Spikes as Detectors: Phase-Conditioned Spiking Dynamics for Time-Series Anomaly Detection

Miao Wang **Gangyi Ding** **Yunlin Lei** **Yu Zhang** **Jingfeng Chen** **Xu Yang***
Beijing Institute of Technology
{miaowang, dgy, leiyunlin, zhangyu54, cjf, pyro_yangxu}@bit.edu.cn

Abstract

Most time-series anomaly detectors ask whether an observation can be predicted or reconstructed. We ask a different question: can a detector's own spiking dynamics become anomaly evidence? We study this question for periodic and quasi-periodic time series and propose SPIRE, a phase-conditioned spiking anomaly detector. SPIRE uses period-aware current formation to expose phase-aligned cross-period residuals to leaky integrate-and-fire (LIF) neurons. We call this conversion eventification. After residual injection, recurrence violations induce abnormal bursts, silences, or phase-shifted spike trains. SPIRE then constructs phase-conditioned normal templates of internal spike states and scores deviations in residual energy, firing-rate patterns, and burst/silence statistics as an evidence source read separately from output-space prediction or reconstruction errors. This separates the operational protocol, including causal current-step detection, one-step prediction, and non-causal reconstruction, from the source of anomaly evidence, namely output errors, internal spike states, or both. Under the official TSB-AD protocol, internal-state evidence is on par with output error on the multivariate split and complementary to it in fusion, and its benefit is regime-dependent. Two matched controls isolate the source of the gain. A ReLU network under an otherwise identical pipeline and a rate-matched stateless binary unit, which keeps the binarization and sparsity of LIF but removes its temporal dynamics, both score significantly lower on both splits. Trained from scratch, SPIRE reaches a VUS-PR of 0.409 on TSB-AD-U with output-error scoring and 0.355 on TSB-AD-M with hybrid scoring, competitive with the strongest from-scratch neural detectors. Phase-conditioned spiking dynamics can thus serve as a first-class, complementary anomaly evidence source for periodic time series, rather than merely an efficient computational substrate. Code is available at `https://github.com/Ray-WM/SPIRE`.

1 Introduction

Time-series anomaly detection (TSAD) is commonly formulated through output-space discrepancies. Forecasting-based detectors score the gap between predicted and observed values [12]. Autoencoder, normalizing-flow, and diffusion-based detectors score reconstruction or likelihood residuals [1, 17, 24]. Representation-learning methods reduce anomalies to embedding distances [27]. The recent foundation-model wave preserves this structure. TSPulse [4], the current TSB-AD leaderboard leader, fuses time-domain, frequency-domain, and prediction-domain reconstruction errors through multi-head triangulation. This convention is effective, but it couples two choices: *how* a detector is operated and *where* its anomaly evidence is read from. In all of the methods above, the evidence is still read from output-space discrepancies. Can the detector's own internal dynamics also serve as anomaly evidence?

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

We study this question with spiking neural networks (SNNs) built from leaky integrate-and-fire (LIF) neurons [8]. LIF neurons accumulate input currents over time and emit spikes when their membrane potentials cross a threshold. A spike train is therefore an observable record of how the input interacts with the detector state. It is structured by temporal integration and thresholding, and it may carry information that the final output head does not fully capture. Periodic and quasi-periodic time series provide a natural setting for this view. Normality is often phase-conditioned recurrence, and observations at the same phase across cycles tend to induce similar hidden states. When this recurrence is broken, the violation may appear as an output error and also as a change in internal spike dynamics.

We propose **SPIRE**, a phase-conditioned spiking anomaly detector for periodic time series. SPIRE first forms input currents from phase-aligned cross-period residuals, comparing the current state with states observed at the same phase in previous cycles. Under intact recurrence the residuals remain small and firing follows stable phase-conditioned patterns, whereas recurrence violations perturb membrane dynamics and may induce abnormal bursts, silences, or phase-shifted spike trains. We call this conversion from recurrence violations to measurable spike-state deviations *eventification*. SPIRE turns eventification into an anomaly score through phase-conditioned spike templates. During training on normal data, SPIRE builds normal templates for residual energy, firing rates, and burst and silence statistics at each recurrent phase. At test time, it measures how much the current spike state deviates from the corresponding phase template.

This design separates two axes that prior TSAD has typically coupled. The first is the *operational protocol*. A detector may evaluate the current internal state after observing $\mathbf{x}_t$, predict the next observation from past observations, or reconstruct an observation from a full window. The second is the *evidence source*. The score may come from output errors, internal spike-state deviations, or both. Existing methods occupy a single column of this design space, the output-error column, differing only in which protocol they use. SPIRE makes the internal-state and hybrid columns operational. The key distinction is not spiking computation itself but the use of phase-conditioned and period-aware spike-state statistics as anomaly evidence.

We evaluate SPIRE on TSB-AD-U and TSB-AD-M [11] under the official TSB-AD protocol with the VUS-PR metric [15]. SPIRE is trained from scratch on the normal prefix of each series with about 34K and 31K parameters. It reaches 0.409 on the univariate split with output-error scoring and 0.355 on the multivariate split with hybrid scoring, which is competitive with the strongest from-scratch neural detectors. On the multivariate split the internal-state readout alone reaches 0.342 against 0.336 for output-error scoring on the same trained backbone, and the two fused reach 0.355. On the univariate split output error alone is already strong and fusion does not improve it, so the benefit of internal-state evidence is regime-dependent. Two matched controls isolate where the gain comes from. A ReLU network that receives the same phase residual, architecture, feature operators, template calibration, and fusion scores lower on both splits. A rate-matched stateless binary unit that keeps binarization and sparsity but removes the membrane dynamics also scores lower. Paired bootstrap intervals exclude zero in all four comparisons. The gain is therefore attributable to the LIF neuron and, within it, to its temporal dynamics. The pre-trained TSPulse and several statistical detectors remain higher on individual splits. SPIRE is a parameter-efficient, from-scratch design point that contributes a new evidence-source axis to the TSAD design space.

Our contributions are as follows.

Principle. We formulate periodic time-series anomaly detection as phase-aligned residual eventification. Recurrence violations are exposed as input-current perturbations and converted by LIF threshold dynamics into spike-state deviations that are observable independently of the detector's output.

Method. We propose SPIRE, which combines period-aware current formation, eventification through LIF neurons, and phase-conditioned spike-state templates for anomaly scoring. Median and MAD statistics give resilience to training contamination, a routing-reliability gate selects a single dominant period, and a global non-phase fallback gives graceful degradation on weakly periodic data.

Design space. We separate the operational protocol of a detector from its evidence source, which exposes internal-state and hybrid scoring as design dimensions orthogonal to multi-view reconstruction in recent foundation models. The framing applies beyond SNNs to any detector with measurable internal state.

Evidence. Under the official TSB-AD protocol with paired bootstrap statistics over series, internal-state evidence is on par with output error on TSB-AD-M and complementary to it in fusion. Two matched controls attribute the gain to the LIF neuron, and specifically to its temporal dynamics, on both splits. Feature-level and sensitivity analyses show that the residual energy anchors the readout and that no single spike-derived statistic, fusion weight, or scaling constant is load-bearing.

2 Related Work

Time-series anomaly detection. Classical methods include Isolation Forest [10], One-Class SVM [16], robust PCA [3], and matrix-profile-based detectors [26], which remain competitive on modern benchmarks. Neural methods usually define anomalies through prediction or reconstruction. LSTM-AD [12] uses prediction errors, and Donut [24], OmniAnomaly [17], USAD [1], Anomaly Transformer [25], and TranAD [18] use reconstruction, association, or latent-space discrepancies. Recent foundation models such as MOMENT [9], GPT4TS [29], UniTS [7], and TSPulse [4] use large-scale pre-training and task-specific heads. TSPulse fuses time-domain reconstruction, FFT-domain reconstruction, and next-point prediction errors through multi-head triangulation. These methods all read evidence from output-space discrepancies. SPIRE is complementary to these methods. It adds a source of evidence instead of another reconstruction target, reading residual energy, firing rates, and burst or silence regimes before the final output head.

Spiking neural networks for temporal data. Spiking neural networks process information through temporal integration and thresholded events [8]. Surrogate-gradient training [13] and spatio-temporal backpropagation [23] have made deep SNNs practical, and recent architectures adapt LIF dynamics to standard deep networks [5, 6, 30]. For time-series modeling, SNNs have mainly been studied as efficient sequence models whose spike trains are pooled into firing-rate features and passed to a continuous output head. SNN-based anomaly detection exists. Online evolving SNNs have been used for unsupervised multivariate anomaly detection [2], and Vacuum Spiker [19] detects anomalies from global changes in neural activity. SPIRE differs in the structure of the evidence. It first exposes phase-aligned recurrence violations as input-current perturbations, then compares the resulting spike state with normal templates at the same recurrent phase. A spike pattern is therefore judged against what normally occurs at that phase of the cycle, and high or low activity alone is not anomalous.

Period-aware time-series modeling. Autoformer [21] and FEDformer [28] use decomposition and frequency-domain modeling for seasonal patterns. TimesNet [22] folds a 1D series into 2D period-phase tensors. PatchTST [14] and TimeMixer [20] use patch-level and multi-scale representations. When period-aware models are used for anomaly detection, the period mechanism remains part of the representation learner and the score is still read from prediction or reconstruction error. SPIRE uses periodicity twice. Period-aware current formation injects phase-aligned cross-period residuals into LIF neurons, and the same phase structure indexes the normal spike-state templates used by the score. This dual use prevents normal phase-dependent firing variation from being mistaken as anomaly evidence and lets recurrence violations appear as phase-conditioned spike-state deviations. SPIRE sits at the intersection of these three lines and is designed for data with phase recurrence.

