nikiforov, 1993; Lai, 1995) motivate our study of performance under heavy-tailed innovations.

Test-time adaptation and non-stationary forecasting. TAFAS (Kim et al., 2025) and PETSA (Chen et al., 2025) adapt pretrained forecasters at test time but do not provide uncertainty quantification. MOOE (Zhao et al., 2025) combines online and offline experts with regret guarantees, while TOT (Li et al., 2025b) provides an online forecasting framework with guarantees based on latent variables. In contrast, we formulate adaptation as a calibration problem and use an anytime-valid detector to determine when recalibration is necessary.

Table 1 positions our method relative to the closest lines of work. CREST combines an explicit scale-change signal, a global anytime-valid guarantee that remains valid under repeated re-baselining, and event-triggered conformal recalibration. To the best of our knowledge, this combination has not been studied in prior work.

Table 1: Positioning against closest work.

Method	Drift	Anytime-valid	State model	Triggered conformal
ACI (Gibbs & Candes, 2021)	general shift	no (avg.)	no	no
Conformal PID (Angelopoulos et al., 2023)	general shift	no (long-run)	no	no
ECI (Wu et al., 2025)	general shift	no (avg.)	no	no
WATCH (Prinster et al., 2025)	changepoint	yes	no	monitoring
CPTC (Sun & Yu, 2025)	hidden-state	no	trained	implicit
ResCP (Neglia et al., 2026)	non-stationary	no	reservoir	no
e-CPD (Shekhar & Ramdas, 2023)	mean/general	yes	no	no
CREST	**scale-regime**	**yes (global)**	**no**	**yes**

3 PROBLEM SETUP

We consider the problem of online one-step-ahead prediction interval construction for a univariate time series $(x_t)_{t\geq 1}$. At time t, given the observed history $x_{1:t} := (x_1, \ldots, x_t)$, the goal is to output a prediction interval $C_{t+1} \subseteq \mathbb{R}$ for the next observation x_{t+1} with nominal coverage $1-\alpha$, using only information available up to time t. A base forecaster first produces a point prediction $\hat{x}_{t+1}$, and conformal calibration then maps the resulting nonconformity score to a prediction interval.

Under exchangeability, a fixed calibration quantile is sufficient to guarantee valid coverage. However, exchangeability is often violated under scale-regime drift: after an increase in variance, fixed-window calibration typically undercovers, whereas continuously adaptive methods such as ACI often respond too slowly and yield overly wide intervals. The central challenge is therefore to detect scale changes rapidly, reconstruct a locally relevant calibration set, and control the false-alarm rate across repeated resets.

To address these challenges, we propose CREST, which combines online scale-drift detection with conformal recalibration to enable rapid adaptation after variance shifts while controlling false alarms across repeated restarts. Details of the procedure are provided in Section 4. For multivariate time series, the same procedure is applied independently to each dimension, and we omit the channel index for notational simplicity.

4 THE METHOD

Figure 2 provides an overview of CREST, which consists of three modules. (A) A seasonal autoregressive (AR) whitening module, together with a local scale estimator, transforms the raw time series into standardized residuals. (B) The core detection module is a block-energy e-process, geometrically mixed over candidate block start times, which triggers an alarm once the evidence for a new scale regime exceeds a threshold determined by the false-alarm budget. (C) Each alarm induces a restart: CREST re-estimates the local scale from a short post-alarm window, resets the conformal calibration buffer, and launches a fresh e-process for the next segment. Operationally, the alarm times partition the stream into consecutive segments; false-alarm control across repeated restarts is handled by a separate budgeted restart scheme. Sections 4.1–4.4 provide the details of these components.

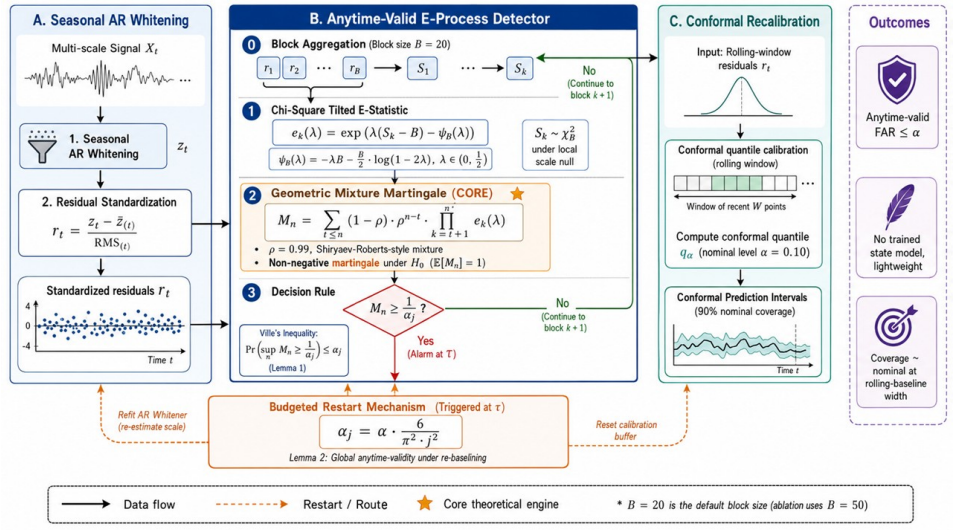


Figure 2: CREST: the anytime-valid detection and recalibration loop.

4.1 MODULE A: SEASONAL WHITENING AND LOCAL STANDARDIZATION

Given the raw series $(x_t)_{t\ge1}$, we first apply a seasonal AR whitening model to remove serial dependence and known seasonal structure, producing whitened residuals $(z_t)_{t\ge1}$. Let $\widehat{\sigma}$ be the RMS of the whitened residuals on the current calibration window; after each alarm it is re-estimated from the post-alarm window. All downstream detection uses the standardized residuals $r_t = z_t/\widehat{\sigma}$.

This separation is important, since a variance jump in the raw series can be confounded with serial dependence or transient spikes, while after whitening and standardization, scale changes are more directly reflected in the second-order behavior of r_t.

4.2 MODULE B: ANYTIME-VALID E-PROCESS DETECTOR

In Module B, we construct a sequential detector for identifying scale changes within the current segment. The local-scale null H_0 is that no new scale change has occurred since the last detector reset. Under this null, the standardized residuals produced by Module A are approximately white with unit variance. We exploit this structure to build a blockwise anytime-valid e-process that accumulates evidence against the no-change hypothesis. We present Module B in three steps: block construction, blockwise e-statistics and geometric mixture, and the alarm rule.

Block construction. Starting from the standardized residuals produced by Module A, the detector accumulates evidence for a scale change at the level of blocks rather than individual time steps. Specifically, we partition the residual stream into non-overlapping blocks of size B and compute the

block energy

$$S_k = \sum_{t \in \text{block } k} r_t^2.$$

This blockwise construction serves two purposes. First, per-step detectors are often overly sensitive to heavy-tailed outliers or transient spikes, whereas aggregating squared residuals within a block yields a more stable summary of second-order behavior. Second, under the local-scale null, the whitening and local standardization in Module A make the residuals approximately white with unit variance, so the block energy S_k approximately follows a χ^2_B distribution (Casella & Berger, 2002).

Blockwise e-statistics and geometric mixture. This approximation yields a tractable working null whose log-moment-generating function (log-MGF) is available in closed form, allowing us to define a blockwise e-statistic

$$e_k(\lambda) = \exp\{\lambda(S_k - B) - \psi_B(\lambda)\},$$

where $\lambda \in (0, 1/2)$ is a tuning parameter and $\psi_B(\lambda) = -\lambda B - \frac{B}{2}\log(1-2\lambda)$ is the log-MGF of $S_k - B$ under the working null. This normalization ensures that $e_k(\lambda)$ has unit mean under the null and is therefore a valid e-statistic (Ramdas et al., 2023).

