KRAFT: KOOPMAN-REFERENCED ADAPTIVE FLOW FOR TIME-SERIES ANOMALY DETECTION UNDER LATENT DOMAIN SHIFTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Time series anomaly detection (TSAD) is challenging not only because anomaly labels are scarce, but also because time series are inherently nonstationary. Changes in normal behavior may be mistaken as anomalies, motivating TSAD under latent domain shifts, where domain labels and shift boundaries are unknown. To address this problem, we propose KRAFT, a Koopman-Referenced Adaptive Flow for TSAD under latent domain shifts. KRAFT conditions flow matching on Koopman-predicted latent dynamics to generate evolving normal references. To accommodate shifts unexplained by this evolution, Koopman-pushed maximum mean discrepancy detects persistent mismatch and triggers validation-guided adapter updates, while residual deviations from the reference provide anomaly scores. Theoretical analysis supports separation between shifted normal behavior and anomalies under some mild conditions, preserving anomaly detection performance under latent domain shifts. Extensive experiments on public TSAD and domain drift benchmarks demonstrate the effectiveness of KRAFT.

1 INTRODUCTION

Time series anomaly detection (TSAD) identifies observations or temporal patterns that deviate from expected system behavior. It plays an important role in industrial monitoring (Deng & Hooi, 2021), infrastructure maintenance (Moallemi et al., 2022), and healthcare (Zhou et al., 2019). In practical deployments, however, normal behavior is nonstationary: system states evolve, operating conditions change, and the resulting distribution shifts can be mistaken for anomalies. Conventional unsupervised TSAD methods learn normal patterns from historical data without anomaly annotations and detect deviations through reconstruction or prediction (Audibert et al., 2020). However, their fixed normal references can become outdated under distribution shifts, causing legitimate changes in normal behavior to be flagged as anomalies.

To accommodate such changes, drift-aware methods explicitly account for nonstationarity through decomposition and transformation of time-varying components (Wang et al., 2023). However, suppressing these variations can also attenuate anomaly signatures when drift and anomalies exhibit similar statistical patterns. Domain adaptation and generalization methods learn transferable representations through cross-domain alignment or domain-invariant learning (Darban et al., 2025; Jacob & Diao, 2025). However, these approaches commonly rely on known domain labels during learning, which are usually unavailable in practice. Test-time adaptation methods use unlabeled deployment data to adjust detectors to changing normality (Kim et al., 2024; 2026). However, anomalies in test data can contaminate model updates and degrade detection performance. These limitations motivate our study of *TSAD under latent domain shifts*, where both domain labels and the timing of domain shifts are unknown. The goal is to adaptively detect domain shifts and distinguish them from anomalies without anomaly labels during testing. This problem presents two closely coupled challenges.

Identifying and modeling latent domain shifts. Domain labels, shift times, and shift types are all unknown. Different sources of shifts can produce distinct statistical patterns: equipment aging often induces gradual evolving shifts, whereas environmental changes or human interventions may cause abrupt shifts in operating behavior. These patterns can coexist, complicating detection across

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

different temporal scales. Drift-aware transformations do not ensure reliable identification of such heterogeneous shifts (Wang et al., 2023), while approaches relying on known domain labels cannot be applied under the latent domain setting (Darban et al., 2025; Jacob & Diao, 2025). Therefore, the challenge is to detect latent domain shifts with diverse statistical patterns.

Preventing anomaly contamination during adaptation. Under such heterogeneous shifts, the detector must adapt to shifted normal without learning anomalies as normal. Both shifted normal samples and true anomalies can deviate from the current reference, making reliable adaptation challenging. Adapting to anomalies can contaminate the model and severely degrade detection performance. Existing test-time adaptation methods remain vulnerable to anomaly contamination (Kim et al., 2024; 2026). Rejecting all deviating samples, however, would prevent adaptation to a new normal domain. Therefore, the challenge is to adapt to shifted normal behavior while preventing anomaly contamination and preserving separability between shifted normal and true anomalies.

To address these challenges, we propose KRAFT, a Koopman-Referenced Adaptive Flow framework for TSAD under latent domain shifts. We use a Koopman operator to predict how latent domain states evolve under normal system dynamics, such as routine equipment aging. These predicted states condition flow matching to generate a normal reference that captures *endogenous shift* arising from the system's own evolution. Building on this reference, we introduce Koopman-pushed maximum mean discrepancy (KP-MMD) to quantify distributional mismatch unexplained by the predicted evolution. Persistent mismatch evidence guides selective updates to a lightweight adapter embedded in the flow network to accommodate *exogenous shift* arising from environmental changes or human interventions. A validation-guided acceptance mechanism admits an adapter update only when it reduces a capped risk on held-out recent observations, helping limit contamination during adaptation. Anomaly scores are computed from discrepancies between observations and the Koopman-referenced adaptive flow reference. By separately modeling endogenous and exogenous shift and evaluating the remaining residuals, the framework helps preserve separability between shifted normal and true anomalies.

Our main contributions are as follows:

- We study TSAD under latent domain shifts, where anomaly labels, domain labels, shift times, and shift types are all unknown, emphasizing adaptation without losing anomaly separability. This problem has rarely been studied.
- We propose KRAFT, which combines Koopman-conditioned flow matching with KP-MMD to model endogenous shift and identify exogenous shifts, enabling the characterization of latent domain shifts with diverse statistical patterns.
- We develop a validation-guided restricted adapter to adapt the exogenous shift while limiting anomaly contamination. We provide theoretical guarantees for preserving anomaly separability under some mild conditions.
- We evaluate KRAFT across diverse public TSAD benchmarks and real drift datasets, and build a new multi-domain TSAD benchmark to systematically assess its ability under latent domain shift scenarios.

2 Related Work

Unsupervised Time-Series Anomaly Detection. Unsupervised TSAD learns normal patterns without anomaly annotations and detects deviations through prediction, reconstruction, or representation learning. Prediction-based methods model temporal and inter-variable dependencies and detect anomalies through forecasting errors (Malhotra et al., 2015; Deng & Hooi, 2021). Reconstruction-based approaches use autoencoders, stochastic recurrent networks, or Transformers to characterize normal behavior and score reconstruction discrepancies (Audibert et al., 2020; Su et al., 2019; Tuli et al., 2022). Representation-based methods exploit temporal one-class objectives, association discrepancies, or contrastive learning to distinguish anomalies (Shen et al., 2020; Xu et al., 2022; Yang et al., 2023). Consequently, in long-horizon monitoring with domain shifts, these models often fail to capture evolving normality, leading to high false alarm rates on new domains.

Time-Series Domain Adaptation and Generalization. Domain adaptation transfers knowledge across domains by aligning temporal dependencies (Cai et al., 2021), matching distributions through

adversarial kernels (Liu & Xue, 2021), or aligning temporal and frequency representations (He et al., 2023). For anomaly detection, contrastive alignment and synthetic anomaly generation further support transfer from labeled source data (Darban et al., 2025). Domain generalization instead learns transferable representations from source domains without target-domain training, including domain-invariant latent representations for TSAD (Jacob & Diao, 2025). However, these approaches typically organize learning around explicit domain labels, limiting their applicability to evolving streams with latent domain shifts.

Drift-Aware Methods and Test-Time Adaptation. Drift-aware TSAD addresses nonstationarity through dynamic decomposition and reconstruction (Wang et al., 2023), adaptive model pools (Yoon et al., 2022), or online learning with drift-adaptive learning rates (Ding et al., 2023). Test-time adaptation for TSAD updates detectors using unlabeled deployment data, combining trend estimation with updates on predicted normal samples (Kim et al., 2024), or curating potential false positives for selective adaptation (Kim et al., 2026). Nevertheless, distinguishing shifted normal from anomalies remains challenging, potentially causing contaminated updates or overly conservative adaptation.

3 Problem Formulation

Let $\mathbf{x}_t \in \mathbb{R}^D$ denote the observation at time t and $\mathcal{H}_{t-1} = (\mathbf{x}_{t-W}, \ldots, \mathbf{x}_{t-1})$ its history of length W. Each observation is associated with an unobserved anomaly label $y_t \in \{0, 1\}$ and latent domain state $d_t \in \mathcal{D}$:

$$\mathbf{x}_t \mid \mathcal{H}_{t-1}, d_t, y_t \sim \begin{cases} P^{\mathrm{N}}(\cdot \mid \mathcal{H}_{t-1}, d_t), & y_t = 0, \\ P^{\mathrm{A}}(\cdot \mid \mathcal{H}_{t-1}, d_t), & y_t = 1, \end{cases} \tag{1}$$

where $\mathcal{D}$ is the latent domain space, and P^{N} and P^{A} denote the normal and anomalous conditional distributions.

Definition 1 (Latent Domain Shift)**.** A latent domain shift is a change in the unobserved domain state d_t that alters the normal conditional law $P^{\mathrm{N}}(\cdot \mid \mathcal{H}_{t-1}, d_t)$. Given unlabeled historical observations $\mathcal{X}_0 = (\mathbf{x}_1, \ldots, \mathbf{x}_{T_0})$, the detector processes $\{\mathbf{x}_t\}_{t>T_0}$ sequentially, with neither $\{y_t\}$ nor $\{d_t\}$ observed and with unknown change times.

Assumption 1 (Endogenous and exogenous shift)**.** To describe shift with diverse statistical patterns, we assume that normal dynamics admit a continuation under a fixed domain. For reference time $r < t$, let $P_r^{\mathrm{N}} = P^{\mathrm{N}}(\cdot \mid \mathcal{H}_{r-1}, d_r)$ and $P_t^{\mathrm{N}} = P^{\mathrm{N}}(\cdot \mid \mathcal{H}_{t-1}, d_t)$ denote the reference and current normal distributions. Let $Q_{t|r}^{\mathrm{N}}$ denote the normal distribution at t under continuation of the reference dynamics with domain d_r held fixed. We define

$$\Delta_{r,t}^{\mathrm{endo}} := Q_{t|r}^{\mathrm{N}} - P_r^{\mathrm{N}}, \qquad \Delta_{r,t}^{\mathrm{exo}} := P_t^{\mathrm{N}} - Q_{t|r}^{\mathrm{N}}, \qquad P_t^{\mathrm{N}} - P_r^{\mathrm{N}} = \Delta_{r,t}^{\mathrm{endo}} + \Delta_{r,t}^{\mathrm{exo}}. \tag{2}$$

Here, $\Delta_{r,t}^{\mathrm{endo}}$ denotes the *endogenous shift* induced by normal evolution within d_r, and $\Delta_{r,t}^{\mathrm{exo}}$ denotes the *exogenous shift* beyond this evolution. Both are signed measures and may coexist; their interpretation depends on the assumed normal continuation.

In practice, endogenous shift may correspond to predictable aging or gradual sensor shift under unchanged operating conditions, whereas exogenous shift may arise from changes in environmental conditions, operating modes, or system configurations beyond the assumed normal evolution. Both can alter normal distribution without constituting anomalies, motivating the following detection task.

Definition 2 (TSAD under Latent Domain Shift)**.** Given $\mathcal{X}_0$ and sequential observations $\{\mathbf{x}_t\}_{t>T_0}$, the objective is to infer anomaly labels and identify latent domain shifts while tracking the evolving normal distribution. Final anomaly decisions are made without access to ground-truth anomaly labels or domain identities.

4 Methodology

As shown in Figure 1, KRAFT constructs a dynamics-conditioned normal reference using Koopman propagation and conditional flow matching. KP-MMD compares current latent observations with the propagated reference to identify mismatch beyond modeled endogenous shift. Persistent mismatch triggers low-rank adapter updates, which are accepted only after validation. Final anomaly decisions use standardized residuals recomputed after validated adaptation relative to the selected normal reference.

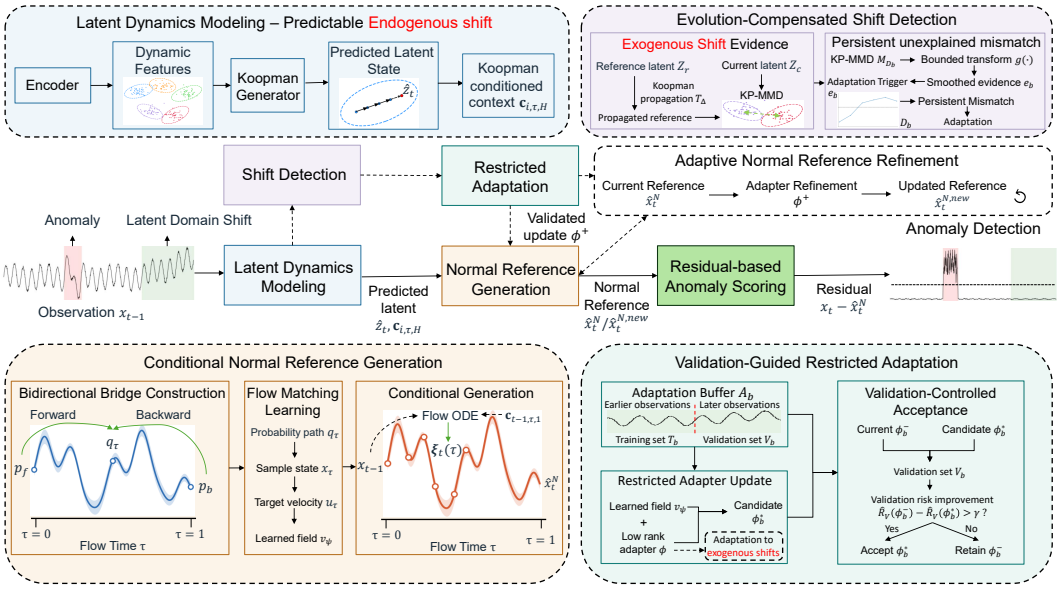


Figure 1: Overview of KRAFT. Koopman-predicted latent evolution guides conditional flow matching to generate a dynamics-conditioned normal reference. Persistent KP-MMD mismatch triggers validation-guided adapter updates to refine the reference, while anomalies are identified through residual discrepancies.

4.1 Dynamics-Conditioned Normal Reference

Modeling endogenous shift. Following Assumption 1, we use Koopman propagation to capture normal endogenous shift and guide reference generation. An encoder summarizes the history window $\mathcal{H}_i = (\mathbf{x}_{i-W+1}, \ldots, \mathbf{x}_i)$ into a temporal context, which is mapped to a dynamical representation:

$$\mathbf{h}_i = E_\theta(\mathcal{H}_i), \qquad \mathbf{z}_i = P_\eta(\mathbf{h}_i), \qquad \widehat{\mathbf{z}}_{i+\Delta} = \exp(K\Delta)\mathbf{z}_i. \tag{3}$$

Here, $\mathbf{h}_i$ provides historical context, $\mathbf{z}_i \in \mathbb{R}^m$ is a dynamical representation learned to support approximately linear evolution, and $K \in \mathbb{R}^{m\times m}$ is a learned Koopman generator. The representation $\mathbf{z}_i$ retains domain-dependent context without observing d_i, while its propagation models local endogenous shift.

We parameterize the finite-dimensional Koopman generator as $K = (A - A^\top) - \mathrm{diag}(\mathrm{softplus}(\mathbf{b}_K))$, where $A \in \mathbb{R}^{m\times m}$ and $\mathbf{b}_K \in \mathbb{R}^m$ are trainable. Its negative-definite symmetric part ensures contractive latent propagation. Starting from each encoded history, this propagation approximates the continuation of normal dynamics over the prediction horizon. The historical context and propagated state then jointly condition the flow to generate a normal reference that accounts for modeled endogenous shift.

Dynamics-conditioned flow matching. Building on the latent dynamics, we use conditional flow matching (Lipman et al., 2022) to construct an observation-space normal reference. For historical training endpoints $\mathbf{x}_i$ and $\mathbf{x}_j$ with $i < j$, let $H = j - i$ be the prediction horizon and $\tau \in [0, 1]$ the normalized path time. Koopman propagation gives the predicted latent state $\mathbf{z}_{i,\tau} = \exp(K\tau H)\mathbf{z}_i$ at elapsed time τH. Its instantaneous rate of change is $K\mathbf{z}_{i,\tau}$ in the original time coordinate. To translate this evolution into a training path informed by both endpoints, forward and backward predictors define diagonal Gaussian densities p_f and p_b using their respective endpoint contexts and the state–rate pair $(\mathbf{z}_{i,\tau}, K\mathbf{z}_{i,\tau})$.

We then combine these densities through a weighted product to define the conditional probability path:

$$q_\tau(\mathbf{x} \mid i, j) \propto p_f(\mathbf{x} \mid i, j, \tau)^{1-\tau} p_b(\mathbf{x} \mid i, j, \tau)^\tau, \qquad q_\tau = \mathcal{N}\big(\boldsymbol{\mu}_\tau, \mathrm{diag}(\boldsymbol{\sigma}_\tau^2)\big). \tag{4}$$

Here, $\boldsymbol{\mu}_\tau$ and $\boldsymbol{\sigma}_\tau$ are the path mean and componentwise standard deviation. The parameterization ensures $\boldsymbol{\mu}_0 = \mathbf{x}_i$ and $\boldsymbol{\mu}_1 = \mathbf{x}_j$. Details and the Gaussian fusion formulas are provided in Appendix A.1.

Along this path, a Gaussian draw $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, I_D)$ held fixed across τ gives the path sample $\mathbf{x}_\tau$ and target velocity $\mathbf{u}_\tau$:

$$\mathbf{x}_\tau = \boldsymbol{\mu}_\tau + \boldsymbol{\sigma}_\tau \odot \boldsymbol{\varepsilon}, \qquad \mathbf{u}_\tau = \partial_\tau \boldsymbol{\mu}_\tau + \partial_\tau \boldsymbol{\sigma}_\tau \odot \boldsymbol{\varepsilon}. \tag{5}$$

Using these targets, we train the conditional velocity field v_ψ, with parameters ψ, by

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{i,j,\tau,\boldsymbol{\varepsilon}} \left[\left\| v_\psi(\mathbf{x}_\tau, \tau; \mathbf{c}_{i,\tau,H}) - \mathbf{u}_\tau \right\|_2^2 \right], \qquad \mathbf{c}_{i,\tau,H} = (\mathbf{x}_i, \mathbf{h}_i, \mathbf{z}_i, \mathbf{z}_{i,\tau}, H), \tag{6}$$

where the expectation is over training endpoint pairs, $\tau \sim \mathcal{U}(0,1)$, and Gaussian noise. Although supervision uses both endpoints, the conditioning vector $\mathbf{c}_{i,\tau,H}$ uses only the starting history, its propagated latent state, and the horizon.

Finally, at time t, we set $i = t-1$ and $H = 1$ and integrate the observation-space trajectory $\boldsymbol{\xi}_t(\tau)$ to obtain the prediction $\widehat{\mathbf{x}}_t$:

$$\frac{\mathrm{d}\boldsymbol{\xi}_t(\tau)}{\mathrm{d}\tau} = v_\psi(\boldsymbol{\xi}_t(\tau), \tau; \mathbf{c}_{t-1,\tau,1}), \qquad \boldsymbol{\xi}_t(0) = \mathbf{x}_{t-1}, \qquad \widehat{\mathbf{x}}_t = \boldsymbol{\xi}_t(1). \tag{7}$$

Conditioning the velocity field on Koopman-predicted latent evolution throughout integration yields a reference reflecting how normal observations would evolve under modeled endogenous shift.

4.2 Evolution-Compensated Shift Detection

Evolution-compensated discrepancy. A direct comparison between historical and current distributions can respond to endogenous evolution even when the normal mechanism is unchanged. We therefore compare current latent observations with a reference propagated to the same temporal positions.

Let $\mathcal{Z}_r = \{\mathbf{z}_{r_\ell}\}_{\ell=1}^n$ and $\mathcal{Z}_c = \{\mathbf{z}_{c_\ell}\}_{\ell=1}^n$ be reference and current windows, each containing n latent states, with corresponding time indices satisfying $c_\ell - r_\ell = \Delta$, and empirical distributions $\widehat{P}_r = n^{-1}\sum_\ell \delta_{\mathbf{z}_{r_\ell}}$ and $\widehat{P}_c = n^{-1}\sum_\ell \delta_{\mathbf{z}_{c_\ell}}$, where $\delta_{\mathbf{z}}$ denotes a point mass at $\mathbf{z}$. Using the propagation map $T_\Delta(\mathbf{z}) = \exp(K\Delta)\mathbf{z}$, we define the Koopman-pushed maximum mean discrepancy (KP-MMD) as

$$D_{\mathrm{KP}} = \mathrm{MMD}_k\left(\widehat{P}_c, T_\Delta(\widehat{P}_r)\right) = \left\| \frac{1}{n}\sum_{\ell=1}^n \phi_k(\mathbf{z}_{c_\ell}) - \frac{1}{n}\sum_{\ell=1}^n \phi_k(T_\Delta(\mathbf{z}_{r_\ell})) \right\|_{\mathcal{H}_k}, \tag{8}$$

where k is a bounded characteristic kernel, ϕ_k is its feature map into the reproducing kernel Hilbert space $\mathcal{H}_k$.

By approximating the latent counterpart of $Q^{\mathrm{N}}_{t|r}$ in Assumption 1, the propagated reference allows D_{KP} to quantify mismatch beyond modeled normal evolution. Together with validation-guided restricted adaptation in Section 4.3, this evidence supports adaptation to exogenous shift while mitigating anomaly contamination of the normal reference.

Persistent mismatch triggering. To reduce sensitivity to isolated discrepancies, we aggregate KP-MMD evidence across successive blocks. For block b, let D_b denote the statistic in Eq. 8. We compute the smoothed evidence $e_b = (1-\alpha)e_{b-1} + \alpha g(D_b)$, where g is a bounded nondecreasing transformation and $\alpha \in (0,1]$ controls smoothing. Adaptation is triggered when hysteresis-filtered evidence persists for p consecutive blocks, with entry and exit thresholds calibrated before deployment. Detailed triggering rules are provided in Appendix A.2. The trigger initiates candidate adaptation; acceptance is determined by the validation procedure below.

4.3 Validation-Guided Restricted Adaptation

Once persistent mismatch triggers adaptation, we seek a shared, low-complexity correction for exogenous shift unexplained by the normal reference. Therefore, we use low-rank adapters to learn small corrections and held-out validation to mitigate anomaly contamination.

Restricted self-supervised updates. To parameterize this correction, we insert low-rank residual adapters after the residual blocks of v_ψ. For hidden activation $\mathbf{a}^{(\ell)} \in \mathbb{R}^h$ at layer ℓ,

$$\widetilde{\mathbf{a}}^{(\ell)} = \mathbf{a}^{(\ell)} + U_\ell V_\ell^\top \mathbf{a}^{(\ell)} + \mathbf{b}_\ell, \qquad U_\ell, V_\ell \in \mathbb{R}^{h \times r_a}, \quad r_a \ll h, \tag{9}$$

where h is the hidden dimension, r_a the adapter rank, and $\mathbf{b}_\ell \in \mathbb{R}^h$ a bias. Initializing U_ℓ and $\mathbf{b}_\ell$ to zero and V_ℓ randomly preserves the pretrained function. Only the adapter parameters $\varphi = \{U_\ell, V_\ell, \mathbf{b}_\ell\}_\ell$ are updated; all pretrained components remain frozen.

For an adaptation trigger at block b, buffer $\mathcal{A}_b$ contains all unlabeled prediction pairs $(\mathcal{H}_{i-1}, \mathbf{x}_i)$ collected over the associated persistent-mismatch interval up to that trigger. We split their target indices chronologically into an earlier training subset $\mathcal{T}_b$ and a later validation subset $\mathcal{V}_b$. Candidate updates use only $\mathcal{T}_b$, while $\mathcal{V}_b$ is reserved for acceptance evaluation.

Using these unlabeled observations, we obtain candidate adapter parameters $\varphi_b^\star$ by approximately minimizing

$$\varphi_b^\star \approx \arg\min_{\varphi} \Big[\lambda_{\mathrm{FM}} \widehat{\mathcal{L}}_{\mathrm{FM},\mathcal{T}_b}(\varphi) + \lambda_{\mathrm{pred}} \widehat{\mathcal{L}}_{\mathrm{pred},\mathcal{T}_b}(\varphi) + \lambda_{\mathrm{reg}} \|\varphi - \varphi_b^0\|_2^2 \Big], \tag{10}$$

where φ_b^0 denotes the adapter parameters at the start of the current episode and remain fixed as its regularization anchor. The flow-matching term applies Eq. 6 to training endpoints with the adapters enabled, and the prediction term is $\widehat{\mathcal{L}}_{\mathrm{pred},\mathcal{T}_b} = |\mathcal{T}_b|^{-1} \sum_{i \in \mathcal{T}_b} D^{-1} \|\mathbf{x}_i - \widehat{\mathbf{x}}_i(\varphi)\|_2^2$, with $\widehat{\mathbf{x}}_i(\varphi)$ generated by Eq. 7. The regularizer discourages large departures from the episode anchor, and a finite update budget limits adaptation within each episode.

Validation-controlled acceptance. To assess whether the correction generalizes, we evaluate each candidate on $\mathcal{V}_b$ using the threshold-capped empirical risk $\widehat{R}_{\mathcal{V}_b}(\varphi) = |\mathcal{V}_b|^{-1} \sum_{i \in \mathcal{V}_b} \min\{s_i(\varphi; b), u\}$, where $s_i(\varphi; b)$ is defined in Eq. 14 and $u > 0$ is the anomaly decision threshold determined before comparison. Observations whose scores remain above u contribute no validation gain, even when their residuals decrease substantially. The candidate and the currently accepted model share the same scoring statistics and threshold, which remain fixed throughout comparison.