3 Method

SPIRE separates two choices that are typically coupled in anomaly detection: how a detector is operated and where its evidence is read from. We define the design space, then introduce phase-aligned residual eventification, phase-conditioned spike-state templates, anomaly scoring, and the two matched controls. A1-E2, causal current-step detection with phase-conditioned internal-state scoring, is the configuration that carries the central claim. A1-E1 and A1-E3 score the same trained backbone from output error and from the two streams fused. The headline numbers of Section 4.2 come from E1 on the univariate split and from E3 on the multivariate split. A2-E1 and A3-E1 are operational controls for forecasting-based and reconstruction-based detectors. A table of abbreviations is given in Appendix A.

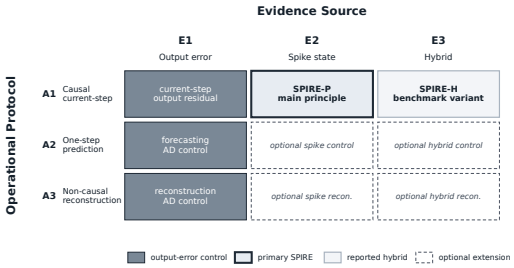


Figure 1: SPIRE separates the operational protocol of a detector from the source of anomaly evidence. A1-E2, where the current observation is scored by phase-conditioned spike-state deviation, carries the central claim. Output-error cells are operational controls. Dashed cells denote optional extensions not required for the main claim.

3.1 Problem setup and design space

Let $\mathbf{X} \in \mathbb{R}^{T \times C}$ be a univariate or multivariate time series. A detector produces a score sequence $s \in \mathbb{R}^T$, where larger s_t indicates that $\mathbf{x}_t$ is more likely to be anomalous. We organize detectors along two axes shown in Figure 1. The *operational protocol* defines the temporal information available when the score is produced. **A1** is causal current-step detection. At time t, the detector has consumed $\mathbf{x}_{1:t}$ and evaluates the internal state induced by the current observation. **A2** is one-step prediction. The output at t predicts $\mathbf{x}_{t+1}$, and the prediction error is assigned to $t+1$. **A3** is non-causal reconstruction. The detector receives a full window and reconstructs each point in it. The *evidence source* defines where the score is read from. **E1** is the output error between the observed signal and the protocol-specific output. **E2** is computed from internal spike-state statistics before the output head. **E3** combines the two through late fusion. A1-E2 is denoted SPIRE-P and A1-E3 SPIRE-H. Output-error cells are protocol controls, with a timeline view in Appendix B.

3.2 Phase-aligned residual eventification

SPIRE forms LIF input currents from phase-aligned cross-period residuals to expose recurrence violations to threshold dynamics. We call this period-aware current formation the spectral residual prior (SRP). Figure 2 shows the full pipeline. A pre-encoder maps the input window to a representation $\mathbf{I} = f_\theta(\mathbf{X}) \in \mathbb{R}^{T \times D}$. SPIRE applies a real FFT along the temporal dimension of $\mathbf{I}$ and aggregates amplitudes over channels. Let $\{k_i\}_{i=1}^K$ be the indices of the top-K nonzero amplitudes. Each frequency index is mapped to an integer period $p_i = \mathrm{round}(T/k_i)$. The routing weights, spectral concentration, and reliability gate are

$$w_i = \frac{\exp(a_{k_i}/\tau)}{\sum_{j=1}^K \exp(a_{k_j}/\tau)}, \qquad r = \frac{\sum_{i=1}^K a_{k_i}}{\sum_k a_k + \varepsilon}, \qquad g = \mathbb{1}[r \geq r_{\min}]. \tag{1}$$

Here r measures the spectral concentration of the learned representation $\mathbf{I}$, not the raw periodicity of $\mathbf{X}$. For each selected period p_i, SPIRE computes the phase-aligned residual $\Delta_{p_i}\mathbf{I}_t = \mathbf{I}_t - \mathbf{I}_{t-p_i}$ with invalid shifted positions masked. The aggregated residual prior $\mathbf{R}_t = \sum_{i=1}^K w_i \, \Delta_{p_i}\mathbf{I}_t$ is injected into the membrane current:

$$\widetilde{\mathbf{I}}_t = \mathbf{I}_t + \alpha g \mathbf{R}_t, \tag{2}$$

where α controls residual strength. A multi-step LIF layer converts $\widetilde{\mathbf{I}}$ into encoder spikes $\mathbf{Z}^{\mathrm{enc}} = \mathrm{LIF}_{T_s}(\widetilde{\mathbf{I}}) \in \{0,1\}^{T_s \times T \times D}$, where T_s is the number of LIF micro-steps. Under intact phase recurrence, $\mathbf{R}_t$ tends to be small and firing follows stable phase-conditioned patterns. Under recurrence violations, $\mathbf{R}_t$ perturbs membrane dynamics and may induce abnormal firing, silence, or phase-shifted spike trains. We call this conversion from phase-aligned residual violations to measurable spike-state deviations *eventification*. The phase-aligned residual itself is a stateless input current that any model can compute identically. Whether the LIF neuron adds anything to it is the question the controls in Section 4.4 answer.

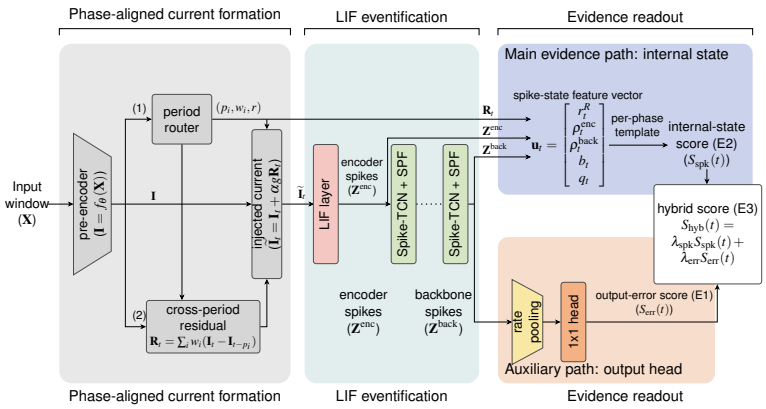


Figure 2: SPIRE architecture. *Phase-aligned current formation*: the period router and the cross-period residual produce $\mathbf{R}_t$, which is injected into the membrane current $\widetilde{\mathbf{I}}_t$ (Eqs. 1–2). *LIF eventification*: the encoder LIF and the Spike-TCN with optional SPF produce encoder spikes $\mathbf{Z}^{\mathrm{enc}}$ and backbone activity $\mathbf{Z}^{\mathrm{back}}$. *Evidence readout*: the spike-state feature vector $\mathbf{u}_t$ (Eqs. 3–4) is matched to the template of its phase for the internal-state score S_{spk}, the rate-pooled output head yields the output-error score S_{err}, and the hybrid score S_{hyb} fuses the two (Eq. 5).

3.3 Phase-conditioned spike-state templates

SPIRE turns eventification into an anomaly score by modeling normal spike states at each recurrent phase, with templates estimated from normal training data without labels and without gradient descent. For each time step, SPIRE extracts a five-dimensional feature vector

$$\mathbf{u}_t = \left[r_t^R,\ \rho_t^{\mathrm{enc}},\ \rho_t^{\mathrm{back}},\ b_t,\ q_t\right] \in \mathbb{R}^5, \tag{3}$$

where $r_t^R = \frac{1}{D}\sum_d |\mathbf{R}_{t,d}|$ is the residual energy, ρ_t^{enc} is the encoder firing rate averaged over LIF micro-steps and channels, ρ_t^{back} is the backbone activity mean, and b_t and q_t are the per-step burst and silence ratios over encoder spikes:

$$b_t = \frac{1}{D}\sum_{d=1}^{D} \mathbb{1}\left[\sum_{\ell=1}^{T_s} \mathbf{Z}^{\mathrm{enc}}_{\ell,t,d} \geq 2\right], \qquad q_t = \frac{1}{D}\sum_{d=1}^{D} \mathbb{1}\left[\sum_{\ell=1}^{T_s} \mathbf{Z}^{\mathrm{enc}}_{\ell,t,d} = 0\right]. \tag{4}$$

The five components are the observable state variables at successive stages of the eventification pathway. Residual energy is the pre-LIF input perturbation, the encoder firing rate is the immediate post-LIF response, and the backbone activity mean is the propagated representation. The burst and silence ratios are the over- and under-excitation threshold regimes. A firing-rate mean alone cannot separate them, since the same mean can arise from steady firing or from bursts interleaved with silences. Three components, ρ_t^{enc}, b_t, and q_t, are strictly spike-derived, whereas r_t^R is a continuous input and ρ_t^{back} aggregates the continuous values of the SEW residual stream. We therefore call the full vector *internal-state evidence* and reserve the term *spike evidence* for the three-dimensional sub-block. Section 4.2 reports both.