Because the onset of a new scale regime is unknown, the detector does not commit to a single candidate changepoint. Instead, it forms a Shiryaev–Roberts-style (Shiryaev, 1963; Roberts, 1966; Shin et al., 2023) geometric mixture over candidate block start times,

$$M_n = \sum_{t \geq 0} (1-\rho)\,\rho^t \prod_{k=t+1}^{n} e_k(\lambda), \tag{1}$$

where $\rho \in (0, 1)$ controls how quickly older evidence is discounted. Smaller values of ρ place more weight on recent blocks and therefore yield shorter effective memory, whereas larger values retain evidence over longer horizons. The parameter λ controls sensitivity to jump size: larger values place more emphasis on substantial variance increases, while smaller values produce a more diffuse response.

Alarm rule and online recursion. The detector raises an alarm once the accumulated evidence statistic $A_n = M_n - \rho^{n+1}$ exceeds the threshold $1/\alpha$, where α denotes the false-alarm budget. This statistic admits the recursion $A_n = (1-\rho)\rho^n + e_n(\lambda)A_{n-1}$, which enables an efficient online update without revisiting past blocks. In practice, we evaluate this recursion in log space for numerical stability. Since $A_n \leq M_n$ for all n, it suffices to control the boundary-crossing probability of the dominating mixture e-process M_n. Equivalently, the alarm condition $A_n \geq 1/\alpha$ can be written as $M_n \geq 1/\alpha + \rho^{n+1}$, and we upper-bound this event by the simpler crossing event $M_n \geq 1/\alpha$.

The following Lemma 1 shows that, within any fixed segment, the proposed alarm rule controls false alarms uniformly over time under the null. The proof of Lemma 1 is provided in Appendix H.

Lemma 1 (Per-segment anytime validity)**.** *Under the null model, (M_n) is a nonnegative martingale with $\mathbb{E}[M_n] = 1$, and $A_n \leq M_n$ for all n. Therefore, $\{\exists n : A_n \geq 1/\alpha\} \subseteq \{\exists n : M_n \geq 1/\alpha\}$. By Ville's inequality,*

$$\mathbb{P}_{H_0}(\exists n : M_n \geq 1/\alpha) \leq \alpha,$$

and hence

$$\mathbb{P}_{H_0}(\exists n : A_n \geq 1/\alpha) \leq \alpha.$$

Thus, alarming whenever $A_n \geq 1/\alpha$ controls the false-alarm probability uniformly over all stopping times within a segment.

4.3 Restart mechanism and global false-alarm control

After the e-process associated with a segment raises an alarm, CREST re-estimates the local scale using a post-alarm window of length R, and then starts a new e-process for the next segment. Let u_j denote the start time of segment j, with $u_1 = 0$, and let τ_j denote the time of the j-th alarm. Segment j consists of a sequence of consecutive, non-overlapping blocks from u_j to τ_j, and its e-process is run at level α_j. The next segment therefore begins at $u_{j+1} = \tau_j + R$.

Restarting the detector introduces a multiple-testing problem across segments. If each segment were monitored at the same fixed nominal level, independently of the others, then the probability of at least one false alarm would accumulate over restarts and could exceed the target level. To avoid this inflation, we allocate a summable false-alarm budget across segments.

The next lemma extends the per-segment anytime validity of Lemma 1 to the full restarted procedure. In particular, it shows that if the segment-wise alarm levels satisfy $\sum_{j\geq 1}\alpha_j \leq \alpha$, then the probability of ever raising a false alarm over the entire monitoring horizon remains controlled by α. The proof of Lemma 2 is provided in Appendix H.

Lemma 2 (Global anytime validity across restarts)**.** *Suppose the local null holds within every segment. If segment j is monitored at level α_j with $\sum_{j\geq 1}\alpha_j \leq \alpha$, and an alarm is raised in segment j whenever $A_n^{(j)} \geq 1/\alpha_j$, then the probability of making at least one false alarm over all segments is at most α.*

In practice, we use the harmonic budget $\alpha_j = \alpha \cdot \frac{6}{\pi^2} j^{-2}$, which ensures $\sum_{j\geq 1}\alpha_j \leq \alpha$ and thus yields global false-alarm control by Lemma 2. Empirically, Appendix E shows that fixed-level restarting inflates the global false-alarm rate above the nominal 0.05, whereas the proposed budgeted restart scheme keeps it below this threshold. It should be also noted that, because both the AR parameters and the local scale are estimated via plug-in procedures, the chi-square null used by the detector is only approximate; the effect of this approximation is evaluated in Sec. 5.2.

4.4 Module C: Recalibration and Conformal Interval Construction

After an alarm, CREST discards pre-alarm conformity scores and rebuilds the calibration buffer from the subsequent observations. The next result shows that, if the post-reset scores are locally exchangeable within the new segment, then the recalibrated split-conformal interval retains the usual finite-sample coverage guarantee conditional on the reset time. The proof of Proposition 1 is provided in Appendix H.

Proposition 1 (Conditional coverage after a reset)**.** *For an alarm time τ, let $c_{\tau+1}, \ldots, c_{\tau+m}$ be the first m conformity scores collected after the reset, and let $c_{\tau+m+1}$ be the next test score. If, conditional on $\mathcal{F}_\tau$, the $m+1$ scores $c_{\tau+1}, \ldots, c_{\tau+m}, c_{\tau+m+1}$ are exchangeable, then the recalibrated split-conformal interval $[\hat{y}_{\tau+m+1} \pm \hat{q}_{1-\alpha}]$ constructed from the buffer satisfies*

$$\mathbb{P}(y_{\tau+m+1} \in [\hat{y}_{\tau+m+1} \pm \hat{q}_{1-\alpha}] \,|\, \mathcal{F}_\tau) \geq 1-\alpha,$$

where $\hat{q}_{1-\alpha}$ is the usual split-conformal empirical quantile computed from $c_{\tau+1}, \ldots, c_{\tau+m}$.

We emphasize that Proposition 1 is a local post-reset guarantee, not an unconditional coverage claim over the entire nonstationary sequence. Since the alarm time is data dependent, global exchangeability need not hold. Instead, the detector identifies a plausible segment boundary, after which the calibration buffer is rebuilt from post-reset observations according to a fixed rule. Under the assumption of local score exchangeability within the new segment, the standard split-conformal argument applies conditional on the reset. We assess this assumption empirically in Sec. 5.

5 Experiments

5.1 Real data analysis

In this section, we organize the real-data evaluation around four questions: (**RQ1**) Does CREST improve the coverage–width trade-off over existing baselines? (**RQ2**) Are its gains concentrated in regimes with strong scale nonstationarity, and does the alarm stream provide an interpretable descriptor of such behavior? (**RQ3**) Does CREST transfer beyond the AR forecaster and remain competitive under an external evaluation protocol? (**RQ4**) How sensitive is CREST to its key design choices?

Datasets and implementation details. We evaluate CREST on five multivariate datasets spanning both financial and periodic domains: Future, CarIndex, Exchange, ETTm2, and Weather. Seasonal-AR whitening is fitted independently for each channel, with 10, 8, 8, 7, and 21 channels respectively.

For each channel, the first 20% of the series (with a minimum of 512 points) is used to fit the seasonal-AR whitener and initialize the conformal calibration quantile. The remaining observations are then processed online. Unless otherwise stated, the detector uses $B = 20$, $\lambda = 0.25$, $\rho = 0.99$, and $R = 5B = 100$. The seasonal lag is set to $\ell = 5$ for Future, CarIndex, and Exchange, to $\ell = 96$ for ETTm2, and to $\ell = 144$ for Weather, with AR order fixed at 5 throughout. The conformal calibration buffer has size $W = 200$, and after each alarm we use a 20-step fallback buffer until enough post-reset observations have accumulated.