Using this risk, we accept a candidate only if it improves upon the currently accepted parameters φ_b^- by at least $\gamma > 0$:

$$\varphi_b^+ = \begin{cases} \varphi_b^\star, & \widehat{R}_{\mathcal{V}_b}(\varphi_b^-) - \widehat{R}_{\mathcal{V}_b}(\varphi_b^\star) \geq \gamma, \\ \varphi_b^-, & \text{otherwise.} \end{cases} \tag{11}$$

Without a triggered update, we set $\varphi_b^+ = \varphi_b^-$. Theorem 1 characterizes normal capped-risk improvement and bounds the possible loss of anomaly recall under validated adaptation.

4.4 Training and Anomaly Detection

Staged training. We first jointly train the encoder, dynamical projection, Koopman generator, and auxiliary decoder D_ω by minimizing

$$\mathcal{L}_{\mathrm{dyn}} = \mathbb{E}_{i,j} \big[\|\mathbf{z}_j - \exp(K(j-i))\mathbf{z}_i\|_2^2 + \lambda_{\mathrm{rec}} \big(\|D_\omega(\mathbf{z}_i) - \mathbf{x}_i\|_2^2 + \|D_\omega(\mathbf{z}_j) - \mathbf{x}_j\|_2^2 \big) \big] + \lambda_K \|K\|_F^2. \tag{12}$$

The reconstruction term encourages the latent states to retain observation information. Next, we freeze these components and learn the probabilistic path from historical triplets $i < k < j$:

$$\mathcal{L}_{\mathrm{path}} = -\mathbb{E}_{i,k,j} \left[\log p_f(\mathbf{x}_k) + \log p_b(\mathbf{x}_k) + \lambda_{\mathrm{path}} \log q_\tau(\mathbf{x}_k \mid i, j) \right], \quad \tau = \frac{k-i}{j-i}. \tag{13}$$

Finally, we freeze the path parameters and train v_ψ using Eq. 6 across multiple prediction horizons. Deployment uses single-step prediction to construct the normal reference.

Anomaly detection after adaptation. Given this reference, we define the prediction residual and anomaly score as

$$\mathbf{r}_i(\varphi) = \mathbf{x}_i - \widehat{\mathbf{x}}_i(\varphi), \qquad s_i(\varphi; b) = \frac{1}{D} \left\| S_b^{-1} \big(\mathbf{r}_i(\varphi) - \mathbf{c}_b \big) \right\|_2^2, \tag{14}$$

where $\mathbf{c}_b$ and S_b are the residual center and positive diagonal scale matrix. Their initial values and the anomaly threshold u are calibrated on historical residuals. The center and scale remain fixed during validation and rescoring. The threshold retains the same calibrated value throughout deployment.

Following Eq. 11, we rescore all samples in $\mathcal{A}_b$ using the selected parameters φ_b^+, giving $s_i^{\text{final}} = s_i(\varphi_b^+; b)$ and $\widehat{y}_i = \mathbf{1}\{s_i^{\text{final}} > u\}$. These scores replace the provisional buffer scores; previously finalized scores remain unchanged. Without a triggered update, detection uses the currently accepted model. Implementation details and the complete procedure are provided in Appendix A.

5 Theoretical Properties

In this section, we analyze KP-MMD estimation, shift distinguishability, and validated adaptation.

Proposition 1 (Evolution-compensated discrepancy)**.** *Under Assumption 2, define* $d_{K,b} := \mathrm{MMD}_k\big(P^{\mathrm{N}}_{z,c}, T_\Delta(P^{\mathrm{N}}_{z,r})\big)$. *With probability at least* $1-\delta_{\mathrm{KP}}$,

$$|D_b - d_{K,b}| \le 2\kappa(\epsilon_r + \epsilon_c) + \zeta_b, \tag{15}$$

where ϵ_r, ϵ_c *are their respective window contamination fractions,* $k(z,z) \le \kappa^2$, *and* ζ_b *bounds the combined empirical kernel-embedding error. Thus, KP-MMD estimates the normal latent discrepancy remaining after the learned propagation, up to contamination and estimation errors.*

Building on Proposition 1, Corollary 1 additionally accounts for propagation mismatch to establish sufficient conditions for distinguishing normal evolution from exogenous shifts.

Corollary 1 (Distinguishability after evolution compensation)**.** *Under Assumption 2, let* $Q^{\mathrm{N}}_{z,c|r}$ *denote normal continuation of the reference distribution, and define* $a_b := \mathrm{MMD}_k(T_\Delta(P^{\mathrm{N}}_{z,r}), Q^{\mathrm{N}}_{z,c|r})$ *and* $d_{\mathrm{exo},b} := \mathrm{MMD}_k(P^{\mathrm{N}}_{z,c}, Q^{\mathrm{N}}_{z,c|r})$. *If* $E_b := a_b + 2\kappa(\epsilon_r+\epsilon_c) + \zeta_b \le \overline{E}$ *and* $\Delta_{\mathrm{exo}} > 2\overline{E}$, *any* $\overline{E} < h < \Delta_{\mathrm{exo}} - \overline{E}$ *separates normal continuation (*$d_{\mathrm{exo},b} = 0$, $D_b < h$*) from exogenous shifts with* $d_{\mathrm{exo},b} \ge \Delta_{\mathrm{exo}}$ *(*$D_b > h$*), simultaneously with probability at least* $1-\delta_{\mathrm{KP}}$.

This result concerns raw KP-MMD: distinguishability requires the exogenous discrepancy to exceed the combined propagation, contamination, and estimation errors. For triggered adaptation, we next quantify normal-risk improvement and the possible loss of anomaly recall.

Theorem 1 (Validated adaptation with recall-loss control)**.** *Under Assumptions 3 and 4, let* R_b^{N} *denote normal capped risk,* $q_b^{\pm}$ *denote anomaly recall before and after adaptation, and* $\omega_b(t)$ *denote the initial anomaly mass within distance* t *above the threshold in root-score space, as defined in Appendix B.2. Set* $c_b = \min\{u, 2\sqrt{u}\, d_b\}$. *With probability at least* $1-\delta_{\mathrm{V}}$, *every accepted update satisfies, for all* $t > 0$,

$$R_b^{\mathrm{N}}(\varphi_b^-) - R_b^{\mathrm{N}}(\varphi_b^+) \ge \frac{\gamma - \eta_b - \epsilon_b c_b}{1-\epsilon_b}, \qquad q_b^+ \ge q_b^- - \omega_b(t) - \frac{d_b^2}{t^2}. \tag{16}$$

When $\gamma > \eta_b + \epsilon_b c_b$, accepted adaptation improves capped risk on shifted normal data. The recall bound quantifies the extent to which this adaptation preserves anomaly detectability: losses are controlled by anomalies near the decision boundary and the magnitude of prediction changes.

Additional properties of Koopman propagation, bidirectional Gaussian fusion, and regularized correction are established in Propositions 2, 3, and 4, respectively. All assumptions, statements, and proofs are collected in Appendix B.

6 Experiments

We evaluate KRAFT on public TSAD and domain-drift datasets, with ablation and sensitivity analyses. Appendix D.5 provides a NASA battery representation study. Appendix D.6 presents an industrial pump case study with aging-related endogenous and temperature-induced exogenous shifts.

6.1 Experimental Setup

Datasets and Evaluation. We evaluate KRAFT on 16 public TSAD datasets, covering 11 univariate datasets (IOPS, MGAB, NAB, NEK, Power, SED, Stock, TODS, UCR, WSD, YAHOO) and five multivariate datasets (MSL, PSM, SMAP, SMD, SWaT). For latent domain shifts, we additionally use INSECTS and three synthetic multi-domain datasets, BDS, MDS, and CDS (Appendices C.1 and C.2). Performance is reported using Affiliation-F, F1-T, Standard-F1, and VUS-PR, covering event-level, point-wise, and range-aware evaluation perspectives (Appendix C.4).

Table 1: Anomaly detection performance on 16 public TSAD datasets (%). Red and blue indicate the best and second-best reported scores for each dataset and metric. The last two columns report each model's average score for that metric and its overall rank among all models.

		Univariate datasets											Multivariate datasets					Summary	
Metric	Model	IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO	MSL	PSM	SMAP	SMD	SWaT	Avg	Rank
Aff-F	KRAFT	92.34	70.02	88.74	97.76	70.42	69.88	94.75	85.53	77.38	93.37	94.94	89.38	81.72	89.62	92.69	77.01	85.35	1
	TranAD	83.19	67.28	90.28	85.02	71.56	61.03	57.94	52.76	73.31	84.34	76.08	79.91	73.83	87.39	92.20	75.37	75.72	3
	USAD	71.08	67.81	91.54	71.13	76.48	55.60	35.92	47.90	76.00	65.10	53.05	81.86	57.86	87.25	85.09	75.06	68.67	9
	OmniAnomaly	80.32	67.35	92.35	86.30	78.16	61.26	75.24	50.73	73.53	78.02	71.31	83.15	58.17	91.38	85.82	73.39	75.41	4
	LOF	81.06	68.44	75.75	84.74	66.76	63.85	69.74	60.58	73.53	81.29	75.63	84.35	61.98	63.32	64.13	56.34	70.72	6
	IForest	52.81	68.82	39.84	71.15	0.00	70.09	0.06	44.17	50.56	41.24	33.30	63.36	63.78	59.96	69.71	0.00	45.55	12
	Sub-PCA	75.39	66.90	89.29	97.10	71.37	67.14	70.63	72.75	76.66	76.45	75.85	84.25	71.49	90.08	85.80	76.29	77.97	2
	DCdetector	71.83	67.91	72.21	62.31	69.75	72.20	55.79	57.81	70.18	72.79	67.77	67.74	67.32	67.10	69.55	71.07	67.71	10
	D3R	76.50	65.87	70.80	71.52	73.36	69.44	68.18	48.72	71.17	74.98	62.18	69.98	71.07	75.86	73.01	75.32	69.87	8
	LSTMAD	74.65	69.09	74.97	80.79	51.29	62.28	87.82	68.87	68.13	74.59	84.51	78.63	62.08	72.55	66.36	44.72	70.08	7
	M2N2	75.72	69.19	75.39	84.89	72.59	62.23	83.28	48.54	67.63	71.69	61.82	75.22	67.35	75.30	84.38	78.72	72.12	5
	DIVAD	65.86	68.37	72.93	63.24	63.47	72.46	76.54	54.45	67.80	67.39	66.58	66.74	65.51	67.97	68.24	44.63	65.76	11
F1-T	KRAFT	51.67	3.73	38.99	56.29	21.26	11.31	80.16	47.54	13.78	43.70	64.37	48.64	36.46	33.42	46.53	35.27	39.57	1
	TranAD	22.63	1.65	37.28	69.97	22.36	9.57	16.73	13.51	7.75	20.94	8.41	39.42	25.49	29.12	37.98	49.58	25.77	8
	USAD	20.99	4.07	61.46	70.64	28.23	9.54	16.86	20.85	14.63	14.18	9.35	48.71	28.96	43.94	50.41	50.41	30.83	4
	OmniAnomaly	51.17	1.61	40.09	82.20	23.48	9.68	36.22	14.33	8.47	34.79	24.16	49.36	30.42	46.63	51.84	46.64	34.44	2
	LOF	27.97	1.15	35.76	63.57	19.80	9.60	66.14	31.63	8.31	24.38	55.93	38.97	25.58	21.81	10.13	30.62	29.46	6
	IForest	7.64	0.84	21.44	65.56	0.00	9.54	1.10	11.06	6.36	4.28	4.90	20.73	25.39	14.32	16.20	0.00	13.08	12
	Sub-PCA	32.75	0.98	54.15	84.18	20.30	9.54	20.10	18.40	18.90	24.62	11.57	49.02	30.38	44.31	51.87	46.65	32.36	3
	DCdetector	6.61	1.32	32.72	29.21	21.13	10.53	16.07	16.40	6.62	7.32	6.81	23.24	25.34	15.73	9.47	28.64	16.07	11
	D3R	14.22	2.07	35.95	53.86	24.03	9.79	16.45	16.49	6.70	12.48	10.23	28.25	26.53	22.96	20.33	33.14	20.84	9
	LSTMAD	29.03	0.41	39.51	67.07	19.72	9.45	66.33	32.10	8.19	22.44	54.84	35.19	27.73	25.94	12.99	26.73	29.85	5
	M2N2	19.61	3.70	35.00	69.02	23.95	7.25	14.10	16.23	5.94	18.67	11.31	40.63	30.08	34.01	42.65	45.12	26.08	7
	DIVAD	19.41	1.07	33.04	31.11	20.18	11.82	22.15	16.48	5.90	19.11	23.85	25.28	25.36	19.39	10.13	28.46	19.55	10
F1	KRAFT	40.95	2.99	30.91	61.93	21.12	10.79	80.55	38.57	13.10	43.07	62.96	33.45	27.25	27.69	46.31	36.04	36.10	1
	TranAD	34.85	1.46	27.33	60.36	22.36	2.63	16.23	11.94	4.40	20.23	5.70	29.60	25.63	25.11	43.99	61.86	24.61	7
	USAD	30.66	3.89	56.15	62.91	28.24	3.41	17.99	23.87	10.74	13.20	7.21	38.71	28.41	38.66	53.06	62.82	30.00	4
	OmniAnomaly	47.05	1.44	28.81	74.03	23.50	0.43	38.59	12.65	5.11	29.57	21.40	39.10	30.43	40.50	57.06	55.93	31.60	2
	LOF	30.28	1.05	24.04	56.92	12.18	4.11	66.20	25.77	4.70	22.62	48.95	30.65	18.80	18.70	8.41	29.08	25.15	6
	IForest	8.37	0.73	29.41	58.10	19.77	3.81	16.91	13.35	4.09	2.07	3.20	14.68	24.15	13.61	16.89	26.76	15.99	10
	Sub-PCA	33.96	0.83	46.71	85.43	16.05	9.56	21.77	18.72	15.12	24.74	11.06	38.29	30.26	38.90	57.22	55.98	31.54	3
	DCdetector	5.19	1.21	24.02	17.37	21.10	10.54	16.97	17.85	3.18	4.64	4.14	14.08	25.33	10.67	8.99	27.02	13.27	12
	D3R	16.26	1.89	23.47	39.80	20.49	8.16	15.26	13.42	3.07	10.66	4.18	17.28	24.43	18.12	23.02	47.25	17.92	9
	LSTMAD	28.76	1.02	22.43	52.39	15.43	5.21	64.11	29.34	4.55	21.78	45.49	35.20	21.94	20.20	6.01	29.01	25.18	5
	M2N2	36.97	3.17	22.53	58.77	22.23	6.95	14.35	9.11	3.35	19.99	4.29	25.28	25.94	21.09	41.97	60.82	23.55	8
	DIVAD	9.48	0.97	23.78	19.74	20.14	11.87	23.32	14.88	2.83	15.53	20.50	15.68	25.39	12.58	9.16	27.08	15.81	11
VUS-PR	KRAFT	24.39	2.05	25.84	65.13	10.78	8.44	99.24	65.82	9.08	24.49	70.20	22.98	20.32	21.15	38.78	23.19	33.24	1
	TranAD	21.61	0.64	24.82	61.63	13.04	5.75	78.08	47.33	2.25	12.20	25.78	14.78	16.49	13.37	28.34	47.37	25.84	6
	USAD	16.58	0.75	55.03	58.53	18.68	4.37	74.53	56.36	8.85	10.00	14.15	29.95	17.59	26.37	34.53	44.73	29.44	4
	OmniAnomaly	25.35	0.64	27.17	74.51	14.32	6.20	91.29	45.55	2.40	16.37	29.26	31.57	18.58	28.07	37.44	42.97	30.73	3
	LOF	19.43	0.57	21.18	58.52	9.31	6.81	83.07	49.14	2.39	12.85	41.37	24.67	13.58	10.59	4.40	14.50	23.27	8
	IForest	8.59	0.62	23.57	56.50	11.56	7.71	70.99	46.62	2.88	2.06	10.47	11.29	15.85	7.55	8.88	15.49	18.79	10
	Sub-PCA	23.02	0.60	46.08	88.91	10.49	3.72	80.86	54.16	12.92	16.41	21.57	31.43	18.52	26.42	37.50	43.02	32.23	2
	DCdetector	5.83	0.59	16.60	14.03	12.32	9.37	74.16	46.66	1.53	3.23	10.17	7.01	14.49	4.21	4.66	15.04	14.99	12
	D3R	15.50	0.57	24.95	34.59	14.16	8.25	80.67	48.82	1.38	8.83	21.44	10.46	16.76	7.77	21.92	28.05	21.51	9
	LSTMAD	16.49	3.53	17.32	61.41	10.57	2.47	84.82	52.95	6.68	15.05	45.80	20.43	14.26	5.82	3.02	11.80	23.28	7
	M2N2	20.65	0.51	29.71	62.38	14.47	6.54	80.28	45.54	1.22	11.62	30.23	14.01	19.25	9.43	30.16	46.01	26.38	5
	DIVAD	5.32	0.64	15.68	17.81	11.90	9.55	73.54	48.79	2.57	5.36	25.93	8.25	14.22	5.76	4.77	15.38	16.59	11

Baselines. We consider 11 representative baselines: classical statistical algorithms LOF (Breunig et al., 2000), IForest (Liu et al., 2008), and Sub-PCA (Liu & Paparrizos, 2024); deep learning methods TranAD (Tuli et al., 2022), USAD (Audibert et al., 2020), OmniAnomaly (Su et al., 2019), DCdetector (Yang et al., 2023), D3R (Wang et al., 2023), LSTMAD (Malhotra et al., 2015), M2N2 (Kim et al., 2024), and DIVAD (Jacob & Diao, 2025). Notably, D3R, M2N2, and DIVAD are specifically designed for anomaly detection in multi-domain time series. Baseline-specific protocols and result sources are documented in Appendix C.3.

Implementation Details. KRAFT uses a temporal convolutional network (TCN) (Bai et al., 2018) encoder with a history window of 64 and a hidden width of 64, whose output is projected into a 16-dimensional Koopman latent space. The flow network contains four residual blocks of width 128, with rank-4 adapters. Full configurations are provided in Appendix C.5.

6.2 Anomaly Detection Performance

As shown in Table 1, KRAFT achieves the strongest overall anomaly detection performance across the 16 public datasets, ranking first by average score on all four metrics. It attains the best Affiliation-F on 11 datasets and the best F1-T, Standard-F1, and VUS-PR on 6, 4, and 6 datasets, respectively, demonstrating consistently superior performance of anomaly detection.

6.3 Detection under Latent Domain Shifts

As shown in Figure 2, KRAFT achieves the best performance in 15 of the 16 dataset–metric combinations, outperforming M2N2, DIVAD, and D3R on all four datasets in Affiliation-F, Standard-F1, and VUS-PR, supporting its effectiveness in detecting anomalies under latent domain shifts. The slightly lower F1-T on MDS may reflect a trade-off between suppressing false alarms and covering entire anomaly intervals, as conservative reference updates can leave weaker portions of anomalous

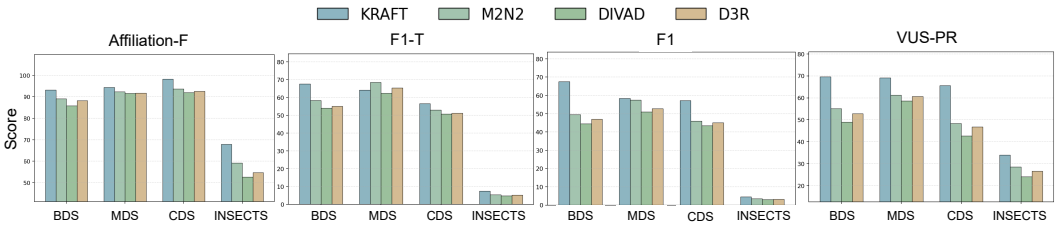


Figure 2: Anomaly detection performance on three constructed multi-domain datasets and the real-world drift dataset INSECTS.

Table 2: Average ablation results (%). Public columns average the 16 public TSAD datasets; Domain Drift columns average BDS, MDS, CDS, and INSECTS.

	Public TSAD Datasets				Domain Drift Datasets			
Variant	Aff-F	F1-T	F1	VUS-PR	Aff-F	F1-T	F1	VUS-PR
KRAFT	**85.35**	**39.57**	**36.10**	**33.24**	**88.41**	**49.01**	**46.91**	**59.58**
w/o Koopman conditioning	$74.12_{\downarrow 11.23}$	$33.77_{\downarrow 5.80}$	$28.88_{\downarrow 7.22}$	$29.95_{\downarrow 3.29}$	$79.45_{\downarrow 8.96}$	$36.45_{\downarrow 12.56}$	$37.48_{\downarrow 9.43}$	$42.34_{\downarrow 17.24}$
w/o evolution compensation	$74.16_{\downarrow 11.19}$	$34.71_{\downarrow 4.86}$	$31.73_{\downarrow 4.37}$	$29.02_{\downarrow 4.22}$	$75.09_{\downarrow 13.32}$	$37.45_{\downarrow 11.56}$	$35.19_{\downarrow 11.72}$	$47.38_{\downarrow 12.20}$
w/o adaptation	$71.19_{\downarrow 14.16}$	$30.36_{\downarrow 9.21}$	$27.21_{\downarrow 8.89}$	$28.00_{\downarrow 5.24}$	$50.05_{\downarrow 38.36}$	$33.52_{\downarrow 15.49}$	$34.86_{\downarrow 12.05}$	$37.70_{\downarrow 21.88}$
w/o validation	$77.25_{\downarrow 8.10}$	$35.38_{\downarrow 4.19}$	$34.23_{\downarrow 1.87}$	$29.98_{\downarrow 3.26}$	$74.75_{\downarrow 13.66}$	$39.39_{\downarrow 9.62}$	$38.14_{\downarrow 8.77}$	$43.38_{\downarrow 16.20}$

events undetected. Per-dataset results, statistical tests, and mechanism-level diagnostics are reported in Appendix D.2.

6.4 Ablation and Sensitivity Analysis

Ablation study. We evaluate four variants by replacing Koopman-predicted conditioning with the current latent state, removing reference evolution compensation in KP-MMD, freezing the adapter, and accepting candidate updates without validation, respectively. Table 2 shows that all components improve detection, with removing adaptation causing the largest overall decline. Larger drops on domain-drift datasets highlight their importance under latent domain shifts (Appendix D.3).

Parameter sensitivity. Figure 3 examines the reference window length n_r, trigger persistence p, and validation margin γ. Across all parameters, the performance follows a consistent inverted-U trend: overly aggressive adaptation introduces unstable updates, while overly conservative settings delay necessary adaptation. The default configuration ($n_r = 100$, $p = 3$, and $\gamma = 0.02$) achieves the best average performance, demonstrating a favorable balance between adaptation responsiveness and anomaly separability preservation. Detailed results are provided in Appendix D.4.

7 Conclusion

We proposed KRAFT for time-series anomaly detection under latent domain shifts, combining Koopman-conditioned flow matching, evolution-compensated shift detection, and validation-guided adaptation to update normal references while limiting anomaly contamination. Theoretical analysis shows that compensation separates exogenous shift from endogenous evolution, providing conditional guarantees supporting anomaly detectability. Extensive experiments on public TSAD and domain-drift benchmarks demonstrate KRAFT's effectiveness, with ablations further highlighting the importance of adaptation for maintaining detection performance under domain shifts.

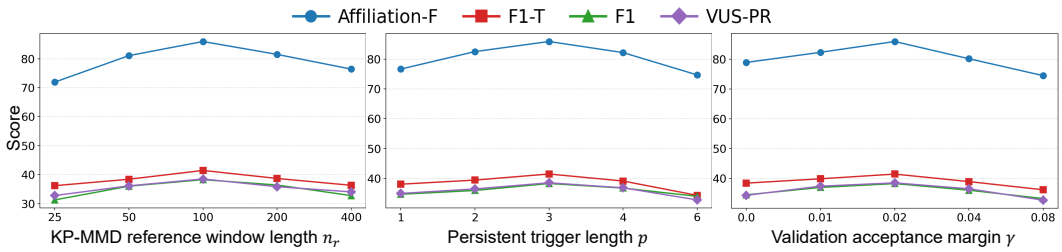


Figure 3: Sensitivity of KRAFT to reference length, persistence length, and validation margin.

Reproducibility Statement

Appendix A describes reference construction, shift detection, restricted adaptation, and anomaly scoring, with Algorithms 1–4 summarizing the computational procedures. The assumptions and proofs supporting Section 5 are provided in Appendix B. Dataset descriptions and preprocessing protocols appear in Appendix C.1, and the construction of BDS, MDS, and CDS is detailed in Appendix C.2. Baseline protocols and evaluation metrics are described in Appendices C.3 and C.4, respectively. Table 6 reports the shared hyperparameter configuration, and Appendix D.1 describes the statistical comparison procedure.

AI Use Statement

In this work, we used generative AI tools for three purposes. First, language editing: improving readability, tightening prose, and formatting LaTeX tables and references. Second, literature discovery: searching for and summarizing candidate related work, which the authors then read in the original before citing. Third, verification of mathematics: the definitions, claims, and proof strategies of Section 5 and Appendix B were derived by the authors, and generative AI was used as a checker to search the derivations for errors; every issue it raised was re-derived by hand, and no claim or proof in this paper originates from an AI tool. We take responsibility for the final content of this work, including text and claims produced with the aid of generative AI.

Ethics Statement

This work studies anomaly detection using existing public benchmark datasets, synthetically constructed time series and industrial equipment test records; it does not involve recruiting human participants or collecting new personal data. Potential applications include monitoring systems in which false alarms and missed anomalies may have practical consequences. Deployment in safety-critical settings therefore requires application-specific validation and appropriate human oversight. The authors have read and adhere to the ICLR Code of Ethics.