Dominant period and per-phase templates. During template fitting, each normal training window provides candidate periods and routing weights. SPIRE sets the dominant period $\bar{p}$ to the most frequent highest-weight period over the reliable normal training windows and averages the corresponding routing weights into a confidence factor $\bar{w}$. Each time step is then assigned to phase $\phi(t) = t \bmod \bar{p}$. For each phase, SPIRE stores robust statistics $\mathbf{m}_\phi = \mathrm{median}\{\mathbf{u}_t : \phi(t) = \phi\}$ and $\mathbf{d}_\phi = \mathrm{MAD}\{\mathbf{u}_t : \phi(t) = \phi\}$, where MAD is the median absolute deviation. Median and MAD are used because normal training slices can contain small contamination, and phase-specific normalization prevents normal phase-dependent variation from being mixed with anomaly evidence. If no reliable period is detected, SPIRE sets $\bar{p} = 1$ and fits a single global template, which avoids returning all-zero spike scores on weakly periodic series. The gate in Eq. 1 and the global fallback keep the score well defined when recurrence is weak, so reliable recurrence is not a precondition for the method to run. A schematic of this construction is in Appendix C.

3.4 Anomaly scoring and operational protocols

SPIRE supports three evidence sources with the same trained backbone. The internal-state (E2) score compares the current feature vector with the template at the same recurrent phase. The output-error (E1) score is the protocol-specific output residual, and the hybrid (E3) score is a late fusion:

$$s_t^{\mathrm{spk}} = \bar{w} \cdot \frac{1}{5} \sum_{j=1}^{5} \left| \frac{u_{t,j} - m_{\phi(t),j}}{d_{\phi(t),j} + \varepsilon} \right|, \quad s_t^{\mathrm{err}} = \frac{1}{C} \|\mathbf{x}_t - \bar{\mathbf{x}}_t\|_2^2, \quad s_t^{\mathrm{hyb}} = \lambda_{\mathrm{spk}} \tilde{s}_t^{\mathrm{spk}} + \lambda_{\mathrm{err}} \tilde{s}_t^{\mathrm{err}}. \quad (5)$$

For the global fallback, $\bar{w} = 1$ and $\phi(t) = 0$. Dividing each feature by its own MAD gives the five features comparable influence, whereas a plain average of absolute deviations would be dominated by the largest-scale feature, the residual energy. The feature weights are equal and are not tuned. Both streams in the hybrid score are normalized with fixed median/MAD scalers fitted on the normal training slice, with scale $1.4826 \cdot \mathrm{MAD}$. The factor 1.4826 is the standard constant that makes the MAD a consistent estimator of the standard deviation under a Gaussian. Fixed normalization prevents a test point's score from depending on other points in the same evaluation batch. The gate threshold $r_{\min}$ was selected on the official tuning splits. The mixing weights, $\lambda_{\mathrm{spk}} = 0.7$ and $\lambda_{\mathrm{err}} = 0.3$ unless otherwise stated, are fixed and shared across all evaluation series; Section 4.2 reports what the tuning splits select instead, and Appendix J the sensitivity to them. The backbone is trained on normal data with a self-supervised, causally masked same-step reconstruction objective $\mathcal{L}_{\mathrm{aux}} = \frac{1}{TC} \sum_t \|\mathbf{x}_t - \bar{\mathbf{x}}_t\|_2^2$, where *causal* refers to the temporal masking of the backbone, not to causal discovery among variables. The reconstruction head only shapes the representation. In the A1-E2 configuration it is not the source of anomaly evidence and can be removed at test time without changing the score. Spike templates and hybrid scalers are fitted by passing the normal training slice through the detector once after training.

The protocols differ in which output target is used. A1 uses a causal temporal backbone and reads the spike-state feature vector from the current time slice. A2 shifts the output target to predict $\mathbf{x}_{t+1}$ and is the operational control for forecasting-based detectors. A3 uses a non-causal backbone with symmetric padding and reconstructs all points in the evaluation window. A1 and A2 use causal backbones but are not fully streaming detectors, since the SRP encoder uses a small symmetric convolution and the period router estimates periods from the input window. A fully streaming version would need a causal encoder and an online period router, a limitation orthogonal to the evidence-source claim.

Backbone, optional SPF, and matched controls. SPIRE uses a Spike-TCN backbone with stacked dilated 1D convolutional blocks containing batch normalization, multi-step LIF, and spike-element-wise (SEW) residual connections [5]. An optional spiking period folding (SPF) block folds the spike tensor according to the detected period and applies spiking convolution in the phase-folded space. SRP is the load-bearing period-aware component, and SPF is an optional refinement whose benefit is dataset-dependent (Section 4.3). The decoder mean-pools spike trains over the LIF micro-step axis and applies a 1×1 convolutional head. To isolate the contribution of the neuron, two controls share everything else with SPIRE: the phase residual, the SRP and SPF architecture, the TCN topology, the five feature operators, template calibration, fusion, and protocol. The *matched ReLU activation-template* control replaces multi-step LIF and spiking convolution with continuous ReLU modules. It reads the same five statistics from activations, with the activation mean and the high-activation and zero-activation proportions in place of rate, burst, and silence. The *rate-matched stateless binary* control replaces every LIF neuron with a binary-threshold unit without membrane state, leak, integration, or reset. Its threshold is calibrated on the training slice so that its activation rate equals the firing rate of the LIF unit it replaces. The control therefore matches LIF in binarization and sparsity and differs only in the temporal dynamics.

4 Experiments

We organize experiments around the two axes of Section 3.1. We ask five questions. How does a from-scratch SPIRE detector compare with statistical, neural, and pre-trained detectors under the official TSB-AD protocol? Do internal spike-state statistics add evidence beyond output errors, and which statistics carry it? Which operational protocol best supports this evidence source? Which period-aware components expose phase recurrence? Are the gains tied to LIF dynamics and to the reliability of phase recurrence?

Table 1: Main results on the official TSB-AD Eval splits (VUS-PR, official per-series training prefix and sliding window). SPIRE is trained from scratch on the normal prefix of each evaluation series. Baseline numbers are from the TSB-AD leaderboard; TSPulse is a pre-trained external reference.

Method	Model type	TSB-AD-U	TSB-AD-M
Pre-trained foundation models			
TSPulse (FT) [4]	pre-trained + fine-tuned	0.520	0.390
TSPulse (ZS) [4]	pre-trained, zero-shot	0.480	0.360
MOMENT (FT) [9]	pre-trained + fine-tuned	0.390	—
From-scratch neural detectors			
USAD [11]	from scratch	0.360	0.300
CNN [11]	from scratch	0.340	0.310
LSTMAD [11]	from scratch	0.330	0.310
OmniAnomaly [11]	from scratch	0.290	0.310
Donut [11]	from scratch	0.200	0.260
Statistical detectors			
Sub-PCA [11]	statistical	0.420	—
KShapeAD [11]	statistical	0.400	—
KMeansAD [11]	statistical	0.370	0.290
PCA [11]	statistical	—	0.310
This work, official protocol			
SPIRE, output error (A1-E1)	from scratch	**0.409**	0.336
SPIRE-P, internal state (A1-E2)	from scratch	0.288	0.342
SPIRE-H, hybrid, fixed $\lambda_{\mathrm{spk}} = 0.7$ (A1-E3)	from scratch	0.394	**0.355**
SPIRE, λ_{spk} selected on the official tuning split	from scratch	0.405	0.336

4.1 Experimental setup

We use TSB-AD-U and TSB-AD-M [11], whose official Eval splits contain 350 univariate and 180 multivariate series. Hyperparameters are selected on the official Tuning splits (48 and 20 series) and then fixed for every evaluation series of a split. All results follow the official TSB-AD protocol. Each series is trained on its official normal training prefix, the per-series training length provided with the benchmark. Spike templates and hybrid scalers are fitted on the same prefix without anomaly labels. VUS-PR [15], the primary metric, uses the official per-series sliding window. The official prefix is shorter than the 50% prefix of the submitted version (median about 25% on TSB-AD-M) and is the leaderboard protocol, so we adopt it throughout. Only the superset regime analysis of Section 4.4 is kept under the 50% prefix, and it is labelled as such. AUC-ROC and AUC-PR are in Appendix F. SPIRE uses window size 50, $L = 3$ TCN levels, backbone width 16, $\alpha = 0.5$, top-$K = 2$, and Adam at learning rate 10^{-3} for at most 50 epochs with patience 10. The split-specific settings ($D = 32$, kernel 16, $T_s = 6$, $r_{\min} = 0.05$ on U; $D = 64$, kernel 8, $T_s = 4$, $r_{\min} = 0.15$ on M) are listed in Appendix L. SPIRE has 34K parameters on U and 31K on M at the mean channel count, against 32K and 28K for the matched ReLU control (Appendix N).

Statistics. Every comparison between two scores or two models is made on the same series and summarized by a paired bootstrap over series with $10{,}000$ resamples. We report the mean difference Δ, its 95% percentile interval, $\Pr(\Delta > 0)$, and win/tie/loss counts. $\Pr(\Delta > 0)$ is the fraction of resamples with a positive mean difference, not a frequentist p-value. The main configuration was trained three times with different seeds, and Table 2 reports the mean and standard deviation over training runs. The matched ReLU control was also trained three times, and every LIF run exceeds every ReLU run on both splits (Appendix K). The rate-matched binary control is a single training run, and all paired statistics use the first run of each model.

4.2 Main results and evidence-source ablation

Table 1 reports the main benchmark results, and Table 2 fixes the trained backbone and the A1 protocol while varying only the score. Under the official protocol SPIRE reaches 0.409 on TSB-AD-U with output-error scoring and 0.355 on TSB-AD-M with hybrid scoring. On M this is above every from-scratch neural and statistical detector listed and within 10% of the fine-tuned TSPulse. On U it is above the from-scratch neural detectors and KMeansAD but below Sub-PCA and the pre-trained models, so we claim no leaderboard record.