Competing methods. We wrap the same fixed one-step-ahead AR forecaster with 90% conformal intervals under seven calibration strategies: (i) fixed split conformal (Fixed); (ii) rolling conformal (Rolling); (iii) rolling conformal with buffer resets triggered by CREST alarms (CREST); (iv) rolling conformal with buffer resets triggered by conformal-martingale alarms (CM; Vovk et al., 2021); (v) adaptive conformal inference (ACI); (vi) conformal PID control (PID; Angelopoulos et al., 2023); and (vii) error-quantified conformal inference (ECI; Wu et al., 2025). All methods share the same base forecaster and differ only in how conformal calibration is updated over time. We additionally evaluated a CUSUM detector, but on these datasets its threshold is so conservative that its trajectory coincides with Rolling, so it is reported only in Appendix A.

Coverage and interval-width comparison (RQ1). Table 2 reports the empirical coverage and median prediction-interval width for seven methods. Across datasets, CREST achieves empirical coverage close to the nominal 90% target while remaining competitive in interval width. The largest gains appear on datasets with pronounced scale nonstationarity. For example, on CarIndex and Weather, Fixed drops to 64.9% and 77.1% coverage, whereas CREST recovers 88.7% and 89.8%, respectively. PID attains nominal coverage but needs intervals 5–87% wider than CREST. ECI produces tighter intervals, yet under-covers by up to 8%. CM, which monitors exchangeability, matches CREST in coverage and width, suggesting that the trigger interface, not the monitored score, drives the result.

Table 2: Empirical coverage and median interval width on five benchmarks. Each entry reports empirical coverage, with median interval width shown in parentheses.

Method	Future	CarIndex	Exchange	ETTm2	Weather
CREST	0.891 (0.96)	0.887 (2.03)	0.891 (1.88)	0.901 (1.05)	0.898 (2.33)
Fixed	0.893 (1.00)	0.649 (1.00)	0.829 (1.00)	0.872 (1.00)	0.771 (1.00)
Rolling	0.887 (0.89)	0.870 (1.88)	0.890 (1.87)	0.900 (1.04)	0.897 (2.30)
CM	0.888 (0.95)	0.883 (1.99)	0.891 (1.89)	0.900 (1.04)	0.897 (2.30)
ACI	0.981 (2.80)	0.976 (3.16)	0.982 (3.83)	0.983 (2.00)	0.981 (5.18)
PID	0.900 (1.79)	0.901 (2.69)	0.900 (2.12)	0.900 (1.12)	0.902 (2.45)
ECI	0.893 (0.91)	0.879 (1.94)	0.876 (2.01)	0.821 (0.80)	0.864 (1.78)

Scale-entanglement analysis (RQ2). To better understand where these gains arise, we further stratify monitoring windows by a scale-entanglement statistic. Define the scale-energy ratio $\theta_t = \frac{E_t^{\mathrm{det}}}{E_t^{\mathrm{det}} + E_t^{\mathrm{tr}}}$, where E_t^{det} and E_t^{tr} are the window-64 variances of the first-difference and level components, respectively. We then define scale-energy entanglement as the window-96 rolling variance of θ_t. Intuitively, this statistic captures the local instability in the relative dominance between short-term fluctuation energy and level-variation energy. Larger values indicate that the series switches more frequently between different scale regimes, and are therefore more likely to cause fixed calibration schemes to fail. As shown in Fig. 3, CREST maintains stable coverage across all entanglement bucket values. The advantage is especially pronounced in the high-entanglement regime, where the coverage of CREST remains much closer to the nominal target of 0.9, while competing methods deviate more substantially. This suggests that CREST retains strong calibration stability even in challenging settings with frequent scale-regime switching.

Alarm stream as a domain descriptor (RQ2). The alarm stream also separates the financial and periodic families. Financial series shift scale regime 8–22$\times$ more often than periodic benchmarks: CarIndex and Future exhibit 1.64 and 1.09 alarms per 1000 steps, against 0.14 and 0.07 for the pe-

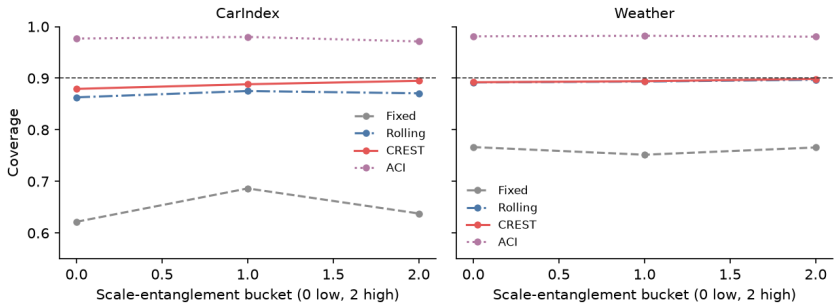


Figure 3: Coverage by scale-entanglement bucket on CarIndex and Weather.

riodic benchmarks (Fig. 4). Alarm frequency is thus a simple, calibration-free descriptor of domain behavior.

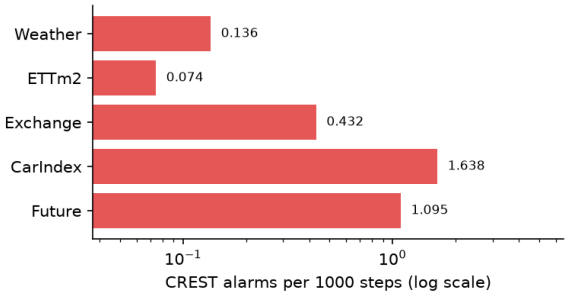


Figure 4: CREST alarm rate per 1000 steps.

Forecaster-agnostic plug-in (RQ3a). Our recalibration module is not tied to the AR forecaster. We rerun the same pipeline with DLinear (Zeng et al., 2023), PatchTST (Wu et al., 2023), TimeMixer (Wang et al., 2024), TimeMixer++ (Wang et al., 2025), and WPMixer (Murad et al., 2025), while keeping the detector unchanged and driven by AR-whitened residuals. Across 18 forecaster–dataset pairs, CREST outperforms rolling recalibration in 15 cases, with gains of up to 2.2 coverage points (median: 0.1), and improves over fixed calibration by as much as 35 points (WPMixer on CarIndex: 42.6%→77.8%). Coverage is near nominal with AR and remains within a few points of nominal with DLinear. Table 3 reports the three drifting datasets; the full comparison is in Appendix E.

Table 3: CREST coverage with AR and five plug-in forecasters on three representative datasets.

Forecaster	CarIndex	Weather	ETTm2
AR	0.887	0.898	0.901
DLinear	0.876	0.895	0.899
PatchTST	0.864	0.866	0.883
TimeMixer	0.790	0.852	0.882
TimeMixer++	0.783	0.853	0.884
WPMixer	0.778	0.852	0.881

External benchmark comparison (RQ3b). We further evaluate CREST under the CPTC benchmark protocol (Sun & Yu, 2025, Appendix C) to assess external validity. On the long real-world CPTC streams, CREST is the only anytime-valid method and improves over fixed calibration by 5.3 and 2.8 coverage points on electricity and traffic, respectively (Appendix G). Conversely, when we run the official CPTC pipeline on our five datasets (Appendix D), its coverage stays within 1.9 points of CREST on all datasets, while CREST yields narrower intervals on Future and Weather.

Event-triggered recalibration is thus most advantageous on long streams with abrupt scale shifts, while remaining competitive with CPTC-style approaches.

Ablation summary (RQ4). Ablations support the main design choices of CREST. A moderate block size ($B = 20$) gives the best balance between false-alarm control and detection delay, staying near nominal false alarm rate (FAR) with a median delay of about 1.3 blocks. The detector responds primarily to scale-regime drift rather than level shifts, and downstream coverage is largely insensitive to the detector's nominal level and to moderate changes in block size, mixture decay, calibration window, and AR order. Component removals confirm the importance of block aggregation, geometric mixing, and harmonic restart budgeting for controlling false alarms and preserving rapid detection. Recalibration also recovers quickly after alarms. Full ablation results are provided in Appendix E.