References

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*, 2020. doi: 10.1145/3394486.3403392. URL https://doi.org/10.1145/3394486.3403392.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104, 2000.

Ruichu Cai, Jiawei Chen, Zijian Li, Wei Chen, Keli Zhang, Junjian Ye, Zhuozhang Li, Xiaoyan Yang, and Zhenjie Zhang. Time series domain adaptation via sparse associative structure alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 6859–6867, 2021. doi: 10.1609/aaai.v35i8.16846.

Zahra Zamanzadeh Darban, Yiyuan Yang, Geoffrey I. Webb, Charu C. Aggarwal, Qingsong Wen, Shirui Pan, and Mahsa Salehi. DACAD: Domain adaptation contrastive learning for anomaly detection in multivariate time series. *IEEE Transactions on Knowledge and Data Engineering*, 37(8):4485–4496, 2025. doi: 10.1109/TKDE.2025.3569909.

Ailin Deng and Bryan Hooi. Graph neural network-based anomaly detection in multivariate time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 4027–4035, 2021. doi: 10.1609/aaai.v35i5.16523. URL https://ojs.aaai.org/index.php/AAAI/article/view/16523.

Chaoyue Ding, Jing Zhao, and Shiliang Sun. Concept drift adaptation for time series anomaly detection via transformer. *Neural Processing Letters*, 55(3):2081–2101, 2023. doi: 10.1007/s11063-022-11015-0.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. URL https://jmlr.org/papers/v13/gretton12a.html.

Huan He, Owen Queen, Teddy Koker, Consuelo Cuevas, Theodoros Tsiligkaridis, and Marinka Zitnik. Domain adaptation for time series under feature and label shifts. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 12746–12774. PMLR, 2023. URL https://proceedings.mlr.press/v202/he23b.html.

Alexis Huet, Jose Manuel Navarro, and Dario Rossi. Local evaluation of time series anomaly detection algorithms. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 635–645, 2022.

Vincent Jacob and Yanlei Diao. Unsupervised anomaly detection in multivariate time series across heterogeneous domains. *Proceedings of the VLDB Endowment*, 18(6):1691–1704, 2025. doi: 10.14778/3725688.3725699.

Dongmin Kim, Sunghyun Park, and Jaegul Choo. When model meets new normals: Test-time adaptation for unsupervised time-series anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 13113–13121, 2024. doi: 10.1609/aaai.v38i12.29210.

HyunGi Kim, Jisoo Mok, Hyungyu Lee, Juhyeon Shin, and Sungroh Yoon. CANDI: Curated test-time adaptation for multivariate time-series anomaly detection under distribution shift. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 15018–15026, 2026. doi: 10.1609/aaai.v40i17.38524.

Tian Lan, Hao Duong Le, Jinbo Li, Wenjun He, Meng Wang, Chenghao Liu, and Chen Zhang. Towards foundation models for zero-shot time series anomaly detection: Leveraging synthetic data and relative context discrepancy. *arXiv preprint arXiv:2509.21190*, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 eighth ieee international conference on data mining*, pp. 413–422. Ieee, 2008.

Qiao Liu and Hui Xue. Adversarial spectral kernel matching for unsupervised time series domain adaptation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pp. 2744–2750, 2021. doi: 10.24963/ijcai.2021/378.

Qinghua Liu and John Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. *Advances in Neural Information Processing Systems*, 37:108231–108261, 2024.

Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, and Puneet Agarwal. Long short term memory networks for anomaly detection in time series. In *Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 89–94, 2015. URL https://www.esann.org/sites/default/files/proceedings/legacy/es2015-56.pdf.

Amirhossein Moallemi, Alessio Burrello, Davide Brunelli, and Luca Benini. Exploring scalable, distributed real-time anomaly detection for bridge health monitoring. *IEEE Internet of Things Journal*, 2022. doi: 10.1109/JIOT.2022.3157532. URL https://doi.org/10.1109/JIOT.2022.3157532.

John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S Tsay, Aaron 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.

Bhaskar Saha and Kai Goebel. Battery data set, 2007. NASA Prognostics Data Repository, NASA Ames Research Center, Moffett Field, CA.

M Saquib Sarfraz, Mei-Yen Chen, Lukas Layer, Kunyu Peng, and Marios Koulakis. Position: quo vadis, unsupervised time series anomaly detection? *arXiv preprint arXiv:2405.02678*, 2024.

Lifeng Shen, Zhuocong Li, and James Kwok. Timeseries anomaly detection using temporal hierarchical one-class network. *Advances in neural information processing systems*, 33:13016–13026, 2020.

Vinicius MA Souza, Denis M Dos Reis, Andre G Maletzke, and Gustavo EAPA Batista. Challenges in benchmarking stream learning algorithms with real-world data. *Data Mining and Knowledge Discovery*, 34(6):1805–1858, 2020.

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 & data mining*, pp. 2828–2837, 2019.

Shreshth Tuli, Giuliano Casale, and Nicholas R Jennings. Tranad: Deep transformer networks for anomaly detection in multivariate time series data. *arXiv preprint arXiv:2201.07284*, 2022.

Chengsen Wang, Zirui Zhuang, Qi Qi, Jingyu Wang, Xingyu Wang, Haifeng Sun, and Jianxin Liao. Drift doesn't matter: Dynamic decomposition with diffusion reconstruction for unstable multivariate time series anomaly detection. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/22f5d8e689d2a011cd8ead552ed59052-Abstract-Conference.html.

Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly Transformer: Time series anomaly detection with association discrepancy. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=LzQQ89U1qm_.

Yiyuan Yang, Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. DCdetector: Dual attention contrastive representation learning for time series anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, 2023. doi: 10.1145/3580305.3599295. URL https://doi.org/10.1145/3580305.3599295.

Susik Yoon, Youngjun Lee, Jae-Gil Lee, and Byung Suk Lee. Adaptive model pooling for online deep anomaly detection from a complex evolving data stream. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, 2022. doi: 10.1145/3534678.3539348. URL https://doi.org/10.1145/3534678.3539348.

Bin Zhou, Shenghua Liu, Bryan Hooi, Xueqi Cheng, and Jing Ye. BeatGAN: Anomalous rhythm detection using adversarially generated time series. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 4433–4439, 2019. doi: 10.24963/ijcai.2019/616. URL https://www.ijcai.org/proceedings/2019/616.

A Additional Methodological Details

This section describes the architecture, temporal alignment, reference generation, and adaptation procedures of KRAFT. We distinguish the observation index t, prediction horizon H, propagation lag Δ, and normalized flow time $\tau \in [0, 1]$. Algorithms 1–4 describe the component procedures and their integration. The following subsections describe each component in detail.

A.1 Dynamics-Conditioned Normal Reference Construction

Input preprocessing. Observations are standardized channelwise using statistics estimated exclusively from the historical fitting subset:

$$\widetilde{x}_{t,d} = \frac{x_{t,d} - \mu_d^{\mathrm{fit}}}{\max\{\sigma_d^{\mathrm{fit}}, \epsilon_{\mathrm{norm}}\}}. \tag{17}$$

These statistics remain fixed during deployment. All model inputs, prediction targets, and residuals below are expressed in this standardized observation space; we omit the tilde for brevity. Input standardization is distinct from residual calibration for anomaly scoring.

TCN encoder. The encoder receives the historical window $\mathcal{H}_i = (\mathbf{x}_{i-W+1}, \ldots, \mathbf{x}_i)$. An input projection maps each observation into a hidden representation, which is processed by temporal residual blocks. Each block contains two causal one-dimensional convolutions with a shared dilation factor. Dilation increases across blocks to capture temporal dependencies at multiple scales.

Each convolution is followed by layer normalization over channels and a GELU activation. Normalization is applied independently at each temporal position. Dropout is applied within residual branches during historical training and disabled during deployment. The terminal temporal activation forms the context, and a two-layer MLP with GELU projects it into the Koopman latent space:

$$\mathbf{h}_i = E_\theta(\mathcal{H}_i), \qquad \mathbf{z}_i = P_\eta(\mathbf{h}_i). \tag{18}$$

No additional positional embedding is used in the encoder. Flow time and prediction horizon are encoded separately in the conditional generator.

Stable Koopman propagation. Following Eq. 3, the continuous-time generator is parameterized as

$$K = (A - A^\top) - \mathrm{diag}\big(\mathrm{softplus}(\mathbf{b}_K)\big), \qquad T_\Delta(\mathbf{z}) = \exp(K\Delta)\mathbf{z}, \tag{19}$$

with trainable $A \in \mathbb{R}^{m\times m}$ and $\mathbf{b}_K \in \mathbb{R}^m$. Its symmetric part is negative definite, ensuring contractive forward propagation in the Euclidean norm. This structural constraint stabilizes propagation without assuming that the learned dynamics explain every shift.

One observation interval is treated as one time unit. For historical endpoints $i < j$, let $H = j - i$. At normalized flow time τ,

$$\mathbf{z}_{i,\tau} = \exp(K\tau H)\mathbf{z}_i, \qquad \partial_\tau \mathbf{z}_{i,\tau} = HK\mathbf{z}_{i,\tau}. \tag{20}$$

Thus, $K\mathbf{z}_{i,\tau}$ is the evolution rate in observation time. Automatic differentiation through $\exp(K\tau H)$ already introduces H, so the resulting teacher velocity is not multiplied by the horizon again.

Historical dynamics learning. The encoder, projection, Koopman generator, and auxiliary decoder are trained jointly by the first stage objective $\mathcal{L}_{\mathrm{dyn}}$ of Eq. 12, written here with the propagation map of Eq. 19:

$$\begin{aligned}\mathcal{L}_{\mathrm{dyn}} = \mathbb{E}_{i,j}\left[\|\mathbf{z}_j - T_H(\mathbf{z}_i)\|_2^2\right] \\ + \lambda_{\mathrm{rec}}\left[\mathrm{MSE}(D_\omega(\mathbf{z}_i), \mathbf{x}_i) + \mathrm{MSE}(D_\omega(\mathbf{z}_j), \mathbf{x}_j)\right] + \lambda_K \|K\|_F^2.\end{aligned} \tag{21}$$

The latent prediction term sums over latent coordinates and averages over samples, whereas reconstruction MSE averages over samples and observation channels; the channel normalization is absorbed into λ_{rec} relative to Eq. 12. The decoder consists of two linear layers separated by a Tanh activation and reconstructs the terminal observation, rather than the entire history window. It encourages informative latent states and is used only in this training stage.

Shared bidirectional predictor. After dynamics learning, the encoder, projection, generator, and decoder are frozen. The forward and backward Gaussian experts share an MLP trunk and are distinguished by a learned direction embedding. The trunk receives the corresponding endpoint context and observation, the propagated state–rate pair $(\mathbf{z}_{i,\tau}, K\mathbf{z}_{i,\tau})$, a direction-dependent time feature, and the direction embedding. The time feature is τ for the forward expert and $1 - \tau$ for the backward expert. Separate output heads predict an endpoint displacement and a variance parameter.

The expert means and diagonal variances are

$$\begin{aligned} \boldsymbol{\mu}_f &= \mathbf{x}_i + \tau\boldsymbol{\delta}_f, & \boldsymbol{\nu}_f &= \epsilon_v + \tau^2 \operatorname{softplus}(\boldsymbol{\rho}_f), \\ \boldsymbol{\mu}_b &= \mathbf{x}_j + (1-\tau)\boldsymbol{\delta}_b, & \boldsymbol{\nu}_b &= \epsilon_v + (1-\tau)^2 \operatorname{softplus}(\boldsymbol{\rho}_b). \end{aligned} \tag{22}$$

The displacements and variance parameters depend on the expert inputs, with their arguments omitted for brevity. Variances are clipped to prescribed numerical bounds before fusion.

Weighted Gaussian bridge. The bridge combines the experts using normalized-time gates:

$$q_\tau(\mathbf{x}) \propto p_f(\mathbf{x})^{1-\tau} p_b(\mathbf{x})^{\tau}. \tag{23}$$

For diagonal Gaussians, its parameters are computed componentwise:

$$\begin{aligned} \boldsymbol{\nu}_\tau &= \left[(1-\tau)\boldsymbol{\nu}_f^{-1} + \tau\boldsymbol{\nu}_b^{-1}\right]^{-1}, \\ \boldsymbol{\mu}_\tau &= \boldsymbol{\nu}_\tau \odot \left[(1-\tau)\boldsymbol{\nu}_f^{-1} \odot \boldsymbol{\mu}_f + \tau\boldsymbol{\nu}_b^{-1} \odot \boldsymbol{\mu}_b\right]. \end{aligned} \tag{24}$$

The gates depend on τ, not τH. The endpoint means satisfy $\boldsymbol{\mu}_0 = \mathbf{x}_i$ and $\boldsymbol{\mu}_1 = \mathbf{x}_j$, with endpoint variance ϵ_v.

The shared predictor is trained on historical triplets $i < k < j$, with $\tau = (k-i)/(j-i)$, by the second stage objective $\mathcal{L}_{\text{path}}$ of Eq. 13:

$$\begin{aligned} \mathcal{L}_{\text{path}} = \mathbb{E}_{i,k,j}\big[&\ell_{\text{G}}(\mathbf{x}_k; \boldsymbol{\mu}_f, \boldsymbol{\nu}_f) + \ell_{\text{G}}(\mathbf{x}_k; \boldsymbol{\mu}_b, \boldsymbol{\nu}_b) \\ &+ \lambda_{\text{path}} \ell_{\text{G}}(\mathbf{x}_k; \boldsymbol{\mu}_\tau, \boldsymbol{\nu}_\tau)\big], \end{aligned} \tag{25}$$

where

$$\ell_{\text{G}}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\nu}) = \frac{1}{2D} \sum_{d=1}^{D} \left[\frac{(x_d - \mu_d)^2}{\nu_d} + \log \nu_d\right] \tag{26}$$

omits the constant Gaussian normalization term. Each ℓ_{G} therefore equals the corresponding negative log-density in Eq. 13 up to this additive constant and the fixed factor $1/D$, which leaves the minimizer and the relative weight λ_{path} unchanged. Only the shared predictor is optimized in this stage.

Conditional velocity network. The velocity network conditions on information from the starting history:

$$\mathbf{c}_{i,\tau,H} = (\mathbf{x}_i, \mathbf{h}_i, \mathbf{z}_i, \mathbf{z}_{i,\tau}, H). \tag{27}$$

Flow time uses a sinusoidal embedding with logarithmically spaced frequencies. The horizon is log-normalized and passed through a learnable MLP:

$$\mathbf{e}_H = \text{MLP}_H\left(\frac{\log(1+H)}{\log(1+H_{\max})}\right). \tag{28}$$

The path state, flow-time embedding, and conditioning features are concatenated and linearly projected into the hidden space.

The trunk consists of pre-normalized residual MLP blocks:

$$\mathbf{a}^{(\ell+1)} = \mathbf{a}^{(\ell)} + W_2^{(\ell)}\,\text{SiLU}\left(W_1^{(\ell)}\,\text{LN}(\mathbf{a}^{(\ell)}) + \mathbf{b}_1^{(\ell)}\right) + \mathbf{b}_2^{(\ell)}. \tag{29}$$

A final layer normalization and linear projection return an observation-space velocity. The future endpoint, backward-expert outputs, and fused bridge parameters are not inputs to the velocity network.

Flow-matching training. After freezing the bridge predictor, a path sample and its target velocity are constructed as

$$\begin{aligned} \mathbf{x}_\tau &= \boldsymbol{\mu}_\tau + \boldsymbol{\sigma}_\tau \odot \boldsymbol{\varepsilon}, \\ \mathbf{u}_\tau &= \partial_\tau \boldsymbol{\mu}_\tau + \partial_\tau \boldsymbol{\sigma}_\tau \odot \boldsymbol{\varepsilon}, \qquad \boldsymbol{\sigma}_\tau = \sqrt{\boldsymbol{\nu}_\tau}, \end{aligned} \tag{30}$$

Algorithm 1 Dynamics-Conditioned Reference Generation

Require: History $\mathcal{H}_i$, causal origin $\mathbf{o}_i$, horizon H, frozen backbone, adapter φ, integration budget N
Ensure: Normal reference $\widehat{\mathbf{x}}_{i+H}$
1: Encode $\mathcal{H}_i$ into $\mathbf{h}_i, \mathbf{z}_i$
2: $\boldsymbol{\xi} \leftarrow \mathbf{o}_i$; $h_\tau \leftarrow 1/N$
3: **for** $q = 0, \dots, N-1$ **do**
4: $\quad \tau_q \leftarrow q h_\tau$
5: $\quad$ Construct conditions at τ_q and $\tau_q + h_\tau$ using Koopman propagation
6: $\quad$ Evaluate $\mathbf{k}_1, \mathbf{k}_2$ and update $\boldsymbol{\xi}$ by Eq. 33
7: **end for**
8: **return** $\boldsymbol{\xi}$

where $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, I_D)$. The flow time of Eq. 6 is drawn uniformly from an interior subinterval $[\tau_{\min}, 1 - \tau_{\min}] \subset [0, 1]$, which realizes $\tau \sim \mathcal{U}(0, 1)$ outside a small neighborhood of the endpoints, where the bridge variance degenerates to ϵ_v and the teacher velocity becomes numerically ill-conditioned. The same noise realization is held fixed during differentiation, including differentiation through the propagated latent state.

The teacher velocity is detached, and only the student velocity network is optimized by the third stage objective $\mathcal{L}_{\mathrm{FM}}$ of Eq. 6:

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}\left[\left\| v_\psi(\mathbf{x}_\tau, \tau; \mathbf{c}_{i,\tau,H}) - \mathbf{u}_\tau \right\|_2^2\right]. \tag{31}$$

This stage uses flow matching alone, without an additional endpoint rollout or reconstruction loss. Historical validation determines the checkpoint retained at each training stage.

Numerical reference generation. For a time-series input, reference generation starts from $\boldsymbol{\xi}(0) = \mathbf{x}_i$ and solves

$$\frac{\mathrm{d}\boldsymbol{\xi}}{\mathrm{d}\tau} = v_{\psi,\varphi}(\boldsymbol{\xi}, \tau; \mathbf{c}_{i,\tau,H}), \qquad \tau \in [0, 1]. \tag{32}$$

The propagated latent state is recomputed at every velocity evaluation. With flow-time step $h_\tau = 1/N_{\mathrm{ODE}}$, Heun integration uses

$$\begin{aligned} \mathbf{k}_1 &= v_{\psi,\varphi}(\boldsymbol{\xi}_q, \tau_q; \mathbf{c}_{i,\tau_q,H}), \\ \mathbf{k}_2 &= v_{\psi,\varphi}(\boldsymbol{\xi}_q + h_\tau \mathbf{k}_1, \tau_q + h_\tau; \mathbf{c}_{i,\tau_q+h_\tau,H}), \\ \boldsymbol{\xi}_{q+1} &= \boldsymbol{\xi}_q + \frac{h_\tau}{2}(\mathbf{k}_1 + \mathbf{k}_2). \end{aligned} \tag{33}$$

The terminal state is the normal reference $\widehat{\mathbf{x}}_{i+H}$. Deployment uses one-step prediction and a single deterministic trajectory without initial noise or sample averaging.

For ordered vector streams, a causal robust center of the preceding window replaces the last observation as the flow origin and origin-conditioning value. The encoder and remaining reference-generation components are unchanged. The bidirectional experts and bridge are used only to construct training targets and are not evaluated at inference. The overall procedure for generating a dynamics-conditioned normal reference is summarized in Algorithm 1.

A.2 Evolution-Compensated Shift Detection

Aligned latent windows. At observed index t, the detector constructs adjacent reference and current windows:

$$\begin{aligned} \mathcal{Z}_{r,t} &= \{\mathbf{z}_{t-2n+1}, \dots, \mathbf{z}_{t-n}\}, \\ \mathcal{Z}_{c,t} &= \{\mathbf{z}_{t-n+1}, \dots, \mathbf{z}_t\}. \end{aligned} \tag{34}$$

These are the windows $\mathcal{Z}_r$ and $\mathcal{Z}_c$, each holding n latent states; n is the reference window length denoted n_r in the sensitivity analysis. Corresponding positions are separated by $\Delta = n$. The reference states are propagated as

$$\overline{\mathbf{z}}_{r,t,\ell} = \exp(Kn)\mathbf{z}_{t-2n+\ell}, \qquad \ell = 1, \dots, n. \tag{35}$$

Both windows use observations available through t and slide with the stream. They are not reset following adapter acceptance. Because the encoder and Koopman generator are frozen, adapter updates do not invalidate cached latent states.

Kernel estimator. KP-MMD uses a Gaussian RBF kernel:

$$k_{\sigma_k}(\mathbf{z}, \mathbf{z}') = \exp\left(-\frac{\|\mathbf{z} - \mathbf{z}'\|_2^2}{2\sigma_k^2}\right). \tag{36}$$

Let $\{\mathbf{a}_\ell\}_{\ell=1}^{M_c}$ and $\{\mathbf{b}_q\}_{q=1}^{M_r}$ denote retained current and propagated reference samples. The biased squared-MMD (Gretton et al., 2012) estimator is

$$\begin{aligned}\widehat{D}^2_{\mathrm{KP},t} = \frac{1}{M_c^2}\sum_{\ell,\ell'} k(\mathbf{a}_\ell, \mathbf{a}_{\ell'}) + \frac{1}{M_r^2}\sum_{q,q'} k(\mathbf{b}_q, \mathbf{b}_{q'}) \\ - \frac{2}{M_c M_r}\sum_{\ell,q} k(\mathbf{a}_\ell, \mathbf{b}_q).\end{aligned} \tag{37}$$

Diagonal terms are included, and the squared statistic is used directly without taking a square root. Since the evidence transformation g in Section 4.2 is bounded and nondecreasing and $x \mapsto x^2$ is increasing on $[0, \infty)$, feeding $\widehat{D}^2_{\mathrm{KP},t}$ rather than its root to g only reparameterizes the calibrated entry and exit thresholds and leaves the triggering rule unchanged. To bound the $O(M^2)$ kernel computation, each window is uniformly subsampled without replacement to at most $M_{\max} = 128$ states, so $M_c = M_r = \min(n, M_{\max})$. At the default $n = 100$ the cap is inactive and both windows are used in full. The discrepancy is set to zero before sufficient history is available.

Evidence and persistence. The smoothing recursion of Section 4.2 is evaluated on the incoming discrepancy sequence, and block-level evidence is obtained by aggregating the resulting hysteresis-filtered signals.

$$e_t = (1 - \alpha)e_{t-1} + \alpha\, g\big(\widehat{D}^2_{\mathrm{KP},t}\big), \qquad \alpha \in (0, 1], \tag{38}$$

where g is the bounded nondecreasing transformation of Section 4.2, implemented as a bounded-influence squashing map, and the recursion is causal. The block statistic D_b of the main text collects the discrepancies computed at the indices of block $\mathcal{B}_b$, and the block evidence Π_b defined below is the hysteresis-filtered evidence at block resolution. Historical calibration determines entry and exit thresholds. For the evidence e_t, the hysteresis latch is

$$\pi_t = \begin{cases}\mathbf{1}\{e_t \geq h_{\mathrm{out}}\}, & \pi_{t-1} = 1, \\ \mathbf{1}\{e_t \geq h_{\mathrm{in}}\}, & \pi_{t-1} = 0.\end{cases} \tag{39}$$

Nonfinite evidence retains the preceding latch state. Block evidence is obtained by aggregating the pointwise latches:

$$\Pi_b = \mathbf{1}\left\{\frac{1}{|\mathcal{B}_b|}\sum_{t \in \mathcal{B}_b} \pi_t \geq \rho_{\mathrm{block}}\right\}. \tag{40}$$

The controller uses `STABLE`, `CANDIDATE`, `ADAPT`, and `COOLDOWN` states. The first block with $\Pi_b = 1$ activates `CANDIDATE`; $\Pi_b = 1$ sustained over p consecutive blocks, the trigger persistence of Section 4.2, confirms `ADAPT`. Evidence disappearance before confirmation returns the controller to `STABLE`. Only `ADAPT` permits adapter optimization. Episode duration and validation-failure limits prevent indefinite adaptation, and cooldown suppresses immediate re-entry. The complete evidence computation and controller transition procedure is summarized in Algorithm 2.

A.3 Validation-Guided Restricted Adaptation

Adapter architecture. A low-rank residual adapter follows each residual block of the conditional velocity network, instantiating Eq. 9. For a hidden representation $\mathbf{a}^{(\ell)} \in \mathbb{R}^h$,

$$\widetilde{\mathbf{a}}^{(\ell)} = \mathbf{a}^{(\ell)} + U_\ell V_\ell^\top \mathbf{a}^{(\ell)} + \mathbf{b}_\ell, \qquad U_\ell, V_\ell \in \mathbb{R}^{h \times r_a}, \quad r_a \ll h. \tag{41}$$

The down-projection and up-projection are linear, with no internal bias or intermediate activation. The separate vector $\mathbf{b}_\ell$ provides an additive correction.

Algorithm 2 Evolution-Compensated Shift Evidence

Require: Observed latent states, frozen K, kernel configuration, evidence and controller states
Ensure: Updated controller state

```
for each newly observed index t do
    if fewer than 2n latent states are available then
        Set the discrepancy to zero
    else
        Construct aligned windows and propagate the reference using exp(Kn)
        Subsample if required and compute Eq. 37
    end if
    Update the evidence using Eq. 38
    Update the latch using Eq. 39
end for
Aggregate block evidence using Eq. 40
Advance the controller using evidence, p-block persistence, episode counters, and validation history
return controller state
```

The up-projection and additive bias are initialized to zero, while the down-projection is randomly initialized. Each adapter therefore starts as an exact identity map. Only $\varphi = \{U_\ell, V_\ell, \mathbf{b}_\ell\}_\ell$ is optimized during adaptation. All pretrained components remain frozen.

