SPARCE: SPARSE COUNTERFACTUAL ATTRIBUTION VIA GRADIENT-GUIDED DIFFUSION FOR TIME-SERIES ROOT CAUSE ANALYSIS

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce a novel approach to root cause analysis (RCA) in multivariate time-series anomaly detection, the task of identifying which features are responsible for a fault, not merely that one occurred. Our key observation is that identifying the sensors responsible for an anomaly can be framed as finding the minimal sparse edit to an anomalous window that restores it to the normal data manifold. Existing attribution methods rely on marginal feature perturbations or finite nearest-neighbor retrieval. Both produce out-of-distribution counterfactuals that violate the conditional dependencies inherent in real monitoring systems. We propose SPARCE (SParse Attribution via Reverse diffusion Counterfactual Explanation), which approximates this minimal-edit problem by injecting gradient-based guidance into the reverse process of a conditional score-based diffusion model trained exclusively on normal system behaviour. At each denoising step, the guidance jointly optimizes a denoising-consistency regularizer that steers the restoration toward normal behaviour and L1 sparsity penalty that encourages concentrated sensor edits. The root-cause attribution map emerges as the pointwise difference between the anomalous input and the diffusion-restored output, providing a jointly coherent explanation without per-sensor inference passes. SPARCE attributes observable deviations, which under propagation include downstream sensors, and is model-agnostic with respect to the anomaly detection backbone. We give sufficient conditions under which the attribution readout separates the deviating sensors. Experiments on SWaT, WADI, MSDS, and SMD demonstrate that SPARCE achieves state-of-the-art performance across Top-K Recall, CW-RCS, and TemporalHM metrics, improving Top-3 Recall by 13.2% over the strongest baseline on SWaT.

1 INTRODUCTION

Root cause analysis in multivariate time-series systems demands more than detecting an anomaly; it requires identifying which sensors are responsible for it, when, and by how much. In safety-critical environments such as industrial control systems, autonomous infrastructure, and distributed computing platforms, this distinction separates actionable diagnostics from uninformative alerts. Despite significant progress in anomaly detection, reliable automated RCA remains an open problem.

Our central idea is *joint sparse restoration: explain an anomalous window through edits that reconcile its sensor trajectories with learned normal behaviour, while penalizing how much the observed window is changed.* If the observable deviation is confined to one sensor, replacing its trajectory with the latent normal counterpart restores the normal window. We formulate this as probabilistic inference under a learned normal prior and an ℓ_1 edit penalty, corresponding to a Laplace prior on deviations, so that the restored window is a sparse counterfactual of the anomalous one. The normal prior captures temporal and cross-sensor dependencies, while the edit penalty favors concentrated corrections. This gives a concrete attribution objective: balance compatibility with normal behaviour against edit cost, and rank sensors by the resulting edit magnitudes. SPARCE targets observable deviations, which under propagation include downstream sensors as well as the initiating one; recovery of the initiating fault is tested separately on synthetic data with known intervention targets.

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.

Existing methods do not directly target this objective. Feature attribution approaches such as SHAP (Lundberg & Lee, 2017) and Integrated Gradients (Sundararajan et al., 2017) explain predictions through feature perturbations or integration along baseline-to-input paths. In physical monitoring systems, where strong inter-sensor dependencies are the rule, marginal perturbations can produce out-of-distribution counterfactuals that violate learned operating constraints and yield unreliable attributions (Mishra et al., 2026). Retrieval-based conditional methods (Mishra et al., 2026) address this by grounding counterfactuals in a reference set of normal windows, improving dependency preservation. However, finite reference coverage limits the restorations they can represent, and the per-sensor approaches considered here assess sensors independently rather than identifying a joint set of responsible sensors. Causal discovery methods (Ikram et al., 2022; Han et al., 2025) impose structural assumptions that can be difficult to satisfy in high-dimensional nonlinear systems and face scaling challenges as the sensor count grows.

We propose **SPARCE** (**SP**arse **A**ttribution via **R**everse diffusion **C**ounterfactual **E**xplanation), which approximately targets this objective by repurposing the reverse process of a score-based diffusion model as a sparse restoration procedure. SPARCE trains a conditional diffusion model only on normal system behaviour, learning an implicit model of the normal data distribution. At inference, given an anomalous window, it injects gradient-based guidance into the reverse process. The guidance combines a self-supervised denoising-consistency regularizer S_θ, computed by the diffusion model itself, with an ℓ_1 penalty that keeps the restoration close to the anomalous window and encourages edits to concentrate on few sensors. The root-cause attribution map is the pointwise absolute difference between the anomalous input and the average of the restored windows, attributing all sensors jointly without per-sensor inference passes. Two design choices are critical in practice: normalizing the guidance gradient, without which Top-3 Recall on SWaT collapses from 0.608 to 0.044, and restricting guidance to the high-noise half of the reverse process.

We evaluate SPARCE on SWaT (Goh et al., 2016), WADI (Ahmed et al., 2017), MSDS (Nedelkoski et al., 2020), and SMD (Su et al., 2019), comparing against feature attribution methods KernelSHAP (Lundberg & Lee, 2017), ShaTS (Franco De La Peña et al., 2026), causal discovery methods (Ikram et al., 2022; Han et al., 2025), conditional attribution (Mishra et al., 2026), and diffusion baselines adapted from CSDI (Tashiro et al., 2021) and WaveStitch (Shankar et al., 2025). We report Top-K Recall, CW-RCS, and TemporalHM (Mishra et al., 2026). SPARCE establishes state-of-the-art root cause attribution on all four benchmarks, with the best Top-3 Recall and the best CW-RCS on every dataset, improving Top-3 Recall by 13.2% over the strongest baseline on SWaT, and five-run comparisons show the gains are significant on every non-tied entry. Beyond ranking metrics, most restorations pass the external detector and concentrate their edits on few sensors. On a controlled Structural-VAR benchmark with known intervention targets, the same readout recovers injected roots with 0.865 Root Recall@1 against 0.417 for deviation ranking.

Our contributions are:

- **Problem formulation.** We formulate RCA as probabilistic restoration under a learned normal prior and an ℓ_1 edit penalty, with sensor attributions derived from joint edits and an explicit distinction between observable deviations and initiating faults.
- **Method.** SPARCE: a gradient-guided, sparse, reverse diffusion framework that approximately targets this objective at inference time without per-sensor enumeration, external anomaly classifiers, or task-specific retraining.
- **Design choices.** We identify gradient normalization and the guidance cutoff as critical choices for diffusion-guided attribution, with ablations isolating their impact.
- **Evidence.** State-of-the-art results on SWaT, WADI, MSDS, and SMD with per-entry confidence intervals and significance tests, direct empirical assessment of restoration validity and edit concentration, and a controlled benchmark with known intervention targets.

2 Related Work

Feature attribution. SHAP (Lundberg & Lee, 2017), Integrated Gradients (Sundararajan et al., 2017), ShaTS (Franco De La Peña et al., 2026) and TimeSHAP (Bento et al., 2021) estimate feature importance through marginal perturbations or path integrals that assume independence or weak dependence. Multivariate time-series violate this. Strong temporal and cross-sensor dependencies

make perturbed samples fall out of distribution, so attributions reflect artifacts of the perturbation rather than the system's response. TimeSHAP adds temporal structure but stays within a marginal intervention framework, leaving joint dynamics out of reach.

Reconstruction and retrieval. OmniAnomaly (Su et al., 2019), InterFusion (Li et al., 2021) and AERCA (Han et al., 2025) learn latent representations of normal behaviour and attribute anomalies from reconstruction error or latent deviation, while CondAttr (Mishra et al., 2026) explains anomalies through nearest-neighbour retrieval of comparable historical windows. Such attributions are bounded by the reference set; rare or novel anomalies have no suitable neighbours, and in high dimension the combinatorial space of normal behaviour cannot be covered exhaustively.

Diffusion models. CSDI (Tashiro et al., 2021) and DDPM adaptations (Ho et al., 2020) made diffusion effective for time-series imputation and forecasting, and guidance steers generation toward desired properties, from classifier guidance (Dhariwal & Nichol, 2021) to posterior sampling for inverse problems (Chung et al., 2022) and guidance on clean estimates (He et al., 2024). This machinery has largely served generation rather than attribution. SPARCE casts RCA as guided restoration under a learned normal prior, so that it produces joint counterfactuals rather than per-sensor perturbations.

3 Problem Formulation

3.1 Setup and deviation model

Let $\boldsymbol{x}_t \in \mathbb{R}^K$ contain K sensor readings standardized using the normal training split. A length-L window is

$$\boldsymbol{W}_t := [\boldsymbol{x}_t, \boldsymbol{x}_{t+1}, \ldots, \boldsymbol{x}_{t+L-1}] \in \mathbb{R}^{K\times L}, \tag{1}$$

with sensors as rows and time as columns. We use the entry-wise norm $\|\boldsymbol{B}\|_1 = \sum_{j,\ell} |\boldsymbol{B}_{j,\ell}|$ and the Frobenius norm $\|\boldsymbol{B}\|_F$.

Event annotations and a detector $f : \mathbb{R}^{K\times L} \to \mathbb{R}$ with threshold δ select attribution windows (Appendix C), where an event is a contiguous labelled anomaly spanning M windows. A restoration $\widetilde{\boldsymbol{Z}}$ is detector-valid when $f(\widetilde{\boldsymbol{Z}}) \le \delta$; the attribution objective excludes f, making attribution agnostic to the upstream detector. Let p_{N} denote the learned normal prior, assumed to have a proper density, from a diffusion model trained on normal windows (Section 4.1) and used without observed-sensor conditioning (Section 4.3).

We model an anomalous window as a latent normal window plus a deviation affecting few sensors,

$$\boldsymbol{W}_{\mathrm{anom}} = \boldsymbol{W}^{\mathrm{N}} + \boldsymbol{D}^{\star}, \qquad \mathbb{J}(\boldsymbol{D}^{\star}) = \{j : \|\boldsymbol{D}^{\star}_{j,:}\|_2 > 0\}, \tag{2}$$

with $|\mathbb{J}(\boldsymbol{D}^{\star})| = s \ll K$. We seek $\mathbb{J}(\boldsymbol{D}^{\star})$ through rankings of restoration edit magnitude. Independence is imposed only in the working inference model below.

Scope. The support describes observable deviations, including downstream effects, so propagation can violate its sparsity assumption (Appendix A). Our real-data labels likewise include direct and downstream effects, consistent with this scope though not identical to $\mathbb{J}(\boldsymbol{D}^{\star})$. Initiating interventions are evaluated separately on a synthetic benchmark with intervention-target labels (Section 5).