5.2 Synthetic experiments

We simulate a time series consisting of a trend component, a smooth periodic component, and fine-scale noise, whose variance changes abruptly at time 1200. The AR whitener is fitted using only the pre-change-point observation window to mimic the deployment setting.

The left panel of Fig. 5 is based on 250 null simulations and shows the empirical FAR as a function of the nominal level α. The FAR increases monotonically with α; at $\alpha = 0.05$, the empirical FAR is 0.044. At the smallest nominal levels, the observed deviations are within the range expected from Monte Carlo sampling error. The right panel is based on 200 simulations for each change magnitude and uses blocks of size $B = 20$. It shows the detection rate and median detection delay as functions of the variance-change magnitude. When the post-change variance increases to 2 and 3 times its pre-change value, the detection probability reaches 99.5%, with a median detection delay of approximately 25 steps (1.3 blocks).

In Appendix H, Table 11 further shows that the FAR of the original tilt statistic increases substantially under heavy-tailed distributions. The FAR is 0.028 under the Gaussian null, rising to 0.736 under a $t(5)$ distribution and further to 0.988 under a $t(3)$ distribution. Replacing the block energy with its Winsorized or Huberized version restores the FAR to a range close to the nominal level (0.000–0.088). The self-normalized variant provides a calibrated approximation to an e-process, but has a comparatively higher empirical FAR of 0.096–0.208.

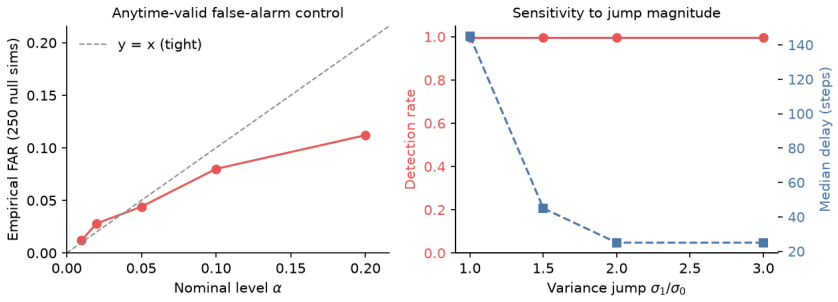


Figure 5: The empirical FAR, the detection rate and median detection delay.

6 Conclusion

We introduced CREST, an event-triggered conformal forecasting framework for scale-regime drift. By monitoring whitened local-scale energy with an e-process, CREST produces anytime-valid alarms and uses a budgeted re-baselining scheme to preserve global false-alarm control under repeated adaptive recalibration. It requires no learned state model and provides an interpretable alarm sequence in which each alarm identifies a candidate regime boundary and consumes a known portion of the global error budget. Experimental results demonstrate that anytime-valid sequential detection can serve as a principled control mechanism for adaptive conformal forecasting under nonstationarity.

REPRODUCIBILITY STATEMENT

Detector parameters, calibration procedures, and the data-split rule are specified in Sections 4 and 5 and in the appendix. The supplementary material provides the core CREST implementation (seasonal-AR whitening, the blockwise e-process detector with the harmonic restart budget, and alarm-triggered conformal recalibration), a self-contained demo reproducing the synthetic false-alarm, detection, and coverage behavior, and the result files behind the main tables and figures, including the per-channel coverage and width CSV.

AI USE STATEMENT

The authors used an AI assistant (OpenAI Codex) for code prototyping, experiment orchestration, and drafting assistance. All scientific claims, experimental results, and the final text were reviewed and verified by the authors, who take full responsibility for the content.

REFERENCES

Anastasios Angelopoulos, Emmanuel J. Candès, and Ryan J. Tibshirani. Conformal pid control for time series prediction. In *Advances in Neural Information Processing Systems 36*, 2023.

Michèle Basseville and Igor V. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. Prentice-Hall, Englewood Cliffs, NJ, 1993.

George Casella and Roger L. Berger. *Statistical Inference*. Duxbury, 2nd edition, 2002.

Mouxiang Chen et al. Accurate parameter-efficient test-time adaptation for time series forecasting. In *International Conference on Machine Learning*, 2025.

Matej Csillag et al. Prediction-powered e-values. In *International Conference on Machine Learning*, 2025.

Dean P. Foster and Robert A. Stine. Alpha-investing: A procedure for sequential control of expected false discoveries. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70 (3):429–444, 2008.

Isaac Gibbs and Emmanuel Candes. Adaptive conformal inference under distribution shift. In *Advances in Neural Information Processing Systems 34*, 2021.

Florian Kalinke and Shakeel Gavioli-Akilagun. Optimal online change detection via random fourier features. In *Advances in Neural Information Processing Systems 38*, 2025.

HyunGi Kim, Siwon Kim, Jisoo Mok, and Sungroh Yoon. Battling the non-stationarity in time series forecasting via test-time adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

Tze Leung Lai. Sequential changepoint detection in quality control and dynamical systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(4):613–658, 1995.

K. K. Gordon Lan and David L. DeMets. Discrete sequential boundaries for clinical trials. *Biometrika*, 70(3):659–663, 1983.

Han Li et al. Kernel-based optimally weighted conformal time-series prediction. In *International Conference on Learning Representations*, 2025a.

Jiajin Li, Zhengyuan Zhou, et al. Online time series forecasting with theoretical guarantees. In *Advances in Neural Information Processing Systems 38*, 2025b.

Jiadong Liang, Zhimei Ren, and Yuxin Chen. Optimal training-conditional regret for online conformal prediction. *arXiv preprint arXiv:2602.16537*, 2026.

Md Mahmuddun Nabi Murad, Mehmet Aktukmak, and Yasin Yilmaz. Wpmixer: Efficient multi-resolution mixing for long-term time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 2025.

Roberto Neglia et al. Rescp: Residual conformal prediction for time series forecasting. In *International Conference on Learning Representations*, 2026. arXiv:2510.05060.

Drew Prinster, Xing Han, Anqi Liu, and Suchi Saria. Watch: Adaptive monitoring for ai deployments via weighted-conformal martingales. In *International Conference on Machine Learning*, 2025.

Aaditya Ramdas, Johannes Ruf, Martin Larsson, and Wouter Koolen. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 2023.

S. W. Roberts. A comparison of some control chart procedures. *Technometrics*, 8(3):411–430, 1966.

Shalev Shaer, Yarin Bar, Drew Prinster, and Yaniv Romano. Testing for distribution shifts with conditional conformal test martingales. In *International Conference on Machine Learning*, 2026.

Shubhanshu Shekhar and Aaditya Ramdas. Sequential change detection via backward confidence sequences. In *International Conference on Machine Learning*, 2023.

Jaehyeok Shin, Aaditya Ramdas, and Alessandro Rinaldo. E-detectors: A nonparametric framework for sequential change detection. *New England Journal of Statistics in Data Science*, 2023. DOI: 10.51387/23-NEJSDS51.

Albert N. Shiryaev. On optimum methods in quickest detection problems. *Theory of Probability & Its Applications*, 8(1):22–46, 1963.

Sophia Sun and Rose Yu. Conformal prediction for time-series forecasting with change points. In *Advances in Neural Information Processing Systems 38*, 2025.

Vladimir Vovk, Ivan Petej, Ilia Nouretdinov, Ernst Ahlberg, Lars Carlsson, and Alex Gammerman. Retrain or not retrain: Conformal test martingales for change-point detection. In *Proceedings of the Tenth Symposium on Conformal and Probabilistic Prediction and Applications*, 2021. PMLR 152.

Ruodu Wang and Aaditya Ramdas. Online multiple testing and e-values. *arXiv preprint arXiv:2202.05263*, 2022.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y. Zhang, and Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations*, 2024.