Candidate generation. At an adaptation block, the target indices of the pending buffer $\mathcal{A}_b$ are split chronologically as in Section 4.3: the most recent observed targets form a validation holdout $\mathcal{V}_b$, and a bounded training pool $\mathcal{T}_b$ retains earlier targets preceding the holdout. The two target sets are disjoint, although their context windows and successive holdouts may overlap. This chronological separation does not imply statistical independence.

A candidate is initialized from the currently accepted adapter φ_b^- and updated by one SGD step on the objective of Eq. 10,

$$J_b(\varphi) = \lambda_{\mathrm{FM}} \widehat{\mathcal{L}}_{\mathrm{FM},\mathcal{T}_b}(\varphi) + \lambda_{\mathrm{pred}} \widehat{\mathcal{L}}_{\mathrm{pred},\mathcal{T}_b}(\varphi) + \lambda_{\mathrm{reg}} \|\varphi - \varphi_b^0\|_2^2, \tag{42}$$

so that $\varphi_b^\star$ is an approximate minimizer of Eq. 10 rather than an exact one. The flow-matching term applies Eq. 6 on $\mathcal{T}_b$ with the adapters enabled, using the frozen bridge to construct teacher targets; the prediction term $\widehat{\mathcal{L}}_{\mathrm{pred},\mathcal{T}_b} = |\mathcal{T}_b|^{-1} \sum_{i \in \mathcal{T}_b} D^{-1} \|\mathbf{x}_i - \widehat{\mathbf{x}}_i(\varphi)\|_2^2$ penalizes reference errors on observed training targets. The proximal term anchors the candidate at φ_b^0, the adapter parameters at the start of the current episode, which are held fixed while the episode is active and are refreshed only when a new episode begins; it supplies the displacement budget used by Proposition 4. Together with the per-block single step, the minimum and maximum episode durations impose the finite update budget of Section 4.3. The default configuration uses uniform sample weights. Gradients are clipped before updating only the adapter. If the training pool is insufficient, the update is skipped.

Flow correction. The candidate modifies the velocity field by

$$\delta v_b(\mathbf{x}, \tau; \mathbf{c}) = v_{\psi,\varphi_b^\star}(\mathbf{x}, \tau; \mathbf{c}) - v_{\psi,\varphi_b^-}(\mathbf{x}, \tau; \mathbf{c}). \tag{43}$$

Its normal reference is obtained by integrating the modified field. The correction acts throughout reference generation, rather than as an additive change to the final anomaly score. Low-rank parameterization limits the correction structure; it does not by itself guarantee a bound on prediction displacement, which instead follows from the proximal term in Eq. 42 through Proposition 4.

Validation criterion. The candidate and currently accepted model are evaluated on the same validation observations using the same fixed residual standardization and threshold. Define the empirical capped risk and its improvement:

$$\begin{aligned} \widehat{R}_{\mathcal{V}_b}(\varphi) &= \frac{1}{|\mathcal{V}_b|} \sum_{i \in \mathcal{V}_b} \min\{s_i(\varphi; b), u\}, \\ \Delta \widehat{R}_b &= \widehat{R}_{\mathcal{V}_b}(\varphi_b^-) - \widehat{R}_{\mathcal{V}_b}(\varphi_b^\star). \end{aligned} \tag{44}$$

Algorithm 3 Restricted Candidate Update and Validation

Require: Observed target history, accepted adapter φ^-, episode anchor φ^0, frozen backbone, block scorer, threshold u, margin γ
Ensure: Selected adapter φ^+ and acceptance flag
Reserve recent targets as $\mathcal{V}$ and construct an earlier training pool $\mathcal{T}$
if $\mathcal{V}$ is empty or $\mathcal{T}$ is insufficient **then**
    return $(\varphi^-, \text{false})$
end if
$\varphi^\star \leftarrow \text{copy}(\varphi^-)$
Sample a minibatch from $\mathcal{T}$ and compute Eq. 42
if the loss or gradients are nonfinite **then**
    return $(\varphi^-, \text{false})$
end if
Clip gradients and take one SGD step on $\varphi^\star$
Evaluate φ^- and $\varphi^\star$ on $\mathcal{V}$ using the same fixed scorer
if candidate predictions are nonfinite **then**
    return $(\varphi^-, \text{false})$
end if
Compute $\Delta\widehat{R}$ by Eq. 44
if $\Delta\widehat{R} \geq \gamma$ **then**
    return $(\varphi^\star, \text{true})$
else
    Discard candidate changes
    return $(\varphi^-, \text{false})$
end if

As in Eq. 11, acceptance requires an improvement of at least $\gamma > 0$:

$$\varphi_b^+ = \begin{cases} \varphi_b^\star, & \Delta\widehat{R}_b \geq \gamma, \\ \varphi_b^-, & \text{otherwise.} \end{cases} \tag{45}$$

The margin γ is an absolute difference in capped-score units, rather than a percentage improvement. Observations that remain above u under both models contribute no improvement. Capping limits their influence but does not alone ensure that anomalous patterns cannot be fitted.

Only the completed candidate is evaluated for acceptance. A missing validation set or nonfinite candidate loss or prediction results in retaining the accepted model.

Candidate rejection. If validation fails, candidate changes are discarded and the deployed parameter set $\Theta_b = (\psi, \varphi_b^-)$ remains unchanged:

$$\Theta_{b+1} = \Theta_b \quad \text{when} \quad \Delta\widehat{R}_b < \gamma. \tag{46}$$

Because scorer statistics are fixed, rejection does not require restoring a separate adaptive scorer. Candidate rejection is distinct from rolling back an already accepted update; no additional post-acceptance rollback mechanism is introduced here. The complete candidate update and validation procedure is summarized in Algorithm 3.

A.4 Scoring Protocol and Overall Algorithm

Fixed residual calibration. Historical calibration observations are excluded from gradient-based model fitting. Their prediction residuals determine the channelwise center and scale:

$$c_d = \text{median}_i\, r_{i,d}, \qquad S_{dd} = 1.4826\, \text{median}_i\, |r_{i,d} - c_d| + \epsilon_s. \tag{47}$$

These calibrated statistics are used as the fixed center $\mathbf{c}_b$ and scale S_b in Eq. 14. Specifically, for every deployment block b, we set $\mathbf{c}_b = \mathbf{c}$ and $S_b = S$, where $\mathbf{c}$ and S are obtained exclusively from historical calibration residuals. The anomaly score is therefore defined as

$$s_i(\varphi; b) = \frac{1}{D} \left\| S_b^{-1} \left(\mathbf{x}_i - \widehat{\mathbf{x}}_i(\varphi) - \mathbf{c}_b \right) \right\|_2^2. \tag{48}$$

The threshold u is a prescribed quantile of historical calibration scores and keeps this value throughout deployment. The center and scale are held fixed within each candidate comparison and the subsequent rescoring. Consequently, validation improvements cannot arise solely from changes in score normalization. Drift evidence and latent prediction errors are not added to the anomaly score.

Initial scores and deferred finalization. For each target i in block $\mathcal{B}_b$, the initial score is $s_i^{\text{init}} = s_i(\varphi_b^-; b)$, where φ_b^- is the adapter accepted before processing that block. Initial scores are retained separately from the refined scores defined below.

If no mismatch episode is active, scores are finalized at the end of the detection block. Otherwise, prediction pairs remain pending while the controller confirms the mismatch and, when permitted, evaluates candidate updates. An unconfirmed episode closes without adaptation. A confirmed episode closes according to the controller's termination rules, including its duration and validation-failure limits.

Let $\mathcal{I}_e$ denote the pending target indices of episode e, and let b_e and t_e denote its closing block and the last observed index at closure. Using the adapter φ_e^{fin} accepted at that time, we compute

$$s_i^{\text{fin}} = s_i(\varphi_e^{\text{fin}}; b_e), \qquad \widehat{y}_i = \mathbf{1}\{s_i^{\text{fin}} > u\}, \qquad i \in \mathcal{I}_e.$$

The release time is $\rho_i = t_e$, and the observation-index delay is $L_i = \rho_i - i$. Only observations available through ρ_i may influence the final decision. Once finalized, a decision is not revised. At the end of a stream, remaining pending observations are rescored with the last accepted model and finalized without an additional adaptation step.

Information available during refinement. For a fixed adapter, the reference for target i uses the original preceding context $\mathcal{H}_{i-1}$. During deferred refinement, however, the selected adapter may depend on $\mathbf{x}_i$ and on later observations within the completed episode. Thus, rescoring permits unlabeled data reuse for adaptation and subsequent detection within the pending buffer. The chronological training–validation split controls candidate fitting and acceptance; it does not make the refined score an arrival-time prediction.

B Theoretical Analysis

This section provides theoretical justification for the proposed framework. We first establish the estimation error bound of evolution-compensated discrepancy, then analyze the effect of validation-guided adaptation on anomaly detectability.

B.1 Evolution-compensated discrepancy

Let $\mu_k(P)$ denote the kernel mean embedding of P, and let $T_\Delta(P)$ denote its pushforward under the learned latent propagation over lag Δ. For $j \in \{r, c\}$, write

$$P_{z,j} = (1 - \epsilon_j) P_{z,j}^{\text{N}} + \epsilon_j P_{z,j}^{\text{C}},$$

where $P_{z,j}^{\text{C}}$ is an arbitrary contaminating law.

Assumption 2 (Bounded kernel and estimation error)**.** The kernel satisfies $k(z, z) \le \kappa^2$. With probability at least $1 - \delta_{\text{KP}}$, the empirical window laws satisfy

$$\begin{aligned} &\|\mu_k(\widehat{P}_c) - \mu_k(P_{z,c})\|_{\mathcal{H}_k} \\ &\quad + \|\mu_k(T_\Delta(\widehat{P}_r)) - \mu_k(T_\Delta(P_{z,r}))\|_{\mathcal{H}_k} \le \zeta_b \end{aligned} \tag{49}$$

simultaneously over the comparisons under consideration.

This assumption concerns kernel estimation error. The error allowance ζ_b must account for temporal dependence and any data-dependent window selection.

Proof of Proposition 1. The kernel bound gives $\|\mu_k(P)\|_{\mathcal{H}_k} \le \kappa$ for every probability law P. By linearity of kernel mean embeddings,

$$\|\mu_k(P_{z,j}) - \mu_k(P_{z,j}^{\text{N}})\|_{\mathcal{H}_k} \le 2\kappa\epsilon_j.$$

Algorithm 4 Overview of KRAFT

Require: Historical fitting, validation, and calibration splits; deployment stream; configuration
Ensure: Initial scores and deferred final scores with release times

Fit input normalization on historical fitting data
Train dynamics, the bridge predictor, and the flow using historical validation for checkpoint selection
Freeze pretrained components and attach identity-initialized adapters
Calibrate drift thresholds and the fixed residual scorer
Initialize the controller, observed history, accepted adapter φ^{acc}, and pending buffer $\mathcal{A} \leftarrow \varnothing$
for each chronological deployment block $\mathcal{B}_b$ **do**
    $\varphi_b^- \leftarrow \varphi^{\mathrm{acc}}$
    for each target $i \in \mathcal{B}_b$ in arrival order **do**
        Generate the reference from $\mathcal{H}_{i-1}$ using φ_b^- and Algorithm 1
        Observe $\mathbf{x}_i$ and record $s_i^{\mathrm{init}} \leftarrow s_i(\varphi_b^-; b)$
        Store the observed prediction pair and update the history
    end for
    Append the block's prediction pairs to $\mathcal{A}$
    Update shift evidence and the controller using Algorithm 2
    if the controller is in `ADAPT` **then**
        if an adaptation episode has just begun **then**
            Set the episode anchor $\varphi^0 \leftarrow \varphi^{\mathrm{acc}}$
        end if
        Obtain $(\varphi^+, \mathrm{accepted})$ using Algorithm 3 with $\mathcal{A}$, φ^{acc}, and φ^0
        $\varphi^{\mathrm{acc}} \leftarrow \varphi^+$
        Report the validation outcome to the controller
    end if
    if the controller is neither `CANDIDATE` nor `ADAPT` **then**
        Finalize $\mathcal{A}$ using φ^{acc}, rescoring if the adapter changed since the initial scores were computed
        Record the current block endpoint as the release time
        $\mathcal{A} \leftarrow \varnothing$
    end if
end for
Rescore and finalize any remaining $\mathcal{A}$ using φ^{acc}, without an additional update
Record the stream endpoint as the remaining release times

Pushforward preserves mixture weights, so the same bound holds for the propagated reference law.

Define

$$\widehat{v}_b = \mu_k(\widehat{P}_c) - \mu_k(T_\Delta(\widehat{P}_r)),$$
$$v_{K,b} = \mu_k(P^{\mathrm{N}}_{z,c}) - \mu_k(T_\Delta(P^{\mathrm{N}}_{z,r})).$$

On the event in Assumption 2, the triangle inequality yields

$$\|\widehat{v}_b - v_{K,b}\|_{\mathcal{H}_k} \le \zeta_b + 2\kappa(\epsilon_r + \epsilon_c).$$

Since $D_b = \|\widehat{v}_b\|_{\mathcal{H}_k}$ and $d_{K,b} = \|v_{K,b}\|_{\mathcal{H}_k}$, the reverse triangle inequality proves the result. □

To establish Corollary 1, let $Q^{\mathrm{N}}_{z,c|r}$ denote the specified normal continuation of the reference distribution. Its mismatch with the learned propagation is

$$a_b := \mathrm{MMD}_k\big(T_\Delta(P^{\mathrm{N}}_{z,r}), Q^{\mathrm{N}}_{z,c|r}\big). \tag{50}$$

The exogenous discrepancy is

$$d_{\mathrm{exo},b} := \mathrm{MMD}_k\big(P^{\mathrm{N}}_{z,c}, Q^{\mathrm{N}}_{z,c|r}\big).$$

By the reverse triangle inequality, $|d_{K,b} - d_{\mathrm{exo},b}| \le a_b$. Combining this with Proposition 1 gives, on the same estimation event,

$$|D_b - d_{\mathrm{exo},b}| \le E_b, \qquad E_b := a_b + 2\kappa(\epsilon_r + \epsilon_c) + \zeta_b. \tag{51}$$

The error budget includes propagation mismatch, contamination, and empirical estimation error. No smallness assumption on a_b is needed for this bound; distinguishability requires separation relative to the resulting total error.

Proof of Corollary 1. Under normal continuation, $P^{\mathrm{N}}_{z,c} = Q^{\mathrm{N}}_{z,c|r}$, and hence $d_{\mathrm{exo},b} = 0$. Equation 51 implies

$$D_b \le E_b \le \overline{E} < h.$$

For an exogenous shift satisfying $d_{\mathrm{exo},b} \ge \Delta_{\mathrm{exo}}$, the same bound yields

$$D_b \ge d_{\mathrm{exo},b} - E_b \ge \Delta_{\mathrm{exo}} - \overline{E} > h.$$

The threshold interval $(\overline{E}, \Delta_{\mathrm{exo}} - \overline{E})$ is nonempty because $\Delta_{\mathrm{exo}} > 2\overline{E}$. Both conclusions hold simultaneously on the estimation event of Assumption 2, with probability at least $1 - \delta_{\mathrm{KP}}$. □

The corollary concerns changes visible in the latent space and provides a sufficient separation condition for raw KP-MMD. It does not directly establish guarantees for the full controller with smoothing, hysteresis, and persistence.

B.2 Validated adaptation and recall-loss control

Let $Z = (\mathcal{H}, \mathbf{x})$ follow the local law

$$P_b = (1 - \epsilon_b) P^{\mathrm{N}}_b + \epsilon_b P^{\mathrm{A}}_b, \qquad 0 \le \epsilon_b < 1.$$

All compared models use the same score statistics and the same threshold $u > 0$. Population quantities evaluate these models with their score statistics held fixed.

For a generic prediction pair $Z = (\mathcal{H}, \mathbf{x})$, the score in Eq. 14 is written as

$$s_b(Z; \varphi) := \frac{1}{D} \left\| S_b^{-1} \big(\mathbf{x} - \widehat{\mathbf{x}}_\varphi(\mathcal{H}) - \mathbf{c}_b \big) \right\|_2^2 .$$

Write

$$\begin{aligned} r_b(Z; \varphi) &:= \sqrt{s_b(Z; \varphi)}, \\ \ell_b(Z; \varphi) &:= \min\{ s_b(Z; \varphi), u \}, \\ R^{\mathrm{N}}_b(\varphi) &:= \mathbb{E}_{P^{\mathrm{N}}_b} \ell_b(Z; \varphi). \end{aligned}$$

Let $\mathcal{C}_b$ contain every compared candidate, and define its standardized prediction displacement from the incumbent by

$$\Delta_b(Z; \varphi) := \frac{1}{\sqrt{D}} \left\| S_b^{-1} \big(\widehat{\mathbf{x}}_\varphi(\mathcal{H}) - \widehat{\mathbf{x}}_{\varphi_b^-}(\mathcal{H}) \big) \right\|_2 . \tag{52}$$

For $J \in \{\mathrm{N}, \mathrm{A}\}$, define the population capped-risk gain by

$$G^J_b(\varphi) := \mathbb{E}_{P^J_b} \big[\ell_b(Z; \varphi^-_b) - \ell_b(Z; \varphi) \big].$$

The corresponding mixture gain is

$$G_b(\varphi) := (1 - \epsilon_b) G^{\mathrm{N}}_b(\varphi) + \epsilon_b G^{\mathrm{A}}_b(\varphi).$$

Using the validation risk from Eq. 11, define

$$\widehat{G}_b(\varphi) := \widehat{R}_{\mathcal{V}_b}(\varphi^-_b) - \widehat{R}_{\mathcal{V}_b}(\varphi).$$

Assumption 3 (Uniform validation accuracy)**.** With probability at least $1 - \delta_{\mathrm{V}}$,

$$\sup_{\varphi \in \mathcal{C}_b} \left| \widehat{G}_b(\varphi) - G_b(\varphi) \right| \le \eta_b \tag{53}$$

simultaneously over the adaptation decisions under consideration.

The uniform event covers all compared candidates, including data-dependent selection. The error allowance η_b must account for temporal dependence and the construction of the validation buffer and shared scoring statistics; a chronological split alone does not establish this bound.

Assumption 4 (Mean-square prediction displacement)**.** For some finite $d_b \geq 0$, the compared candidates satisfy

$$\sup_{\varphi \in \mathcal{C}_b} \mathbb{E}_{P_b^{\mathrm{A}}}\left[\Delta_b(Z;\varphi)^2\right] \leq d_b^2. \tag{54}$$

This condition allows large changes on a small fraction of inputs. The bound d_b need not be small; smaller values yield more informative guarantees.

Define the initial anomaly mass immediately above the decision boundary by

$$\omega_b(t) := \Pr_{P_b^{\mathrm{A}}}\left(\sqrt{u} < r_b(Z;\varphi_b^-) \leq \sqrt{u} + t\right), \qquad t > 0, \tag{55}$$

and let

$$q_b^{\pm} := \Pr_{P_b^{\mathrm{A}}}\big(s_b(Z;\varphi_b^{\pm}) > u\big).$$

No upper bound on $\omega_b(t)$ or lower bound on q_b^- is assumed. The quantity t is an analysis parameter and does not change the algorithm.

Proof of Theorem 1. Fix a compared candidate φ. Because score statistics are shared, the reverse triangle inequality for standardized residual norms gives

$$\left|r_b(Z;\varphi) - r_b(Z;\varphi_b^-)\right| \leq \Delta_b(Z;\varphi). \tag{56}$$

The function $f(r) = \min\{r^2, u\}$ is $2\sqrt{u}$-Lipschitz on $[0,\infty)$: below $\sqrt{u}$ its derivative is at most $2\sqrt{u}$, and above $\sqrt{u}$ it is constant. Its range is also contained in $[0,u]$. Therefore,

$$\left|\ell_b(Z;\varphi_b^-) - \ell_b(Z;\varphi)\right| \leq \min\{u, 2\sqrt{u}\,\Delta_b(Z;\varphi)\}.$$

By Cauchy–Schwarz and Assumption 4,

$$\begin{aligned}
|G_b^{\mathrm{A}}(\varphi)| &= \left|\mathbb{E}_{P_b^{\mathrm{A}}}[\ell_b(Z;\varphi_b^-) - \ell_b(Z;\varphi)]\right| \\
&\leq \min\left\{u, 2\sqrt{u}\,\mathbb{E}_{P_b^{\mathrm{A}}}\Delta_b(Z;\varphi)\right\} \\
&\leq \min\{u, 2\sqrt{u}\,d_b\} = c_b.
\end{aligned}$$

On the event in Assumption 3, an accepted candidate φ_b^+ satisfies

$$G_b(\varphi_b^+) \geq \widehat{G}_b(\varphi_b^+) - \eta_b \geq \gamma - \eta_b,$$

where the last inequality follows from Eq. 11.

Using the mixture decomposition and $G_b^{\mathrm{A}}(\varphi_b^+) \leq c_b$, we obtain

$$\begin{aligned}
(1-\epsilon_b)G_b^{\mathrm{N}}(\varphi_b^+) &= G_b(\varphi_b^+) - \epsilon_b G_b^{\mathrm{A}}(\varphi_b^+) \\
&\geq \gamma - \eta_b - \epsilon_b c_b.
\end{aligned}$$

Since $\epsilon_b < 1$ and

$$G_b^{\mathrm{N}}(\varphi_b^+) = R_b^{\mathrm{N}}(\varphi_b^-) - R_b^{\mathrm{N}}(\varphi_b^+),$$

division by $1-\epsilon_b$ proves the first inequality.

Conversely, on the uniform validation event, an actually generated candidate $\varphi^\star$ is accepted whenever

$$(1-\epsilon_b)G_b^{\mathrm{N}}(\varphi^\star) > \gamma + \eta_b + \epsilon_b c_b,$$

because $\widehat{G}_b(\varphi^\star) \geq (1-\epsilon_b)G_b^{\mathrm{N}}(\varphi^\star) - \epsilon_b c_b - \eta_b > \gamma$.

For recall, fix any $t > 0$. If an anomaly is detected before adaptation but missed afterward, then either

$$\sqrt{u} < r_b(Z;\varphi_b^-) \leq \sqrt{u} + t,$$

or $\Delta_b(Z;\varphi_b^+) > t$. Indeed, outside these two events, (56) gives

$$r_b(Z;\varphi_b^+) \geq r_b(Z;\varphi_b^-) - \Delta_b(Z;\varphi_b^+) > \sqrt{u}.$$

Thus, Markov's inequality yields

$$\begin{aligned}\Pr_{P_b^{\mathrm{A}}}\big(s_b(Z;\varphi_b^-) > u,\ s_b(Z;\varphi_b^+) \le u\big) &\le \omega_b(t) + \Pr_{P_b^{\mathrm{A}}}\big(\Delta_b(Z;\varphi_b^+) > t\big) \\ &\le \omega_b(t) + \frac{d_b^2}{t^2}.\end{aligned}$$

Subtracting this possible loss from the initial recall proves the second inequality.

Both bounds apply to every accepted candidate on the uniform validation event, whose probability is at least $1 - \delta_{\mathrm{V}}$. □

The recall bound characterizes how validated adaptation affects anomaly detectability. Specifically, it quantifies the potential recall degradation induced by reference updates through the prediction-displacement budget and the initial near-threshold anomaly mass $\omega_b(t)$. When the adaptation magnitude is bounded, the separation between shifted normal behavior and anomalous deviations is preserved up to the derived bound, indicating that the adapted reference improves normal-risk fitting while maintaining anomaly detection capability.

B.3 Stable Koopman propagation

Proposition 2 (Contractive propagation with modeling error)**.** *For*

$$K = (A - A^\top) - \mathrm{diag}\big(\mathrm{softplus}(\mathbf{b}_K)\big), \qquad \lambda_* = \min_j \mathrm{softplus}((\mathbf{b}_K)_j) > 0,$$

we have $\|\exp(Kt)\|_{\mathrm{op}} \le e^{-\lambda_* t}$ *for* $t \ge 0$*. If an ideal normal trajectory satisfies*

$$\dot{\mathbf{z}}^{\mathrm{N}} = K\mathbf{z}^{\mathrm{N}} + \mathbf{r}(t), \qquad \|\mathbf{r}(t)\|_2 \le \varepsilon_K,$$

and $\|\widehat{\mathbf{z}}_0 - \mathbf{z}^{\mathrm{N}}(0)\|_2 \le e_0$*, then*

$$\big\|\exp(Kt)\widehat{\mathbf{z}}_0 - \mathbf{z}^{\mathrm{N}}(t)\big\|_2 \le e^{-\lambda_* t} e_0 + \frac{\varepsilon_K}{\lambda_*}(1 - e^{-\lambda_* t}). \tag{57}$$

Proof. The symmetric part satisfies $(K + K^\top)/2 \preceq -\lambda_* I$. For $\mathbf{w}(t) = \exp(Kt)\mathbf{w}_0$,

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{w}(t)\|_2^2 = 2\mathbf{w}(t)^\top K\mathbf{w}(t) \le -2\lambda_*\|\mathbf{w}(t)\|_2^2.$$

Integration proves the contraction bound.