3.2 Sparse restoration as probabilistic inference

The working model pairs a candidate normal window drawn from p_{N} with an independent edit having an entrywise Laplace prior:

$$\pi_{\lambda_m}(\boldsymbol{D}) = \Big(\frac{\lambda_m}{2}\Big)^{KL} \exp\big(-\lambda_m \|\boldsymbol{D}\|_1\big), \qquad \lambda_m > 0. \tag{3}$$

This working prior has no point mass at zero, imposes no channel budget, and is not a distributional assumption on $\boldsymbol{D}^{\star}$. For $\boldsymbol{Z} = \boldsymbol{W}_{\mathrm{anom}} - \boldsymbol{D}$, the unit absolute Jacobian determinant gives the base posterior

$$p(\boldsymbol{Z} \mid \boldsymbol{W}_{\mathrm{anom}}) = \frac{p_{\mathrm{N}}(\boldsymbol{Z})\, \pi_{\lambda_m}(\boldsymbol{W}_{\mathrm{anom}} - \boldsymbol{Z})}{\int p_{\mathrm{N}}(\boldsymbol{Z}')\, \pi_{\lambda_m}(\boldsymbol{W}_{\mathrm{anom}} - \boldsymbol{Z}')\, d\boldsymbol{Z}'}. \tag{4}$$

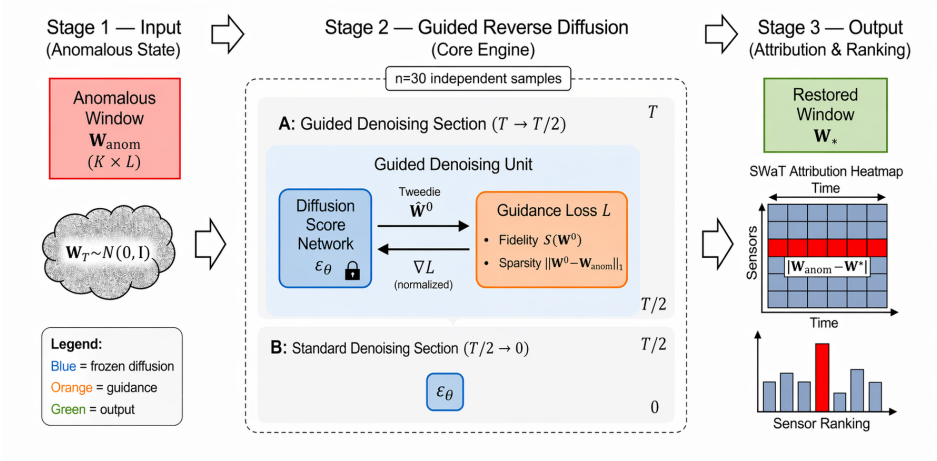


Figure 1: SPARCE framework overview. Stage 1: Gaussian noise initializes the reverse process, while $\boldsymbol{W}_{\text{anom}}$ enters through the edit penalty. Stage 2: high-noise steps use normalized guidance from the fidelity term $S(\widehat{\boldsymbol{W}}_0)$ and the ℓ_1 edit penalty; later steps use standard denoising. Stage 3: the $n = 30$ restorations are averaged to obtain $\boldsymbol{W}^\star$, and $|\boldsymbol{W}_{\text{anom}} - \boldsymbol{W}^\star|$ yields the attribution map and sensor ranking.

Its MAP restoration defines the base objective for a sparsity-promoting counterfactual:

$$\widehat{\boldsymbol{Z}}_{\text{MAP}} \in \underset{\boldsymbol{Z} \in \mathbb{R}^{K \times L}}{\arg\min} \Big\{ -\log p_{\text{N}}(\boldsymbol{Z}) + \lambda_m \|\boldsymbol{W}_{\text{anom}} - \boldsymbol{Z}\|_1 \Big\}. \tag{5}$$

Any restoration $\widetilde{\boldsymbol{Z}}$ induces the attribution map and sensor scores

$$\boldsymbol{\Phi}_{j,\ell}(\widetilde{\boldsymbol{Z}}) = \big|(\boldsymbol{W}_{\text{anom}})_{j,\ell} - \widetilde{\boldsymbol{Z}}_{j,\ell}\big|, \qquad a_j(\widetilde{\boldsymbol{Z}}) = \frac{1}{L} \sum_{\ell=1}^{L} \boldsymbol{\Phi}_{j,\ell}(\widetilde{\boldsymbol{Z}}). \tag{6}$$

Sensors are ranked by decreasing a_j. SPARCE uses the average $\widetilde{\boldsymbol{Z}} = \boldsymbol{W}^\star$ of n algorithmic restorations, not an assumed posterior mode (Section 4.4).

Well-posedness and sparsity. The posterior is proper because the Laplace density is bounded and everywhere positive. If p_{N} is also continuous and bounded, the MAP exists, with $-\log 0 = +\infty$ (Appendix A). The entrywise ℓ_1 penalty promotes sparsity but does not enforce sparse sensor support. MAP edits may contain zeros; base-posterior edit draws and finite averages of independent draws almost surely have no zero entries, by absolute continuity. This statement excludes the population posterior mean. SPARCE's average has no exact-sparsity guarantee; we evaluate edit concentration.

3.3 From the objective to an algorithm

Section 4.2 introduces the denoising-consistency regularizer S_θ, yielding

$$q(\boldsymbol{Z} \mid \boldsymbol{W}_{\text{anom}}) \propto p(\boldsymbol{Z} \mid \boldsymbol{W}_{\text{anom}}) \exp\big(-\lambda_s S_\theta(\boldsymbol{Z})\big), \qquad \lambda_s \geq 0. \tag{7}$$

The regularizer adds $\lambda_s S_\theta$ to the mode objective, modifying the inference target in equation 5. Under the conditions in Appendix A, q is proper and attains a mode. The regularized objective also admits a conditional support-separation guarantee: Appendix A gives sufficient conditions on curvature, prior accuracy, regularization strength, optimization error and minimum deviation magnitude under which approximate minimizers rank the sensors in $\mathbb{J}(\boldsymbol{D}^\star)$ above all others according to equation 6.

Section 4.3 derives practical guidance from an exact identity for the noisy target scores of equation 7, using learned scores, a plug-in approximation and a heuristic schedule. Section 5 evaluates the resulting restorations directly through detector validity, edit cost and edit concentration.

4 METHOD

SPARCE approximates the target equation 7 with guidance injected into the reverse process of a diffusion model trained on normal windows; Figure 1 gives an overview.

4.1 DIFFUSION MODEL FOR NORMAL BEHAVIOUR

We train a conditional noise-prediction model $\boldsymbol{\epsilon}_\theta$ on normal training windows with the standard forward kernel $\boldsymbol{W}_k = \sqrt{\bar{\alpha}_k}\boldsymbol{W}_0 + \sqrt{1-\bar{\alpha}_k}\boldsymbol{\epsilon}$, where $\mathrm{vec}(\boldsymbol{\epsilon}) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_{KL})$ and $\bar{\alpha}_k = \prod_{i \le k}(1-\beta_i)$ for a fixed schedule (Ho et al., 2020). We use conditional denoising (Tashiro et al., 2021) with complete sensor channels as hidden targets: for a hidden set $\Omega \subseteq \{1, \dots, K\}$, between one and three sensors in our runs (Appendix B), the mask $c_\Omega \in \{0,1\}^{K \times L}$ marks visible entries, $h_\Omega = \mathbf{1} - c_\Omega$, $\boldsymbol{W}_{\mathrm{obs}} = c_\Omega \odot \boldsymbol{W}_0$, and the loss is taken over hidden channels,

$$\mathcal{L}_{\mathrm{diff}} = \mathbb{E}_{\boldsymbol{W}_0, \boldsymbol{\epsilon}, k, \Omega}\, d_\Omega^{-1} \big\| h_\Omega \odot \big(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{W}_k, k; \boldsymbol{W}_{\mathrm{obs}}, c_\Omega)\big)\big\|_F^2, \tag{8}$$

with the reduction denominator d_Ω as implemented (Appendix B). Table 3 compares this choice with random cell masking.

Attribution queries the model in the all-hidden mode, abbreviating $\boldsymbol{\epsilon}_\theta(\boldsymbol{W}, k; \mathbf{0}, \mathbf{0})$ by $\boldsymbol{\epsilon}_\theta(\boldsymbol{W}, k)$. Writing p_{N} for the model's unguided output density in that mode and $p_{\mathrm{N},k}$ for its forward-noised marginal,

$$\nabla_{\boldsymbol{W}} \log p_{\mathrm{N},k}(\boldsymbol{W}) \approx -\frac{\boldsymbol{\epsilon}_\theta(\boldsymbol{W}, k)}{\sqrt{1-\bar{\alpha}_k}}, \qquad \widehat{\boldsymbol{W}}_0(\boldsymbol{W}, k) := \frac{\boldsymbol{W} - \sqrt{1-\bar{\alpha}_k}\,\boldsymbol{\epsilon}_\theta(\boldsymbol{W}, k)}{\sqrt{\bar{\alpha}_k}}. \tag{9}$$

The definition of $\widehat{\boldsymbol{W}}_0$ is exact. Its conditional-mean interpretation and the score relation are approximate; the all-hidden mode used at attribution is not directly trained (Appendix B).

4.2 SELF-SUPERVISED NORMALITY REGULARIZER

No external detector is queried during restoration: the model scores its own candidates at a fixed level $k' = T/5$,

$$S_\theta(\boldsymbol{Z}) = \mathbb{E}_{\boldsymbol{\epsilon}} \big\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\big(\sqrt{\bar{\alpha}_{k'}}\boldsymbol{Z} + \sqrt{1-\bar{\alpha}_{k'}}\boldsymbol{\epsilon},\, k'\big)\big\|_F^2 \ge 0, \tag{10}$$

in the same all-hidden mode, with parameters frozen and input gradients flowing through the denoiser. Guidance differentiates a Monte Carlo estimate $\widehat{S}_\theta$ from n_ε draws held fixed within each differentiation. S_θ is denoising-consistency regularization, not the detector f and not a log-likelihood, although a positively weighted S_θ matches one variational-bound term up to a constant under the conditions in Appendix A. Low S_θ alone does not certify normality; Appendix E reports its ROC-AUC and its rank correlation with f.

4.3 GRADIENT-GUIDED REVERSE DIFFUSION

Attribution starts from $\mathrm{vec}(\boldsymbol{W}_T) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_{KL})$ and keeps $c = \mathbf{0}$ throughout, so $\boldsymbol{W}_{\mathrm{anom}}$ is not supplied as observed conditioning and enters only through the cost

$$C(\boldsymbol{Z}) = \lambda_s S_\theta(\boldsymbol{Z}) + \lambda_m \|\boldsymbol{W}_{\mathrm{anom}} - \boldsymbol{Z}\|_1, \qquad \lambda_s \ge 0,\ \lambda_m > 0, \tag{11}$$

whose weights multiply sums, making equation 7 read $q(\boldsymbol{Z} \mid \boldsymbol{W}_{\mathrm{anom}}) \propto p_{\mathrm{N}}(\boldsymbol{Z}) e^{-C(\boldsymbol{Z})}$.

Target score and its approximation. Let q_k be the target pushed through the forward kernel and $\Psi_k(\boldsymbol{W}) = \mathbb{E}_{p_{\mathrm{N}}}[e^{-C(\boldsymbol{W}_0)} \mid \boldsymbol{W}_k = \boldsymbol{W}]$. Factoring $q_k \propto p_{\mathrm{N},k}\Psi_k$ gives, for $k \ge 1$ and under conditions permitting differentiation under the integral,

$$\nabla_{\boldsymbol{W}} \log q_k(\boldsymbol{W} \mid \boldsymbol{W}_{\mathrm{anom}}) = \nabla_{\boldsymbol{W}} \log p_{\mathrm{N},k}(\boldsymbol{W}) + \nabla_{\boldsymbol{W}} \log \Psi_k(\boldsymbol{W}), \tag{12}$$

the decomposition of Chung et al. (2022) with e^{-C} as pseudo-likelihood (Appendix A). The first term is estimated by equation 9. For the second we use the clean-estimate plug-in of DPS, evaluated on the stopped computation $\widehat{\boldsymbol{W}}_0^{\mathrm{sg}}$, in which $\mathrm{sg}[\boldsymbol{\epsilon}_\theta]$ keeps the value and zeroes the derivative:

$$\begin{aligned} \nabla_{\boldsymbol{W}} \log \Psi_k(\boldsymbol{W}) &\approx -\nabla_{\boldsymbol{W}} C\big(\widehat{\boldsymbol{W}}_0(\boldsymbol{W}, k)\big), \\ \widetilde{\nabla}_{\boldsymbol{W}} C\big(\widehat{\boldsymbol{W}}_0^{\mathrm{sg}}(\boldsymbol{W}, k)\big) &= \frac{1}{\sqrt{\bar{\alpha}_k}} \nabla_{\boldsymbol{Z}} C(\boldsymbol{Z})\big|_{\boldsymbol{Z} = \widehat{\boldsymbol{W}}_0(\boldsymbol{W}, k)}, \end{aligned} \tag{13}$$

Algorithm 1 SPARCE attribution for one anomalous window, with guidance cost $\widehat{C}(\boldsymbol{Z}) = \lambda_s \widehat{S}_\theta(\boldsymbol{Z}) + \lambda_m \|\boldsymbol{W}_{\text{anom}} - \boldsymbol{Z}\|_1$ evaluated from n_ε fixed scoring draws.