Shiyu Wang, Jiawei Li, Xiaoming Shi, Zhou Ye, Baichuan Mo, Wenze Lin, Shengtong Ju, Zhixuan Chu, and Ming Jin. Timemixer++: A general time series pattern machine for universal predictive analysis. In *International Conference on Learning Representations*, 2025.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

Junxi Wu, Dongjian Hu, Yajie Bao, Shu-Tao Xia, and Changliang Zou. Error-quantified conformal inference for time series. In *International Conference on Learning Representations*, 2025.

Cheng Yu, Zhoufan Zhu, and Ke Zhu. Conditional quantile adjusted conformal prediction for time series. In *International Conference on Machine Learning*, 2026.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

Zirui Zhao, Liya Cao, and Yuanyu Wan. Mixture of online and offline experts for non-stationary time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

A CUSUM THRESHOLD CALIBRATION

CUSUM baselines use $C_t = \max(0, C_{t-1} + r_t^2 - 1.5)$ on standardized squared residuals, where the reference value 1.5 sits above the unit null mean so that the statistic drifts downward between scale increases, with a refractory period of 100 steps for recalibration use. To compare fairly with CREST's anytime-valid 5% budget, the threshold h is ARL-calibrated *per dataset*: residuals of all channels are permuted 60 times each (destroying temporal structure while preserving the heavy-tailed marginal), the CUSUM running maximum is recorded, and h is the 95% quantile. Resulting thresholds: Future 758, CarIndex 9791, Exchange 19349, ETTm2 76617, Weather 309674. Heavy-tailed datasets therefore receive very conservative thresholds, and their CUSUM detector degenerates toward a rolling baseline; CREST achieves the same FAR budget while still recalibrating on scale events.

B PER-CHANNEL DETAILS

All dataset-level entries in the main tables are means over per-channel coverage and medians over per-channel widths (Future 10, CarIndex 8, Exchange 8, ETTm2 7, Weather 21 channels); the synthetic FAR and detection tables report single-channel simulations, and alarm events are defined per channel. When a single global budget is needed across parallel channels, Appendix H gives the α/G allocation. Per-channel coverage and width for every dataset and method are provided in the supplementary CSV (`downstream_w4_per_channel.csv`), together with the synthetic FAR/delay sweeps. Figure 6 visualizes the per-channel-averaged coverage in Table 2.

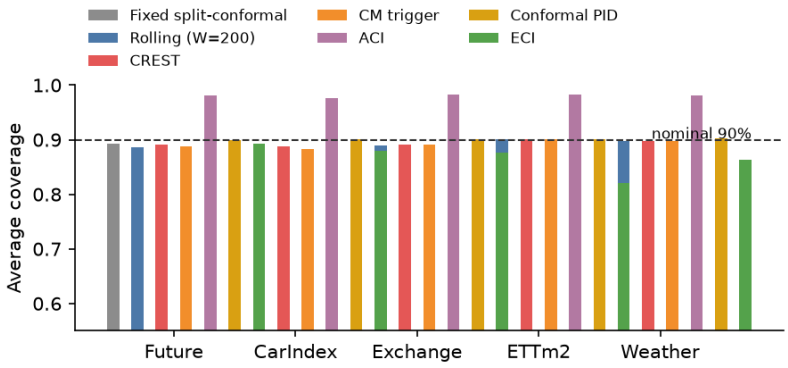


Figure 6: Average conformal coverage by dataset and calibration strategy. The dashed line is the nominal 90%.

C EVALUATION PROTOCOL ON THE CPTC BENCHMARK SUITE

We use the official CPTC repository (Rose-STL-Lab/CPTC) and its six benchmark datasets (three-mode system, bee, bouncing ball with observation and dynamics noise, electricity, traffic). Conformity scores follow the harness: $|y - \hat{y}|$ for univariate series and the L2 norm over the first two dimensions for the multivariate bee dataset. The warm start is $T = \min(\lfloor L/2 \rfloor, 100)$ where L is the forecast horizon. Width is reported as diameter ($2\times$ mean radius). Our methods reuse the same scores and warm start; the only difference from the official baselines is the calibration rule (fixed warm-start quantile, rolling window $W = \min(200, L)$, or rolling with CREST/CUSUM buffer resets). CUSUM thresholds are ARL-calibrated per dataset by the permutation null described in Appendix A.

D CPTC ON OUR REAL DATASETS

The official CPTC algorithm (Sun & Yu, 2025) requires per-step state probabilities z_{prob} and per-state prediction means z_{mean}. We supply a 2-state Gaussian mixture fitted to $\log(|z_t|)$ of the

whitened calibration residuals (states ordered by mean log-scale: low/high variance), with posteriors smoothed by a 96-step moving average and normalized; the point forecast is state-independent (the same seasonal-AR forecaster), so z_{mean} is constant across states. The warm start is the calibration window and the monitoring horizon is capped at 3000 steps (CPTC recomputes per-state quantiles at every step, so it is designed for short-to-medium horizons). Coverage is the fraction of steps with $|y_t - \hat{y}_t| \leq w_t$; width is the interval diameter. Our methods are evaluated on the identical window for a like-for-like comparison; Table 4 reports the result. CPTC hyper-parameters: $\alpha = 0.10$, $\gamma = 0.2$, $\text{min_residuals} = 25$.

Table 4: CPTC (official code, 2-state GMM) vs. CREST on the identical 3000-step window; coverage at nominal 90%, width = median interval diameter.

	CPTC		CREST	
Dataset	Cov.	Width	Cov.	Width
Future	0.903	148.7	0.891	76.4
CarIndex	0.906	508.1	0.887	534.2
Exchange	0.903	0.004	0.888	0.011
ETTm2	0.901	1.085	0.895	2.652
Weather	0.916	1.169	0.905	0.679

E Ablation details

Conformal PID is run on scores normalized by the calibration quantile with $lr = 0.05$, $C_{\text{sat}} = 1.0$; coverage stays at 0.900–0.904 across $K_I \in \{0.2, 2, 20, 200\}$ while median widths grow with K_I, so we set $K_I = 2$, the smallest gain at nominal coverage. ECI likewise uses a direct port of the authors' core update with $lr = 0.05$, the smallest rate in the released sweep, and no per-dataset tuning; the reported under-coverage on drifting series (ETTm2 0.821, Weather 0.864) is the method's behavior at this fixed configuration. Alarm diagnostics detail: the entanglement jump at alarm times is $2.4\times$ the random baseline on Weather but not on CarIndex (few alarms).

Table 5: Block-size tradeoff (FAR under H0, 300 null sims; $2\times$ variance jump at step 1200, 200 sims, $\alpha = 0.05$).

B	FAR	Detection rate	Median delay (steps)
20	0.053	0.995	25
50	0.017	1.000	55
100	0.000	1.000	105
200	0.000	1.000	205

Table 6: Alarm rate in the 1000 steps after a shift (200 sims each). `bias` is the mean standardized residual after the shift.

Shift	Alarm rate / 1k	SD	Residual bias
scale $2\times$	1.005	0.07	-0.02
level 1σ	0.000	0.00	0.08
level 2σ	0.005	0.07	0.17
both	1.005	0.07	0.17

Forecaster plug-in. Table 9 reports CREST coverage with each base forecaster on all three drifting datasets, complementing the subset shown in the main text.

Robustness sweeps and deployment details

Winsorization level and Huber constant are continuous knobs: the weaker versions (W01, $q = 0.99$; H20, $a = 2.0$) give intermediate trade-offs (e.g., $t(3)$ FAR 0.604 and 0.152), letting operators tune

Table 7: Coverage of detection-triggered recalibration under hyper-parameter variation (nominal 90%).