Table 2: Evidence-source ablation under the A1 protocol (official protocol, VUS-PR): the same trained backbone scored with output error, internal-state deviation, or their hybrid. First block: mean $\pm$ s.d. over three training runs; decomposition and paired differences (95% bootstrap intervals over series): first run.

Score (same trained backbone, A1)	TSB-AD-U	TSB-AD-M
E1: output error	0.408 ± 0.001	0.336 ± 0.001
E2: internal state, full five-dimensional vector	0.288 ± 0.001	0.343 ± 0.001
E3: hybrid, $\lambda_{\mathrm{spk}} = 0.7$	0.394 ± 0.000	0.353 ± 0.001
Decomposition of the internal-state vector (first run)		
residual energy r_t^R only	0.313	0.324
spike sub-block $(\rho_t^{\mathrm{enc}}, b_t, q_t)$ only	0.180	0.275
E3 $-$ E1	-0.014 $[-0.026, -0.003]$	$+0.019$ $[-0.001, +0.042]$
full vector $-$ residual energy only	-0.025 $[-0.038, -0.013]$	$+0.018$ $[+0.008, +0.031]$

Internal-state evidence. On the same A1 backbone, the internal-state readout alone (E2) is on par with output error on M (0.342 against 0.336) and below it on U (0.288 against 0.409), where output error is already strong. The M mean is driven by large gains on a subset of series, since E2 is higher on 49 of the 180 series and E1 on 131. Fixed-weight fusion improves output error on M ($+0.019$, 95% interval $[-0.001, +0.042]$, $\Pr(\Delta > 0) = 0.97$, win/tie/loss $73/3/104$) and lowers it on U (-0.014, $[-0.026, -0.003]$), so the benefit of the internal-state readout is regime-dependent. The fixed weight $\lambda_{\mathrm{spk}} = 0.7$ is carried over from the submission. Selecting it per split on the official tuning splits does not reproduce the M gain. The 20 tuning series of M favor output error alone (0.359 at $\lambda_{\mathrm{spk}} = 0$ against 0.317 at 0.7), and the selection returns 0.336 on Eval. On U the selection returns $\lambda_{\mathrm{spk}} = 0.25$ and 0.405 (Table 1, Appendix J). The M benefit of fusion is thus a regime effect that the small tuning split does not predict, and we do not present it as a tuned result. The most direct evidence that the spike-derived statistics carry information of their own is the contrast within the vector. On M, scoring the full vector instead of the residual energy alone raises VUS-PR from 0.324 to 0.342 ($\Pr(\Delta > 0) = 1.00$). Fusing the full vector instead of the residual energy with output error raises the fused score from 0.348 to 0.355. The strictly spike-derived sub-block alone scores 0.275 on M and 0.180 on U, below output error. Single-feature and leave-one-out results (Appendix I) show that on M the full vector exceeds every single feature and output error, and that removing any one of the four spike-derived features changes the score by at most 0.001, so the four are mutually redundant and the readout has no single point of failure. On U the residual energy alone scores 0.313 against 0.288 for the full vector, and output error is far above both. This is the regime reading of Section 4.4 seen inside the readout. Where output error already captures the violations, the spike-derived statistics have little to add. Where it degrades, as on M, they lift the readout to parity with output error, with one fixed readout on both splits and no per-split feature selection. Fusion improves over output error on M for every λ_{spk} from 0.25 to 0.75 and all three scalers, and lowers it on U for every scaler (Appendix J), so no fusion weight or scaling constant is load-bearing.

4.3 Operational protocol and period-aware components

Table 6 (Appendix D) fixes the evidence source to E1 and varies the operational protocol under the official prefix. Under the official prefix the three protocols are close, with A1/A2/A3 at $0.409/0.413/0.413$ on U and $0.336/0.337/0.338$ on M, the latter within seed variation. The protocol is therefore not the source of SPIRE's gains. We keep A1 because it scores the state produced after the current observation enters the causal backbone. That is the state the eventification chain in Section 3.2 reads. Table 5 (Appendix D) ablates SRP and SPF. Under E1, the ladder from the Spike-TCN baseline is $0.398 \rightarrow 0.402$ (SRP) $\rightarrow 0.405$ (SPF) $\rightarrow 0.409$ (SRP+SPF) on U and $0.329 \rightarrow 0.337 \rightarrow 0.334 \rightarrow 0.336$ on M. SRP is the main period-aware contributor on both splits, and SPF adds a small gain on U and is neutral to slightly negative on M. Under hybrid scoring, removing SRP and SPF together lowers the M score from 0.355 to 0.325 (-0.029, 95% interval $[-0.051, -0.010]$) and the U score from 0.394 to 0.358 (-0.036, $[-0.053, -0.019]$). The architectural contribution ($+0.029$ on M, $+0.036$ on U) and the readout contribution ($+0.019$ from E1 to E3 on M within the full architecture) are therefore measured separately. We keep SRP+SPF as the default on both benchmarks instead of enabling SPF only where it helps.

Table 3: Isolating the neuron. All arms share the phase residual, SRP+SPF architecture, feature operators, template calibration, fusion, and protocol; only the unit differs. Hybrid VUS-PR, paired differences with 95% bootstrap intervals over series, $\Pr(\Delta > 0)$, and win/tie/loss counts; three-run repetitions in Appendix K.

Unit (same pipeline)	TSB-AD-U	TSB-AD-M
LIF (SPIRE-H)	**0.394**	**0.355**
Matched ReLU activation-template (control)	0.362	0.340
Rate-matched stateless binary (control)	0.340	0.344
LIF − ReLU: Δ [95% interval]	$+0.032$ $[+0.011, +0.053]$	$+0.014$ $[+0.001, +0.028]$
$\Pr(\Delta > 0)$; win/tie/loss	1.00; 210/5/135	0.98; 120/2/58
LIF − binary: Δ [95% interval]	$+0.055$ $[+0.040, +0.071]$	$+0.011$ $[+0.002, +0.020]$
$\Pr(\Delta > 0)$; win/tie/loss	1.00; 258/10/82	0.99; 118/4/58

4.4 LIF dynamics and regime analysis

Isolating the neuron. Table 3 compares SPIRE with the two controls of Section 3.4, which differ from it only in the unit. The matched ReLU activation-template network scores 0.340 against 0.355 on M and 0.362 against 0.394 on U, with $\Pr(\Delta > 0) = 0.98$ and 1.00 and bootstrap intervals excluding zero on both splits. The residual, the features, and the scoring rule are shared. The difference is therefore attributable to the neuron, not to the phase-residual construction, the feature design, or the hybrid scoring. The rate-matched stateless binary unit keeps the binarization and sparsity of LIF and removes only its temporal dynamics, and it is also significantly worse than LIF on both splits (0.344 and 0.340). A continuous activation is worse than LIF, so a genuine spiking neuron is needed. A stateless binary unit at the same firing rate is also worse. Emitting binary events at the right sparsity is therefore not enough, and the membrane integration, leak, and reset contribute on top. The gain is thus localized to the temporal dynamics of the LIF neuron and is significant on both splits, although its magnitude is moderate ($+0.011$ to $+0.055$ VUS-PR). At the stream level (Appendix K) the advantage of LIF on M is a fusion-level property. The ReLU control has a higher output-error score (0.358 against 0.336) and an equal internal-state score (0.343 against 0.342) but a lower fused score, so the two streams of the LIF model are more complementary. On U, LIF is higher on output error and in fusion.

Phase recurrence and regime. Two stratified analyses measure different quantities and we present them separately. The first is the architectural gap on a superset under the submission protocol. On an 870-series superset of TSB-AD-U, we group series by the spectral concentration $\bar{r}$ of the representation. These runs use the 50% normal prefix of the submitted version and are kept separate from the official Eval results. The SPIRE-H minus ANN gap rises monotonically with $\bar{r}$, from $+0.005$ on weakly periodic series ($\bar{r} \leq 0.1$) to $+0.025$ at medium $\bar{r}$ and $+0.120$ on strongly periodic series ($\bar{r} > 0.3$). The gap under the same E1 score rises from approximately zero to $+0.070$ (Appendix H). When phase recurrence is reliable, the phase-aligned residual is a meaningful stimulus and the period-aware spiking architecture gains most. When it is weak, the period-aware path is gated off or falls back to the global template. The second analysis is the readout gain on the official Eval splits. We stratify the official series into terciles of router reliability r. The within-tercile gain of hybrid over output-error scoring is largest on the low-reliability tercile ($+0.08$ on M, $+0.01$ on U) and slightly negative on the mid and high terciles (Appendix H). The two directions do not conflict. The first is the architecture axis, SPIRE against a continuous counterpart. The second is the evidence-source axis, fusion against output error. Reliable recurrence is thus the regime where the spiking architecture gains most and where the internal-state readout is least needed. The per-domain breakdown under the official protocol (Appendix G) shows the same pattern with no per-domain tuning. Figure 3 visualizes the eventification chain on one strongly periodic series (Appendix E).