Require: $\boldsymbol{W}_{\text{anom}}$; frozen $\boldsymbol{\epsilon}_\theta$; $\lambda_s, \lambda_m, \eta, T, k', n, n_\varepsilon$; normalization norm
Ensure: sensor scores $a_1, \ldots, a_K$
for $i = 1, \ldots, n$ independently **do**
  $\mathrm{vec}(\boldsymbol{W}_T) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_{KL})$, $c = \mathbf{0}$ throughout
  for $k = T, \ldots, 1$ **do**
    $\hat{\boldsymbol{\epsilon}}_k \leftarrow \boldsymbol{\epsilon}_\theta(\boldsymbol{W}_k, k)$, $\widetilde{\boldsymbol{W}}_k \leftarrow \boldsymbol{W}_k$
    if $k \geq \lceil T/2 \rceil$ **then**
      $\widehat{\boldsymbol{W}}_0 \leftarrow (\boldsymbol{W}_k - \sqrt{1-\bar{\alpha}_k}\,\mathrm{sg}[\hat{\boldsymbol{\epsilon}}_k])/\sqrt{\bar{\alpha}_k}$, $\widetilde{\boldsymbol{W}}_k \leftarrow \boldsymbol{W}_k - \eta\,\mathrm{norm}\left(\widetilde{\nabla}_{\boldsymbol{W}_k} \widehat{C}(\widehat{\boldsymbol{W}}_0)\right)$
    end if
    $\boldsymbol{W}_{k-1} \leftarrow \mathrm{DDPMReverse}(\widetilde{\boldsymbol{W}}_k, \hat{\boldsymbol{\epsilon}}_k, k)$
  end for
end for
$\boldsymbol{W}^\star \leftarrow n^{-1} \sum_i \boldsymbol{W}_0^{(i)}$, $a_j \leftarrow L^{-1} \sum_\ell \left|(\boldsymbol{W}_{\text{anom}})_{j,\ell} - \boldsymbol{W}^\star_{j,\ell}\right|$

where $\widetilde{\nabla}$ denotes differentiation through the stopped computation, using subgradients at nondifferentiable points. The second line is the chain rule for that computation, not for the full gradient in the first: dropping the denoiser Jacobian is a second approximation, and it does not detach the denoiser inside S_θ. He et al. (2024) also guide on clean estimates, without assuming their manifold guarantees.

Guided step. With $\widehat{C}$ the Monte Carlo cost and $\mathrm{norm}(\cdot)$ the implemented normalization, each step $k \geq \lceil T/2 \rceil$ applies

$$\begin{aligned} \boldsymbol{G}_k &= \nabla_{\boldsymbol{Z}} \widehat{C}(\boldsymbol{Z})\big|_{\boldsymbol{Z} = \widehat{\boldsymbol{W}}_0(\boldsymbol{W}_k, k)}, \qquad \hat{\boldsymbol{\epsilon}}_k = \boldsymbol{\epsilon}_\theta(\boldsymbol{W}_k, k), \\ \widetilde{\boldsymbol{W}}_k &= \boldsymbol{W}_k - \eta\,\mathrm{norm}\Big(\boldsymbol{G}_k/\sqrt{\bar{\alpha}_k}\Big), \qquad \boldsymbol{W}_{k-1} = \mathrm{DDPMReverse}\big(\widetilde{\boldsymbol{W}}_k, \hat{\boldsymbol{\epsilon}}_k, k\big), \end{aligned} \tag{14}$$

reusing the prediction $\hat{\boldsymbol{\epsilon}}_k$ computed before the nudge; later steps apply the same reverse update without a nudge (Algorithm 1).

Normalization and cutoff. For nonzero gradients, exact per-window Frobenius normalization fixes the nudge at length η and removes the $1/\sqrt{\bar{\alpha}_k}$ factor of equation 13, leaving relative sensor components untouched; at fixed η its removal collapses attribution (Table 3). For fixed draws the guided trajectory then depends on (λ_s, λ_m) only through their ratio, though q depends on both. The correction in equation 12 is generally nonzero below $\lceil T/2 \rceil$, so the cutoff is a design choice and later states still carry earlier guidance.

4.4 Attribution readout and aggregation

We average the n restorations before differencing, $\boldsymbol{W}^\star = n^{-1} \sum_{i=1}^n \boldsymbol{W}^{\star(i)}$, and apply the readout equation 6 with $\widetilde{\boldsymbol{Z}} = \boldsymbol{W}^\star$. Averaging restorations is not averaging absolute edits, and opposing edits can cancel; $\boldsymbol{W}^\star$ is a restored window, not an established mode or mean of q, so detector validity is measured on $\boldsymbol{W}^\star$ itself. Over the M windows of an event, with $\boldsymbol{a}^{(m)}$ the per-window scores,

$$\boldsymbol{r} = \mathrm{zscore}(\bar{\boldsymbol{a}} + \boldsymbol{a}_{90}) + \tfrac{1}{2}\,\mathrm{zscore}(\boldsymbol{v}), \qquad v_j = \sum_{m=1}^{M} \mathbf{1}\{j \in \mathrm{Top}_3(\boldsymbol{a}^{(m)})\}, \tag{15}$$

where $\bar{\boldsymbol{a}}$ and $\boldsymbol{a}_{90}$ are the sensorwise mean and 90th percentile, standardization is across sensors within the event, with the convention for constant vectors given in Appendix B, and votes are counted across windows rather than across restorations. Section 5 includes a matched-aggregation comparison applying this routine to every baseline.

4.5 Complexity

Per anomalous window, SPARCE requires nT reverse forward evaluations and $nG_T n_\varepsilon$ scoring forward and input-backward evaluation equivalents, where $G_T = T - \lceil T/2 \rceil + 1$. With $n = 30$,

$T = 50$ and $n_\varepsilon = 1$, this totals 2,280 forward and 780 backward equivalents. SPARCE avoids per-sensor enumeration, although each evaluation depends on K and L. Per-sensor baselines repeat reconstruction K times; Shapley costs depend on coalition budgets. Table 7 reports wall-clock times.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Datasets. We evaluate on SWaT (Goh et al., 2016) (51 sensors), WADI (Ahmed et al., 2017) (127 sensors), MSDS (Nedelkoski et al., 2020) (10 metrics) and SMD (Su et al., 2019) (38 sensors), z-score normalized and segmented into windows of length $L = 50$.

Baselines. We compare against the marginal attribution methods KernelSHAP (Lundberg & Lee, 2017) and ShaTS (Franco De La Peña et al., 2026), the causal discovery methods RCD (Ikram et al., 2022) and AERCA (Han et al., 2025), the retrieval-based conditional method CondAttr (Mishra et al., 2026), and diffusion baselines that adapt CSDI (Tashiro et al., 2021) and WaveStitch (Shankar et al., 2025) to RCA by masking each sensor independently and scoring its reconstruction error. All methods share the same VAE detector and normal training split, with hyperparameters as published where available (Appendix D).

Evaluation. Following Mishra et al. (2026), we aggregate per-window attributions into one sensor ranking per event for Recall@K (Han et al., 2025), defined as $|\mathrm{Top}_K \cap \mathbb{J}|/\min(K, |\mathbb{J}|)$ against the annotated set $\mathbb{J}$. CW-RCS@K and TemporalHM@K (Mishra et al., 2026) are computed per window and per event respectively (Appendix C). All three use $K \in \{3, 5\}$.

Implementation. The diffusion model is trained for 200 epochs on normal windows, randomly masking one to three complete sensor channels per training window. Attribution uses $T = 50$ reverse steps, $\eta = 0.5$, $\lambda_m = 0.1$, $\lambda_s = 1.0$, $k' = 10$ and $n = 30$ restorations per window. On an NVIDIA RTX A6000, training takes 40 minutes and attribution over all SWaT attack windows 20 minutes at batch size 256. We report the mean over 5 runs with standard deviations and 95% t-intervals per entry, and test SPARCE against CondAttr with Welch's t-test on 5 seeds per method.

5.2 MAIN RESULTS

Table 1: Root cause identification (Recall@K). Best in **bold**, second best underlined. SPARCE is the mean over 5 runs; baseline values are as published or from a single run under our protocol. Per-entry deviations, intervals and significance tests are in Appendix E.

	SWaT		WADI		MSDS		SMD	
Method	R@3	R@5	R@3	R@5	R@3	R@5	R@3	R@5
ϵ-Diagnosis (Shan et al., 2019)	0.125	0.125	0.125	0.175	0.266	0.452	0.182	0.290
RCD (Ikram et al., 2022)	0.000	0.000	0.083	0.083	0.573	0.984	0.131	0.155
AERCA (Han et al., 2025)	0.290	0.330	0.344	0.380	0.908	0.974	0.460	0.477
KernelSHAP (Lundberg & Lee, 2017)	0.055	0.064	0.011	0.011	0.311	0.467	0.219	0.241
ShaTS (Franco De La Peña et al., 2026)	0.393	0.513	0.402	0.458	0.915	0.986	0.511	0.528
CSDI-RCA (Tashiro et al., 2021)	0.432	0.514	0.402	0.418	0.855	0.916	0.472	0.498
WaveStitch-RCA (Shankar et al., 2025)	0.400	0.457	0.382	0.396	0.832	0.921	0.423	0.463
CondAttr (Mishra et al., 2026)	<u>0.537</u>	<u>0.601</u>	<u>0.462</u>	<u>0.476</u>	<u>0.948</u>	**1.000**	<u>0.529</u>	<u>0.535</u>
SPARCE (ours)	**0.608**	**0.656**	**0.486**	**0.492**	**0.956**	**1.000**	**0.609**	**0.652**

Table 1 reports root cause identification across four benchmarks. SPARCE attains the best Recall@3 on all four datasets and the best or tied-best Recall@5, improving Recall@3 by 13.2% over the strongest baseline on SWaT and by 15.1% on SMD. The gains are largest where inter-sensor structure is richest and where per-sensor scoring has the most to lose: on MSDS, with ten metrics, Recall@5 saturates at 1.000 for SPARCE and CondAttr and no longer separates the two methods. Marginal attribution methods trail throughout, consistent with the view that perturbing sensors independently produces counterfactuals that the system's dependencies never permit, and the diffusion baselines that mask one sensor at a time recover part of the gap but stay below joint restoration.

The gaps in Table 1 are larger than run-to-run variation: repeating both SPARCE and the strongest baseline with five seeds, per-entry standard deviations are at most 0.020 and every non-tied entry favours SPARCE under Welch's t-test at $p < 0.01$, the largest gains being $+0.085$ Recall@3 on SWaT and $+0.120$ Recall@5 on SMD. Appendix E lists the per-entry means, 95% t-intervals and p-values; the rerun places CondAttr slightly below its published values, so Table 1 retains the published numbers as the more conservative comparison.

Table 2 evaluates attribution confidence and temporal localization. SPARCE achieves the highest CW-RCS on both datasets, by a margin larger than on Recall, which indicates that the restoration concentrates edit mass on annotated sensors instead of dispersing it across correlated channels; the sparsity term, rather than the ranking alone, is what this metric rewards. TemporalHM shows the same ordering, so the attribution not only identifies the responsible sensors but flags them early and holds that attribution throughout the event. Results on SMD and MSDS are consistent (Appendix E).