B	CarIndex	Weather	ρ	CarIndex	Weather
20	0.887	0.898	0.90	0.891	0.898
50	0.883	0.899	0.99	0.887	0.898
100	0.879	0.899	0.999	0.886	0.898
W	CarIndex	Weather	AR order	CarIndex	Weather
100	0.885	0.875	3	0.888	0.899
200	0.887	0.898	5	0.887	0.898
400	0.896	0.898	7	0.887	0.898

Table 8: Component ablations (2× variance jump at step 1200; FAR under H0 over 300 sims; detection over 200 sims). The budget rows report the global false-alarm probability over five *independent* null segments (fixed level vs. harmonic levels).

Variant	FAR (H0)	Det. rate	Median delay	Early alarms
full	0.017	1.000	55	0
no block aggregation	0.030	0.970	24	6
no geometric mixture	0.007	0.985	205	3
no whitening	0.000	1.000	50	0
budget: fixed level	0.093	—	—	—
budget: harmonic	0.023	—	—	—

Table 9: Coverage (nominal 90%) across forecasters (pipeline as in Fig. 2; detector on AR-whitened forecast residuals). AR is the paper's main forecaster; DLinear, PatchTST, TimeMixer, TimeMixer++, and WPMixer are trained per channel on the same calibration window.

Dataset	Forecaster	Fixed	Rolling	CREST	ACI
CarIndex	AR	0.649	0.870	**0.887**	0.976
CarIndex	DLinear	0.555	0.854	**0.876**	0.962
CarIndex	PatchTST	0.912	0.875	0.864	0.965
CarIndex	TimeMixer	0.474	0.781	**0.790**	0.800
CarIndex	TimeMixer++	0.510	0.781	**0.783**	0.799
CarIndex	WPMixer	0.426	0.766	**0.778**	0.802
Weather	AR	0.771	0.897	**0.898**	0.981
Weather	DLinear	0.759	0.894	**0.895**	0.971
Weather	PatchTST	0.582	0.866	0.866	0.916
Weather	TimeMixer	0.569	0.851	**0.852**	0.883
Weather	TimeMixer++	0.560	0.852	**0.853**	0.885
Weather	WPMixer	0.551	0.852	**0.852**	0.886
ETTm2	AR	0.872	0.900	**0.901**	0.983
ETTm2	DLinear	0.883	0.898	**0.899**	0.978
ETTm2	PatchTST	0.765	0.883	0.883	0.942
ETTm2	TimeMixer	0.692	0.882	**0.882**	0.929
ETTm2	TimeMixer++	0.694	0.883	**0.884**	0.933
ETTm2	WPMixer	0.687	0.880	**0.881**	0.929

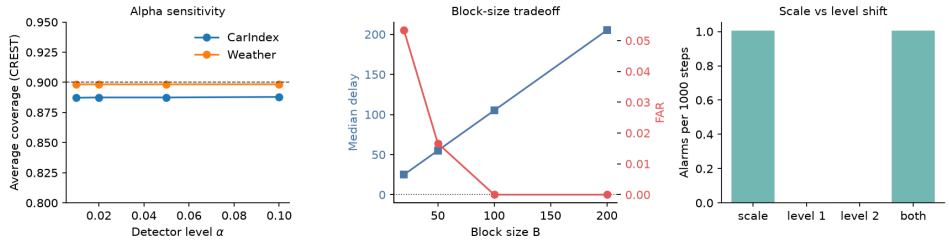


Figure 7: Left: coverage of detection-triggered recalibration vs. detector level α (CarIndex, Weather). Middle: block-size tradeoff: median detection delay (steps) and FAR under the null. Right: alarm rate for a $2\times$ scale shift vs. 1–2σ level shifts (scale specificity).

Table 10: Empirical robustness of CREST under the raw chi-square tilt ($B = 20$, $\alpha = 0.05$, 250 null and 250 jump simulations, with the AR fit on the pre-change window as in deployment; FAR is the fraction of null simulations with at least one alarm). Squared t-innovations are polynomial-tailed, so the tilt's moment-generating function diverges and the empirical FAR inflates sharply under heavy tails; winsorized, Huberized, and self-normalized variants restore near-nominal behavior (Table 11). The symmetric $\lambda = -0.25$ tilt detects $2\times$ scale decreases with lower power.

Setting	FAR (null)	Det. rate	Median delay
Gaussian, $2\times$ jump	0.028	0.992	25
$t(5)$ innovations, $2\times$ jump	0.736	0.488	25
$t(3)$ innovations, $2\times$ jump	0.988	0.052	25
scale decrease $0.5 \to 0.25$, $\lambda = -0.25$	—	0.328	385

Table 11: Robust block-energy constructions under heavy tails (same protocol as Table 10). W05: squares winsorized at the 95% calibration quantile and re-centered to unit mean; H13: Huberized residuals ($a = 1.345$) standardized by the calibration Huber scale; SN: winsorized squares with the per-term tilt moment-generating function estimated empirically on the calibration window. Bounding the terms restores a finite moment-generating function under arbitrary tails; the formal status of each variant is discussed below.

Setting	Variant	FAR (null)	Det. rate	Median delay
$t(5)$	W05	0.004	0.992	45
$t(5)$	H13	0.000	1.000	105
$t(5)$	SN	0.096	0.976	45
$t(3)$	W05	0.088	0.968	45
$t(3)$	H13	0.008	0.988	85
$t(3)$	SN	0.208	0.868	45

FAR against delay. In heavy-tailed deployments we recommend W05: near-nominal FAR, detection ≥ 0.97, median delay 45. SN additionally preserves an e-process by construction, at the cost of a higher empirical FAR under heavy tails (0.096 at $t(5)$, 0.208 at $t(3)$); it is not a strict anytime-valid guarantee until its calibration error is bounded.

Tilt sensitivity: the tilt trades false alarms against detection speed. Larger λ concentrates the e-statistic on large variance increases and lowers the empirical false-alarm rate, while smaller λ gives a more diffuse response. At the default $\lambda = 0.25$ the Gaussian-null false-alarm rate at $\alpha = 0.05$ is 0.044, and a $2\times$ variance jump is detected with probability 0.995 at a median delay of 25 steps (Fig. 5). Deployment constants: detector block $B = 20$; recalibration buffer $W = 200$ with a 20-step post-alarm fallback before switching to the reset buffer; CUSUM alarms use a 100-step refractory period; PID/ECI use a 100-step learning-rate window.

For the CUSUM baseline, standard ARL calibration on the Gaussian null ($h = 31.7$, the 95% quantile of the running maximum) gives empirical FAR 0.080 and median delay 13 for $2\times$ jumps, versus FAR 0.065 and delay 25 for CREST: CREST's anytime-valid guarantee is not an artifact of a conservative CUSUM threshold, but comes at a modest delay cost on this synthetic protocol (a near-matched-FAR comparison; a GLR-of-squares head-to-head under matched anytime budgets is left as future work).

Formal status of the robust variants: W05 and H13 bound the block terms (winsorized squares and Huberized residuals are bounded), so the tilt's moment-generating function is finite and no longer diverges under heavy tails. Bounding changes the null distribution of the block sums, however, so the original chi-square normalization is approximate: we re-center each variant to unit calibration mean and report its FAR empirically. SN goes one step further and estimates the per-term log-moment-generating function from the calibration window, so that $\mathbb{E}[e_k \mid H_0] \approx 1$ holds by construction (up to the calibration error of that estimate); a fully conservative, analytic normalization for the bounded variants is left as future work. Alpha wealth: after j alarms the remaining level is $\alpha\,(1 - \frac{6}{\pi^2}\sum_{k\leq j} k^{-2})$, which practitioners can track when choosing α.

F Pareto analysis of adaptive baselines

To verify that the comparisons in Table 2 are not artifacts of a single favorable baseline configuration, we report a coverage–width Pareto analysis for the two continuous adaptive baselines, ACI and ECI (Figure 8). We sweep ACI's learning rate $\gamma \in \{0.01, 0.02, 0.05, 0.1, 0.2\}$ and memory $W \in \{100, 300, 1000\}$, and ECI's learning rate $\mathrm{lr} \in \{0.005, \ldots, 0.2\}$, on the identical downstream pipeline and aggregation used in Table 2. Default operating points from that table are overlaid.