5 Discussion and Conclusion

When does internal-state evidence help? The experiments identify a bounded regime for SPIRE. The spiking architecture gains most over its continuous counterpart on phase-recurrent data. There the SRP residual is a meaningful phase-aligned stimulus, and LIF thresholding turns recurrence

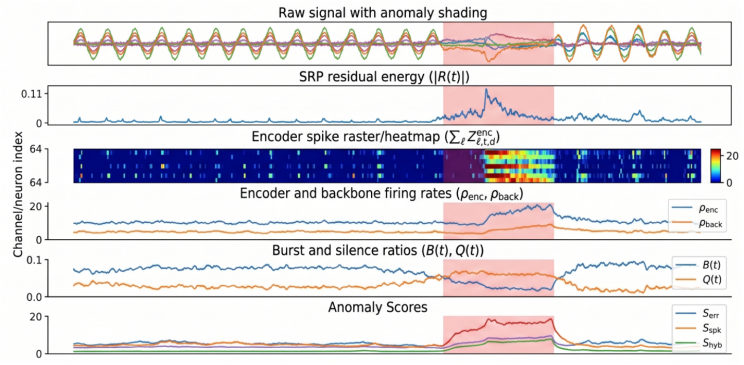


Figure 3: Mechanism visualization on a strongly periodic series. The recurrence violation first raises the phase-aligned residual $|R(t)|$, then alters encoder firing and the burst and silence regimes $B(t)$, $Q(t)$, and finally the internal-state score S_{spk}; the hybrid score S_{hyb} tracks the anomaly more cleanly than S_{err} alone.

violations into abnormal firing, silence, or phase-shifted patterns. The per-phase templates then make these deviations visible to the score. The benefit of the internal-state readout follows a different rule. Fixed-weight late fusion interpolates the ranking quality of two score streams, so it helps when the two are of comparable quality and dilutes the stronger one when one dominates. On the official univariate split output error alone is already strong and fusion lowers it. On the multivariate split output error degrades, most of all on the series with the least reliable recurrence. The internal-state stream keeps working through the global fallback and compensates. Reliable recurrence is thus the regime where the readout is least needed, not a precondition for it to help. The evidence source is a design axis to be chosen, or weighted, by regime.

Limitations, future steps, and conclusion. Three limitations concern the central claim. First, internal-state evidence is complementary, not uniformly strongest. It is on par with output error only on the multivariate split, where its mean is driven by large gains on a subset of series. The strictly spike-derived sub-block alone is weaker than output error on both splits. Second, the SNN-specific gain is isolated to the temporal dynamics of the LIF neuron and is significant on both splits, but its magnitude is moderate, between 0.01 and 0.06 VUS-PR. Third, on the multivariate split the top-line result is the hybrid score at a fixed weight carried over from the submission, and the small official tuning split does not select that weight. On the univariate split the top-line result is output-error scoring alone. Beyond these, SPIRE is designed for periodic structure and is less suited to cross-channel anomalies such as correlation shifts, the implementation is not fully streaming, and the rate-matched control is a single training run. The three operations of SPIRE extend naturally to confounded multivariate series, legitimate drift, and missing entries (Appendix M), none of which is evaluated here. Broader impact is discussed in Appendix O. Within these limits, SPIRE separates how a detector is operated from where its anomaly evidence is read. Internal spike states yield an evidence source that can be read independently of output errors. On the multivariate split it adds information beyond the residual current and beyond output error, and its value is tied to the temporal dynamics of the spiking neuron. For phase-recurrent time series, a detector's own spiking dynamics can serve as a first-class, complementary anomaly evidence source.

Acknowledgments and Disclosure of Funding

This work was supported by the National Science and Technology Major Project on Brain Science and Brain-like Intelligence Technology (2022ZD0208700) and by the Fundamental Research Funds for the Central Universities (2025XC11020). The authors declare no competing interests.

References

[1] Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A. Zuluaga. USAD: UnSupervised anomaly detection on multivariate time series. In *Proceedings of the 26th*

ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 3395–3404. ACM, 2020.

[2] Dennis Bäßler, Tobias Kortus, and Gabriele Gühring. Unsupervised anomaly detection in multivariate time series with online evolving spiking neural networks. *Machine Learning*, 111 (4):1377–1408, 2022.

[3] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM*, 58(3):1–37, 2011.

[4] Vijay Ekambaram, Subodh Kumar, Arindam Jati, Sumanta Mukherjee, Tomoya Sakai, Pankaj Dayama, Wesley M. Gifford, and Jayant Kalagnanam. TSPulse: Tiny pre-trained models with disentangled representations for rapid time-series analysis. In *Proceedings of the International Conference on Learning Representations*, 2026. Accepted at ICLR 2026; arXiv:2505.13033v3.

[5] Wei Fang, Zhaofei Yu, Yanqi Chen, Tiejun Huang, Timothée Masquelier, and Yonghong Tian. Deep residual learning in spiking neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pages 21056–21069, 2021.

[6] Wei Fang, Zhaofei Yu, Yanqi Chen, Timothée Masquelier, Tiejun Huang, and Yonghong Tian. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2661–2671, 2021.

[7] Shanghua Gao, Teddy Koker, Owen Queen, Thomas Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. UniTS: A unified multi-task time series model. In *Advances in Neural Information Processing Systems*, 2024.

[8] Wulfram Gerstner and Werner M. Kistler. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002. ISBN 9780521890793.

[9] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 16115–16152. PMLR, 2024.

[10] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *Proceedings of the 8th IEEE International Conference on Data Mining*, pages 413–422. IEEE, 2008.

[11] Qinghua Liu and John Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. In *Advances in Neural Information Processing Systems*, 2024.

[12] Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. LSTM-based encoder-decoder for multi-sensor anomaly detection. *arXiv preprint arXiv:1607.00148*, 2016.

[13] Emre O. Neftci, Hesham Mostafa, and Friedemann Zenke. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6):51–63, 2019.

[14] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *Proceedings of the International Conference on Learning Representations*, 2023.

[15] John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S. Tsay, Aaron J. Elmore, and Michael J. Franklin. Volume under the surface: A new accuracy evaluation measure for time-series anomaly detection. *Proceedings of the VLDB Endowment*, 15(11):2774–2787, 2022.

[16] Bernhard Schölkopf, Robert C. Williamson, Alex J. Smola, John Shawe-Taylor, and John C. Platt. Support vector method for novelty detection. In *Advances in Neural Information Processing Systems*, volume 12, 1999.

[17] Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2828–2837. ACM, 2019.

[18] Shreshth Tuli, Giuliano Casale, and Nicholas R. Jennings. TranAD: Deep transformer networks for anomaly detection in multivariate time series data. *Proceedings of the VLDB Endowment*, 15(6):1201–1214, 2022.

[19] Iago Xabier Vázquez, Javier Sedano, Muhammad Afzal, and Ángel Miguel García-Vico. Vacuum spiker: A spiking neural network-based model for efficient anomaly detection in time series. arXiv preprint arXiv:2510.06910, 2025.

[20] 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 *Proceedings of the International Conference on Learning Representations*, 2024.

[21] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, volume 34, pages 22419–22430, 2021.

[22] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2d-variation modeling for general time series analysis. In *Proceedings of the International Conference on Learning Representations*, 2023.

[23] Yujie Wu, Lei Deng, Guoqi Li, Jun Zhu, and Luping Shi. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in Neuroscience*, 12:331, 2018.

[24] Haowen Xu, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, Dan Pei, Yang Feng, Jie Chen, Zhaogang Wang, and Honglin Qiao. Unsupervised anomaly detection via variational auto-encoder for seasonal KPIs in web applications. In *Proceedings of the 2018 World Wide Web Conference*, pages 187–196. ACM, 2018.

[25] Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly transformer: Time series anomaly detection with association discrepancy. In *Proceedings of the International Conference on Learning Representations*, 2022.

[26] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. Matrix profile I: All pairs similarity joins for time series: A unifying view that includes motifs, discords and shapelets. In *Proceedings of the 16th IEEE International Conference on Data Mining*, pages 1317–1322. IEEE, 2016.

[27] Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. TS2Vec: Towards universal representation of time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8980–8987, 2022.

[28] Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 27268–27286. PMLR, 2022.

[29] Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One Fits All: Power general time series analysis by pretrained LM. In *Advances in Neural Information Processing Systems*, 2023.

[30] Zhaokun Zhou, Yuesheng Zhu, Chao He, Yaowei Wang, Shuicheng Yan, Yonghong Tian, and Li Yuan. Spikformer: When spiking neural network meets transformer. In *Proceedings of the International Conference on Learning Representations*, 2023.

A Abbreviations

Table 4: Abbreviations used in the paper.

Abbreviation	Meaning
TSAD	time-series anomaly detection
SNN / LIF	spiking neural network / leaky integrate-and-fire neuron
TCN / SEW	temporal convolutional network / spike-element-wise residual connection
SRP	spectral residual prior: period-aware current formation (Section 3.2)
SPF	spiking period folding: optional phase-folded spiking convolution block
A1 / A2 / A3	operational protocols: causal current-step / one-step prediction / non-causal reconstruction
E1 / E2 / E3	evidence sources: output error / internal spike state / hybrid late fusion
internal-state evidence	the full vector $\mathbf{u}_t$ of five features; *spike evidence* is its three strictly spike-derived features (Section 3.3)
SPIRE-P / SPIRE-H	A1-E2 (internal-state score) / A1-E3 (hybrid score)
VUS-PR	volume under the precision-recall surface (primary TSB-AD metric)
AUC-PR / AUC-ROC	area under the precision-recall / receiver-operating-characteristic curve
MAD / IQR	median absolute deviation / interquartile range
FT / ZS	fine-tuned / zero-shot (pre-trained baselines)
$\Pr(\Delta > 0)$	fraction of paired-bootstrap resamples over series with a positive mean difference

B Operational Protocol Timeline

Figure 4 gives a timeline view of the three operational protocols defined in Section 3.1. A1 reads the spike-state feature vector from the current time slice after observing $\mathbf{x}_{\leq t}$ and asks whether the current state is abnormal. A2 produces a one-step prediction $\hat{\mathbf{x}}_t$ from past observations and assigns the squared prediction error to t. A3 reconstructs $\tilde{\mathbf{x}}_t$ from a window $[t-L, t+L]$ that contains both past and future observations. The three protocols differ only in the temporal access pattern; all three can be combined with any of the three evidence sources E1, E2, or E3.