Table 2: Attribution confidence (CW-RCS@K) and temporal localization (TemporalHM@K) on SWaT and WADI. Mean over 5 runs. Best in **bold**, second best underlined. SMD and MSDS are reported in Appendix E.

	SWaT				WADI			
	CW-RCS		TemporalHM		CW-RCS		TemporalHM	
Method	@3	@5	@3	@5	@3	@5	@3	@5
KernelSHAP (Lundberg & Lee, 2017)	0.004	0.004	0.064	0.078	0.001	0.001	0.006	0.020
ShaTS (Franco De La Peña et al., 2026)	0.122	0.134	0.422	0.519	0.031	0.035	0.275	0.351
CSDI-RCA (Tashiro et al., 2021)	0.189	0.195	0.421	0.505	0.126	0.135	0.245	0.325
WaveStitch-RCA (Shankar et al., 2025)	0.178	0.184	0.413	0.515	0.102	0.121	0.248	0.318
CondAttr (Mishra et al., 2026)	<u>0.245</u>	<u>0.253</u>	<u>0.503</u>	<u>0.562</u>	<u>0.189</u>	<u>0.192</u>	<u>0.265</u>	<u>0.360</u>
SPARCE (ours)	**0.394**	**0.400**	**0.645**	**0.692**	**0.388**	**0.389**	**0.514**	**0.520**

5.3 Analysis

Components. Table 3 isolates each design choice on SWaT. Guidance, not the diffusion prior alone, drives attribution, and gradient normalization matters most at a fixed step size, since the unnormalized gradient scale varies by orders of magnitude across noise levels. Channel masking and the guidance cutoff each contribute more than either loss term on its own.

Aggregation. Table 4 applies the routine of Section 4.4 to every method. The ordering is unchanged, and SPARCE stays ahead of the strongest baseline with its voting term removed, so the gain comes from the restorations rather than from the readout.

Table 3: Component ablation on SWaT.

Variant	R@3	R@5
No guidance	0.484	0.539
Uncond. DDPM + SDEdit	0.524	0.542
Random cell masking	0.445	0.489
w/o gradient normalization	0.044	0.091
w/o sparsity (λ_m=0)	0.588	0.612
w/o fidelity (λ_s=0)	0.564	0.607
Guidance at all steps	0.534	0.559
SPARCE	**0.608**	**0.656**

Table 4: Matched aggregation on SWaT.

Method	R@3	R@5
KernelSHAP	0.178	0.165
ShaTS	0.385	0.503
WaveStitch-RCA	0.415	0.468
CSDI-RCA	0.441	0.524
CondAttr	<u>0.540</u>	<u>0.610</u>
SPARCE, no voting	0.569	0.618
SPARCE	**0.608**	**0.656**

Restoration diagnostics. Table 5 assesses detector validity, edit magnitude, and edit concentration independently of the ranking metrics. The three sensors with the largest ℓ_1 edit mass receive between 2.7 and 14.2 times the mass that three sensors would receive under a uniform allocation, indicating concentration relative to uniform allocation rather than strictly sparse sensor support.

Interventions. Table 6 evaluates the unchanged readout on a Structural-VAR benchmark whose labels are injected intervention targets rather than all deviating sensors (Appendix F). SPARCE

outperforms deviation ranking in both settings, with a substantially larger gap when two roots are injected.

Table 5: Counterfactual restoration diagnostics: detector-valid restorations (%), median per-element edit (std. deviations), and top-3 ℓ_1 edit-mass share relative to uniform.

Dataset	Valid (%)	Med. edit	Top-3 mass
SWaT	81.0	0.48	$8.1\times$
WADI	67.5	0.54	$14.2\times$
MSDS	92.3	0.31	$2.7\times$
SMD	83.0	0.45	$6.2\times$

Table 6: Structural-VAR benchmark with known intervention targets, against deviation ranking.

	SPARCE	Deviation
Root Recall@1	**0.865**	0.417
Root-child sep.	**0.768**	0.368
Multi-root R@2	**0.728**	0.080

Inference latency. SPARCE processes all sensors together, while the compared baselines repeat attribution across sensors. Its runtime is comparable on SWaT (51 sensors) and approximately $3.5\times$ faster on WADI (127 sensors), showing its practical advantage on the larger sensor set (Table 7).

Table 7: Inference latency on one NVIDIA RTX A6000, batch size 1. Totals cover all attack windows.

	SWaT (51 sensors, 1,059 windows)		WADI (127 sensors, 194 windows)	
Method	Per window (s)	Full dataset (min)	Per window (s)	Full dataset (min)
ShaTS	21.8	385	71.3	231
CondAttr	22.0	388	71.6	232
SPARCE (ours)	**20.0**	**354**	**20.3**	**66**

Appendix E reports these and further studies. Recall@3 stays within 0.06 of the default across sweeps over η, λ_m and n; SPARCE attains the best Recall@3 in 17 of 20 detector-dataset pairs across five detector backbones; Recall@3 falls by 0.016 under 40% synthetic training contamination; and replacing S_θ by the external detector in equation 11 changes Recall@3 by at most 0.010 while raising detector validity, confirming the surrogate does nothing the detector would not.

5.4 Limitations

SPARCE attributes observable deviations, so under fault propagation, edited set also contains downstream sensors; separating these from initiating faults is evaluated only on synthetic interventions. The attribution inherits the learned normal prior and degrades as that prior does under training-set contamination or distribution shift, and the guided sampler approximates equation 7 without guarantee of sampling it or reaching its mode. Binary and categorical sensors remain difficult for a continuous diffusion process (Austin et al., 2021), which may place plausible mass on several discrete states.

6 Conclusion

We frame root cause analysis in multivariate time series as a sparse counterfactual restoration problem. A learned model of normal behaviour acts as a prior, an ℓ_1 penalty keeps the edit small, and the sensors that get edited are the attributed ones. This replaces a ranking heuristic with an explicit inference objective, one whose scope we state and whose recovery conditions we state. SPARCE approximates that objective with gradient-guided reverse diffusion, attributing every sensor in a single pass, with no per-sensor enumeration and no detector queries. On SWaT, WADI, MSDS and SMD it sets a new state-of-the-art in Recall, CW-RCS and TemporalHM, with significant Recall gains over the strongest baseline. Two design choices are decisive: normalizing the guidance gradient and guiding only the high-noise half of the reverse process. The restorations also pass the external detector they were never optimized against, and concentrate their edits on few sensors. On a benchmark with known intervention targets, the same readout recovers initiating faults where deviation ranking fails. The harder open problem is real systems, where annotations cannot separate initiating faults from their downstream effects.

AI USE STATEMENT

In this work, we used generative AI tools to assist with language editing, including improvements to grammar, sentence structure, clarity, and readability. We used generative AI tools to check the consistency of mathematical notation and proof steps against standard theorem and lemma statements; all mathematical arguments and proofs were developed and verified by the authors. We did not use generative AI tools for research ideation, experimental design, execution of experiments, generation of experimental results, or the retrieval and citation of related work. We additionally used generative AI tools for limited assistance with LaTeX formatting and table generation. All AI-assisted material was carefully reviewed, edited where necessary, and verified by the authors for correctness and consistency with the underlying work. The authors take full responsibility for the final content of this manuscript, including any text or artifacts produced with the assistance of generative AI.

ETHICS STATEMENT

This work uses four publicly available benchmarks (SWaT, WADI, MSDS, SMD) collected from physical testbeds and computing infrastructure under their standard research-use terms. It involves no human subjects, no personal data and no new data collection, and we release no new dataset.

The intended application is operational: helping engineers localize faults in safety-critical monitoring systems, where reliable identification can shorten diagnosis, prevent cascading failures and improve oversight of automated detectors. The corresponding risk is over reliance. SPARCE attributes observable deviations, which under fault propagation include downstream sensors as well as the one where the fault originated. In addition, its restorations approximate the minimal-edit objective rather than solving it exactly. Acting on an attribution as if it identified a physical cause could therefore direct an action at the wrong component. We state this scope in Section 3.1 and its limits in Section 5.4, and we recommend that the method be used as decision support alongside domain expertise rather than as an autonomous diagnostic system.

The released artifacts, a diffusion model trained on public sensor data and an inference pipeline, carry no dual-use concern beyond that of the underlying benchmarks. We report negative and inconclusive results alongside positive ones, including the cases where a baseline is stronger and the conditions under which our analysis does not apply. The authors declare no conflicts of interest.

REPRODUCIBILITY STATEMENT

The method is specified completely in Section 4 and Algorithm 1, with the architecture, training and inference settings in Appendix B.1 and the remaining conventions, covering loss reductions, the masking distribution, gradient normalization and the standardization used in the readout, in Appendix B. Appendix C gives the dataset statistics, pre-processing, window selection and the exact metric definitions. Appendix D documents the baseline implementations and the matched-aggregation protocol. The assumptions and complete proofs for every theoretical claim are in Appendix A, and the generator, fault injection, labels and evaluation metrics for the synthetic benchmark are in Appendix F. Appendix E reports per-entry standard deviations, confidence intervals and significance tests over repeated runs, together with the robustness studies. The code is not released at submission time to preserve anonymity; it will be made publicly available upon acceptance, including training and evaluation scripts and the configurations behind every reported number.

REFERENCES

Chuadhry Mujeeb Ahmed, Venkata Reddy Palleti, and Aditya P Mathur. Wadi: a water distribution testbed for research in the design of secure cyber physical systems. In *Proceedings of the 3rd international workshop on cyber-physical systems for smart water networks*, pp. 25–28, 2017.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, pp. 17981–17993, 2021.

João Bento, Pedro Saleiro, André F Cruz, Mário AT Figueiredo, and Pedro Bizarro. Timeshap: Explaining recurrent models through sequence perturbations. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 2565–2573, 2021.

Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Manuel Franco De La Peña, Ángel Luis Perales Gómez, and Lorenzo Fernández Maimó. Shats: a shapley-based explainability method for time series artificial intelligence models. *Future Generation Computer Systems*, 176:108178, 2026. ISSN 0167-739X. doi: https://doi.org/10.1016/j.future.2025.108178.

Jonathan Goh, Sridhar Adepu, Khurum Nazir Junejo, and Aditya Mathur. A dataset to support research in the design of secure water treatment systems. In *International Conference on Critical Information Infrastructures Security*, pp. 88–99. Springer, 2016.

Xiao Han, Saima Absar, Lu Zhang, and Shuhan Yuan. Root cause analysis of anomalies in multivariate time series through granger causal discovery. In *The Thirteenth International Conference on Learning Representations*, 2025.

Yutong He, Naoki Murata, Chieh-Hsin Lai, Yuhta Takida, Toshimitsu Uesaka, Dongjun Kim, WeiHsiang Liao, Yuki Mitsufuji, Zico Kolter, Ruslan Salakhutdinov, et al. Manifold preserving guided diffusion. In *International Conference on Learning Representations*, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Azam Ikram, Sarthak Chakraborty, Subrata Mitra, Shiv Saini, Saurabh Bagchi, and Murat Kocaoglu. Root cause analysis of failures in microservices through causal discovery. In *Advances in Neural Information Processing Systems*, volume 35, pp. 31158–31170, 2022.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Zhihan Li, Youjian Zhao, Jiaqi Han, Ya Su, Rui Jiao, Xidao Wen, and Dan Pei. Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 3220–3230, 2021.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Shashank Mishra, Karan Patil, Cedric Schockaert, Didier Stricker, and Jason Rambach. Conditional attribution for root cause analysis in time-series anomaly detection. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 165–181. Springer, 2026.

Sasho Nedelkoski, Jasmin Bogatinovski, Ajay Kumar Mandapati, Steffen Becker, Jorge Cardoso, and Odej Kao. Multi-source distributed system data for ai-powered analytics. In *European Conference on Service-Oriented and Cloud Computing*, pp. 161–176. Springer, 2020.