For $\mathbf{e}(t) = \exp(Kt)\widehat{\mathbf{z}}_0 - \mathbf{z}^{\mathrm{N}}(t)$, variation of constants yields

$$\mathbf{e}(t) = \exp(Kt)\mathbf{e}(0) - \int_0^t \exp(K(t-s))\mathbf{r}(s)\,\mathrm{d}s.$$

Hence

$$\|\mathbf{e}(t)\|_2 \le e^{-\lambda_* t} e_0 + \varepsilon_K \int_0^t e^{-\lambda_*(t-s)}\,\mathrm{d}s,$$

which gives Eq. 57. □

Proposition 2 provides a sufficient bound on the propagation mismatch a_b in Eq. 50, relative to the normal continuation underlying Assumption 1. Specifically, suppose the kernel feature map is L_k-Lipschitz and the laws $T_\Delta(P_{z,r}^{\mathrm{N}})$ and $Q_{z,c|r}^{\mathrm{N}}$ admit a coupling $(Z, \widetilde{Z})$ for which the trajectory error bound in Eq. 57 holds almost surely with common constants. Then

$$\begin{aligned}a_b &= \Big\|\mathbb{E}[\phi_k(Z) - \phi_k(\widetilde{Z})]\Big\|_{\mathcal{H}_k} \\ &\le L_k \mathbb{E}\|Z - \widetilde{Z}\|_2 \\ &\le L_k\left[e^{-\lambda_* \Delta} e_0 + \frac{\varepsilon_K}{\lambda_*}(1 - e^{-\lambda_* \Delta})\right].\end{aligned}$$

This bound can be substituted into the error budget in Eq. 51 to assess the separation condition in Corollary 1. It controls the propagation component of that budget; the statistical estimation bound remains governed by Assumption 2.

B.4 BIDIRECTIONAL GAUSSIAN FUSION

Fix the training endpoints (i, j) and suppress their conditioning in the notation. Write the directional predictors as

$$p_{\ell,\tau} = \mathcal{N}\big(\boldsymbol{\mu}_{\ell,\tau}, \mathrm{diag}(\boldsymbol{\sigma}^2_{\ell,\tau})\big), \qquad \ell \in \{f, b\},$$

with strictly positive component variances; in the notation of Eq. 24, $\sigma^2_{f,\tau,d} = \nu_{f,d}$, $\sigma^2_{b,\tau,d} = \nu_{b,d}$, and $\sigma^2_{\tau,d} = \nu_{\tau,d}$. The path in Eq. 4 is

$$q_\tau(\mathbf{x}) = Z_\tau^{-1} p_{f,\tau}(\mathbf{x})^{1-\tau} p_{b,\tau}(\mathbf{x})^{\tau}, \qquad \tau \in [0, 1].$$

Proposition 3 (Variational and precision-weighted fusion)**.** *The fused path is Gaussian, with component parameters*

$$\begin{aligned} \sigma^2_{\tau,d} &= \left(\frac{1-\tau}{\sigma^2_{f,\tau,d}} + \frac{\tau}{\sigma^2_{b,\tau,d}} \right)^{-1}, \\ \mu_{\tau,d} &= \sigma^2_{\tau,d} \left(\frac{(1-\tau)\mu_{f,\tau,d}}{\sigma^2_{f,\tau,d}} + \frac{\tau \mu_{b,\tau,d}}{\sigma^2_{b,\tau,d}} \right). \end{aligned} \tag{58}$$

It uniquely minimizes

$$\mathcal{J}_\tau(q) = (1-\tau)\,\mathrm{KL}(q \| p_{f,\tau}) + \tau\,\mathrm{KL}(q \| p_{b,\tau}) \tag{59}$$

over probability densities with finite objective. Moreover, each $\mu_{\tau,d}$ is a convex combination of the directional means, and

$$\min_{\ell \in \{f,b\}} \sigma^2_{\ell,\tau,d} \le \sigma^2_{\tau,d} \le \max_{\ell \in \{f,b\}} \sigma^2_{\ell,\tau,d}. \tag{60}$$

The endpoint laws satisfy $q_0 = p_{f,0}$ and $q_1 = p_{b,1}$.

Proof. For each coordinate d, the log of the unnormalized product has the form

$$-\frac{1}{2}\lambda_{\tau,d} x_d^2 + \beta_{\tau,d} x_d + \text{constant},$$

where

$$\lambda_{\tau,d} = \frac{1-\tau}{\sigma^2_{f,\tau,d}} + \frac{\tau}{\sigma^2_{b,\tau,d}} > 0, \qquad \beta_{\tau,d} = \frac{(1-\tau)\mu_{f,\tau,d}}{\sigma^2_{f,\tau,d}} + \frac{\tau \mu_{b,\tau,d}}{\sigma^2_{b,\tau,d}}.$$

Completing the square gives $\sigma^2_{\tau,d} = \lambda^{-1}_{\tau,d}$ and $\mu_{\tau,d} = \beta_{\tau,d}/\lambda_{\tau,d}$. This proves Gaussian closure and Eq. 58, and also establishes a finite positive normalizer Z_τ.

By the definition of q_τ,

$$\log q_\tau = (1-\tau)\log p_{f,\tau} + \tau \log p_{b,\tau} - \log Z_\tau.$$

Substitution into Eq. 59 gives

$$\mathcal{J}_\tau(q) = \mathrm{KL}(q \| q_\tau) - \log Z_\tau.$$

Nonnegativity of KL divergence, with equality only when $q = q_\tau$ almost everywhere, proves uniqueness.

Define the effective weights

$$w_{f,\tau,d} = \frac{(1-\tau)/\sigma^2_{f,\tau,d}}{\lambda_{\tau,d}}, \qquad w_{b,\tau,d} = \frac{\tau/\sigma^2_{b,\tau,d}}{\lambda_{\tau,d}}.$$

They are nonnegative and sum to one, and

$$\mu_{\tau,d} = w_{f,\tau,d}\mu_{f,\tau,d} + w_{b,\tau,d}\mu_{b,\tau,d}.$$

The fused precision $\lambda_{\tau,d}$ is itself a convex combination of the directional precisions. Taking reciprocals proves Eq. 60. Finally, setting $\tau = 0$ or $\tau = 1$ in the normalized product gives the endpoint laws. □

The effective weights combine path time with directional precision. In particular, equal directional variances are preserved by fusion; multiplying the two densities without normalized exponents would not have this property. Endpoint mean constraints are inherited from the directional parameterization: $\boldsymbol{\mu}_{f,0} = \mathbf{x}_i$ and $\boldsymbol{\mu}_{b,1} = \mathbf{x}_j$ imply $\boldsymbol{\mu}_0 = \mathbf{x}_i$ and $\boldsymbol{\mu}_1 = \mathbf{x}_j$. These mean constraints do not require zero endpoint variance.

Consistency with the flow-matching target. Suppose the directional parameters are continuously differentiable and their variances remain positive. The fused parameters in Eq. 58 are then differentiable. For a fixed $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, I_D)$,

$$\mathbf{X}_\tau = \boldsymbol{\mu}_\tau + \boldsymbol{\sigma}_\tau \odot \boldsymbol{\varepsilon} \sim q_\tau, \qquad \dot{\mathbf{X}}_\tau = \dot{\boldsymbol{\mu}}_\tau + \dot{\boldsymbol{\sigma}}_\tau \odot \boldsymbol{\varepsilon}.$$

Thus Eq. 5 is exactly the derivative of a sample trajectory from the designed fused path. Equivalently, its endpoint-conditioned velocity field is

$$u_\tau(\mathbf{x}) = \dot{\boldsymbol{\mu}}_\tau + \frac{\dot{\boldsymbol{\sigma}}_\tau}{\boldsymbol{\sigma}_\tau} \odot (\mathbf{x} - \boldsymbol{\mu}_\tau),$$

where division is componentwise. For any smooth compactly supported test function f, whenever differentiation under expectation is valid,

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\mathbb{E}[f(\mathbf{X}_\tau)] = \mathbb{E}[\nabla f(\mathbf{X}_\tau)^\top u_\tau(\mathbf{X}_\tau)].$$

This proves $\partial_\tau q_\tau + \nabla \cdot (q_\tau u_\tau) = 0$ in the weak sense.

All derivatives include the dependence of the directional predictors on the propagated state and its rate:

$$\partial_\tau \mathbf{z}_{i,\tau} = HK\mathbf{z}_{i,\tau}, \qquad \partial_\tau(K\mathbf{z}_{i,\tau}) = HK^2\mathbf{z}_{i,\tau}.$$

Under the stated differentiability and positivity conditions, Proposition 3 and the continuity-equation argument establish that the fused Gaussian path admits the training velocity used in Eq. 5. This justifies the supervision associated with the designed probability bridge. The regularity of the learned velocity field required by Proposition 4 remains a separate condition.

B.5 Regularization and prediction displacement

Proposition 4 (Regularization-controlled prediction displacement)**.** *Write the adaptation objective as*

$$J_b(\varphi) = F_b(\varphi) + \lambda_{\mathrm{reg}}\|\varphi - \varphi_b^0\|_2^2, \qquad F_b \geq 0, \quad \lambda_{\mathrm{reg}} > 0.$$

Suppose each considered candidate satisfies $J_b(\varphi) \leq J_b(\varphi_b^0) + \xi_b$, where $\xi_b \geq 0$, and define

$$\rho_b = \sqrt{\frac{F_b(\varphi_b^0) + \xi_b}{\lambda_{\mathrm{reg}}}} + \|\varphi_b^- - \varphi_b^0\|_2.$$

For identical initial values and frozen conditioning c_τ, suppose uniformly over the relevant trajectories and candidates that

$$\|v_\varphi(x, \tau; c_\tau) - v_\varphi(y, \tau; c_\tau)\|_2 \leq L_x\|x - y\|_2,$$
$$\|v_\varphi(x, \tau; c_\tau) - v_{\varphi_b^-}(x, \tau; c_\tau)\|_2 \leq L_\varphi\|\varphi - \varphi_b^-\|_2.$$

For exact integration over $\tau \in [0, 1]$,

$$D^{-1/2}\left\|S_b^{-1}\left(\widehat{\mathbf{x}}_\varphi - \widehat{\mathbf{x}}_{\varphi_b^-}\right)\right\|_2 \leq D^{-1/2}\|S_b^{-1}\|_{\mathrm{op}} L_\varphi \rho_b \chi(L_x), \tag{61}$$

where $\chi(L) = (e^L - 1)/L$ for $L > 0$ and $\chi(0) = 1$.

Proof. Nonnegativity gives

$$\lambda_{\mathrm{reg}}\|\varphi - \varphi_b^0\|_2^2 \leq J_b(\varphi) \leq F_b(\varphi_b^0) + \xi_b.$$

Taking square roots and applying the triangle inequality yields $\|\varphi - \varphi_b^-\|_2 \leq \rho_b$.

Let $w(\tau)$ be the distance between the two reference trajectories with the same initial value. Their integral equations imply

$$w(\tau) \leq \int_0^\tau \left(L_x w(s) + L_\varphi \rho_b\right) \mathrm{d}s.$$

For $L_x > 0$, denote the right side by $H(\tau)$. Then $w \leq H$, $H(0) = 0$, and $H' \leq L_x H + L_\varphi \rho_b$. Multiplying by $e^{-L_x\tau}$ and integrating gives

$$w(\tau) \leq L_\varphi \rho_b \frac{e^{L_x\tau} - 1}{L_x}.$$

For $L_x = 0$, directly $w(\tau) \leq L_\varphi \rho_b \tau$. Evaluate at $\tau = 1$ and multiply by $D^{-1/2}\|S_b^{-1}\|_{\mathrm{op}}$ to obtain Eq. 61. □

Proposition 4 provides a sufficient condition for Assumption 4. Define

$$B_b^{\mathrm{pred}} := D^{-1/2}\|S_b^{-1}\|_{\mathrm{op}} L_\varphi \rho_b \chi(L_x).$$

If the proposition's conditions hold with common constants for all compared candidates and P_b^{A}-almost every prediction history, Eq. 61 gives

$$\Delta_b(Z;\varphi) \le B_b^{\mathrm{pred}}.$$

Consequently,

$$\sup_{\varphi\in\mathcal{C}_b} \mathbb{E}_{P_b^{\mathrm{A}}}[\Delta_b(Z;\varphi)^2] \le (B_b^{\mathrm{pred}})^2,$$

so Assumption 4 holds with $d_b = B_b^{\mathrm{pred}}$. This supplies a displacement budget for the anomaly-contribution and recall-loss bounds in Theorem 1. The assumption also allows nonuniform displacements whose second moments satisfy Eq. 54; the uniform bound above is sufficient but not necessary.

C Detailed Experimental Setup

This appendix documents the benchmark datasets, the construction of the multi-domain datasets, the baseline comparison protocol, the evaluation metrics, and the complete configuration of KRAFT.

C.1 Datasets

We follow the dataset collection and preprocessing conventions of the TSB-AD benchmark (Liu & Paparrizos, 2024), which spans service monitoring, spacecraft telemetry, industrial control, energy, and financial domains. Our evaluation suite consists of three groups.

- **Univariate benchmarks.** Eleven single-channel datasets are used: IOPS, MGAB, NAB, NEK, Power, SED, Stock, TODS, UCR, WSD, and YAHOO. They differ widely in sequence length, sampling regularity, and anomaly density, and therefore probe robustness rather than a single operating regime.
- **Multivariate benchmarks.** Five multi-channel datasets are used: MSL, PSM, SMAP, SMD, and SWaT. These originate from spacecraft and industrial control systems, where anomalies are expressed jointly across correlated channels.
- **Domain-drift benchmarks.** To evaluate detection when normal behavior itself changes, we use the real-world drift benchmark INSECTS (Souza et al., 2020) together with three synthetic multi-domain datasets, BDS, MDS, and CDS, whose generation is described in Appendix C.2.

Every sequence is partitioned chronologically into a historical segment and a deployment stream. The historical segment is further split into a fitting subset, used to estimate the input standardization statistics and to run the three training stages of Appendix C.5, and a disjoint calibration subset, used only for the drift entry and exit thresholds and the fixed residual scorer of Eq. 48. Anomaly labels are never accessed during fitting, calibration, or deployment, and are used exclusively when computing the reported metrics.

Tables 3 and 4 summarize the univariate and multivariate benchmarks, reporting the application field, the number of time series, the average sequence length, and the anomaly ratio. Table 5 reports the corresponding statistics for the four domain-drift benchmarks, together with the average number of latent domains per sequence.

C.2 Construction of the Multi-Domain Benchmark

Existing TSAD benchmarks rarely expose a controllable notion of latent domain, so we build three synthetic suites in which the timing, the smoothness, and the *mechanism* of each shift are known by construction while remaining unobserved by the detector. The symbols introduced in this subsection describe the data generator only and are local to it. In particular, the transition centers τ_k and widths

Table 3: Statistics of the eleven univariate benchmarks.

Dataset	Field	# TS	Avg. length	Anomaly ratio (%)
UCR	Misc.	228	67818.7	0.6
NAB	Web	28	5099.7	10.6
YAHOO	Web	259	1560.2	0.6
IOPS	Operations	17	72792.3	1.3
MGAB	Sensor	9	97777.8	0.2
SED	Energy	3	23332.3	4.1
Stock	Finance	20	15000.0	9.4
TODS	Traffic	15	5000.0	6.3
NEK	Weather	9	1073.0	8.0
Power	Power grid	1	35040.0	8.5
WSD	Weather	111	17444.5	0.6

Table 4: Statistics of the five multivariate benchmarks.

Dataset	Field	# TS	Avg. length	Anomaly ratio (%)
MSL	Space	16	3119.4	5.1
PSM	Sensor	1	217624.0	11.2
SMAP	Space	27	7855.9	2.9
SMD	Server	22	25466.4	3.8
SWaT	ICS	2	207457.5	12.7

ω_k below index the generator timeline and are unrelated to the normalized flow time τ of Section 4.1 or to the quantities ω appearing in the analysis. Conversely, we write M for the number of domains and $q_{t,i}$ for the latent causal process, because K and $\mathbf{z}$ denote the Koopman generator and the Koopman latent state throughout the paper.

Base generator. We start from the additive trend–seasonality–noise synthesis of Lan et al. (2025), which produces a single-channel signal

$$x_{t,i}^{\text{base}} = \text{tr}(t; \theta_i^{\text{tr}}) + \text{se}(t; \theta_i^{\text{se}}) + \zeta_{t,i}, \tag{62}$$

where θ_i^{tr} and θ_i^{se} collect the trend and seasonality parameters of channel i and $\zeta_{t,i}$ is observation noise. In our base configuration, the trend and seasonality parameters remain fixed within each sequence, without explicit switches between latent domains. Figure 4 illustrates this configuration before controlled domain changes are introduced.

Explicit drift latent. We therefore attach a latent domain trajectory to the generator. Assuming M underlying domains, we define a scalar state $s_t \in [0, M-1]$ as a superposition of $M-1$ smooth transitions,

$$s_t = \sum_{k=1}^{M-1} \text{sig}\left(\frac{t-\tau_k}{\omega_k}\right), \qquad \text{sig}(a) = \frac{1}{1+e^{-a}}, \tag{63}$$

in which τ_k places the center of the k-th domain change and ω_k sets the width of the surrounding transition band: a large ω_k produces a gradual handover, whereas a small ω_k produces an abrupt one. Writing $m_t = \lfloor s_t \rfloor$ for the active domain index and $\{s_t\} = s_t - m_t$ for the fractional progress within a transition, the effective generator parameters are obtained by interpolating between the two adjacent domain configurations,

$$\theta_t = (1 - \{s_t\})\, \theta_{m_t} + \{s_t\}\, \theta_{m_t+1}. \tag{64}$$

Figure 5 visualizes such a trajectory: s_t behaves as a cumulative domain counter whose k-th step is centered at τ_k and spread over a band controlled by ω_k. The configuration $\{(\tau_k, \omega_k)\}_k$ is therefore a direct handle on the drift scenario, and the band $[\tau_k - \omega_k,\ \tau_k + \omega_k]$ serves as the ground-truth region of a domain change.

Table 5: Statistics of the four domain-drift benchmarks.

Dataset	Avg. # domains	# TS	Avg. length	Anomaly ratio (%)
BDS	3.2	5	1294.2	3.8
MDS	4.2	5	1192.4	7.2
CDS	4.2	5	1422.0	4.9
INSECTS	Not annotated	4	45119.2	1.2

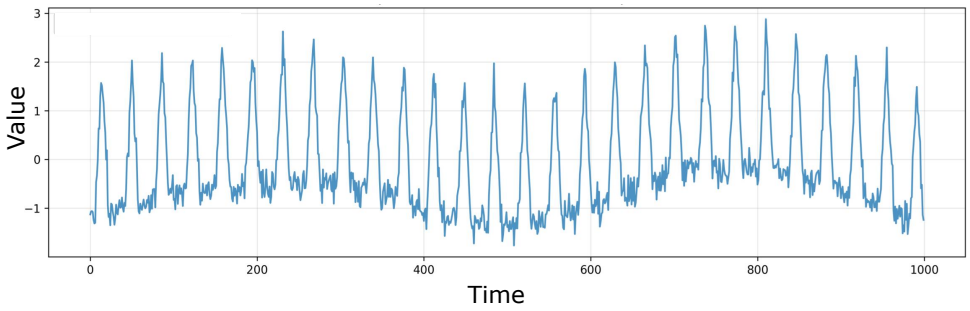


Figure 4: A synthetic sequence generated by Eq. 62 with fixed trend and seasonality parameters and no explicitly introduced domain switches.

Multivariate mixing. For multi-channel sequences, each observed channel is a convex mixture of its independent baseline and a shared latent causal process $q_{t,i}$,

$$x_{t,i} = \alpha_i \, x_{t,i}^{\text{base}} + (1 - \alpha_i) \, q_{t,i}, \qquad q_{t,i} = f_i \left(\sum_{j \in \text{PA}(i)} b_{ij} q_{t,j} + a_i q_{t-1,i} + c_i \epsilon_{t,i} \right), \tag{65}$$

where $\alpha_i \in [0, 1]$ is a mixing coefficient, $\text{PA}(i)$ is the parent set of channel i in a directed acyclic graph, b_{ij} is the cross-channel causal strength, a_i is an autoregressive coefficient, c_i scales the innovation $\epsilon_{t,i}$, and f_i is a fixed nonlinearity. Anomalies are injected independently of the domain trajectory, so that a domain change never coincides with a label by construction.

Basic domain shift (BDS). In the mildest suite, only the per-channel baseline parameters $(\theta_i^{\text{tr}}, \theta_i^{\text{se}})$ follow s_t, while the mixing coefficients and the causal coefficients are held fixed. Amplitude and periodicity therefore change across domains, whereas the waveform family, the smoothness, and the dependency structure persist, as shown in Figure 6.

Mixed domain shift (MDS). The second suite lets the baseline parameters and the mixing coefficients α_i evolve jointly. As a domain transition proceeds, a channel moves between being dominated by its own seasonal baseline and being dominated by the shared causal process of Eq. 65, so what changes is the texture of the signal rather than only its scale. Figure 7 illustrates the resulting behavior, where a regular seasonality-dominated segment gives way to a volatile, noise-rich one. This suite therefore tests whether a detector can distinguish an increase in intrinsic variability from a genuine anomaly.

Causal domain shift (CDS). The hardest suite freezes the baselines and mixing weights and lets only the structural coefficients $\{a_i, b_{ij}, c_i\}$ follow s_t. Interpolating these coefficients changes the temporal dynamics and, when a coefficient crosses zero, removes or creates an edge of the dependency graph, so the topology itself differs across domains. Figure 8 shows the resulting sequence, which exhibits sharp transitions and structural discontinuities rather than a smooth statistical drift, and Figure 9 shows the underlying graph. Under CDS it is the generating mechanism, not merely its surface statistics, that differs across domains, which is precisely the setting in which a fixed reference model is expected to fail.

C.3 Baselines and Comparison Protocol

We compare KRAFT against eleven representative detectors covering classical, deep, and drift-oriented families.

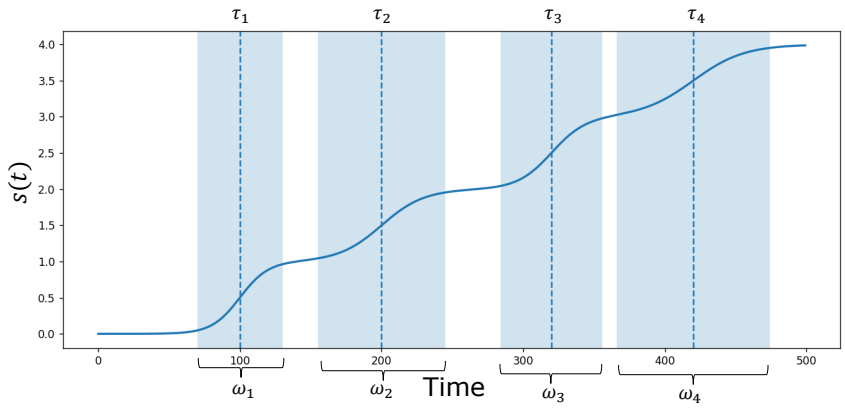


Figure 5: A trajectory of the explicit drift latent s_t of Eq. 63. The state accumulates $M-1$ sigmoid transitions, each corresponding to one domain change: τ_k marks the timestamp of the k-th switch and ω_k the width of the surrounding buffer zone, which governs whether the transition is abrupt or gradual. Between two consecutive plateaus the generator parameters are interpolated according to Eq. 64.

- **LOF** (Breunig et al., 2000) scores a point by the ratio between its local density and that of its neighbors.
- **IForest** (Liu et al., 2008) isolates outliers through randomized recursive partitioning of the feature space.
- **Sub-PCA** (Liu & Paparrizos, 2024) reconstructs subsequences from their leading principal components and scores the residual.
- **LSTMAD** (Malhotra et al., 2015) forecasts the next observation with a recurrent network and scores the prediction error.
- **OmniAnomaly** (Su et al., 2019) couples a stochastic recurrent network with normalizing flows to model the distribution of normal patterns.
- **USAD** (Audibert et al., 2020) trains two autoencoders adversarially to sharpen the reconstruction gap between normal and anomalous inputs.
- **TranAD** (Tuli et al., 2022) applies a Transformer with adversarial training and a focus score to amplify small deviations.
- **DCdetector** (Yang et al., 2023) contrasts two attention views of the same series and scores their representation discrepancy.
- **D3R** (Wang et al., 2023) targets nonstationarity by dynamically decomposing and reconstructing time-varying components.
- **M2N2** (Kim et al., 2024) performs test-time adaptation, combining trend estimation with self-supervised updates on observations predicted to be normal.
- **DIVAD** (Jacob & Diao, 2025) learns domain-invariant latent representations so that a single detector transfers across domains.

The last three are explicitly designed for multi-domain or drifting time series and therefore form the most informative comparison for the setting of Definition 2.

Comparison protocol. All methods see exactly the same sequences and the same chronological splits; what differs is only the inference regime implied by each design, which we preserve rather than modify. No method observes anomaly labels at any stage, and no method observes the deployment stream during fitting. Baseline hyperparameters follow the values recommended by their authors, and each baseline is tuned only over the search space reported in its original paper.

C.4 Evaluation Metrics

Point-wise accuracy alone is a poor summary for time-series anomaly detection, because anomalies occur as contiguous events and a detector may recover an event while misaligning its exact extent.