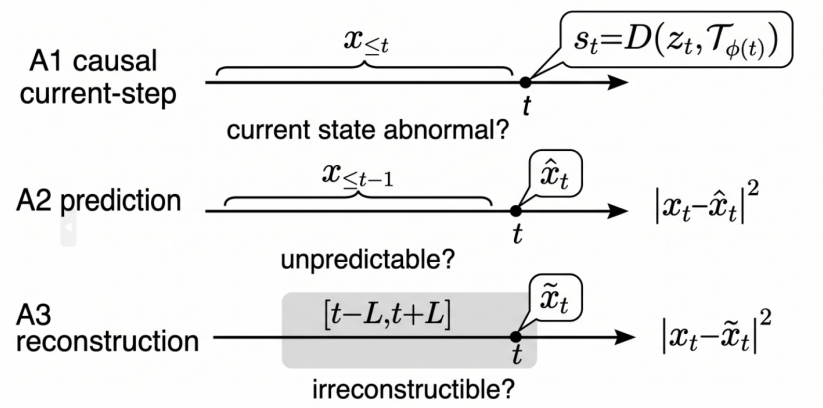
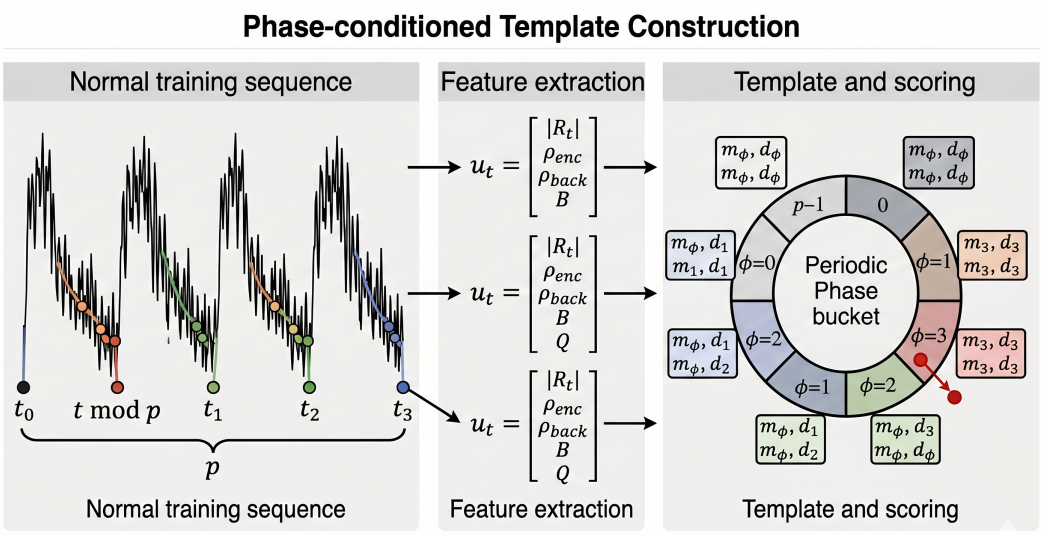


Figure 4: Operational protocol timeline. A1 reads the current state, A2 issues a one-step forecast, and A3 reconstructs a point from a centered window.

C Phase-conditioned Template Construction

Figure 5 schematizes how phase-conditioned spike-state templates are built. The normal training sequence is split into segments of length $\bar{p}$. At each step t in a normal slice, the five-dimensional feature vector $\mathbf{u}_t$ is recorded under the assigned phase $\phi(t) = t \bmod \bar{p}$. After all reliable normal windows have been processed, each phase bucket holds a population of $\mathbf{u}_t$ vectors that come from the same recurrent point in the cycle. The robust statistics $\mathbf{m}_\phi$ and $\mathbf{d}_\phi$ stored per phase are the median

and MAD of this population. At test time, a feature vector at time t is scored against the template at $\phi(t)$. This phase-specific normalization prevents normal phase-dependent firing variation from being mistaken as anomaly evidence: a residual energy spike at phase $\phi = 2$ is only abnormal if it exceeds the MAD of normal phase-2 spikes, not if it merely exceeds the global average. Under the official protocol the router reports no reliable period, and the global template is therefore used, on 30 of the 180 TSB-AD-M series and 3 of the 350 TSB-AD-U series; no series fails catastrophically in this fallback.



Phase conditioning prevents normal phase-dependent firing variation from being mistaken as anomaly evidence.

Figure 5: Phase-conditioned template construction. Each normal training step t is mapped to a phase bucket $\phi(t) = t \bmod \bar{p}$, contributes to the bucket population, and the per-phase median and MAD of $\mathbf{u}_t$ form the template. At test time, a sample at phase ϕ is scored against the matching bucket.

D Operational Protocol and Component Comparisons

Table 6 fixes the evidence source to E1 and varies the operational protocol of Section 3.1 under the official prefix. The three protocols are within 0.004 of each other on U and within 0.002 on M, the latter within the seed variation of Table 2, so the protocol is not where SPIRE's gains come from. A1 is kept as the shell of SPIRE because it scores the internal state produced after the current observation enters the causal backbone, which is the state that the eventification chain of Section 3.2 reads. Table 5 ablates the two period-aware components under the official protocol (Section 4.3). Under output-error scoring SRP is the main contributor on both splits, and SPF adds a small gain on U and is neutral to slightly negative on M. Under hybrid scoring, removing SRP and SPF together lowers the M score from 0.355 to 0.325 and the U score from 0.394 to 0.358 (second block). We therefore treat SRP as the load-bearing period-aware component and SPF as an optional refinement whose benefit is dataset-dependent, and we keep SRP+SPF as the default on both benchmarks for consistency instead of enabling SPF only where it helps.

E Mechanism Visualization

Figure 3 shows a representative strongly periodic series with a shaded anomaly. The recurrence violation first increases the SRP residual energy, then alters encoder firing and the burst or silence regimes, and finally elevates the internal-state and hybrid scores. The output-error score also rises but with noisier timing. The example illustrates the mechanism behind Table 2: internal-state evidence is informative on its own and, on series of this kind, more effective when fused with output-error evidence.

Table 5: Component ablation of the period-aware front-end (official protocol, VUS-PR). SRP is the dominant contributor; SPF helps on U and is approximately neutral on M.

Configuration	TSB-AD-U	TSB-AD-M
A1-E1, output-error scoring		
Spike-TCN baseline, no SRP/SPF	0.398	0.329
+ SRP only	0.402	**0.337**
+ SPF only	0.405	0.334
+ SRP + SPF	**0.409**	0.336
A1-E3, hybrid scoring		
Spike-TCN baseline, no SRP/SPF	0.358	0.325
+ SRP + SPF (SPIRE-H)	**0.394**	**0.355**

Table 6: Operational protocol comparison under output-error scoring (official protocol, VUS-PR).

Configuration	TSB-AD-U	TSB-AD-M
A1: causal current-step	0.409	0.336
A2: one-step prediction, warmup 25	0.413	0.337
A3: non-causal reconstruction	0.413	0.338
A1 + E3 hybrid, SPIRE-H	0.394	0.355

F Full Metric Report

Table 7 reports VUS-PR, AUC-ROC, and AUC-PR for the SPIRE configurations on both official Eval splits. VUS-PR is from the first training run, as in the main text; AUC-ROC and AUC-PR are computed from the saved score streams of the second training run, the run for which all three evidence sources were stored per series, so the three rows of each split are scored on the same streams.

Table 7: Full metric report for the main SPIRE configurations on the official TSB-AD Eval splits.

	TSB-AD-U			TSB-AD-M		
Configuration	VUS-PR	AUC-ROC	AUC-PR	VUS-PR	AUC-ROC	AUC-PR
SPIRE-E1 (A1-E1)	0.409	0.780	0.391	0.336	0.689	0.301
SPIRE-P (A1-E2)	0.288	0.722	0.255	0.342	0.688	0.333
SPIRE-H (A1-E3)	0.394	0.778	0.382	0.355	0.719	0.352

G Per-domain Breakdown under the Official Protocol

Tables 8 and 9 report hybrid (E3) and output-error (E1) VUS-PR by domain on the official Eval splits. One architecture and one set of hyperparameters per split is used across all domains, and the per-series normal template is the only adaptation to a domain. Performance ranges from near-perfect on strongly recurrent domains such as TAO and Exathlon to low where periodic structure is weak, and no per-domain tuning is used anywhere.

H Regime Analyses

Readout gain by router reliability (official protocol). Table 10 stratifies the official Eval series into equal-count terciles of the router reliability r (Eq. 1, the spectral concentration of the learned representation, measured on the normal prefix) and reports output-error VUS-PR, hybrid VUS-PR, and the within-tercile difference, which is the fair comparison since absolute scores are not comparable across terciles of different difficulty. The internal-state readout adds the most on the low-reliability tercile. This is the behaviour expected of fixed-weight late fusion, which interpolates the ranking quality of two streams. It helps where the two are of comparable quality and dilutes the stronger one where one dominates. On the high-reliability tercile output error is already strong and fusion dilutes it. On the low-reliability tercile output error degrades and the internal-state stream, which

Table 8: Per-domain VUS-PR on the official TSB-AD-M Eval split (17 domains, 180 series), sorted by the hybrid score.