Sahand N. Negahban, Pradeep Ravikumar, Martin J. Wainwright, and Bin Yu. A Unified Framework for High-Dimensional Analysis of M-Estimators with Decomposable Regularizers. *Statistical Science*, 27(4):538 – 557, 2012. doi: 10.1214/12-STS400.

Huasong Shan, Yuan Chen, Haifeng Liu, Yunpeng Zhang, Xiao Xiao, Xiaofeng He, Min Li, and Wei Ding. ϵ-diagnosis: Unsupervised and real-time diagnosis of small-window long-tail latency in large-scale microservice platforms. In *The World Wide Web Conference*, pp. 3215–3222, 2019.

Aditya Shankar, Lydia Chen, Arie van Deursen, and Rihan Hai. Wavestitch: Flexible and fast conditional time series generation with diffusion models. *Proceedings of the ACM on Management of Data*, 3(6):1–25, 2025.

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.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328. PMLR, 2017.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in neural information processing systems*, 34:24804–24816, 2021.

APPENDIX OVERVIEW

- **Appendix A** states the formulation and its analysis: regularity conditions, proofs, and the recovery result and its assumptions.
- **Appendix B** gives method details: conventions, masking, normalization, and the approximations behind the implemented guidance.
- **Appendix C** describes the datasets, the evaluation protocol, and the metric definitions.
- **Appendix D** documents the baseline implementations and the matched-aggregation protocol.
- **Appendix E** reports extended results: uncertainty and significance, robustness studies, further ablations, and qualitative examples.
- **Appendix F** specifies the Structural-VAR benchmark and its full results.

A FORMULATION AND ANALYSIS

Fix $\boldsymbol{W}_{\text{anom}} \in \mathbb{R}^{K\times L}$. Throughout, $\|\cdot\|_1$ is the entrywise absolute sum, $\|\cdot\|_F$ the Frobenius norm, and $\|\boldsymbol{B}\|_{\max} = \max_{j,\ell} |\boldsymbol{B}_{j,\ell}|$. Write

$$C(\boldsymbol{Z}) = \lambda_m \|\boldsymbol{W}_{\text{anom}} - \boldsymbol{Z}\|_1 + \lambda_s S_\theta(\boldsymbol{Z}), \qquad \lambda_m > 0, \quad \lambda_s \geq 0.$$

A.1 WELL-POSEDNESS

Lemma A.1 (Properness and mode existence)**.** *For any proper density p_{N}, the base posterior in equation 4 has a finite, positive normalizer. If $S_\theta \geq 0$ is finite p_{N}-almost everywhere, then $\mathcal{Z} = \int p_{\text{N}}(\boldsymbol{Z}) e^{-C(\boldsymbol{Z})} d\boldsymbol{Z}$ satisfies $0 < \mathcal{Z} \leq 1$. If, additionally, p_{N} is continuous and bounded, the base MAP objective attains its minimum; if S_θ is also continuous and finite, so does the regularized mode objective, with $-\log 0 = +\infty$.*

Proof. The Laplace density is bounded and everywhere positive, so its integral against p_{N} is finite and positive. Likewise, $0 < e^{-C} \leq 1$ p_{N}-almost everywhere gives $0 < \mathcal{Z} \leq 1$. Under the continuity assumptions, $g(\boldsymbol{Z}) = p_{\text{N}}(\boldsymbol{Z}) e^{-C(\boldsymbol{Z})}$ is continuous, positive somewhere, and

$$0 \leq g(\boldsymbol{Z}) \leq \|p_{\text{N}}\|_\infty e^{-\lambda_m \|\boldsymbol{W}_{\text{anom}} - \boldsymbol{Z}\|_F} \longrightarrow 0 \quad \text{as } \|\boldsymbol{Z}\|_F \to \infty.$$

It therefore attains its positive maximum on a compact set, equivalently minimizing $-\log g$. Omitting the S_θ term proves base MAP existence. Uniqueness is not asserted. □

A.2 THE REGULARIZER AND THE VARIATIONAL BOUND

Proposition A.2 (S_θ as one bound term)**.** *For a Gaussian DDPM with fixed positive isotropic reverse variances σ_k^2, the same conditioning mode as p_{N}, and scoring level $k' > 1$, the corresponding term in the variational upper bound on $-\log p_{\text{N}}(\boldsymbol{Z})$ is*

$$\mathcal{L}_{k'-1}(\boldsymbol{Z}) = w_{k'} S_\theta(\boldsymbol{Z}) + c_{k'}, \qquad w_{k'} = \frac{\beta_{k'}^2}{2\sigma_{k'}^2 \alpha_{k'} (1 - \bar{\alpha}_{k'})},$$

where $c_{k'}$ is independent of $\boldsymbol{Z}$ and S_θ uses the summed squared norm in equation 10.

Proof. Let q_{f} denote the forward law, with Gaussian distributions understood on vectorized windows. The forward posterior covariance is $\tilde{\beta}_{k'} \boldsymbol{I}$, where $\tilde{\beta}_{k'} = \beta_{k'}(1 - \bar{\alpha}_{k'-1})/(1 - \bar{\alpha}_{k'})$; the reverse covariance is $\sigma_{k'}^2 \boldsymbol{I}$. The Gaussian KL formula gives

$$\begin{aligned}\mathcal{L}_{k'-1}(\boldsymbol{Z}) &= \mathbb{E}_{\boldsymbol{W}_{k'}|\boldsymbol{Z}} \mathrm{KL}(q_{\text{f}}(\boldsymbol{W}_{k'-1} \mid \boldsymbol{W}_{k'}, \boldsymbol{Z}) \,\|\, p_\theta(\boldsymbol{W}_{k'-1} \mid \boldsymbol{W}_{k'})) \\ &= \frac{\mathbb{E}_{\boldsymbol{W}_{k'}|\boldsymbol{Z}} \|\tilde{\boldsymbol{\mu}} - \boldsymbol{\mu}_\theta\|_2^2}{2\sigma_{k'}^2} + c_{k'}.\end{aligned}$$

The covariance contribution $c_{k'}$ is independent of $\boldsymbol{Z}$. For $\boldsymbol{W}_{k'} = \sqrt{\bar{\alpha}_{k'}}\boldsymbol{Z} + \sqrt{1 - \bar{\alpha}_{k'}}\boldsymbol{\epsilon}$, the noise parameterization gives

$$\tilde{\boldsymbol{\mu}} - \boldsymbol{\mu}_\theta = \frac{\beta_{k'}}{\sqrt{\alpha_{k'}(1 - \bar{\alpha}_{k'})}} \operatorname{vec}(\boldsymbol{\epsilon}_\theta(\boldsymbol{W}_{k'}, k') - \boldsymbol{\epsilon}).$$

Substitution and expectation over $\boldsymbol{\epsilon}$ yield the claim (Ho et al., 2020). □

This identifies a single weighted bound term, not the full negative log density; consequently, equation 7 is not generally a tempered normal prior.

A.3 TARGET-SCORE IDENTITY

Proposition A.3 (Score of the noised target)**.** *Let $q(\boldsymbol{Z} \mid \boldsymbol{W}_{\mathrm{anom}}) = \mathcal{Z}^{-1} p_{\mathrm{N}}(\boldsymbol{Z}) e^{-C(\boldsymbol{Z})}$, with $C \geq 0$ finite p_{N}-almost everywhere. For $0 < \bar{\alpha}_k < 1$, let q_k and $p_{\mathrm{N},k}$ be the densities after applying the Gaussian forward kernel $\kappa_k(\boldsymbol{W} \mid \boldsymbol{Z}) = \mathcal{N}(\boldsymbol{W}; \sqrt{\bar{\alpha}_k}\boldsymbol{Z}, (1-\bar{\alpha}_k)\boldsymbol{I})$. Define $\Psi_k(\boldsymbol{W}) = \mathbb{E}_{p_{\mathrm{N}}}[e^{-C(\boldsymbol{W}_0)} \mid \boldsymbol{W}_k = \boldsymbol{W}]$. Then, for every $\boldsymbol{W}$,*

$$q_k(\boldsymbol{W} \mid \boldsymbol{W}_{\mathrm{anom}}) = \mathcal{Z}^{-1} p_{\mathrm{N},k}(\boldsymbol{W}) \Psi_k(\boldsymbol{W}),$$
$$\nabla_{\boldsymbol{W}} \log q_k = \nabla_{\boldsymbol{W}} \log p_{\mathrm{N},k} + \nabla_{\boldsymbol{W}} \log \Psi_k.$$

Proof. Bayes' rule gives

$$\begin{aligned} q_k(\boldsymbol{W}) &= \mathcal{Z}^{-1} \int p_{\mathrm{N}}(\boldsymbol{Z}) e^{-C(\boldsymbol{Z})} \kappa_k(\boldsymbol{W} \mid \boldsymbol{Z})\, d\boldsymbol{Z} \\ &= \mathcal{Z}^{-1} p_{\mathrm{N},k}(\boldsymbol{W}) \Psi_k(\boldsymbol{W}). \end{aligned}$$

The Gaussian kernel and e^{-C} are positive almost everywhere under the relevant measures, so both factors are positive. Gaussian kernel derivatives are uniformly bounded in the kernel's displacement, allowing differentiation under the integrals against p_{N}, since $e^{-C} \leq 1$. Taking the log gradient and using that $\mathcal{Z}$ is independent of $\boldsymbol{W}$ proves the identity. □

This is the diffusion posterior decomposition of Chung et al. (2022), with e^{-C} as a pseudo-likelihood. The identity is exact; the Tweedie plug-in, stopped differentiation, and subsequent sampling choices in Section 4.3 are approximations detailed in Appendix B.

A.4 OBSERVABLE DEVIATIONS AND INITIATING INTERVENTIONS

Proposition A.4 (Propagation of deviation support)**.** *Let $\boldsymbol{x}_t^{\mathrm{N}} = \boldsymbol{A}\boldsymbol{x}_{t-1}^{\mathrm{N}} + \boldsymbol{u}_t$ and $\boldsymbol{x}_t^{\mathrm{A}} = \boldsymbol{A}\boldsymbol{x}_{t-1}^{\mathrm{A}} + \boldsymbol{u}_t + \boldsymbol{b}_t$ share innovations and initial state. Their difference satisfies*

$$\boldsymbol{d}_t = \sum_{r=1}^{t} \boldsymbol{A}^{t-r} \boldsymbol{b}_r.$$

For $\boldsymbol{b}_r = \beta \boldsymbol{e}_j \mathbf{1}\{r = 1\}$, $\beta \neq 0$, and the window $t = 1, \ldots, L$,

$$\mathbb{J}(\boldsymbol{D}^\star) = \bigcup_{t=1}^{L} \mathrm{supp}(\boldsymbol{A}^{t-1} \boldsymbol{e}_j).$$

Proof. Subtracting the recursions gives $\boldsymbol{d}_t = \boldsymbol{A}\boldsymbol{d}_{t-1} + \boldsymbol{b}_t$ with $\boldsymbol{d}_0 = \mathbf{0}$. Unrolling gives the first expression; substituting the intervention gives $\boldsymbol{d}_t = \beta \boldsymbol{A}^{t-1} \boldsymbol{e}_j$ and hence the support identity. □

The observable support contains the root and can include downstream sensors, subject to exact cancellations. Full support recovery thus includes but does not distinguish the root; a top-r ranking may omit it when $r < |\mathbb{J}(\boldsymbol{D}^\star)|$. Propagation can also violate sparse support, motivating the separate intervention-recovery evaluation in Appendix F.