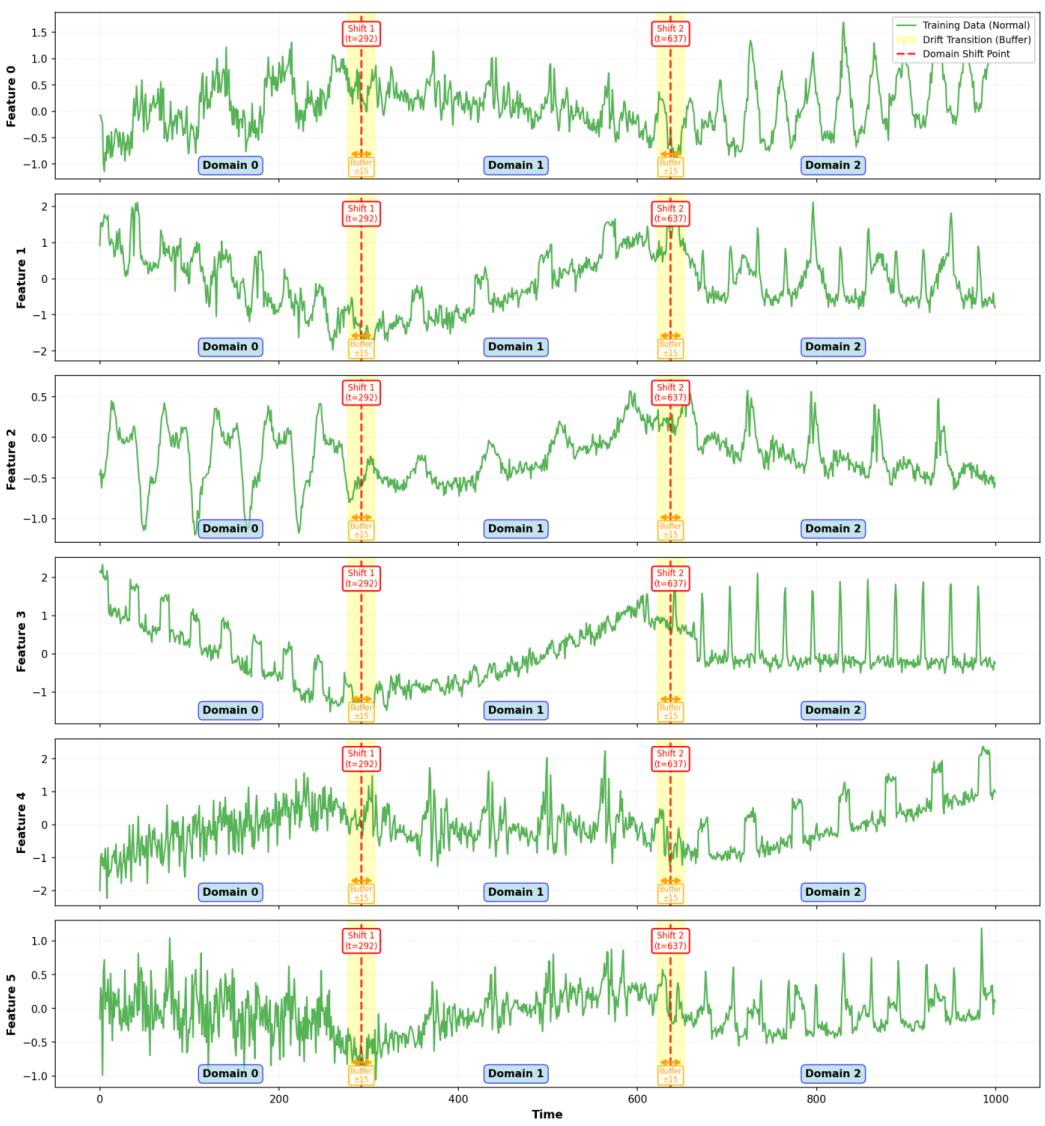


Figure 6: A representative BDS sequence. Amplitude and periodicity differ visibly between successive domains, while the shape of the underlying waveform is preserved throughout the stream.

We therefore report four complementary scores, all computed on binary decisions obtained with the fixed threshold of Appendix A.4 unless the metric is itself threshold-free.

- **Standard F1 (F1).** The harmonic mean of point-wise precision and recall, evaluated independently at each timestamp. It is reported without point adjustment, which is known to inflate scores substantially.
- **Temporal F1 (F1-T)** (Sarfraz et al., 2024). A range-aware variant that credits a prediction according to its overlap with a ground-truth anomaly segment, rewarding early and contiguous detections and penalizing fragmented ones.
- **Affiliation F1 (Affiliation-F)** (Huet et al., 2022). An event-level score built from the average directed distance between each predicted event and its affiliated ground-truth event. Because it is normalized against a random-prediction baseline, it remains informative when anomalies are extremely rare.
- **VUS-PR** (Paparrizos et al., 2022). A threshold-free score given by the volume under the surface traced by precision and recall as both the decision threshold and a temporal buffer width vary. It removes the dependence on any single threshold choice and tolerates small detection lags.

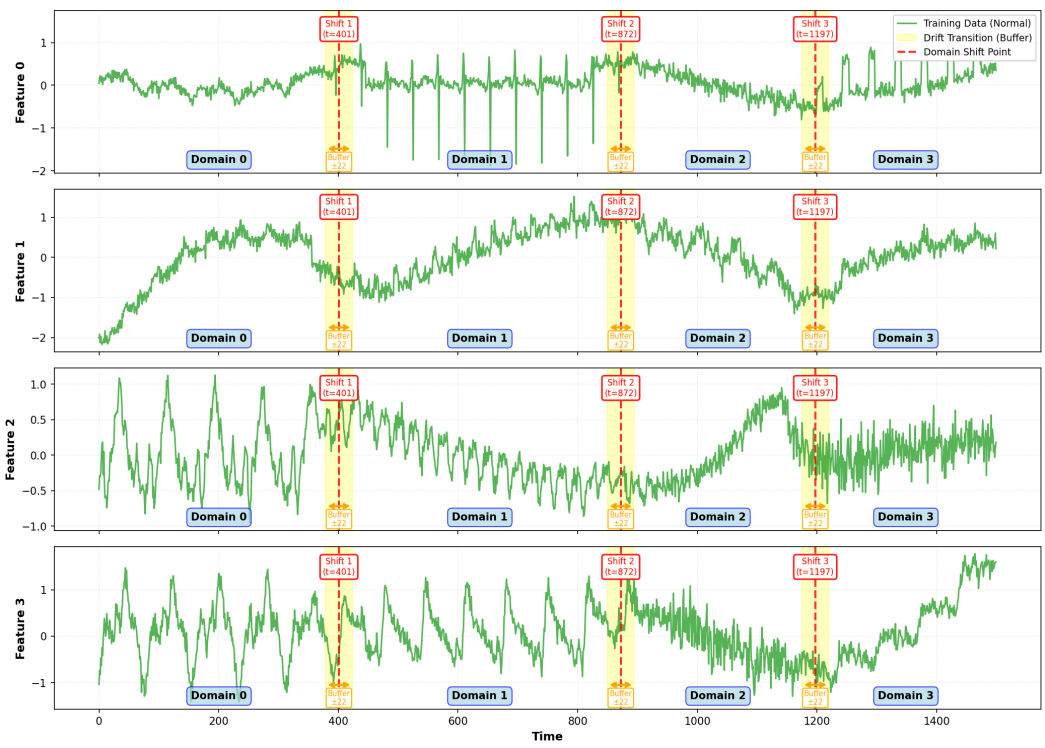


Figure 7: A representative MDS sequence. As the mixing coefficients follow the drift latent, the sequence transitions from a regular, seasonality-dominated regime to a volatile regime governed by the shared causal process.

Together these cover point-wise accuracy (F1), range consistency (F1-T), event-level affiliation (Affiliation-F), and threshold-independent ranking quality (VUS-PR). For every metric we report the mean over the sequences of a dataset, and the per-dataset values are given in Table 1.

C.5 Architecture and Optimization Settings

KRAFT is run with a single configuration on every dataset; no per-dataset tuning is performed. The reference model is fitted in the three historical stages of Section 4.4 — Koopman dynamics learning, bidirectional bridge learning, and conditional flow matching — each optimized with Adam under early stopping on a held-out historical split, with the best checkpoint restored before the next stage begins. The calibration subset is used only for threshold and scorer calibration, never for training or model selection. At deployment the backbone, the bridge, and the residual scorer are frozen, and the only trainable quantities are the rank-4 adapters inserted after each residual block of the velocity network, which the validation-guided procedure of Section 4.3 updates at most once per block of observations. Table 6 reports the complete architectural, optimization, detection, adaptation, and scoring settings.

D Additional Experimental Results

This section provides detailed results complementing the main-text evaluation, including a statistical comparison protocol, detection under domain shifts, component ablations, and hyperparameter sensitivity. A supplementary NASA battery case study is included to examine whether the learned Koopman representation captures predictable variation in hidden operating states.

D.1 Statistical Comparison Protocol and Public Benchmark Results

Comparisons by average score leave open whether a margin reflects a consistent advantage or a few favorable sequences. We therefore assess the main comparisons of this paper for statistical

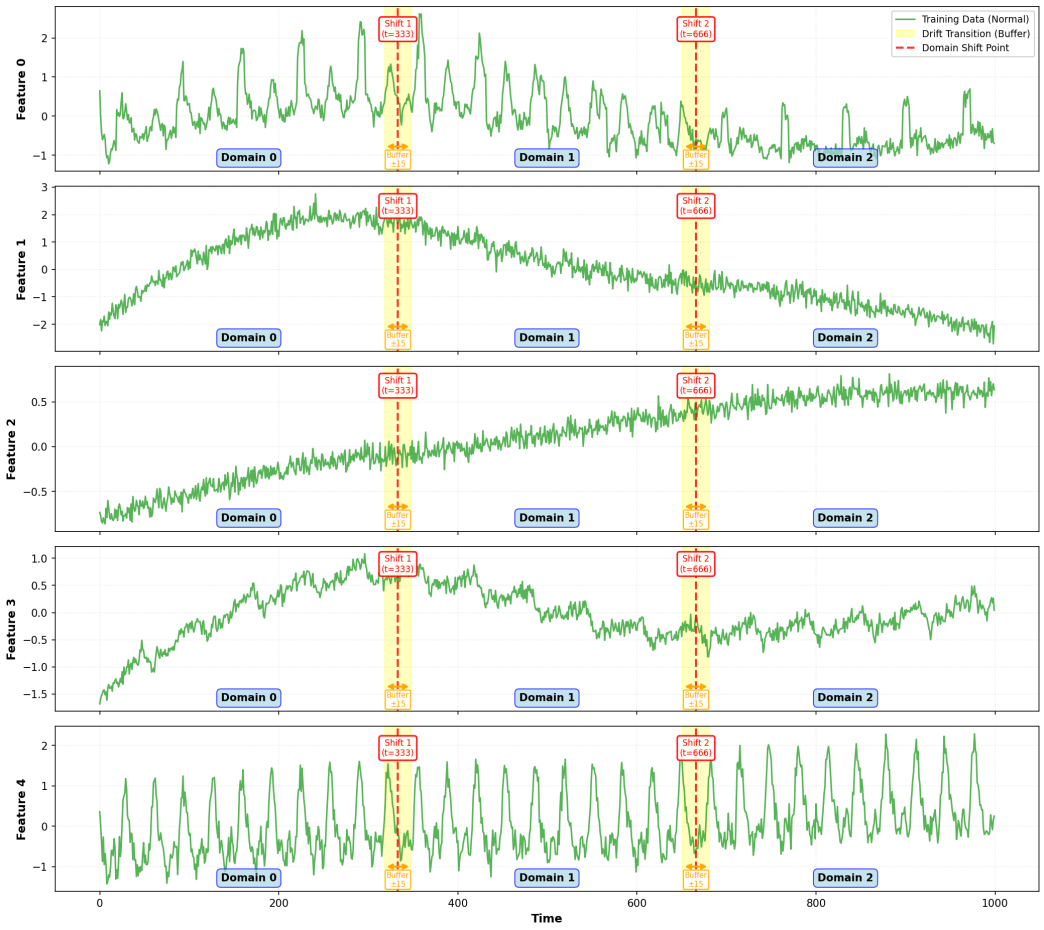


Figure 8: A representative CDS sequence. Because the structural coefficients themselves are interpolated across domains, the stream shows abrupt changes in temporal dependence and structural discontinuities instead of a gradual statistical drift.

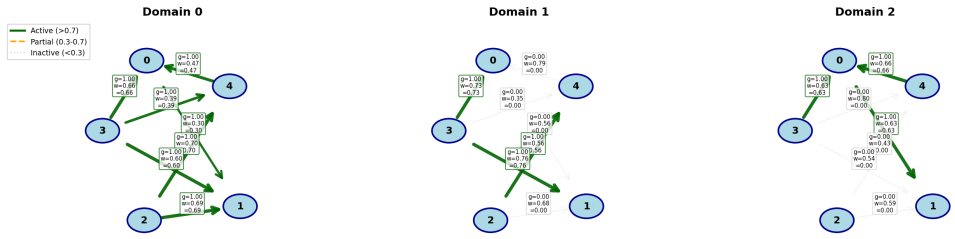


Figure 9: The directed acyclic graph of the latent causal process underlying Figure 8.

significance, using the following protocol throughout. One observation is a single dataset–metric pair. For each such pair the competing methods are ranked from best to worst, each method's rank is averaged over all observations, and every pair of methods is compared with a two-sided Wilcoxon signed-rank test on the raw scores at $\alpha = 0.10$. The results are summarized as critical difference diagrams in which a lower average rank is better, a bracket joins a group of methods that the test cannot separate, and a method drawn without any bracket differs significantly from other method. The protocol is applied to the public benchmark comparison below, to the domain-drift comparison in Appendix D.2, and to the ablation in Appendix D.3.

Figure 10 applies it to Table 1, aggregating the 16×4 entries into 64 paired observations per model pair. KRAFT attains the best average rank, 2.83 out of 12, and lies outside every bracket. All eleven of its pairwise comparisons against the baselines are significant, the weakest being against Sub-PCA

Table 6: Complete configuration of KRAFT. The same settings are used for all univariate, multivariate, and domain-drift benchmarks.

Parameter	Setting
Backbone and latent dynamics	
Input window length / TCN hidden width	64 / 64
TCN blocks / convolutions per block / kernel size	4 / 2 / 3
TCN dilation factors / receptive field	$1, 2, 4, 8$ / 61
TCN normalization / activation / dropout	Channelwise LayerNorm / GELU / 0.1
Positional embedding	None
Projection MLP hidden width / Koopman latent dimension	64 / 16
Auxiliary decoder hidden width / activation	64 / Tanh
Reconstruction weight λ_{rec} / Koopman weight λ_K	1 / 10^{-4}
Bridge and velocity networks	
Bridge hidden layers / width / activation	2 / 128 / SiLU
Direction embedding dimension	8
Expert variance bounds / bridge weight λ_{path}	$[10^{-4}, 10^{2}]$ / 1
Velocity residual blocks / hidden width	4 / 128
Flow-time embedding dimension / frequency range	32 / $[1, 10^{3}]$, log-spaced
Horizon embedding dimension / normalizer $H_{\max}$	16 / 32
Adapter rank / trainable parameters	4 / 4,608
Adapter up-projection / down-projection initialization	Zero / $\mathcal{N}(0, 128^{-1})$
Historical optimization	
Optimizer / batch size / weight decay	Adam / 128 / 10^{-5}
Stage 1–2 / Stage 3 learning rate	10^{-3} / 10^{-4}
Maximum epochs / patience / minimum improvement	50 / 10 / 10^{-5}
Resampled tuples per epoch / gradient clipping	20,000 / 5
Stage 1 horizon range	4–32
Stage 2 horizon set	$\{2, 4, 8, 16\}$, uniform
Stage 3 horizon set / proportions	$\{1, 2, 4, 8\}$ / $\{1/2, 1/6, 1/6, 1/6\}$
Offline flow-time range	$[0.02, 0.98]$
Stage 3 checkpoint weights (one-step / mean)	0.7 / 0.3
Shift detection and episode control	
KP-MMD window length n_r / induced propagation lag Δ	100 / 100
Subsampling cap $M_{\max}$ / bandwidth σ_k	128 / Median heuristic
Bounded-influence map g / scale β_g	$x/(x+\beta_g)$ / Median historical discrepancy
Evidence smoothing α / block quorum ρ_{block}	0.1 / 0.5
Entry / exit calibration quantile	0.99 / 0.90
Detection block size / persistence p	32 / 3 blocks
Minimum / maximum adaptation duration	4 / 64 blocks
Validation patience / cooldown	4 / 16 blocks
Testing adaptation	
Optimizer / learning rate / steps per active block	SGD / 0.03 / 1
Minibatch / training pool / validation holdout	32 / 512 / 128
Self-supervised loss weights / sample weights	1, 1 / Uniform
Testing flow-time range / gradient clipping	$[0.05, 0.95]$ / 1
Proximal weight λ_{reg} / acceptance margin γ	10^{-4} / 0.02
Inference and scoring	
Solver / reported / Testing integration steps	Heun / 16 / 4
Reference prediction horizon	1 step
Standardization floor / residual scale	10^{-8} / $1.4826\,\mathrm{MAD} + 10^{-8}$
Anomaly calibration quantile / deployment statistics	0.99 / Fixed

($p = 0.080$), and they remain jointly significant when the eleven KRAFT-versus-baseline tests are Holm-corrected as a single family.

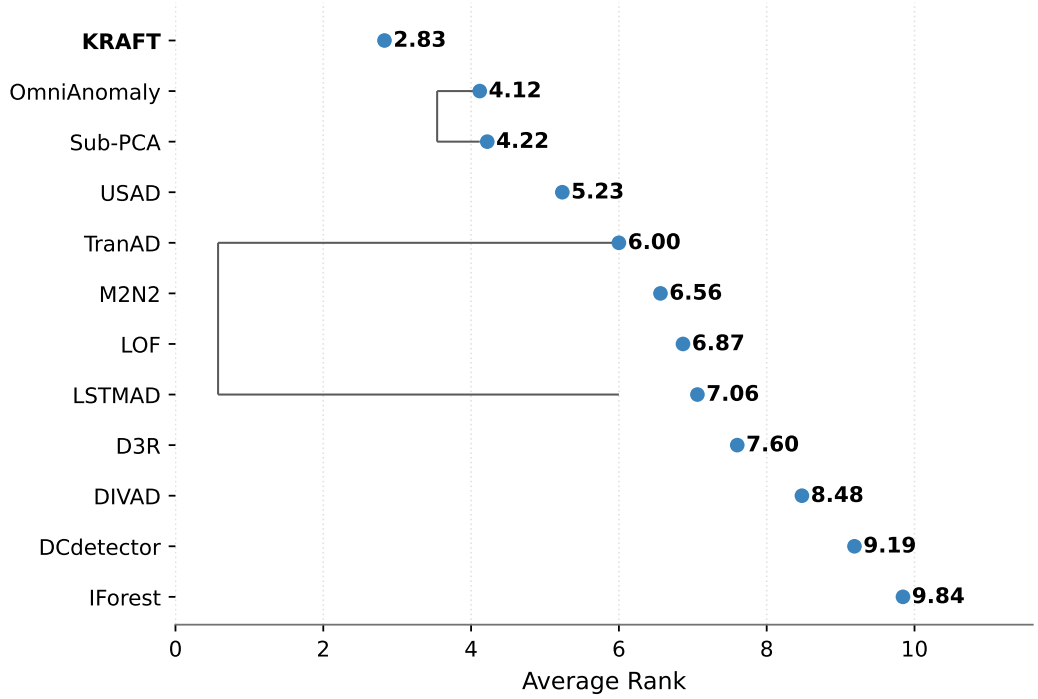


Figure 10: Critical difference diagram for the 16 public TSAD datasets of Table 1, over 64 paired dataset–metric observations per model pair.

D.2 Detailed Results under Latent Domain Shifts

Table 7 reports detection performance on BDS, MDS, CDS, and INSECTS, with means and standard deviations computed across sequences. KRAFT achieves the highest mean score in 15 of the 16 dataset–metric combinations, leading in Affiliation-F, Standard-F1, and VUS-PR on all four datasets. These results support the effectiveness of combining dynamics-conditioned reference generation with restricted adaptation under different forms of domain variation.

The exception is F1-T on MDS, where KRAFT performs below M2N2 and D3R despite achieving the highest scores on the other three metrics. This difference indicates that improvements in point-wise detection and range-aware ranking do not necessarily yield the best temporal event coverage. On INSECTS, KRAFT improves over the compared methods across all metrics, although the low absolute F1-T and Standard-F1 scores indicate that detection under real-world drift remains challenging.

Statistical significance. Figure 11 applies the protocol of Appendix D.1 to the 16 dataset–metric entries of Table 7. KRAFT reaches an average rank of 1.12 out of 4, and the diagram contains no bracket at all, so every pair of methods is separated: $p = 5.8 \times 10^{-4}$ against M2N2, $p = 6.1 \times 10^{-5}$ against D3R, and $p = 3.1 \times 10^{-5}$ against DIVAD.

Mechanism-level diagnostics. The aggregates above establish that KRAFT detects better under latent domain shift, but not that it does so for the reason the method claims. We therefore report a stream-level diagnostic that runs the deployed configuration under a prequential protocol in which every anomaly label and every domain label is withheld from the testing stage and attached only for offline analysis. Because this diagnostic calibrates its thresholds per stream, its absolute values are not directly comparable with Table 9; what it resolves is the placement in time of the adaptation, which no aggregate can show. The traces below are drawn from the INSECTS stream carrying recurring drift.

Figures 12 and 13 share a time axis and follow one chain: the observations depart from the distribution the model was trained on, the detector registers that departure, the controller opens episodes,

Table 7: Average detection performance (mean ± std over sequences) of KRAFT, M2N2, DIVAD, and D3R on the four drift datasets. Best result is in red, second-best is in blue.

Metric	Model	BDS	MDS	CDS	INSECTS
Affiliation-F	KRAFT	93.17 ± 12.65	94.37 ± 8.42	98.25 ± 1.44	67.86 ± 0.28
	M2N2	89.05 ± 11.56	92.41 ± 10.44	93.73 ± 1.47	59.22 ± 0.74
	DIVAD	85.78 ± 12.25	91.58 ± 11.38	92.01 ± 2.40	52.61 ± 1.88
	D3R	88.25 ± 12.10	91.80 ± 10.39	92.54 ± 2.65	54.70 ± 1.45
F1-T	KRAFT	67.70 ± 21.78	64.22 ± 21.24	56.63 ± 12.03	7.50 ± 1.49
	M2N2	58.42 ± 13.93	68.56 ± 13.03	52.98 ± 7.37	5.40 ± 1.26
	DIVAD	54.08 ± 13.82	62.48 ± 19.55	50.72 ± 9.37	4.69 ± 2.23
	D3R	55.12 ± 13.71	65.43 ± 16.20	51.22 ± 7.74	5.25 ± 1.16
Standard-F1	KRAFT	67.59 ± 26.85	58.36 ± 24.13	57.15 ± 6.77	4.54 ± 1.72
	M2N2	49.56 ± 25.12	57.55 ± 15.99	46.00 ± 5.00	3.42 ± 1.65
	DIVAD	44.54 ± 21.24	50.93 ± 19.45	43.37 ± 6.07	3.08 ± 1.41
	D3R	47.04 ± 24.00	52.70 ± 16.47	45.08 ± 5.41	3.21 ± 1.52
VUS-PR	KRAFT	69.59 ± 31.01	69.15 ± 28.11	65.60 ± 12.79	33.99 ± 3.47
	M2N2	55.13 ± 28.01	61.23 ± 26.21	48.41 ± 8.28	28.62 ± 3.86
	DIVAD	48.96 ± 21.69	58.59 ± 28.07	42.75 ± 14.85	24.10 ± 0.85
	D3R	52.77 ± 25.08	60.62 ± 27.30	46.74 ± 11.88	26.68 ± 1.77

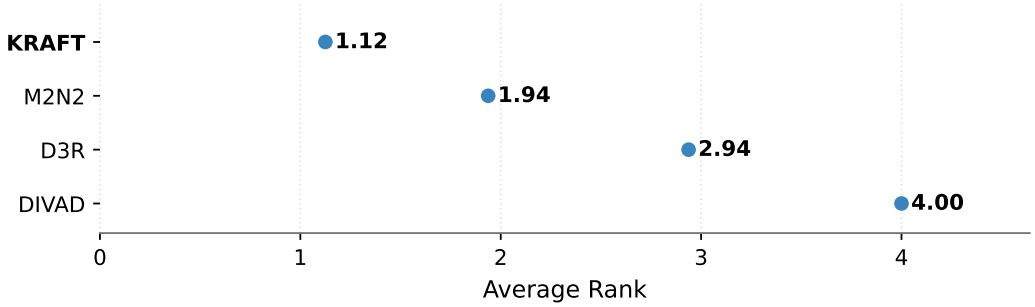


Figure 11: Critical difference diagram for the four domain-drift datasets of Table 7, over 16 paired dataset–metric observations per model pair.

and the adapter moves inside them. The excerpt covers a quiet stretch, the cluster of episodes that accounts for most of this run's adaptation, and a second quiet stretch.

The first link is established without reference to the model. The raw departure from the training window tracks the episode structure, with a median of 0.334 before the cluster, 0.443 within it and 0.223 after; averaged over the excerpt it reads 0.466 inside episodes against 0.308 outside them, a separation that survives a circular block permutation of the episode mask, which preserves its autocorrelation ($p = 0.049$, 2000 permutations). The INSECTS streams carry no per-point drift annotation, so this is a statement about measurable movement of the input distribution rather than about agreement with labelled boundaries. The detector recovers the same structure from the Koopman latent space: its evidence sits near 0.13 before the cluster, rises to roughly 0.35 at $t \approx 19{,}000$, holds a median of 0.227 until $t \approx 28{,}000$ and falls back to 0.114. The two curves agree closely over the excerpt, at a Spearman correlation of 0.956, and at 0.963 over the complete stream. The evidence of Eq. 8 is therefore not an artefact of the representation: it follows a change that is present in the raw sensor readings and can be measured without the encoder, the Koopman operator or the kernel that define it.