Domain	n	E3	E1	Domain	n	E3	E1
TAO	11	0.994	0.778	OPPORTUNITY	7	0.165	0.157
Exathlon	25	0.905	0.912	SVDB	28	0.161	0.189
MSL	14	0.509	0.323	Genesis	1	0.124	0.316
SWaT	2	0.366	0.363	CATSv2	5	0.102	0.099
LTDB	4	0.307	0.322	CreditCard	1	0.095	0.109
SMD	20	0.303	0.368	MITDB	11	0.079	0.086
Daphnet	1	0.292	0.356	GECCO	1	0.035	0.035
SMAP	25	0.269	0.225	GHL	23	0.026	0.030
PSM	1	0.207	0.208				

Table 9: Per-domain VUS-PR on the official TSB-AD-U Eval split (23 domains, 350 series), sorted by the hybrid score.

Domain	n	E3	E1	Domain	n	E3	E1
TAO	2	0.994	0.974	Daphnet	1	0.388	0.386
Exathlon	30	0.939	0.943	IOPS	15	0.363	0.355
Stock	8	0.910	0.863	SMAP	17	0.343	0.361
NEK	8	0.799	0.733	CATSv2	1	0.330	0.341
YAHOO	30	0.610	0.615	WSD	20	0.281	0.255
OPPORTUNITY	27	0.587	0.670	NAB	23	0.269	0.273
SWaT	1	0.556	0.563	SVDB	18	0.178	0.210
TODS	13	0.492	0.649	Power	1	0.146	0.147
SMD	33	0.444	0.446	MITDB	7	0.145	0.153
MSL	7	0.394	0.396	SED	2	0.118	0.214
LTDB	8	0.393	0.414	UCR	70	0.057	0.065
				MGAB	8	0.008	0.012

keeps working through the global fallback of Section 3.3, compensates. We do not over-interpret the direction across terciles: r is the spectral concentration of the representation rather than raw periodicity, and the high-r tercile happens to contain the intrinsically hardest series, where output error itself is lowest.

Table 10: Within-tercile gain of hybrid scoring over output-error scoring on the official Eval splits, by router reliability r.

Split	Tercile	n	E1	E3 (hybrid)	within-tercile gain
M	low r	60	0.479	0.561	$+0.082$
M	mid r	60	0.375	0.371	-0.004
M	high r	60	0.153	0.132	-0.021
U	low r	116	0.482	0.493	$+0.011$
U	mid r	116	0.473	0.456	-0.017
U	high r	118	0.273	0.236	-0.037

Architectural gap on the 870-series superset (submission protocol, 50% prefix). Tables 11 and 12 report the periodicity-stratified and per-domain analysis on the 870-series TSB-AD-U superset that supports the architecture-axis regime claim in Section 4.4. These numbers were obtained under the protocol of the submitted version, with the first 50% of each series as the normal training prefix, and are kept separate from the official-protocol results; they are not comparable in absolute level to the other tables. They measure the gap between SPIRE and its ANN counterpart, a different quantity from the readout gain of Table 10. The SPIRE-H minus ANN gap grows monotonically with spectral concentration, from $+0.005$ on weakly periodic series to $+0.120$ on strongly periodic series, and phase-recurrent domains such as YAHOO, OPPORTUNITY, NEK, and ECG-style datasets benefit most, while weakly periodic or heterogeneous domains such as UCR and NAB show small or

near-zero gains. These results define the intended regime of the spiking architecture rather than a universal dominance claim.

Table 11: Periodicity-stratified analysis on the 870-series TSB-AD-U superset under the 50%-prefix submission protocol. The gap between SPIRE and the ANN counterpart grows with spectral concentration.

Periodicity stratum	SNN A1-E1	ANN A1-E1	SPIRE-H	SPIRE-H − ANN
High, $\bar{r} > 0.3$	0.345	0.275	**0.395**	+0.120
Medium, $0.1 < \bar{r} \leq 0.3$	0.290	0.280	0.305	+0.025
Low, $\bar{r} \leq 0.1$	0.250	0.250	0.255	+0.005

Table 12: Per-domain VUS-PR on the 870-series TSB-AD-U superset under the 50%-prefix submission protocol. The SPIRE-H minus ANN column tracks the periodicity-stratified pattern in Table 11.

Domain	Series	SPIRE-H	ANN A1-E1	SPIRE-H − ANN
YAHOO	367	0.408	0.290	+0.118
OPPORTUNITY	46	0.385	0.282	+0.103
NEK	88	0.376	0.286	+0.090
ECG-style	57	0.355	0.291	+0.064
Daphnet	39	0.295	0.272	+0.023
MITDB	31	0.288	0.277	+0.011
UCR	152	0.265	0.260	+0.005
NAB	90	0.250	0.249	+0.001

I Feature-level Ablation of the Internal-state Vector

Table 13 reports, on both official Eval splits, the VUS-PR of each feature of $\mathbf{u}_t$ scored alone through the same phase template, the leave-one-out score when that feature is removed from the five-dimensional vector, and the sub-block scores. On M the full vector (0.342) exceeds every single feature and every sub-block, the residual energy is load-bearing (removing it costs 0.072), and removing any single spike-derived feature changes the score by at most 0.001: the four spike-derived features are mutually redundant, which is a robustness property rather than a sign of irrelevance, since adding them to the residual raises the score from 0.324 to 0.342 with $\Pr(\Delta > 0) = 1.00$. On U the residual remains load-bearing, the four spike-derived features are individually weak, and dropping any one of them raises the score slightly, so on U they are mildly redundant rather than complementary. The full readout at 0.288 sits below the residual energy alone at 0.313 and far below output error at 0.409, which is why fusion does not help on this split.

J Sensitivity of the Fusion Weight and Scaler

Table 15 sweeps the mixing weight λ_{spk} with $\lambda_{\text{err}} = 1 - \lambda_{\text{spk}}$ and compares the three robust scalers at $\lambda_{\text{spk}} = 0.7$. All scalers are fitted only on the normal training prefix, so there is no test-set leakage. On M, fusion improves over output error for every weight between 0.25 and 0.75 and for every scaler. On U, adding internal-state weight lowers the score monotonically for every scaler. The constant 1.4826 that scales the MAD applies uniformly to both streams; rankings within a stream are unaffected by it. Table 14 reports the same sweep on the official Tuning splits, which is the principled way to select the weight. On the 20 multivariate tuning series output error alone is best and every positive weight lowers the score, so a tuning-split selection returns $\lambda_{\text{spk}} = 0$ and the Eval score 0.336 rather than the hybrid 0.355; on the 48 univariate tuning series the scores are nearly flat for $\lambda_{\text{spk}} \leq 0.5$ (0.312 at 0.25 against 0.310 at 0), so the selection returns $\lambda_{\text{spk}} = 0.25$ and 0.405 on Eval, slightly below output error alone (0.409). The fixed weight 0.7 used for SPIRE-H is carried over from the submission, and the M Eval gain of fusion is a regime effect that the small tuning split does not predict.

Table 13: Single-feature, leave-one-out, and sub-block VUS-PR of the internal-state readout on the official Eval splits. Fusion rows use the fixed median/MAD scalers and $\lambda_{\mathrm{spk}} = 0.7$.

	TSB-AD-U		TSB-AD-M	
Feature	alone	leave-one-out	alone	leave-one-out
residual energy r_t^R	0.313	0.207	0.324	0.270
encoder firing rate ρ_t^{enc}	0.199	0.292	0.282	0.341
backbone activity mean ρ_t^{back}	0.206	0.290	0.227	0.343
burst ratio b_t	0.170	0.292	0.271	0.343
silence ratio q_t	0.168	0.292	0.248	0.342
full five-dimensional vector		0.288		0.342
four spike-derived features (without r_t^R)		0.207		0.270
strictly spike-derived sub-block $(\rho_t^{\mathrm{enc}}, b_t, q_t)$		0.180		0.275
output error E1		0.409		0.336
E1 fused with residual energy only		0.406		0.348
E1 fused with the spike-derived sub-block		0.389		0.336
E1 fused with the full vector (E3)		0.394		0.355

Table 14: Mixing-weight sweep on the official Tuning splits (VUS-PR, median/MAD scaler), used for per-split weight selection.

Split	n	$\lambda_{\mathrm{spk}} = 0$ (E1)	0.25	0.5	0.7	0.75	1 (E2)
TSB-AD-U Tuning	48	0.310	**0.312**	0.307	0.305	0.300	0.258
TSB-AD-M Tuning	20	**0.359**	0.321	0.318	0.317	0.317	0.308

K Stream-level Decomposition of the Neuron Controls

Table 16 reports, for SPIRE and the matched ReLU activation-template control, the three evidence sources separately. The matched ReLU control reads the same five statistics from activations. On M the continuous control has a higher output-error score and an equal internal-state score, but a lower fused score, so the LIF advantage on M is a fusion-level property: its two streams are more complementary. On U the LIF model is higher on output error and in fusion, and slightly lower on the internal-state stream alone. The rate-matched stateless binary control is reported at the fused level in Table 3. Both SPIRE and the matched ReLU control were trained three times with different seeds: at the fused level the ReLU control reaches 0.357 ± 0.005 on U and 0.345 ± 0.004 on M against 0.394 ± 0.000 and 0.353 ± 0.001 for LIF (Table 2), and every LIF run exceeds every ReLU run on both splits, by about 0.04 on U and about 0.01 on M, so the seed variation of either model does not account for the paired differences in Table 3. The rate-matched binary control is a single training run.

L Hyperparameters

Table 17 lists the SPIRE hyperparameters used in all experiments. Split-specific values were selected on the official Tuning splits and are shared by every evaluation series of that split; all other values are common to both splits.