Remark A.5 (Per-sensor restriction)**.** For a fixed sensor j, restricting edits to $\{\boldsymbol{D} : \mathbb{J}(\boldsymbol{D}) \subseteq \{j\}\}$ restricts the objective through $\boldsymbol{Z} = \boldsymbol{W}_{\mathrm{anom}} - \boldsymbol{D}$, so the restricted infimum is at least the unrestricted minimum, with equality if and only if some unrestricted minimizer has edit support contained in $\{j\}$, including the zero edit. This is an optimization comparison, not a performance guarantee for approximate algorithms.

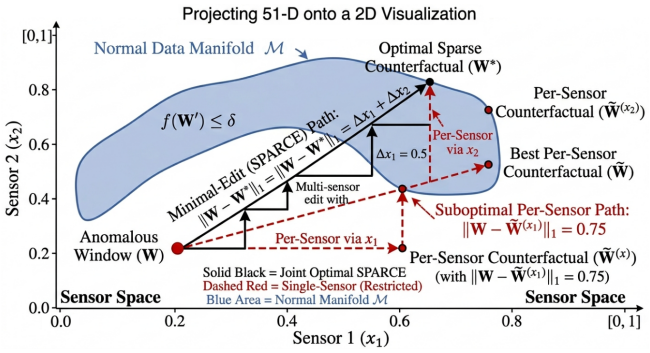


Figure 2: Illustrative ℓ_1 edit geometry in two sensor dimensions. A joint edit can reach a candidate normal region at lower edit cost than an edit confined to one coordinate. The shaded region is not a hard constraint in equation 5, which also accounts for prior density; the picture is not used in any proof.

A.5 Conditional support recovery

We specialize the basic-inequality argument for decomposable regularizers (Negahban et al., 2012) to characterize when the existing attribution readout separates observable deviations.

Setting. In edit coordinates $\boldsymbol{D} = \boldsymbol{W}_{\text{anom}} - \boldsymbol{Z}$, write

$$\ell_{\text{N}}(\boldsymbol{D}) = -\log p_{\text{N}}(\boldsymbol{W}_{\text{anom}} - \boldsymbol{D}) + \lambda_s S_\theta(\boldsymbol{W}_{\text{anom}} - \boldsymbol{D}),$$
$$F_{\text{N}}(\boldsymbol{D}) = \ell_{\text{N}}(\boldsymbol{D}) + \lambda_m \|\boldsymbol{D}\|_1.$$

Define ℓ_{ref} by replacing p_{N} with a reference density p_{ref}, retaining the same S_θ and λ_s. Let $\mathcal{I} = \{(j,\ell) : \boldsymbol{D}^\star_{j,\ell} \neq 0\}$, $\nu = |\mathcal{I}| \le sL$, and let $\mathcal{D}$ be a convex region containing $\boldsymbol{D}^\star$ and an estimate $\widetilde{\boldsymbol{D}}$. The analysis is conditional on the given normal reference and deviation; it imposes no Laplace law or independence on the physical fault process.

Theorem A.6 (Stable recovery under approximate optimization)**.** *Assume ℓ_{ref} and ℓ_{N} are continuously differentiable on an open neighbourhood of $\mathcal{D}$, both densities are positive at $\boldsymbol{W}_{\text{anom}} - \boldsymbol{D}$ there, and $\inf_{\mathcal{D}} F_{\text{N}}$ is finite. Suppose:*

(H1) Reference curvature. *For every $\boldsymbol{D} \in \mathcal{D}$ and some $\kappa > 0$,*

$$\ell_{\text{ref}}(\boldsymbol{D}) - \ell_{\text{ref}}(\boldsymbol{D}^\star) - \langle \nabla \ell_{\text{ref}}(\boldsymbol{D}^\star), \boldsymbol{D} - \boldsymbol{D}^\star \rangle \ge \frac{\kappa}{2} \|\boldsymbol{D} - \boldsymbol{D}^\star\|_F^2.$$

(H2) Residual and prior error. *For $\gamma, \varepsilon_p \ge 0$,*

$$\|\nabla \ell_{\text{ref}}(\boldsymbol{D}^\star)\|_{\max} \le \gamma, \qquad \sup_{\boldsymbol{D} \in \mathcal{D}} \|\nabla \ell_{\text{N}}(\boldsymbol{D}) - \nabla \ell_{\text{ref}}(\boldsymbol{D})\|_{\max} \le \varepsilon_p.$$

(H3) Optimization error. *For some $\xi \ge 0$, $\widetilde{\boldsymbol{D}} \in \mathcal{D}$ and $F_{\text{N}}(\widetilde{\boldsymbol{D}}) \le \inf_{\mathcal{D}} F_{\text{N}} + \xi$.*

If $\lambda_m \ge 2(\gamma + \varepsilon_p)$, then

$$\begin{aligned} \|\widetilde{\boldsymbol{D}} - \boldsymbol{D}^\star\|_F &\le B_\nu(\xi) := \frac{3\lambda_m\sqrt{\nu} + \sqrt{9\lambda_m^2\nu + 8\kappa\xi}}{2\kappa} \\ &\le \frac{3\lambda_m\sqrt{\nu}}{\kappa} + \sqrt{\frac{2\xi}{\kappa}}. \end{aligned} \tag{16}$$

Proof. Set $\boldsymbol{H} = \widetilde{\boldsymbol{D}} - \boldsymbol{D}^\star$, $u = \|\boldsymbol{H}\|_F$, and $e = \ell_{\text{N}} - \ell_{\text{ref}}$. Convexity of $\mathcal{D}$, the fundamental theorem of calculus, and (H2) give

$$e(\widetilde{\boldsymbol{D}}) - e(\boldsymbol{D}^\star) = \int_0^1 \langle \nabla e(\boldsymbol{D}^\star + t\boldsymbol{H}), \boldsymbol{H} \rangle \, dt \ge -\varepsilon_p \|\boldsymbol{H}\|_1. \tag{17}$$

Combining this with (H1) and the residual bound in (H2),

$$\ell_{\mathrm{N}}(\widetilde{\boldsymbol{D}}) - \ell_{\mathrm{N}}(\boldsymbol{D}^\star) \geq \frac{\kappa}{2}u^2 - (\gamma + \varepsilon_p)\|\boldsymbol{H}\|_1. \tag{18}$$

Since $\boldsymbol{D}^\star \in \mathcal{D}$, (H3) also gives

$$\ell_{\mathrm{N}}(\widetilde{\boldsymbol{D}}) - \ell_{\mathrm{N}}(\boldsymbol{D}^\star) \leq \lambda_m\big(\|\boldsymbol{D}^\star\|_1 - \|\boldsymbol{D}^\star + \boldsymbol{H}\|_1\big) + \xi.$$

Decomposability yields $\|\boldsymbol{D}^\star\|_1 - \|\boldsymbol{D}^\star + \boldsymbol{H}\|_1 \leq \|\boldsymbol{H}_{\mathcal{I}}\|_1 - \|\boldsymbol{H}_{\mathcal{I}^c}\|_1$. Thus

$$\frac{\kappa}{2}u^2 \leq (\lambda_m + \gamma + \varepsilon_p)\|\boldsymbol{H}_{\mathcal{I}}\|_1 - (\lambda_m - \gamma - \varepsilon_p)\|\boldsymbol{H}_{\mathcal{I}^c}\|_1 + \xi.$$

Under $\lambda_m \geq 2(\gamma + \varepsilon_p)$, the first coefficient is at most $3\lambda_m/2$ and the second term is nonpositive. Dropping that term and using $\|\boldsymbol{H}_{\mathcal{I}}\|_1 \leq \sqrt{\nu}\,u$ gives $\kappa u^2 - 3\lambda_m\sqrt{\nu}\,u - 2\xi \leq 0$. Solving this quadratic proves the first bound; the second follows from $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$. □

The bound also holds with ν replaced by sL and does not require $\widetilde{\boldsymbol{D}}$ to be sparse.

Corollary A.7 (Per-window ranking)**.** *Let $1 \leq s = |\mathbb{J}(\boldsymbol{D}^\star)| < K$ and $\widetilde{\boldsymbol{Z}} = \boldsymbol{W}_{\mathrm{anom}} - \widetilde{\boldsymbol{D}}$. For the readout equation 6, write $a_j = L^{-1}\|\widetilde{\boldsymbol{D}}_{j,:}\|_1$, $b_j = L^{-1}\|\boldsymbol{D}^\star_{j,:}\|_1$, and $b_{\min} = \min_{j\in\mathbb{J}(\boldsymbol{D}^\star)} b_j$. Under Theorem A.6, if*

$$b_{\min} > \frac{2B_\nu(\xi)}{\sqrt{L}}, \tag{19}$$

then $a_j > a_i$ whenever $j \in \mathbb{J}(\boldsymbol{D}^\star)$ and $i \notin \mathbb{J}(\boldsymbol{D}^\star)$. Consequently, $\mathrm{Recall@}r = 1$ *for every $s \leq r \leq K$.*

Proof. For $\boldsymbol{H} = \widetilde{\boldsymbol{D}} - \boldsymbol{D}^\star$, the reverse triangle inequality and Cauchy–Schwarz give

$$|a_j - b_j| \leq \frac{\|\boldsymbol{H}_{j,:}\|_1}{L} \leq \frac{\|\boldsymbol{H}_{j,:}\|_2}{\sqrt{L}} \leq \frac{B_\nu(\xi)}{\sqrt{L}}.$$

An inactive sensor has $b_i = 0$, hence $a_i \leq B_\nu(\xi)/\sqrt{L}$; an active sensor has $a_j \geq b_{\min} - B_\nu(\xi)/\sqrt{L}$, strictly above this threshold by equation 19. □

Corollary A.8 (Event-level aggregation)**.** *Let $K \geq 3$. If Corollary A.7 holds in each of an event's M windows with the same observable support, the aggregation in equation 15 also ranks every active sensor above every inactive sensor, provided standardization maps constant vectors to zero.*

Proof. For any active j and inactive i, all differences $a_j^{(m)} - a_i^{(m)}$ are positive and their finite minimum is positive. Means and corresponding order statistics preserve this separation, as does the empirical 90th percentile with common convex interpolation weights. Hence $\bar{a}_j + a_{90,j} > \bar{a}_i + a_{90,i}$. This vector has positive standard deviation, so standardization preserves the strict inequality. For top-3 votes, if $s \leq 3$, every active sensor receives all M votes; if $s > 3$, every inactive sensor receives zero votes. In either case $v_j \geq v_i$, and standardization preserves this weak inequality, including the constant-vector case. Adding it with coefficient $1/2$ to the strict inequality proves the claim. □

Nonvacuity and limitations. Take $p_{\mathrm{N}} = p_{\mathrm{ref}} = \mathcal{N}(\mathbf{0}, \sigma^2\boldsymbol{I})$ on vectorized windows, $\boldsymbol{W}^{\mathrm{N}}$ drawn from this density, S_θ constant, and $\mathcal{D} = \mathbb{R}^{K\times L}$. Then $\kappa = 1/\sigma^2$, $\varepsilon_p = 0$, and $\gamma = \|\boldsymbol{W}^{\mathrm{N}}\|_{\max}/\sigma^2$. The condition $\lambda_m \geq 2\gamma$ holds on the positive-probability event $\|\boldsymbol{W}^{\mathrm{N}}\|_{\max} \leq \lambda_m\sigma^2/2$. The exact edit minimizer is

$$\widehat{\boldsymbol{D}}_{\mathrm{MAP}} = \mathrm{soft}_{\lambda_m\sigma^2}(\boldsymbol{W}_{\mathrm{anom}}), \qquad \widehat{\boldsymbol{Z}}_{\mathrm{MAP}} = \boldsymbol{W}_{\mathrm{anom}} - \widehat{\boldsymbol{D}}_{\mathrm{MAP}},$$

where $\mathrm{soft}_\tau(x) = \mathrm{sign}(x)\max(|x| - \tau, 0)$ entrywise. Thus $\xi = 0$, and sufficiently large sustained deviations satisfy equation 19. The bound is conservative: it identifies quantities controlling recovery rather than predicting benchmark performance. With general positive-definite Gaussian covariance $\boldsymbol{\Sigma}$ and constant S_θ, $\kappa = 1/\lambda_{\max}(\boldsymbol{\Sigma})$ and $\gamma = \|\boldsymbol{\Sigma}^{-1}\,\mathrm{vec}(\boldsymbol{W}^{\mathrm{N}})\|_{\max}$; weak curvature can substantially loosen the bound, and multimodality can invalidate (H1).