Figure 13 closes the chain. The displacement curve climbs within each shaded episode and is flat between them, and once the evidence falls back it is flat for the remainder of the excerpt. Adaptation is therefore neither continuous nor scheduled: it is placed by the detector on the parts of the stream where the input distribution has actually moved, which is the behavior Section 4.2 specifies.

Table 8 summarizes the same behavior over all twenty runs and quantifies the restriction that Section 4.3 places on it. Testing adaptation carries a failure mode specific to anomaly detection: an

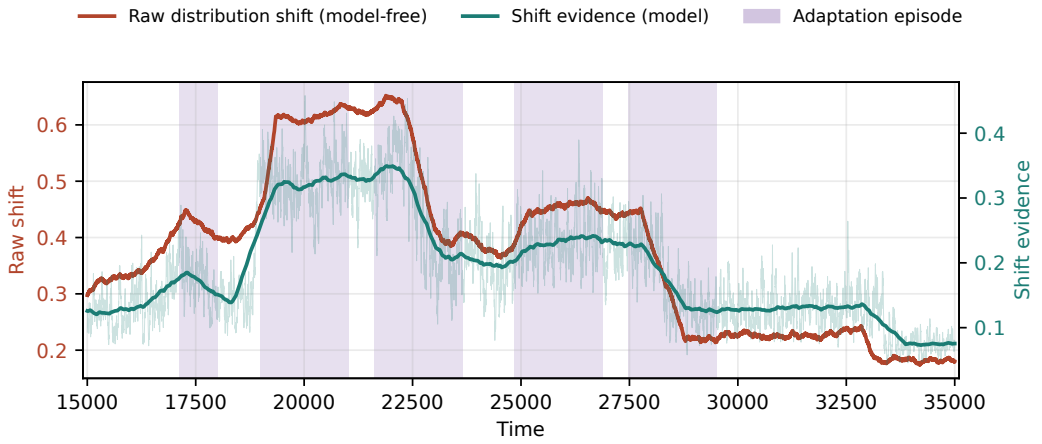


Figure 12: Movement of the input distribution and the detector's response to it, over an excerpt of the INSECTS recurring-drift stream. The left axis carries a model-free statistic: for each of the 33 sensor features the standardized mean is taken in a sliding window of 1000 steps and its absolute departure from the window the backbone was fitted on is averaged across features, using no labels and no component of the model. The right axis carries the evolution-compensated shift evidence of Eq. 8, shown unsmoothed in the background and under the same 1000-step average in the foreground. Shading marks the adaptation episodes the evidence opens. The two axes differ in scale; what the panel shows is that the curves rise and fall together.

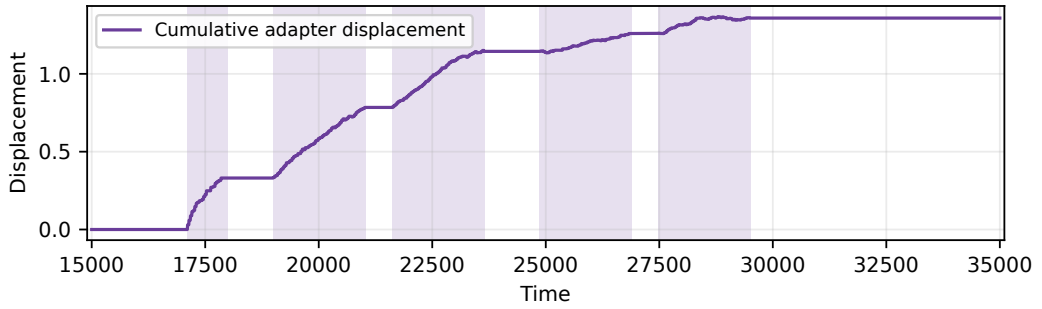


Figure 13: Cumulative displacement of the adapter over the same excerpt, on the same time axis and with the same episodes shaded.

adapter that refits whatever it currently observes will, on an anomalous window, learn to treat that window as normal and thereby remove the very event it is meant to expose. The validation criterion of Eq. 44 is what bounds this, and the last two columns show it doing so. Per observation, the adapter travels 0.87×10^{-5} inside annotated anomalous segments against 2.20×10^{-5} outside them, a factor of 2.5; among the sixteen of twenty runs on which the adapter moves at all, displacement is lower inside anomalous segments on fifteen (Wilcoxon signed rank, $p = 0.005$). Nothing in the controller has access to the labels that define these segments, so the asymmetry follows from the update rule rather than from a schedule. The effect is clearest on the two streams where adaptation is actually exercised, abrupt and incremental-recurrent, at 0.36 and 0.41 of the normal rate; on the incremental stream the controller adapts on only 6.2% of blocks and rejects nearly half of its candidates, so both of its displacement figures are small and their ratio is not informative.

What this restraint buys is visible on the constructed sequences, where the domain boundaries are known exactly. Figure 14 plots the fraction of normal observations flagged as anomalous within a sliding window of 64 steps; unlike an overall false-positive rate this quantity is local in time, so it reveals whether spurious alarms are spread evenly over a run or concentrated in particular stretches of it. They are concentrated. Within the first domain, before any shift has occurred, KRAFT and the *w/o adaptation* variant of Table 9 are indistinguishable (0.052 against 0.056 on average), which rules out the reading that adaptation simply lowers the alarm rate everywhere. After the first transition

Table 8: Behavior of the episode controller and the adapter on the four INSECTS streams, averaged over five seeds each. Displacement is reported per observation, since normal observations outnumber anomalous ones by roughly eighty to one and the raw totals would reflect that imbalance rather than the update rule.

		Blocks	Candidates		Displacement ($\times 10^{-5}$)	
Drift form	Episodes	adapting	Accept	Reject	Normal	Anomalous
Abrupt	5.4	18.5%	195.8	10.4	2.47	0.89
Incremental	8.4	6.2%	40.2	34.8	0.49	1.02
Incremental-gradual	1.0	3.2%	15.6	1.2	2.03	0.00
Incremental-recurrent	14.6	38.6%	613.6	38.0	3.81	1.57
All streams	7.4	16.6%	216.3	21.1	2.20	0.87

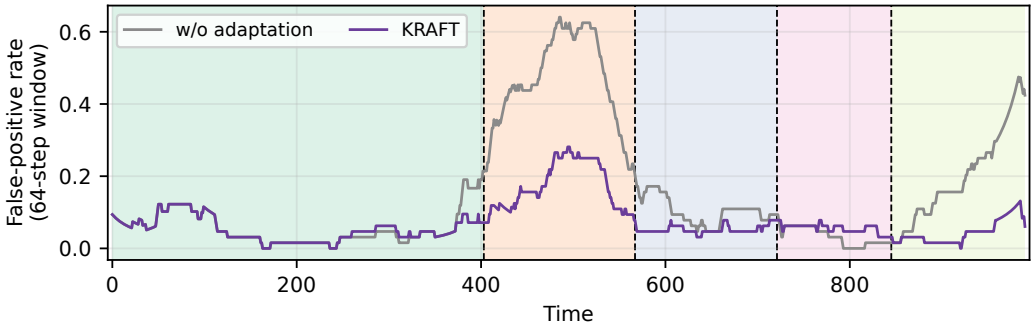


Figure 14: Rate at which normal observations are flagged, measured in a sliding window of 64 steps, on a representative CDS sequence with five latent domains. Shading marks the true domains and the dashed lines their boundaries, none of which is visible to either model.

the frozen variant flags 46.3% of normal observations on average and up to 64.1% at its peak, mistaking the new domain's ordinary behavior for anomalous roughly two times in three, while KRAFT averages 17.3% over the same domain. The pattern repeats at the last transition, 19.8% against 4.4%, and over the whole sequence the average is halved, from 14.9% to 7.1%. Taken with the preceding two figures, the diagnostic locates the gain reported in Table 7 at the domain transitions and attributes it to episodes the detector opened on its own.

D.3 Detailed Ablation Results

Ablated variants. All variants use the same encoder and flow-backbone architectures, historical data splits, and three-stage training schedule. Unless directly affected by the ablation, hyperparameters follow Appendix C.5. Input standardization is fitted on the historical fitting subset and shared across variants. Residual centers, scales, and anomaly thresholds are calibrated separately for each variant using the same historical calibration subset and calibration rules, and remain fixed during deployment. Each ablation changes one designated mechanism while retaining the remaining model structure and evaluation protocol. Historical training budgets are matched, and variants that perform adaptation use the same per-candidate optimization budget.

- **w/o Koopman conditioning.** The propagated latent state $\mathbf{z}_{i,\tau} = \exp(K\tau H)\mathbf{z}_i$ in the velocity-field conditioning vector of Eq. 6 is replaced by $\mathbf{z}_i$, yielding $(\mathbf{x}_i, \mathbf{h}_i, \mathbf{z}_i, \mathbf{z}_i, H)$. This substitution preserves the conditioning dimension and is applied consistently during flow training, reference generation, and candidate optimization. The Koopman generator, its training objective, and the bidirectional probability path of Eq. 4 are retained. Since the training path remains dynamics-informed, this variant specifically tests the contribution of explicit propagated-state conditioning to the velocity field, rather than removing all Koopman information from reference learning. Evolution compensation in the shift detector is unchanged.

Table 9: Ablation study of KRAFT on 20 datasets (16 public TSAD benchmarks and 4 domain-drift datasets, in %). Red and blue indicate the best and second-best score for each dataset and metric. The last two columns report each variant's average score for that metric and its rank among the five variants.

		Univariate datasets											Multivariate datasets					Domain-drift datasets				Summary	
Metric	Variant	IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO	MSL	PSM	SMAP	SMD	SWaT	BDS	MDS	CDS	INSECTS	Avg	Rank
Affiliation-F	KRAFT	92.34	70.02	88.74	97.76	70.42	69.88	94.75	85.53	77.38	93.37	94.94	89.38	81.72	89.62	92.69	77.01	93.17	94.37	98.25	67.86	85.96	1
	w/o Koopman conditioning	80.62	65.38	67.15	97.75	70.58	57.48	77.91	68.42	68.17	83.12	87.60	75.58	70.24	87.99	66.77	61.19	79.82	85.48	93.34	59.17	75.19	3
	w/o evolution compensation	77.48	62.81	76.32	79.34	71.79	61.75	80.39	67.08	68.75	89.31	81.69	69.31	77.26	72.93	78.88	71.54	78.99	73.46	99.57	48.32	74.35	4
	w/o adaptation	78.52	51.11	70.50	76.69	64.09	54.72	84.28	75.77	59.82	81.87	75.16	90.39	64.17	68.99	75.94	67.06	50.15	49.19	62.95	37.92	66.96	5
	w/o validation	85.43	60.64	78.55	84.08	66.02	66.48	72.26	82.04	68.10	74.91	96.76	82.74	73.96	73.88	93.05	77.10	92.53	69.09	84.85	52.54	76.75	2
F1-T	KRAFT	51.67	3.73	38.99	56.29	21.26	11.31	80.16	47.54	13.78	43.70	64.37	48.64	36.46	33.42	46.53	35.27	67.70	64.22	56.63	7.50	41.46	1
	w/o Koopman conditioning	42.78	3.00	29.79	43.58	20.58	9.28	72.91	44.18	13.42	31.62	59.27	39.83	31.57	26.11	41.34	31.05	59.06	51.58	29.28	5.87	34.30	4
	w/o evolution compensation	43.67	3.12	32.66	57.69	15.97	9.36	73.20	38.24	12.27	42.05	53.39	42.25	31.28	27.90	42.33	29.95	52.67	47.59	43.49	6.04	35.26	3
	w/o adaptation	35.47	3.62	39.29	47.65	16.59	9.26	59.02	33.19	11.62	31.31	46.89	39.70	28.38	21.69	38.67	23.44	43.07	41.68	41.96	7.35	30.99	5
	w/o validation	43.94	2.97	37.07	46.07	19.39	9.14	71.97	44.48	12.73	43.76	53.51	41.54	32.81	29.21	38.44	38.98	51.87	54.09	45.30	6.29	36.18	2
F1	KRAFT	40.95	2.99	30.91	61.93	21.12	10.79	80.55	38.57	13.10	43.07	62.96	33.45	27.25	27.69	46.31	36.04	67.59	58.36	57.15	4.54	38.27	1
	w/o Koopman conditioning	31.09	2.31	27.22	46.57	17.62	8.40	60.87	29.61	10.87	34.23	46.34	28.43	25.35	28.45	37.70	26.95	56.40	36.04	54.03	3.45	30.60	4
	w/o evolution compensation	39.86	2.27	25.66	61.67	16.57	9.37	72.78	34.76	10.86	41.22	51.64	32.49	20.58	21.25	39.23	27.43	47.88	48.07	41.51	3.29	32.42	3
	w/o adaptation	34.23	2.94	22.62	47.59	17.81	9.04	46.95	31.55	12.67	39.73	43.91	25.46	20.19	23.22	32.19	25.20	45.10	54.43	36.87	3.05	28.74	5
	w/o validation	33.73	2.78	34.45	63.13	18.69	8.39	79.89	32.54	13.37	43.83	61.39	25.85	24.91	26.26	45.46	32.96	52.56	48.36	48.64	2.98	35.01	2
VUS-PR	KRAFT	24.39	2.05	25.84	65.13	10.78	8.44	99.24	65.82	9.08	24.49	70.20	22.98	20.32	21.15	38.78	23.19	69.59	69.15	65.60	33.99	38.51	1
	w/o Koopman conditioning	20.87	1.87	26.53	56.49	7.43	7.32	99.06	55.90	9.00	22.81	67.12	20.86	21.35	16.49	28.31	17.84	39.39	58.97	46.17	24.84	32.43	4
	w/o evolution compensation	19.43	1.66	20.53	57.46	9.45	8.28	92.08	53.68	8.45	23.06	59.73	23.79	19.30	17.35	32.08	17.98	66.68	51.03	47.15	24.67	32.69	2
	w/o adaptation	16.97	2.05	24.31	57.53	8.43	7.43	73.37	53.65	6.78	23.70	73.18	17.57	16.33	16.29	30.61	19.87	43.73	32.67	51.14	23.24	29.94	5
	w/o validation	18.81	2.02	23.28	57.40	8.84	6.87	83.02	65.45	8.58	22.27	61.34	21.19	19.03	18.48	41.37	21.74	38.11	55.46	52.41	27.52	32.66	3

- **w/o evolution compensation.** The propagated reference distribution in Eq. 8 is replaced by the unpropagated reference distribution, giving $\mathrm{MMD}_k(\widehat{P}_c, \widehat{P}_r)$. Consequently, the discrepancy includes predictable latent evolution as well as changes unexplained by the normal dynamics. The bounded transformation g, smoothing coefficient α, hysteresis mechanism, and persistence requirement p are retained. Detector thresholds follow the same historical calibration procedure, applied to the ablated statistic. Reference generation still uses the propagated latent state, so this variant isolates evolution compensation in shift detection.

- **w/o adaptation.** Adapter parameters remain at their initialization throughout deployment; candidate optimization and validation-based acceptance are disabled. With U_ℓ and $\mathbf{b}_\ell$ initialized to zero, the velocity field remains identical to its pretrained form. The generated reference still varies with the observed history and Koopman-predicted state, but its parameters receive no deployment-time correction. The KP-MMD detector and fixed residual-scoring protocol are retained, although detected mismatch episodes do not induce parameter updates. This variant assesses the contribution of testing adaptation beyond the pretrained reference generator.

- **w/o validation.** The acceptance rule of Eq. 11 is bypassed, and every successfully generated candidate is committed, $\varphi_b^+ = \varphi_b^\star$, without evaluating its held-out capped risk. The same chronological buffer-splitting rule is retained: candidate optimization uses only $\mathcal{T}_b$, while $\mathcal{V}_b$ remains excluded from training. Triggering, the per-candidate optimization budget, the episode-anchor definition, proximal regularization, and adapter rank are unchanged. This variant isolates the contribution of held-out validation to screening candidate updates, while preserving the restrictions on candidate generation.

The four variants partition the method along two axes. The first two remove Koopman propagation, from reference generation and from shift detection respectively, while leaving the testing adaptation machinery intact; the last two remove the testing adaptation machinery, entirely and partially, while leaving reference generation intact.

Table 9 presents the dataset-level results of four ablations: replacing Koopman-predicted conditioning with the current latent state, removing evolution compensation from KP-MMD, disabling adapter updates, and removing candidate validation. The full model achieves the highest average score across the 20 datasets for all four metrics, while disabling adaptation produces the largest overall degradation.

The domain-drift results further clarify the roles of these components. Removing Koopman conditioning or evolution compensation weakens detection when the normal reference must account for evolving dynamics, whereas removing validation reduces the benefits of online adaptation. Although individual procedures occasionally improve a particular dataset–metric combination, the aggregate results support the complementary contributions of the proposed components.

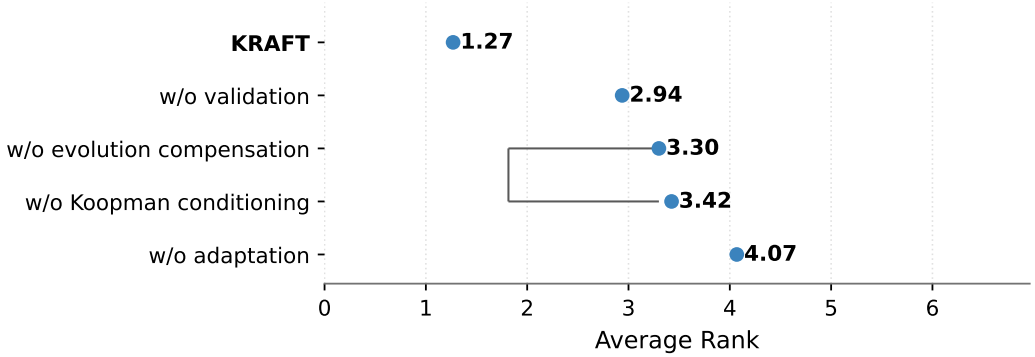


Figure 15: Critical difference diagram for the ablation of Table 9, over 80 paired dataset–metric observations per variant pair.

Statistical significance. Figure 15 pools the 20 datasets and four metrics of Table 9 into 80 paired observations per variant pair. The full model attains an average rank of 1.27 and is significantly better than all four variants, with $p < 10^{-11}$ throughout, so the isolated dataset–metric entries on which an ablation wins do not accumulate into an advantage. The ordering among the variants sharpens the reading above. Freezing the adapter is the most damaging intervention, at rank 4.07, and is significantly worse than every other variant, which identifies adaptation as the load-bearing component rather than an incremental refinement. Removing validation is the least damaging, at 2.94, yet still degrades the model significantly, so the acceptance rule contributes beyond simply enabling adaptation. Removing Koopman conditioning (3.42) and removing evolution compensation (3.30) are the only pair in this ablation that the test cannot separate ($p = 0.60$), and they are therefore drawn inside a common bracket. Both disable one half of the same mechanism, namely the use of a propagated reference in place of the current latent state, the first inside the generative path and the second inside the shift statistic; their near-equivalence indicates that the benefit is carried by the propagation step itself rather than by the place where it is applied.

D.4 Hyperparameter Sensitivity

Tables 10, 11, and 12 report the effects of the KP-MMD reference length n_r, persistence length p, and validation margin γ, respectively. The default setting of each parameter achieves the highest average score across the 20 datasets for all four metrics, although the best setting varies across individual datasets.

Reference window length. Table 10 evaluates $n_r \in \{25, 50, 100, 200, 400\}$. Performance generally improves toward the default $n_r = 100$ and decreases for substantially shorter or longer windows. This pattern is consistent with a trade-off between reliable distribution estimation and sensitivity to local changes: short windows provide limited samples, whereas long windows can mix observations from different operating conditions. Because the two windows are adjacent, varying n_r also varies the propagation lag $\Delta = n_r$, and for $n_r \in \{200, 400\}$ it activates the subsampling cap of Appendix A.2. The trend therefore reflects the combined effect of window size, propagation horizon, and estimator resolution rather than window size in isolation.

Persistence length. Table 11 evaluates $p \in \{1, 2, 3, 4, 6\}$. The default $p = 3$ provides the strongest average performance. Short persistence requirements can react to transient discrepancies, while long requirements can postpone adaptation to sustained changes. The results favor an intermediate level of temporal confirmation.

Validation margin. Table 12 evaluates $\gamma \in \{0, 0.01, 0.02, 0.04, 0.08\}$. The default $\gamma = 0.02$ achieves the best average performance. Smaller margins provide weaker evidence requirements for accepting candidates, whereas larger margins may reject useful updates with moderate validation gains. The observed pattern supports balancing update selectivity with adaptation responsiveness.

Table 10: Sensitivity of KRAFT to the KP-MMD reference window length n_r, on 20 datasets (16 public TSAD benchmarks and 4 domain-drift datasets, in %). Red/blue indicate the best/second-best score for each dataset and metric among the five settings; the default setting is in bold.

		Univariate datasets											Multivariate datasets					Domain-drift datasets				Summary	
Metric	Setting	IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO	MSL	PSM	SMAP	SMD	SWaT	BDS	MDS	CDS	INSECTS	Avg	Rank
Affiliation-F	25	80.45	46.27	75.53	83.21	59.73	74.19	86.50	74.10	61.92	69.81	88.17	75.86	62.89	73.28	72.72	55.69	79.88	77.78	82.18	59.15	71.97	5
	50	84.62	68.18	90.10	98.67	66.78	65.23	98.02	73.66	64.79	91.31	91.14	89.18	69.44	84.72	84.26	81.41	84.86	82.39	89.53	63.55	81.09	3
	100	92.34	70.02	88.74	97.76	70.42	69.88	94.75	85.53	77.38	93.37	94.94	89.38	81.72	89.62	92.69	77.01	93.17	94.37	98.25	67.86	85.96	1
	200	92.16	63.13	86.99	82.52	74.59	64.87	88.59	74.13	77.11	87.33	93.53	85.50	79.02	85.00	84.62	81.74	89.43	85.65	94.02	60.33	81.51	2
	400	84.36	64.13	76.46	89.69	68.66	49.82	89.57	80.52	64.23	77.73	97.67	75.78	71.25	80.87	87.21	69.75	80.03	80.29	87.72	52.97	76.44	4
F1-T	25	48.29	2.93	30.24	51.03	21.84	10.08	75.58	39.19	12.47	43.04	54.23	40.36	28.97	30.00	44.35	30.89	56.81	49.34	48.51	5.89	36.20	5
	50	45.58	3.68	35.02	49.57	18.81	9.89	83.44	46.64	13.69	38.80	63.11	45.03	32.53	32.19	40.74	31.10	62.94	60.30	48.84	6.46	38.42	3
	100	51.67	3.73	38.99	56.29	21.26	11.31	80.16	47.54	13.78	43.70	64.37	48.64	36.46	33.42	46.53	35.27	67.70	64.22	56.63	7.50	41.46	1
	200	50.53	3.47	36.10	47.87	20.58	11.22	71.42	44.94	13.71	42.76	59.53	42.28	33.14	32.89	44.56	31.42	63.96	61.75	55.14	6.56	38.69	2
	400	53.63	3.43	36.44	48.98	17.55	11.61	65.12	50.67	10.52	45.77	54.00	41.93	26.70	31.12	37.88	31.20	59.33	52.70	42.23	5.73	36.33	4
F1	25	32.61	2.35	27.89	63.55	16.97	9.39	55.54	33.34	11.29	36.56	52.37	24.22	27.17	27.76	34.60	28.88	48.91	48.91	40.10	3.30	31.29	5
	50	38.13	2.89	30.44	57.47	20.58	10.22	78.67	39.96	12.92	39.20	57.75	31.48	24.70	27.87	44.46	32.90	64.03	53.55	49.43	3.94	36.03	3
	100	40.95	2.99	30.91	61.93	21.12	10.79	80.55	38.57	13.10	43.07	62.96	33.45	27.25	27.69	46.31	36.04	67.59	58.36	57.15	4.54	38.27	1
	200	36.19	2.82	29.42	57.19	20.98	9.78	79.07	36.98	12.63	46.12	60.15	34.13	25.71	26.15	44.88	33.42	60.43	54.79	53.62	4.23	36.43	2
	400	37.10	2.83	28.63	53.36	19.98	10.83	73.17	34.34	11.59	35.99	57.51	22.14	21.44	23.84	36.31	33.46	49.85	50.79	48.37	3.93	32.77	4
VUS-PR	25	25.92	1.90	22.51	54.67	8.58	6.96	96.21	52.74	8.19	19.92	63.25	21.94	16.48	19.14	24.34	18.56	57.14	54.96	57.46	25.09	32.80	5
	50	21.97	1.82	23.01	58.87	10.67	8.72	96.34	64.10	8.72	23.50	72.15	19.05	19.57	19.01	34.84	22.69	64.82	61.84	60.76	30.41	36.14	2
	100	24.39	2.05	25.84	65.13	10.78	8.44	99.24	65.82	9.08	24.49	70.20	22.98	20.32	21.15	38.78	23.19	69.59	69.15	65.60	33.99	38.51	1
	200	23.61	2.00	25.21	63.38	9.88	8.05	93.89	60.37	8.19	20.52	65.01	21.35	19.09	19.77	36.17	20.91	63.23	65.44	61.41	29.47	35.85	3
	400	21.77	2.03	23.20	59.39	8.94	8.02	88.46	57.78	8.58	21.61	68.13	21.16	17.58	17.79	40.94	21.78	59.62	58.40	52.88	23.13	34.06	4

Table 11: Sensitivity of KRAFT to the persistent trigger length p, on 20 datasets (16 public TSAD benchmarks and 4 domain-drift datasets, in %). Red/blue indicate the best/second-best score for each dataset and metric among the five settings; the default setting is in bold.