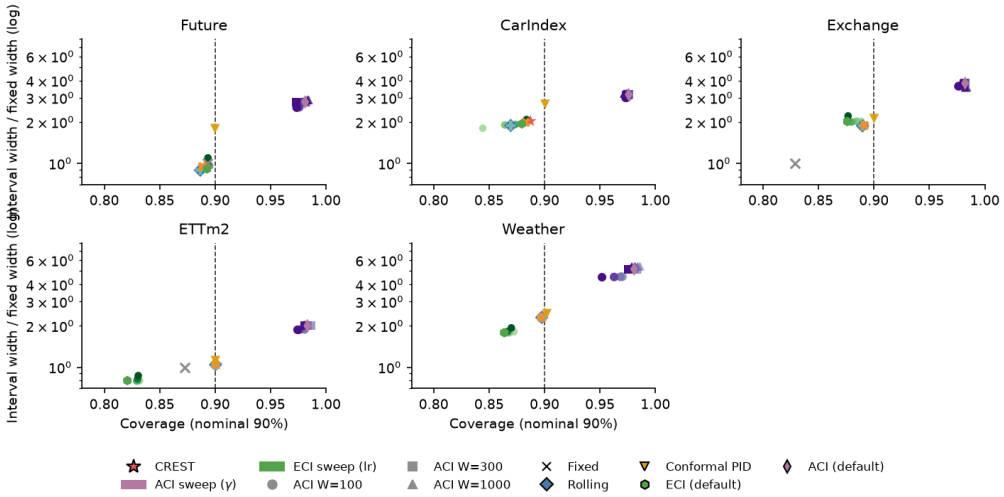


Figure 8: Coverage–width Pareto analysis for ACI and ECI. Colored markers are hyper-parameter sweeps (ACI: purple shades = γ, marker shape = W; ECI: green shades = learning rate); reference markers are the default operating points from Table 2. The dashed line marks nominal 90% coverage.

Across all five datasets, the ACI front lies entirely at coverage at least 0.95 (and at least 0.97 on four datasets), with relative widths between $1.9\times$ and $5.4\times$ fixed; even its most aggressive hyper-parameters never approach nominal 90% coverage at widths competitive with ours, confirming that ACI's over-coverage on these drift-heavy streams is intrinsic rather than a tuning artifact. The ECI front spans coverage 0.82–0.90. Our CREST operating point is not dominated (in coverage and width) by any ECI configuration on four of five datasets; on Future the default ECI point ($\mathrm{lr} = 0.05$) weakly dominates it, at coverage 0.893 versus our 0.891 and 5% narrower intervals. On ETTm2 and Weather the ECI front undercovers (0.82–0.87 versus nominal 0.90) while CREST stays at nominal coverage. These sweeps therefore support the conclusion of Table 2: event-triggered recalibration matches the best ECI trade-offs without per-step quantile updates, and no ACI operating point is competitive at nominal coverage on these streams.

G FULL CPTC BENCHMARK COMPARISON

Coverage percentages are in Table 12; interval diameter is in Table 13. REDSDS, CP, ACI and CPTC are the official implementations; our methods use the same conformity scores and warm start. CPTC entries use the released per-dataset hyper-parameters. The release's AgACI/DtACI metric helper flips the coverage indicator, so those two methods are omitted here and reported only in the raw per-method files.

Table 12: Coverage % (90% target) on the CPTC benchmark suite. Our methods use the same scores and warm start as the harness.

Dataset	REDSDS	CP	ACI	CPTC	Ours-fixed	Ours-roll	Ours-CREST
3-mode	82.9	88.8	89.9	**95.0**	87.6	82.3	82.5
bee	69.5	79.9	85.9	**93.1**	74.9	71.0	71.0
b.ball dyn	56.2	85.8	89.4	**90.5**	85.5	65.9	66.4
b.ball obs	20.6	87.7	89.8	**90.1**	89.6	58.2	58.2
electricity	62.7	85.8	89.8	**91.3**	81.3	85.9	86.6
traffic	23.1	87.0	90.0	**92.3**	85.4	87.8	88.2

Table 13: Interval diameter on the CPTC benchmark suite (official methods as reported by the harness; ours computed identically).

Dataset	REDSDS	CP	ACI	CPTC	Ours-fixed	Ours-roll	Ours-CREST
3-mode	2.45	7.24	6.41	2.46	7.21	6.30	6.43
bee	0.47	1.65	4.87	0.82	1.53	1.40	1.40
b.ball dyn	0.99	4.29	2.28	1.76	4.35	3.18	3.22
b.ball obs	1.17	10.54	3.85	3.76	10.83	7.95	7.95
electricity	82.1	366.6	47.9	156.4	249.0	420.4	436.3
traffic	0.024	0.219	0.282	0.079	0.209	0.225	0.228

H THEORETICAL DETAILS AND PROOFS

Setup and the chi-square tilt. Let z_t denote the seasonal-AR whitened residuals and let $r_t = z_t/\widehat{\sigma}$ denote their standardization by the calibration-window scale estimate $\widehat{\sigma}$. Let $S_k = \sum_{t\in\text{block } k} r_t^2$ be the sum over block k of size B. Under the null hypothesis H_0 that the local scale is unchanged, the r_t are approximately i.i.d. with mean zero and unit variance, so S_k is approximately χ^2_B-distributed. The moment-generating function of a χ^2_B variable is

$$M(\lambda) = \mathbb{E}_{H_0}\left[e^{\lambda S_k}\right] = (1-2\lambda)^{-B/2}, \qquad \lambda < \tfrac{1}{2},$$

so its log-moment-generating function is

$$\log M(\lambda) = -\tfrac{B}{2}\log(1-2\lambda) = \lambda B + \psi_B(\lambda),$$

with $\psi_B(\lambda) = -\lambda B - \frac{B}{2}\log(1-2\lambda)$ as in Sec. 4. The block e-statistic $e_k(\lambda) = \exp(\lambda(S_k - B) - \psi_B(\lambda))$ then satisfies, for every k,

$$\mathbb{E}_{H_0}\big[e_k(\lambda)\big] = \exp(-\lambda B - \psi_B(\lambda))\,\mathbb{E}_{H_0}\big[e^{\lambda S_k}\big] = \exp(-\lambda B - \psi_B(\lambda) + \lambda B + \psi_B(\lambda)) = 1.$$

Moreover S_k is a function of the residuals in block k only; under H_0 the blocks are independent, so $\mathbb{E}_{H_0}[e_k(\lambda) \mid \mathcal{F}_{k-1}] = 1$ where $\mathcal{F}_{k-1}$ is the filtration generated by blocks $1, \ldots, k-1$. The tilt is fixed at $\lambda = 0.25$. This is the design point for a c-fold variance increase, $\lambda = (c-1)/(2c)$ evaluated at $c = 2$: it maximizes the per-block expected log-growth $\lambda c B + \frac{B}{2}\log(1-2\lambda)$ of the e-statistic under that alternative.