Table 15: Sensitivity to the mixing weight and to the scaler on the official Eval splits (VUS-PR).

	λ_{spk} (median/MAD scaler)					scaler at $\lambda_{\mathrm{spk}} = 0.7$		
Split	0 (E1)	0.25	0.5	0.75	1 (E2)	median/MAD	mean/std	median/IQR
TSB-AD-U	0.409	0.405	0.399	0.392	0.288	0.394	0.402	0.404
TSB-AD-M	0.336	0.359	0.357	0.353	0.342	0.355	0.339	0.341

Table 16: Evidence sources of SPIRE (LIF) and of the matched ReLU activation-template control, official Eval splits (VUS-PR).

	TSB-AD-U			TSB-AD-M		
Unit	E1	E2	E3	E1	E2	E3
LIF (SPIRE)	0.409	0.288	0.394	0.336	0.342	0.355
Matched ReLU activation-template	0.374	0.306	0.362	0.358	0.343	0.340

Table 17: SPIRE hyperparameters (official protocol).

Module	Hyperparameter	TSB-AD-U	TSB-AD-M
Pre-encoder	kernel size	3	3
	encoder width D	32	64
Period router	top-K frequencies	2	2
	reliability threshold $r_{\min}$	0.05	0.15
	softmax temperature τ	1.0	1.0
SRP	residual strength α	0.5	0.5
LIF	micro-steps T_s	6	4
	membrane time constant τ_m	1.5	2.0
	threshold V_{th}	1.0	1.0
	surrogate gradient	ATan	ATan
Backbone	TCN levels L	3	3
	backbone width D_{ch}	16	16
	kernel size	16	8
	dropout	0.1	0.2
	SPF position / mode	level 2 / spiking	level 2 / spiking
Hybrid score	λ_{spk}, λ_{err}	0.7, 0.3	0.7, 0.3
	robust scale (MAD) factor	1.4826	1.4826
Training	optimizer / learning rate	Adam / 10^{-3}	Adam / 10^{-3}
	maximum epochs / patience	50 / 10	50 / 10
	window size / training stride	50 / 5	50 / 5
	normal training prefix	official per-series	official per-series

M Extended Future Directions

The three operations of SPIRE, current formation, eventification, and phase-conditioned readout, extend to three settings raised during review, applied in conditional, adaptive, and masked variants; we claim no results for them. For confounding variables in a multivariate series, the current phase template is fit on the joint representation, so a shared shift driven by a hidden common cause can enter the normal template and mask a genuine anomaly. The natural extension is a cross-variable conditional template, in which each channel's phase-conditioned normal is estimated given the co-observed channels, so that a deviation is judged relative to what the other variables imply rather than in isolation; this separates a channel-specific anomaly from a confound-driven co-movement. For strongly correlated series with legitimate changes in scale and amplitude, the templates are fit once on the normal prefix and are therefore static, so slow legitimate drift in amplitude could be read as anomalous. The phase residual already cancels part of a slow drift because it is a cross-period difference, but scale changes need an explicit mechanism: an adaptive template refresh that updates

the per-phase median and MAD online with a slow forgetting factor, so that the normal tracks legitimate slow drift while the fast phase residual still exposes abrupt violations. For missing or inaccurate entries, the method currently assumes complete input. The extension is a mask-conditioned readout, in which a missing or unreliable entry is gated out of both the residual and the template comparison so that the score is computed only over reliably observed channels, while the LIF membrane carries state across short gaps so that a gap neither triggers a false spike nor blanks the score.

N Model Size and Compute

Table 18 gives the parameter counts of SPIRE and of the matched ReLU control. The LIF model is size-matched to its continuous counterpart (the difference is the learnable membrane parameters) and is two to three orders of magnitude smaller than pre-trained TSPulse, so the gain is not bought with capacity. On TSB-AD-M the count depends on the number of channels C of each series; we report the count at the mean channel count ($C = 29$) and the range over the split. All experiments ran on a single NVIDIA RTX 4090 GPU. One full pass of SPIRE over a split, including per-series training, template and scaler fitting, scoring, and metric computation, takes about 4.8 hours on TSB-AD-U (350 series) and 2.6 hours on TSB-AD-M (180 series); the matched ReLU control takes about 1.5 hours per split, and the rate-matched binary control about 5.5 hours per split because it trains a LIF reference before the calibrated binary model. The experiments reported in this paper, including the controls, the component ladder, the protocol comparison, and the training repetitions, amount to about 60 GPU-hours; the full project, including development and the runs of the submitted version, used several times more.

Table 18: Parameter counts of SPIRE and of the matched ReLU control under the configurations of Table 17.

Model	TSB-AD-U ($C = 1$)	TSB-AD-M (at $C = 29$; range over the split)
SPIRE (LIF)	34.4K	30.7K (25.0K–76.5K)
Matched ReLU activation-template	32.0K	28.3K (22.7K–74.1K)

O Broader Impact

SPIRE is a from-scratch detector for periodic time series and is not tied to a particular deployment. Its direct positive use is the monitoring of industrial, infrastructure, and sensor time series with a model of about 30K parameters that is trained per series on commodity hardware. Like any anomaly detector, it can cause harm when it is used as intended but gives incorrect results: a false alarm can trigger unnecessary intervention, and a missed anomaly can delay a response. SPIRE is designed for phase-recurrent structure and is less suited to weakly periodic data and to cross-channel anomalies (Section 5), so it should be validated on the target domain and combined with domain expertise before its scores drive decisions. We see no direct path from this work to malicious use; the released asset is research code evaluated on a public benchmark.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Introduction state the contribution as phase-aligned residual eventification for time-series anomaly detection, give the official-protocol results without a leaderboard-record claim, state that the benefit of internal-state evidence is regime-dependent, and restrict the architectural claim to phase-recurrent data. The experimental claims are tied to the evidence-source ablation in Section 4.2, the operational and component comparisons in Section 4.3, and the matched controls and regime analyses in Section 4.4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations in Section 5. In particular, it states that internal-state evidence is complementary rather than uniformly strongest, that the SNN-specific gain is significant but moderate in magnitude, that the TSB-AD-M result depends on hybrid scoring, that SPIRE is designed for periodic structure and is less suited to pure cross-channel correlation shifts, that it is not a fully streaming detector, and that the rate-matched control is a single training run.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present formal theoretical results. The equations in Sections 3.2–3.4 define the residual injection, spike-state features, templates, and scores, but they are not stated as theorems requiring proofs.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 4.1 specifies the datasets, official tuning/evaluation splits, training prefix, metric, optimizer, and default hyperparameters. Appendix L gives the full per-split configuration, Appendix N the model size and compute, and the released code specifies the remaining implementation details needed to reproduce the main tables.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code for SPIRE, the two matched controls, the score ablations, the offline feature and sensitivity analyses, and the visualization is released at `https://github.com/Ray-WM/SPIRE`, with instructions to obtain the public TSB-AD benchmark data and to reproduce the reported tables. The paper uses the public TSB-AD benchmark rather than introducing a new dataset.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4.1 gives the training/test protocol, official split usage, primary and secondary metrics, optimizer, training budget, normal-prefix construction, and model hyperparameters. Appendix L and Appendix J provide the full default configuration and sensitivity sweeps.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Every comparison that carries a claim is accompanied by a paired bootstrap over series (10,000 resamples) with the mean difference, its 95% percentile interval, $\Pr(\Delta > 0)$, and win/tie/loss counts (Section 4.1, Tables 2 and 3); the factor of variability captured is the choice of evaluation series. The main configuration and the matched ReLU control were additionally trained three times with different seeds (Table 2 and Appendix K), and Table 2 reports the mean and standard deviation over training runs. The rate-matched binary control is a single training run, which is stated in the text.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix N reports the GPU type (a single NVIDIA RTX 4090), the wall-clock time of one full pass over each split, the cost of the controls, and an estimate of the total compute of the reported experiments and of the full project. Section 4.1 reports the model scale and training budget.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The work uses public benchmark data and does not involve private personal data, human-subject experiments, unsafe content generation, or deceptive use. The code release and the paper text comply with the NeurIPS Code of Ethics.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Appendix O discusses the broader impact. The positive impact is the monitoring of industrial, infrastructure, and sensor time series with a small per-series model. The negative impact is that false alarms or missed anomalies can cause unnecessary intervention or delayed response, especially outside the periodic regime SPIRE is designed for; domain validation and the regime limits of Section 5 are named as the mitigation.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release high-risk generative models, scraped datasets, pre-trained language models, or assets with substantial dual-use misuse risk. The released asset is research code for reproducing time-series anomaly detection experiments on public benchmarks.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Existing datasets, metrics, and external baselines are cited in the paper, including TSB-AD, VUS-PR, and the reported comparison methods. The TSB-AD package is used under its Apache-2.0 license with data from the official archives of the benchmark authors; the backbone builds on SeqSNN (MIT) and SpikingJelly (its own open-source license). The released repository names these sources with their URLs.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The new asset is the implementation and experiment scripts for SPIRE, including code for the internal-state score, the matched ReLU and rate-matched binary controls, the periodicity analysis, and the visualization. The repository includes a README with environment setup, data preparation, the commands that reproduce the reported tables, and the license (MIT).

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The work does not involve crowdsourcing or human-subject experiments.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The work does not involve crowdsourcing or human-subject experiments and therefore does not require IRB approval or an equivalent human-subject review.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core methodology does not use LLMs as a model component, data-generation mechanism, evaluator, or source of scientific claims. Any use of language tools for writing, editing, or formatting does not affect the method, experiments, or originality of the research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.