		Univariate datasets											Multivariate datasets					Domain-drift datasets				Summary	
Metric	Setting	IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO	MSL	PSM	SMAP	SMD	SWaT	BDS	MDS	CDS	INSECTS	Avg	Rank
Affiliation-F	1	73.84	63.03	77.72	92.39	67.20	59.94	85.18	82.80	75.80	85.84	81.36	76.93	71.51	85.93	79.23	67.58	85.33	79.52	83.72	59.41	76.71	4
	2	91.32	61.13	84.42	92.65	72.23	64.96	93.42	81.33	76.17	93.91	89.87	90.91	81.36	88.40	90.21	72.59	85.79	92.04	86.73	61.97	82.57	2
	3	92.34	70.02	88.74	97.76	70.42	69.88	94.75	85.53	77.38	93.37	94.94	89.38	81.72	89.62	92.69	77.01	93.17	94.37	98.25	67.86	85.96	1
	4	85.75	64.02	92.90	90.37	67.12	68.90	89.36	82.79	72.96	90.35	88.29	83.09	75.78	86.10	87.24	73.24	88.36	97.87	94.85	65.25	82.23	3
	6	81.78	66.68	75.50	90.19	62.35	64.10	79.91	71.34	59.98	85.88	86.47	69.10	74.63	80.15	87.57	62.96	78.89	81.66	79.07	56.55	74.74	5
F1-T	1	44.86	3.37	35.63	53.43	19.35	9.50	77.93	41.76	13.00	43.28	57.66	46.20	34.05	33.68	40.34	31.75	64.08	52.49	52.63	5.78	38.04	4
	2	46.74	3.70	37.64	57.41	22.04	10.30	70.97	43.57	12.47	42.23	63.46	44.35	33.25	34.16	41.52	34.17	66.67	65.04	51.78	6.83	39.41	2
	3	51.67	3.73	38.99	56.29	21.26	11.31	80.16	47.54	13.78	43.70	64.37	48.64	36.46	33.42	46.53	35.27	67.70	64.22	56.63	7.50	41.46	1
	4	49.63	3.51	38.06	52.00	22.30	11.70	71.70	40.73	14.36	39.42	65.19	48.86	33.20	30.42	44.73	30.80	61.43	62.10	54.50	7.30	39.10	3
	6	41.98	3.36	32.09	47.68	19.52	8.47	61.80	43.06	13.21	36.18	56.92	42.14	33.74	26.21	36.86	28.25	50.14	48.46	49.51	6.11	34.28	5
F1	1	38.36	2.63	28.19	52.96	20.74	9.66	75.96	36.86	12.32	40.13	52.44	31.13	26.51	26.56	42.04	34.75	61.11	50.83	45.78	4.04	34.65	4
	2	36.24	3.06	29.06	57.78	18.78	10.37	74.61	34.80	11.63	41.76	60.41	31.60	26.37	26.50	44.74	33.98	66.12	52.64	53.45	4.59	35.92	3
	3	40.95	2.99	30.91	61.93	21.12	10.79	80.55	38.57	13.10	43.07	62.96	33.45	27.25	27.69	46.31	36.04	67.59	58.36	57.15	4.54	38.27	1
	4	38.91	3.00	30.76	60.35	20.66	10.24	79.60	39.61	12.77	37.67	61.51	31.15	24.29	26.50	45.68	33.91	60.33	56.16	57.28	4.25	36.73	2
	6	37.60	2.71	28.11	52.69	18.15	9.81	71.39	37.84	11.70	42.27	57.76	30.71	24.32	25.17	42.76	33.81	58.60	48.50	41.49	3.66	33.95	5
VUS-PR	1	21.05	1.78	25.07	62.55	9.15	7.06	94.20	64.96	8.63	20.35	62.63	21.67	19.35	19.46	36.12	23.78	60.81	59.80	52.39	27.92	34.94	4
	2	22.32	2.03	24.92	65.20	10.32	7.92	93.42	59.90	9.17	23.17	63.80	22.54	19.07	20.10	35.94	23.23	65.94	66.73	60.28	32.86	36.44	3
	3	24.39	2.05	25.84	65.13	10.78	8.44	99.24	65.82	9.08	24.49	70.20	22.98	20.32	21.15	38.78	23.19	69.59	69.15	65.60	33.99	38.51	1
	4	23.03	2.00	25.79	66.39	10.70	8.04	92.06	63.41	8.54	23.48	65.35	19.72	18.40	18.92	38.19	21.68	69.58	66.47	62.80	31.85	36.82	2
	6	20.75	1.62	21.43	59.10	9.82	6.47	88.59	51.48	7.51	21.63	59.65	19.28	20.62	18.83	34.50	20.75	57.19	52.31	56.51	27.12	32.76	5

D.5 Supplementary Study: Koopman Representation of Endogenous Shift

Motivation. Assumption 1 posits that part of the movement of normal behavior is endogenous and admits a compact description in a learned latent space. On TSAD benchmarks this cannot be inspected directly, because the latent state is by definition unobserved and no ground truth for it exists. We therefore run a diagnostic on a dataset in which the endogenous state *is* measurable: NASA battery aging data (Saha & Goebel, 2007), where each discharge cycle carries a capacity retention $H_t = C_t/C_0$ that reads out the degradation state directly. The study isolates two properties that Section 4.1 relies on, namely whether the learned latent state carries the endogenous shift variable in a form that transfers across devices, and whether the constrained generator parameterization behaves as its design intends. It evaluates the Koopman component on its own and involves no flow sampling, adapters, shift detection, or anomaly scoring.

Protocol. We use the four room-temperature cells B0005, B0006, B0007 and B0018 in a leave-one-battery-out design: the encoder, the Koopman generator and a linear probe are fitted on three cells and evaluated on the held-out fourth, over five seeds, giving 20 runs across four held-out-cell folds and five seeds. Each discharge cycle is encoded into a 16-dimensional latent state. Capacity is excluded from representation learning and is used only to fit the offline linear probe on training cells, evaluate it on the held-out cell, and color the visualizations.

The latent state carries the endogenous shift variable. Figure 16 projects the held-out latent trajectories onto their first two principal components and colours each point by its capacity reten-

Table 12: Sensitivity of KRAFT to the validation acceptance margin γ, on 20 datasets (16 public TSAD benchmarks and 4 domain-drift datasets, in %). Red/blue indicate the best/second-best score for each dataset and metric among the five settings; the default setting is in bold.

		Univariate datasets											Multivariate datasets					Domain-drift datasets				Summary	
Metric	Setting	IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO	MSL	PSM	SMAP	SMD	SWaT	BDS	MDS	CDS	INSECTS	Avg	Rank
Affiliation-F	0.0	89.36	61.56	80.40	91.87	64.60	64.54	94.18	73.65	70.82	87.36	87.01	78.84	67.22	88.77	86.58	67.90	92.06	83.05	89.02	59.95	78.94	4
	0.01	95.58	66.67	84.79	98.69	70.37	68.11	84.13	77.57	75.14	93.82	94.16	88.42	79.16	80.81	84.20	73.76	89.11	88.96	88.12	65.14	82.34	2
	0.02	92.34	70.02	88.74	97.76	70.42	69.88	94.75	85.53	77.38	93.37	94.94	89.38	81.72	89.62	92.69	77.01	93.17	94.37	98.25	67.86	85.96	1
	0.04	82.64	65.56	85.07	85.61	66.03	67.12	90.94	77.95	71.56	92.14	90.65	78.69	81.16	82.95	91.82	72.69	86.17	85.77	84.10	65.06	80.18	3
	0.08	83.67	65.11	75.63	86.47	50.06	64.31	90.01	76.14	60.17	86.04	84.11	84.59	66.55	71.96	79.70	73.52	75.72	81.43	80.61	54.39	74.51	5
F1-T	0.0	47.15	3.51	35.38	49.15	21.17	10.57	72.43	43.36	12.14	40.91	60.88	50.23	35.57	30.21	43.83	33.86	61.15	58.97	51.39	6.61	38.42	4
	0.01	49.61	3.55	38.99	53.47	20.93	10.44	77.32	45.73	13.26	42.31	63.75	45.72	35.30	31.50	42.94	33.10	64.62	62.19	55.66	7.12	39.88	2
	0.02	51.67	3.73	38.99	56.29	21.26	11.31	80.16	47.54	13.78	43.70	64.37	48.64	36.46	33.42	46.53	35.27	67.70	64.22	56.63	7.50	41.46	1
	0.04	49.76	3.72	38.66	49.53	20.97	11.20	80.85	46.53	13.07	41.50	59.23	44.34	34.10	30.69	44.38	29.86	62.81	58.94	51.48	7.15	38.94	3
	0.08	41.48	3.68	39.03	54.63	18.33	10.23	70.33	39.98	13.18	35.67	55.41	43.24	32.10	28.25	38.81	33.68	59.61	52.33	47.49	6.24	36.19	5
F1	0.0	36.22	2.65	26.06	53.91	20.50	10.19	76.77	33.00	12.86	41.45	57.78	31.21	24.91	25.15	37.87	29.62	61.96	51.44	52.57	3.86	34.50	4
	0.01	40.10	2.94	30.08	58.84	21.16	10.17	75.77	38.19	12.41	40.82	63.19	31.96	26.82	29.78	43.38	34.87	65.50	55.73	53.32	4.33	36.97	2
	0.02	40.95	2.99	30.91	61.93	21.12	10.79	80.55	38.57	13.10	43.07	62.96	33.45	27.25	27.69	46.31	36.04	67.59	58.36	57.15	4.54	38.27	1
	0.04	39.19	2.62	30.34	60.97	20.00	9.61	72.64	36.16	11.59	43.65	60.36	30.99	26.25	26.15	44.56	32.16	60.22	54.91	54.41	4.14	36.05	3
	0.08	33.84	2.70	29.54	54.88	17.71	9.33	74.32	32.37	10.47	42.28	51.05	31.98	24.53	22.17	39.54	31.84	59.15	45.83	47.36	3.88	33.24	5
VUS-PR	0.0	21.27	1.98	22.97	56.49	9.81	7.32	92.23	57.27	7.63	20.52	65.59	22.54	17.24	20.44	36.83	19.39	59.20	59.90	57.70	30.23	34.33	4
	0.01	23.61	2.06	24.28	66.76	10.41	8.70	99.97	62.82	8.90	24.15	69.23	22.35	20.02	20.02	37.58	22.91	62.06	68.90	62.89	30.40	37.40	2
	0.02	24.39	2.05	25.84	65.13	10.78	8.44	99.24	65.82	9.08	24.49	70.20	22.98	20.32	21.15	38.78	23.19	69.59	69.15	65.60	33.99	38.51	1
	0.04	21.91	1.94	24.09	63.44	10.58	8.72	98.80	61.19	9.21	23.33	73.94	22.54	20.39	18.58	37.00	20.29	61.75	63.88	56.35	32.51	36.52	3
	0.08	20.84	1.65	22.51	54.67	9.23	7.29	76.46	57.28	8.26	22.83	59.73	18.13	16.56	18.42	34.95	21.21	62.55	60.37	52.19	28.89	32.70	5

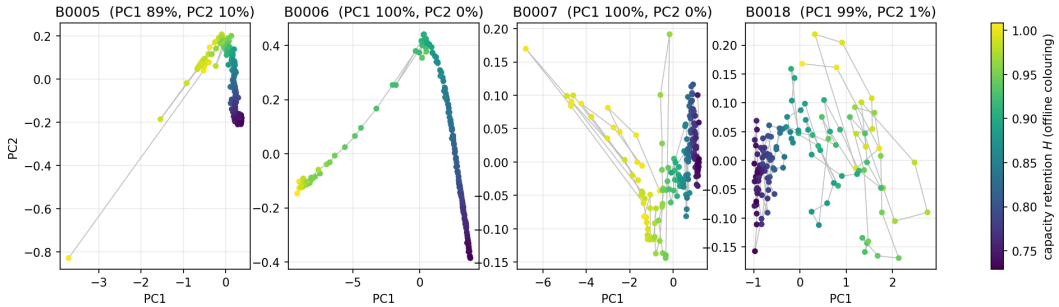


Figure 16: Principal-component projection of the held-out latent trajectories, one panel per left-out cell. Colour encodes capacity retention and is applied offline, after training. The first component explains 89%–100% of the variance and orders each cell monotonically by degradation state.

tion. The first component absorbs 89%–100% of the variance depending on the cell, and the colour gradient is monotone along it, so the learned latent space is effectively one-dimensional and that dimension is the degradation state. A linear probe fitted on the training cells and applied frozen to the unseen cell attains $R^2 = 0.777$ and a Spearman correlation of 0.984, with per-cell R^2 between 0.703 and 0.812, so the recovery is not carried by a single favorable cell. The slow state is thus linearly readable from $\mathbf{z}$ and the reading transfers to a device never seen during training, which is the property that Section 4.1 invokes when it conditions reference generation on $\mathbf{z}_i$ without observing the domain label.

The structure is induced by the temporal objective. Table 13 shows that this structure follows from learning a transition operator over correctly ordered cycles rather than from the encoder alone. Retraining with the cycle order shuffled, which leaves the architecture and the optimization budget unchanged, collapses the probe from $R^2 = 0.777$ to 0.106 and the Spearman correlation from 0.984 to 0.638. Replacing the constrained Koopman generator with a TCN-based latent transition model or an unconstrained linear transition map also degrades the probe, to 0.701 and 0.333, respectively. The shuffled-order control supports the contribution of temporal ordering to the learned representation.

The stability parameterization behaves as designed. Figure 17 shows the spectra of the learned generators. All 320 eigenvalues produced by $K = (A - A^\top) - \mathrm{diag}(\mathrm{softplus}(\mathbf{b}_K))$ across the 20 runs lie strictly in the left half-plane, with a global maximum real part of -0.041 and a spectral radius $\rho(\exp K) \in [0.912, 0.960]$; removing the constraint places positive real parts in all 20 runs, with $\rho(\exp K)$ up to 1.081. The operational consequence matters more than the spectra themselves. Contractivity bounds the propagated state uniformly in the elapsed time, so a reference built on it can lose informativeness but cannot blow up. Under the unconstrained parameterization the propagated state diverges instead, which for a residual-based detector would inflate scores indiscriminately and

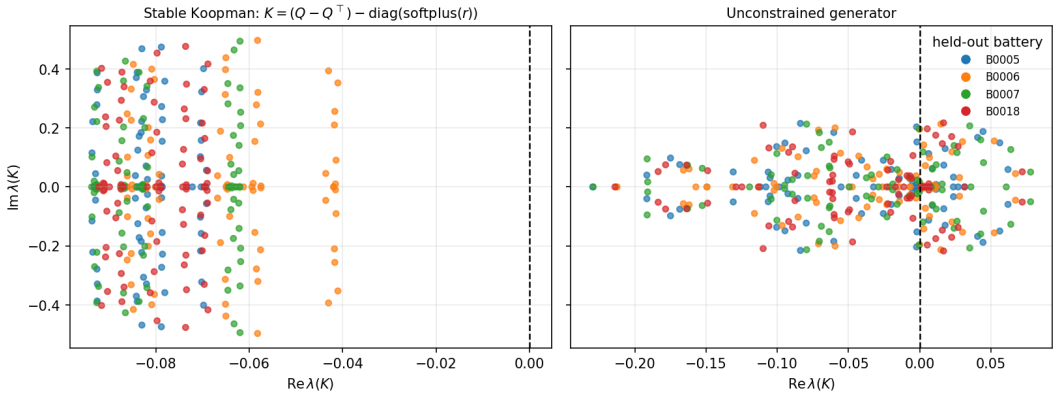


Figure 17: Eigenvalues of the learned Koopman generator in the complex plane, pooled over 4 folds, 5 seeds and 16 latent dimensions. Left: the constrained parameterization used by KRAFT, with every eigenvalue strictly left of the dashed stability boundary. Right: the same model without the constraint.

Table 13: Frozen linear probe on the held-out cell (mean over 4 folds $\times$ 5 seeds) and spectral diagnostics of the learned generator. The probe never sees the test cell during training and capacity labels never enter the encoder.

Transition model	Probe R^2	Spearman	$\max \mathrm{Re}\,\lambda(K)$	$\rho(\exp K)$
Constrained Koopman (ours)	0.777	0.984	-0.041	$[0.912, 0.960]$
TCN-based latent transition	0.701	0.975	–	–
Unconstrained linear	0.333	0.914	$+0.078$	$[1.009, 1.081]$
Shuffled-order Koopman	0.106	0.638	–	–

manufacture false alarms. The constraint is therefore not a regularity preference but a precondition for using propagation inside a detection pipeline at all.

Implications for the propagation used by KRAFT. The two properties established above map onto the two places where KRAFT relies on Koopman propagation. In reference generation, the propagated state enters Eq. 6 as one component of the conditioning vector $\mathbf{c}_{i,\tau,H}$ supplied to a trained velocity field, alongside the observation, the history embedding and the current latent state. What this asks of the propagation is that the coordinate it acts on carry the endogenous state in a stable and transferable form, which is exactly what the probe results of Table 13 exhibit on a dataset where that state is measurable. In shift detection, the propagated reference enters Eq. 8 through a kernel mean embedding over a window of n_r samples, so the statistic depends on the aggregate location of the pushed reference cloud.

Two further properties make this use robust. Contractivity, verified across all 20 runs in Figure 17, bounds the pushed reference uniformly in the elapsed time, so the statistic remains well behaved at any propagation lag. And because the map is frozen, it is applied identically when the entry and exit thresholds are calibrated on historical data and when the statistic is evaluated on the deployment stream; the thresholds are therefore quantiles of the statistic as it is actually computed, and the calibration of the detector rests on the consistency of the map rather than on its exact specification. The realized contribution of the compensation term at the operating point of KRAFT is then measured directly on the target task, by the *w/o evolution compensation* variant of Table 9.

D.6 Industrial Pump Case Study under Coexisting Drift Sources

Dataset and motivation. We additionally evaluate KRAFT on three fluorine-pump test sequences collected from Huawei industrial equipment, denoted as `error1`, `error2`, and `error3`. The test records document operating conditions and equipment measurements, including flow rate, rotational speed, wear, cavitation status, sampling information, and ambient temperature. The equipment undergoes aging-related evolution, while deliberate changes in ambient temperature introduce different

Table 14: Anomaly detection performance on the industrial fluorine-pump sequences (%). Bold indicates the best result for each sequence and metric.

Sequence	Method	Aff-F	Standard-F1	VUS-PR
error1	M2N2	87.21	83.82	87.94
	DIVAD	90.13	87.57	90.82
	D3R	93.38	90.68	94.11
	KRAFT	**98.62**	**98.58**	**99.61**
error2	M2N2	85.76	82.63	86.51
	DIVAD	91.72	89.34	92.16
	D3R	91.24	89.71	93.03
	KRAFT	**98.61**	**98.47**	**99.51**
error3	M2N2	84.07	79.83	84.62
	DIVAD	88.63	85.14	89.35
	D3R	90.79	86.40	91.52
	KRAFT	**98.64**	**98.02**	**99.37**

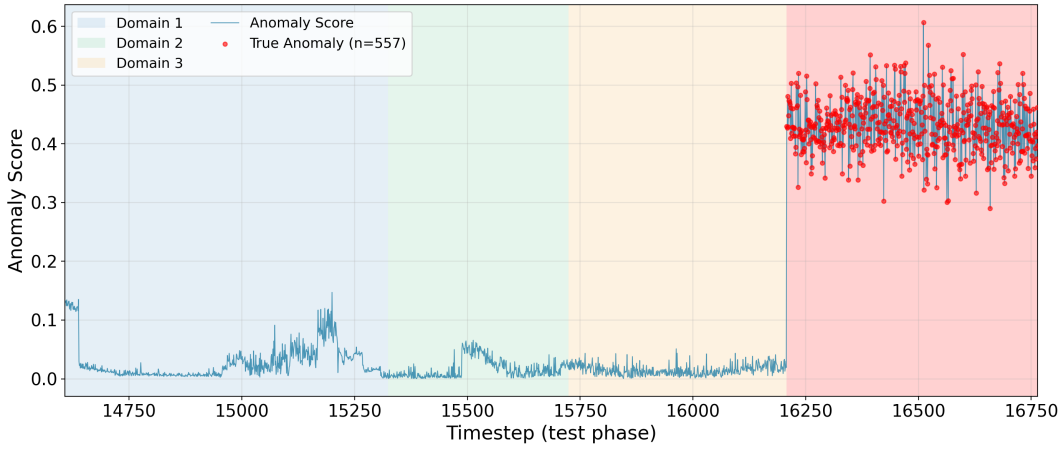
operating domains. This setting therefore provides an industrial case study of coexisting endogenous evolution and exogenous domain shifts.

The two sources of variation correspond to the modeling motivation of KRAFT: Koopman-conditioned reference construction represents predictable normal evolution (endogenous shifts), while evolution-compensated detection and restricted adaptation address changes not explained by the learned propagation (exogenous shifts). The experiment evaluates detection in this combined setting. Temperature-domain annotations and ground-truth anomaly labels are used only for evaluation and visualization, not for model fitting, shift triggering, or adaptation.

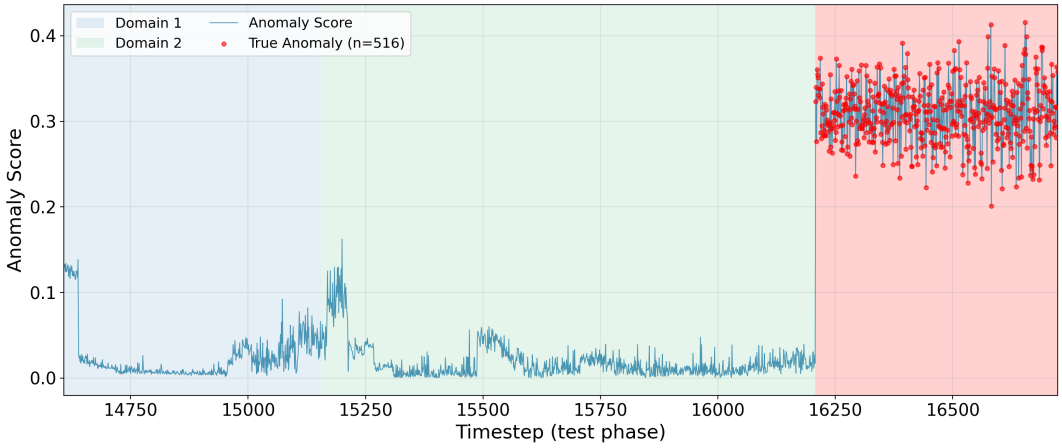
Evaluation protocol. All methods use the same chronological splits. KRAFT follows the fitting and calibration protocol of Appendix C.1 and the configuration of Appendix C.5.

Detection performance. Table 14 compares KRAFT with M2N2, DIVAD, and D3R using Aff-F, Standard-F1, and VUS-PR. KRAFT achieves the highest scores across all three sequences and metrics, supporting its effectiveness in industrial settings with coexisting endogenous and exogenous shifts.

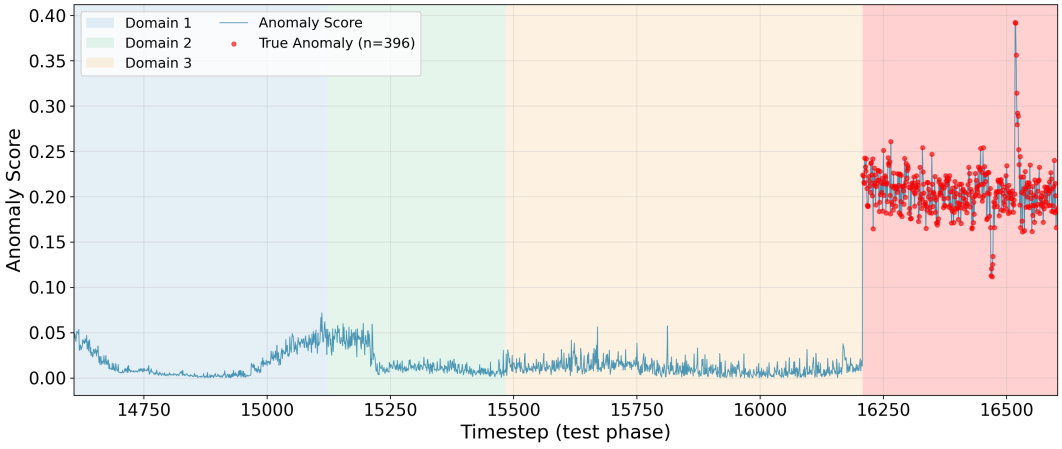
Score trajectories. Figure 18 visualizes KRAFT's anomaly scores together with temperature-domain annotations and ground-truth anomalies. Scores remain comparatively low across the displayed normal operating domains and become persistently elevated within the anomalous intervals. Although the score magnitude varies across sequences, the separation between normal and anomalous intervals is consistent with the quantitative results. These observations support the applicability of KRAFT to industrial data containing both aging-related evolution and changes in operating conditions.



(a) `error1`



(b) `error2`



(c) `error3`

Figure 18: KRAFT anomaly scores on three industrial fluorine-pump test sequences. Blue curves show anomaly scores, and red markers identify ground-truth anomalous samples. Blue, green, and yellow backgrounds distinguish the displayed temperature domains, whereas red shading denotes the anomalous interval rather than an additional domain.