Proof of Lemma 1. Fix the filtration $\mathcal{F}_n = \sigma(S_1, \ldots, S_n)$ (equivalently, of the residuals up to block n). For $t \geq 0$ define the component process $C_n^{(t)} = \prod_{k=t+1}^{n} e_k(\lambda)$, with empty products equal to one, so $C_n^{(t)} = 1$ for $t \geq n$. Each $C_n^{(t)}$ is $\mathcal{F}_n$-measurable and nonnegative, and by the mean-one property

$$\mathbb{E}_{H_0}\big[C_n^{(t)} \mid \mathcal{F}_{n-1}\big] = C_{n-1}^{(t)}\,\mathbb{E}_{H_0}\big[e_n(\lambda) \mid \mathcal{F}_{n-1}\big] = C_{n-1}^{(t)},$$

where the factor $C_{n-1}^{(t)}$ is $\mathcal{F}_{n-1}$-measurable and can be pulled out of the conditional expectation, and $\mathbb{E}_{H_0}[e_n(\lambda) \mid \mathcal{F}_{n-1}] = 1$ by the block independence established above. Hence every component is a nonnegative martingale (integrability follows from $\mathbb{E}_{H_0}[C_n^{(t)}] = 1$). With the fixed geometric weights $w_t = (1-\rho)\rho^t$ (which sum to one over all $t \geq 0$, since

$$\sum_{t \geq 0} w_t = (1-\rho)\sum_{s=0}^{\infty} \rho^s = 1,$$

the mixture $M_n = \sum_{t \geq 0} w_t C_n^{(t)}$ satisfies

$$\mathbb{E}_{H_0}\big[M_n \mid \mathcal{F}_{n-1}\big] = \sum_{t < n} w_t C_{n-1}^{(t)} + \sum_{t \geq n} w_t = M_{n-1},$$

so M_n is a nonnegative martingale with $\mathbb{E}_{H_0}[M_n] = 1$; this is the standard fixed-weight mixture construction for e-processes (Ramdas et al., 2023). The detector monitors the partial sum $A_n = \sum_{t=0}^{n} w_t C_n^{(t)} = M_n - \rho^{n+1}$, which obeys the recursion $A_n = w_n + e_n(\lambda) A_{n-1}$ and satisfies $A_n \leq M_n$. Ville's inequality for nonnegative supermartingales states that for every $c > 0$, $\mathbb{P}_{H_0}(\sup_{n \geq 0} M_n \geq c) \leq \mathbb{E}_{H_0}[M_0]/c$; with $c = 1/\alpha$ and $\mathbb{E}_{H_0}[M_0] = 1$,

$$\mathbb{P}_{H_0}\big(\exists n :\ A_n \geq 1/\alpha\big) \leq \mathbb{P}_{H_0}\big(\exists n :\ M_n \geq 1/\alpha\big) \leq \alpha.$$

Finally, the alarm time $T = \inf\{n : A_n \geq 1/\alpha\}$ is a stopping time, and $\{T \leq N\} = \{\exists n \leq N : A_n \geq 1/\alpha\}$ is $\mathcal{F}_N$-measurable for every N, so the bound holds at *every* stopping time, not only at a pre-fixed horizon.

Proof of Lemma 2. After alarm j, the detector re-estimates the local scale from the post-alarm window and starts a fresh e-process at level α_j. Let $\mathcal{E}_j$ be the event that segment j produces a false alarm, i.e., that its e-process reaches $1/\alpha_j$ while the segment's own null holds. Conditional on $\mathcal{F}_{u_j}$ (the information available at the start of segment j, where $u_j = \tau_{j-1} + R$) and assuming the local scale remains constant throughout segment j, Lemma 1 applies to the e-process restarting at u_j, so the per-segment bound holds conditionally. By Lemma 1, $\mathbb{P}(\mathcal{E}_j) \leq \alpha_j$. With $\sum_{j \geq 1} \alpha_j \leq \alpha$, the union bound gives

$$\mathbb{P}(\text{at least one false alarm}) = \mathbb{P}\Big(\bigcup_{j \geq 1} \mathcal{E}_j\Big) \leq \sum_{j \geq 1} \mathbb{P}(\mathcal{E}_j) \leq \alpha.$$

The harmonic schedule $\alpha_j = \alpha \cdot \frac{6}{\pi^2} j^{-2}$ is admissible for an *unbounded* number of restarts because the Basel sum equals $\sum_{j \geq 1} j^{-2} = \pi^2/6$, so $\sum_j \alpha_j = \alpha$. It spends $6/\pi^2 \approx 0.61$ of the total level on the first segment and decays quadratically, concentrating the false-alarm budget where it is most likely to be used; a uniform split would require a known bound K on the number of segments, and an exponential split wastes level on late segments that rarely occur. The guarantee is uniform over stopping times: the alarm times are stopping times of the data stream, each segment's bound holds at every stopping time by Lemma 1, and the union is over the (data-dependent but countable) set of segments.

Proof of Proposition 1. After an alarm at τ, the recalibration buffer collects the m post-alarm scores $c_{\tau+1}, \ldots, c_{\tau+m}$, and the interval for the next observation uses the split-conformal quantile $\hat{q}_{1-\alpha}$, defined as the $\lceil (m+1)(1-\alpha) \rceil$-th smallest value of $\{c_{\tau+1}, \ldots, c_{\tau+m}\} \cup \{+\infty\}$ (the $+\infty$ convention keeps the quantile defined even when the buffer is small). Conditional on the information $\mathcal{F}_\tau$ available at the alarm, the $m+1$ scores $c_{\tau+1}, \ldots, c_{\tau+m+1}$ are exchangeable: the buffer is refilled by a fixed rule with residuals observed after τ, and within a stable post-regime these residuals are i.i.d. given $\mathcal{F}_\tau$. Under exchangeability, the rank of the test score $c_{\tau+m+1}$ among the $m+1$ scores is uniformly distributed on $\{1, \ldots, m+1\}$. Let $k = \lceil (m+1)(1-\alpha) \rceil$ and let $a_{(1)} \le \cdots \le a_{(m+1)}$ be the sorted scores. The interval covers $y_{\tau+m+1}$ exactly when $\mathrm{rank}(c_{\tau+m+1}) \le k$: if the rank is $r \le k$, then $c_{\tau+m+1} \le a_{(k)}$, and replacing the test score by $+\infty$ in the buffer cannot decrease order statistics, so $c_{\tau+m+1} \le a_{(k)} \le \hat{q}_{1-\alpha}$; if $r > k$, the $+\infty$ replacement leaves the first k order statistics unchanged, so $\hat{q}_{1-\alpha} = a_{(k)} < c_{\tau+m+1}$. Therefore

$$\mathbb{P}\big(y_{\tau+m+1} \in [\hat{y}_{\tau+m+1} \pm \hat{q}_{1-\alpha}] \mid \mathcal{F}_\tau\big) = \frac{k}{m+1} \ge 1-\alpha,$$

which is the claim.

Remarks. (i) The geometric mixture assigns weight $(1-\rho)\rho^{n-t}$ to a change that began $n-t$ blocks ago, so the detector averages over candidate change locations rather than committing to a single one; the tilt is fixed at the design point $\lambda = 0.25$ throughout. (ii) The AR coefficients and the residual scale are estimated from data, so the chi-square null is approximate: under standard consistency conditions the standardized block sums converge in distribution to χ^2_B and the statements above hold asymptotically, which is why the empirical FAR tables in Sec. 5.2 accompany the theoretical guarantees. (iii) The tilt $\lambda > 0$ makes the detector one-sided for variance increases, the direction in which fixed calibration under-covers; a symmetric mixture over $\pm\lambda$ yields an analogous two-sided e-process, which we leave as an extension (scale decreases inflate intervals but do not threaten coverage). (iv) The guarantees are per segment and per channel; when a global false-alarm budget is needed across G parallel channels, allocate α/G per channel (Bonferroni) or use an e-BH-style procedure, and run the harmonic budget within each channel. (v) The Shiryaev–Roberts mixture admits an $O(1)$-per-block recursion evaluated in log space via log-sum-exp, so the detector is numerically stable (no underflow or overflow) and its cost is negligible at deployment. (vi) The scale-energy entanglement statistic θ_t is used for analysis and bucketing (Figs. 1, 3) and as a domain descriptor (Fig. 4); the detector itself operates on the block energy S_k of whitened residuals, and adapting λ or B from θ_t is a possible extension left for future work. (vii) Automatic lag and AR-order selection is also left for future work; Table 7 covers the operating range used here.