Relation to SPARCE. The theorem does not establish (H3) for the guided sampler or its averaged restoration $\boldsymbol{W}^\star$: the induced edit must itself lie in $\mathcal{D}$ and satisfy the objective-gap condition. As an analytical benchmark, an exactly Gaussian target with a quadratic negative log density has an expected gap above its minimum of $KL/2$ per independent draw and $KL/(2n)$ for an average of n draws, about 42.5 nats at the SWaT dimensions with $n = 30$; applying this to a merely locally Gaussian target is heuristic, and an expected gap does not bound the realized ξ. Individual sample guarantees need not transfer to an average under a nonconvex objective. Moreover, ε_p measures clean-score error on $\mathcal{D}$, not the denoising training loss or noisy-score error. The theorem uses the absolute weights (λ_s, λ_m), whereas exact normalized guidance depends only on their ratio; rescaling the weights to satisfy the theorem changes the target. Finally, observable-support recovery identifies initiating interventions only when the two supports coincide.

B Method Details

B.1 Architecture and configuration

The diffusion model follows a residual attention architecture. Each residual block applies full self-attention along the temporal and sensor axes. Sinusoidal time embeddings, learned sensor embeddings and the observation mask form a $(B, 145, K, L)$ context tensor, where B is the batch size. Hyperparameters were selected on held-out normal windows by a criterion that balances the normality score S_θ against the edit magnitude $\|\boldsymbol{W}_{\mathrm{anom}} - \boldsymbol{Z}\|_1$, since S_θ alone favours weaker anchoring; no anomaly labels entered the choice. The sensitivity results in Appendix E are post-hoc.

Table 8: Diffusion model architecture hyperparameters.

Component	**Value**
Residual layers	6
Attention heads	8 per layer
Channel dimension	64
Time embedding	128-dim sinusoidal
Feature embedding	16-dim learned
Diffusion step embedding	128-dim
Input channels	2 (noisy target + conditioning)

Table 9: Training.

Setting	Value
Epochs	200
Batch size	256
Optimizer	Adam (Kingma & Ba, 2014)
Learning rate	10^{-3}
LR decay	$\times 0.1$ at $75\%, 90\%$
Weight decay	10^{-6}
Noise schedule	Quadratic
β range	$[10^{-4}, 0.5]$
Hidden sensors	1–3 per window
Diffusion steps	$T = 50$

Table 10: Inference.

Setting	Value
Restorations	$n = 30$
Guidance scale	$\eta = 0.5$
Sparsity weight	$\lambda_m = 0.1$
Regularizer weight	$\lambda_s = 1.0$
Scoring level	$k' = T/5 = 10$
Guided steps	$k \geq \lceil T/2 \rceil$
Scoring draws	$n_\varepsilon = 1$
Hardware	NVIDIA RTX A6000

B.2 Reductions and coefficient scale

The norms in equation 8, equation 10 and equation 11 are summed. In equation 8, $d_\Omega = 1$ for a sum and $d_\Omega = |\Omega|L$ for a mean over hidden entries. In guidance, dividing the regularizer and edit losses by d_s and d_m corresponds to summed-form weights $(\lambda_s/d_s, \lambda_m/d_m)$; a mean over all entries uses $d_s = KL$ or $d_m = KL$. The implementation uses a mean over hidden entries in equation 8, $d_\Omega = |\Omega|L$, and summed reductions for both guidance terms, $d_s = d_m = 1$, so $\lambda_s = 1.0$ and $\lambda_m = 0.1$ in Table 10 are stated in summed form.

B.3 Masking and the all-hidden mode

Training randomly hides one, two, or three complete sensor channels per window. Since all datasets have $K > 3$, the all-hidden configuration has zero probability under the training masking distribution and is not directly trained. Partial masking alone does not ensure coverage of that configuration: it trains conditionals of hidden channels given visible ones, not the joint.

We therefore define p_{N} as the generative law the trained model induces with no observed sensors, and treat consistency of equation 9 with its forward-marginal scores as a modeling approximation. Two measurements support the mode. Table 24 is computed in this same all-hidden configuration, and S_θ separates normal from anomalous windows with ROC-AUC 0.896 to 0.942 and rank correlation 0.84 to 0.89 with the external detector, so the model yields a usable normality signal without conditioning. Independently, 67.5 to 92.3% of averaged restorations fall below the detector threshold (Table 15), so the windows it generates in this mode are ones an independent detector accepts as normal.

B.4 Normalization and standardization

Guidance is normalized as $\boldsymbol{G}_k/\|\boldsymbol{G}_k\|_F$ over the $K \times L$ entries of each window, with no stabilizer and no clipping, so for a nonzero gradient the nudge has length exactly η, relative sensor components are unchanged, and the guided trajectory is invariant to a common rescaling of (λ_s, λ_m).

In equation 15, standardization is across sensors within each event: $(x_j - \mathrm{mean}(\boldsymbol{x}))/\,\mathrm{std}(\boldsymbol{x})$ for positive standard deviation. Constant vectors are mapped to the zero vector by a guarded division, the convention assumed by Corollary A.8.

B.5 Approximations

Relative to the exact target-score identity equation 12, SPARCE uses learned scores in the all-hidden mode, a Gaussian terminal initialization, and a Tweedie plug-in for the intractable conditional expectation (Chung et al., 2022). It stops differentiation through the denoiser in the Tweedie estimate, while retaining input gradients through the Monte Carlo estimate of S_θ. The update further uses normalized guidance with scale η, a cutoff below $\lceil T/2 \rceil$ where the exact correction is generally nonzero, the denoiser prediction computed before the nudge, and finite-step DDPM updates. Their combined error is not bounded here; neither exact sampling from q nor convergence to its mode is guaranteed.

C Datasets and Evaluation Protocol

Datasets. SWaT (Goh et al., 2016) and WADI (Ahmed et al., 2017) are water treatment and distribution testbeds with labelled attack scenarios. MSDS (Nedelkoski et al., 2020) collects metrics from a distributed computing system, and SMD (Su et al., 2019) resource-utilization metrics from server machines.

Table 11: Dataset statistics. Windows are the anomalous windows on which attribution is evaluated.

Dataset	Sensors	Timesteps	Windows
SWaT	51	449,919	1,059
WADI	127	172,801	194
MSDS	10	29,500	13
SMD	38	708,377	1,680

Preprocessing. We use the standard train and test splits released by the original authors. Sensor values are z-score normalized with statistics computed on the training split only, and the test split is segmented into non-overlapping windows of length $L = 50$, following the protocol of Mishra et al. (2026).

Events and window selection. An event is a contiguous labelled anomaly, which spans M windows. Anomalous windows are identified from the ground-truth labels and confirmed by the VAE detector

trained on the normal split, the same detector for every method. For each event, the ground-truth sensor set is taken from the dataset annotations, which mark every sensor behaving anomalously during the event without distinguishing primary causes from downstream effects. This is the observable-deviation support discussed in Section 3.1.

Metrics. The three metrics operate at different levels. Recall@K is computed on the event ranking obtained from equation 15: for an event with annotated set $\mathbb{J}$ and predicted top-K set Top_K,

$$\mathrm{Recall@}K = \frac{|\mathrm{Top}_K \cap \mathbb{J}|}{\min(K, |\mathbb{J}|)},$$

as defined by Han et al. (2025). The denominator means Recall@5 can fall below Recall@3 for events with more than three annotated sensors.

CW-RCS@K is computed per window from the attribution magnitudes, before any ranking. With $\phi_{t,j} = a_j$ the attribution of sensor j in window t, the normalized confidence weight is $\tilde{\phi}_{t,j} = |\phi_{t,j}| / \sum_\ell |\phi_{t,\ell}|$, and with $\widehat{R}_t^{(K)}$ the window's top-K set,

$$\text{CW-RCS}(t) = \frac{1}{|\mathbb{J}_t|} \sum_{j \in \mathbb{J}_t \cap \widehat{R}_t^{(K)}} \tilde{\phi}_{t,j},$$

averaged over anomalous windows. Because $\tilde{\phi}_{t,j} \leq 1$, it is bounded above by Recall@K: it rewards a method only when the correct sensors also carry a dominant share of the attribution mass, so a correct ranking with diffuse attribution scores lower than a correct ranking with concentrated attribution.

TemporalHM@K is computed per event from the sequence of per-window top-K sets. With t_0 the anomaly onset, T_a its duration and $t^\star$ the first window in which any annotated sensor enters $\widehat{R}_t^{(K)}$, early identification and persistence are

$$E = \max\Big(0,\, 1 - \frac{t^\star - t_0}{T_a}\Big), \qquad A = \frac{1}{T_a} \sum_{t=t_0}^{t_0+T_a-1} \mathbf{1}\{\mathbb{J} \cap \widehat{R}_t^{(K)} \neq \emptyset\},$$

combined as $\mathrm{TemporalHM}_\beta = (1+\beta^2)EA/(\beta^2 E + A + \epsilon)$ with $\beta = 1$, so that a method must both flag the responsible sensors early and keep flagging them. We use the definitions and implementation of Mishra et al. (2026) unchanged, and report all three at $K \in \{3, 5\}$.

D Baseline Implementations

Sources and settings. All baselines use their official open-source implementations. Where the original authors report dataset-specific hyperparameters we use those directly; for datasets not covered in the original work we use the authors' SWaT settings as the default, so that no baseline is tuned by us on a dataset its authors did not address. Every attribution method is evaluated on the same windows, with the same VAE detector and the same normal training split used for the SPARCE diffusion model, so the methods differ only in how they attribute a window that all of them receive.

Diffusion baselines. CSDI (Tashiro et al., 2021) and WaveStitch (Shankar et al., 2025) are generative models rather than attribution methods, so we adapt them with the per-sensor protocol they naturally support: each sensor is masked in turn, the model reconstructs it from the remaining sensors, and the reconstruction error becomes that sensor's attribution score. This requires K conditional passes per window, whereas SPARCE avoids per-sensor enumeration (Table 7), and scores each sensor in isolation rather than as part of a joint edit.

Matched aggregation. The comparison in Section 5.3 removes the readout as a confound. Every method produces a per-window sensor score vector, and we then apply the identical routine of equation 15 to all of them: the same standardization across sensors within an event, the same top-3 nomination, the same voting weight, and the same combination of mean and 90th percentile. Only the per-window scores differ. For SPARCE we additionally report the variant with the voting term removed, so that the contribution of the restorations can be separated from the contribution of the readout.

E Extended Results

E.1 Uncertainty and significance

Table 1 reports the mean over 5 runs for SPARCE, and single-run or published values for the baselines. To quantify run-to-run variability we rerun SPARCE and the strongest baseline five times each under our protocol, with seeds controlling model initialization, the masking draws during training, and the restoration noise at attribution time. Table 12 reports per-entry means, standard deviations, 95% t-intervals and Welch's t-test for SPARCE and CondAttr (Mishra et al., 2026) on Recall@3 and Recall@5. Variability is small relative to the gaps: the widest interval spans 0.05, and every non-tied entry favours SPARCE at $p < 0.01$, with the two largest margins on SWaT Recall@3 and SMD Recall@5. Our rerun places CondAttr slightly below its published values on all four datasets, so Table 1 keeps the published numbers, which is the more conservative comparison.

Table 12: SPARCE against CondAttr over 5 seeds per method. MSDS Recall@5 is tied at 1.000 and is not tested.

Dataset	Metric	SPARCE (95% CI)	CondAttr (95% CI)	Δ	Welch p
SWaT	R@3	0.608 ± 0.007 $[0.600, 0.616]$	0.523 ± 0.020 $[0.498, 0.548]$	$+0.085$	$3.2 \cdot 10^{-4}$
	R@5	0.656 ± 0.020 $[0.631, 0.681]$	0.597 ± 0.018 $[0.575, 0.619]$	$+0.059$	$1.2 \cdot 10^{-3}$
WADI	R@3	0.486 ± 0.013 $[0.470, 0.502]$	0.452 ± 0.008 $[0.442, 0.463]$	$+0.034$	$1.6 \cdot 10^{-3}$
	R@5	0.492 ± 0.005 $[0.485, 0.499]$	0.465 ± 0.009 $[0.453, 0.476]$	$+0.027$	$9.7 \cdot 10^{-4}$
MSDS	R@3	0.956 ± 0.005 $[0.950, 0.962]$	0.938 ± 0.009 $[0.927, 0.949]$	$+0.018$	$5.9 \cdot 10^{-3}$
	R@5	1.000 ± 0.000	1.000 ± 0.000	tied	tied
SMD	R@3	0.609 ± 0.004 $[0.604, 0.614]$	0.527 ± 0.005 $[0.521, 0.533]$	$+0.082$	$4.3 \cdot 10^{-9}$
	R@5	0.652 ± 0.004 $[0.647, 0.657]$	0.532 ± 0.019 $[0.509, 0.555]$	$+0.120$	$8.4 \cdot 10^{-5}$

E.2 Additional benchmark results

Tables 13 and 14 report CW-RCS@K and TemporalHM@K on SMD and MSDS, complementing the SWaT and WADI results in Table 2. The findings match the main paper: SPARCE attains the highest attribution confidence on both benchmarks and the highest temporal localization on SMD, so the gains observed on the industrial testbeds carry over to distributed-system and server data. The one exception across all four datasets is TemporalHM@5 on MSDS, where CondAttr is higher; MSDS has ten metrics, so a top-5 prediction already covers half the system and the metric saturates for both methods.

Table 13: CW-RCS@K on SMD and MSDS.

	SMD		MSDS	
Method	@3	@5	@3	@5
KernelSHAP	0.054	0.071	0.050	0.093
ShaTS	0.346	0.383	0.462	0.489
CSDI-RCA	0.329	0.333	0.437	0.441
CondAttr	<u>0.366</u>	<u>0.388</u>	<u>0.551</u>	<u>0.607</u>
SPARCE	**0.435**	**0.459**	**0.595**	**0.632**

Table 14: TempHM@K on SMD and MSDS.

	SMD		MSDS	
Method	@3	@5	@3	@5
KernelSHAP	0.298	0.321	0.343	0.498
ShaTS	0.679	0.698	0.911	0.916
CSDI-RCA	0.650	0.673	0.887	0.904
CondAttr	<u>0.695</u>	<u>0.702</u>	<u>0.927</u>	**0.959**
SPARCE	**0.722**	**0.734**	**0.930**	<u>0.954</u>

E.3 Restoration validity, edit cost and concentration

Beyond sensor ranking, Table 15 assesses restoration behaviour through detector validity, edit magnitude and concentration. A restoration is counted valid when the averaged window falls below the detector threshold; validity is measured on the average itself, since an average of valid samples need not be valid. Between two thirds and nine tenths of restorations are valid, and the edits that achieve this are small: the median per-element change is about half a standard deviation, and the 90th percentile reaches at most 1.84. Concentration is the sharper result. A uniform edit would place a

$3/K$ share of its ℓ_1 mass on any three sensors; SPARCE places between 2.7 and 14.2 times that share on its top three, so the restoration is not diffusing small corrections across the system but rewriting a few channels.

Table 15: Restoration validity, edit cost and edit concentration, per dataset. Edits are in standard deviations of the normalized data.

	Validity		Per-element edit			Top-3 edit mass
Dataset	count	%	p10	median	p90	share vs. uniform
SWaT	858/1059	81.0	0.32	0.48	1.61	0.476 vs. 0.059
WADI	131/194	67.5	0.36	0.54	1.84	0.335 vs. 0.024
MSDS	12/13	92.3	0.21	0.31	1.05	0.800 vs. 0.300
SMD	1395/1680	83.0	0.29	0.45	1.54	0.490 vs. 0.079

E.4 Cross-detector consistency

Table 16 evaluates SPARCE, CondAttr and ShaTS across five anomaly detection backbones on all four datasets, testing whether attribution quality depends on the upstream detector. The detector only selects which windows are attributed and is never queried during restoration, so the guided diffusion attribution is decoupled from the internal representations of the detector. SPARCE holds the best Recall@3 in 17 of 20 combinations and the best CW-RCS@3 in 18 of 20, and what variation there is falls within the range expected from differences in detection sensitivity, which change the set of windows each backbone presents. This supports the detector-agnostic property claimed in Section 4.

Table 16: Cross-detector results (Recall@3 / CW-RCS@3). Best values for each dataset–detector combination and metric are bold.

Dataset	Method	VAE	AE	LSTM	TCN	Transformer
SWaT	ShaTS	0.393 / 0.122	0.332 / 0.158	0.407 / 0.126	0.462 / 0.138	0.475 / 0.196
	CondAttr	0.537 / 0.245	0.425 / 0.216	0.490 / 0.229	0.555 / 0.284	0.564 / 0.325
	SPARCE	**0.608 / 0.394**	**0.570 / 0.377**	**0.574 / 0.357**	**0.582 / 0.390**	**0.589 / 0.388**
WADI	ShaTS	0.402 / 0.031	0.340 / 0.071	0.416 / 0.035	0.470 / 0.049	0.483 / 0.113
	CondAttr	0.462 / 0.189	0.366 / 0.167	0.422 / 0.177	**0.483** / 0.231	**0.493** / 0.275
	SPARCE	**0.486 / 0.388**	**0.469 / 0.377**	**0.477 / 0.369**	0.460 / **0.385**	0.471 / **0.382**
MSDS	ShaTS	0.915 / 0.462	0.773 / 0.484	0.917 / 0.464	0.925 / 0.472	0.920 / 0.507
	CondAttr	0.948 / 0.551	0.750 / 0.486	0.865 / 0.515	0.904 / 0.574	**0.951 / 0.599**
	SPARCE	**0.956 / 0.595**	**0.923 / 0.578**	**0.939 / 0.566**	**0.950 / 0.590**	0.926 / 0.586
SMD	ShaTS	0.511 / 0.346	0.432 / 0.373	0.522 / 0.349	0.567 / 0.358	0.577 / 0.401
	CondAttr	0.529 / 0.366	0.419 / 0.323	0.483 / 0.342	0.547 / 0.399	0.556 / **0.433**
	SPARCE	**0.609 / 0.435**	**0.588 / 0.423**	**0.598 / 0.414**	**0.576 / 0.432**	**0.590** / 0.428

E.5 Contaminated training data

SPARCE assumes the training split contains only normal behaviour, which rarely holds exactly in deployment. We test how far that assumption can be violated. A fraction c of randomly chosen SWaT training windows is replaced by synthetic faults: a 6σ mean shift, of random sign, applied to one to four randomly chosen sensors over a random contiguous segment covering at least half the window. Shifts are injected in raw units, scaled by each sensor's standard deviation, so they are 6σ after normalization, and the scaler is fitted on the clean data before contamination so that normalization is identical at every level. Faults are generated procedurally from a disjoint random stream and are never copied from test attacks; only the training split is contaminated, leaving validation and test clean. The diffusion model is retrained from scratch at each level and attribution is run on the clean test events with the default configuration.

Table 17 shows that the two quantities degrade at different rates. The self-supervised score is affected directly, since it is trained on the corrupted data: its ROC-AUC falls by 0.082 between a clean training split and one in which two windows in five carry an injected fault. Attribution is more stable over the same range, losing 0.016 Recall@3 and 0.049 R@10. Standard deviations over the three seeds are at most 0.020 for Recall and 0.014 for the AUC.

Table 17: Contamination of the SWaT training split, mean over 3 seeds.

Contam.	AUC of S_θ	R@3	R@5	R@10
0%	0.942	0.608	0.656	0.704
5%	0.938	0.607	0.654	0.704
10%	0.926	0.602	0.646	0.695
20%	0.904	0.604	0.641	0.685
30%	0.893	0.597	0.631	0.665
40%	0.860	0.592	0.621	0.655

Table 18: Guiding with the detector f instead of S_θ.

Dataset	Guide	R@3	R@5	Valid.
SWaT	S_θ	0.608	0.656	81.0
	f	0.612	0.676	91.0
WADI	S_θ	0.486	0.492	67.5
	f	0.481	0.492	88.0
MSDS	S_θ	0.956	1.000	92.3
	f	0.966	1.000	100.0
SMD	S_θ	0.609	0.652	83.0
	f	0.601	0.663	94.2

E.6 Guiding with the detector

The guidance optimizes S_θ, while feasibility is judged by f. Table 18 replaces S_θ by the detector inside equation 11: Recall@3 changes by at most 0.010 on any dataset and the ordering of methods never changes, while validity rises, as expected when the restoration is guided by the criterion that scores it. The surrogate therefore costs little in ranking quality and buys independence from the detector at attribution time.

E.7 Comparison with deviation ranking

Since SPARCE restores from noise with the anomalous window entering only through the guidance cost, one might ask whether the averaged restoration collapses toward the training mean, in which case the attribution would reduce to z-score deviation ranking. Table 19 shows it does not. Deviation ranking, run through the identical event aggregation of equation 15, trails SPARCE by 0.08 to 0.10 Recall@3 on every dataset, and the two rankings correlate only moderately per window (Spearman 0.57 to 0.71). The restoration is also not uniformly near the mean: on annotated sensors the median per-element edit is about 1.1σ, against about 0.25σ on the rest, so the edit concentrates where the anomaly is by a factor of roughly four rather than spreading evenly. This separation is what a magnitude-only baseline cannot produce and is consistent with the concentration reported in Table 15.

Table 19: SPARCE against z-score deviation ranking under the same event aggregation. Edit columns are the median per-element edit $\mathrm{median}_\ell |\boldsymbol{W}^\star - \boldsymbol{W}_{\mathrm{anom}}|$ in standard deviations, split by annotated and non-annotated sensors; their all-sensor median matches Table 15, which these two groups bracket.

Dataset	SPARCE R@3	Deviation R@3	Spearman ρ	Edit (annot.)	Edit (non-annot.)
SWaT	0.608	0.521	0.64	1.11σ	0.23σ
WADI	0.486	0.397	0.57	1.18σ	0.27σ
MSDS	0.956	0.872	0.71	0.96σ	0.18σ
SMD	0.609	0.514	0.61	1.07σ	0.25σ

E.8 Hyperparameter sensitivity

Tables 20 and 21 vary one setting at a time around the default on SWaT, with all other settings fixed. Recall@3 stays within 0.06 of the default across the guidance settings, and the two ends of each sweep fail for interpretable reasons: too small a step or too few restorations leaves the restoration short of the normal manifold, while too large a step overshoots it and edits sensors that were not anomalous. The number of diffusion steps is the least forgiving parameter, since too few denoising

steps cannot reach the manifold at all, and the window length trades temporal context against dilution of the edit over a longer window. Values are single-run; differences under $\sim$0.01, such as n=30 versus n=50, are within the seed variation of Table 12.

Table 20: Guidance settings on SWaT (Recall@3).

Parameter	Value	R@3
η	0.3	0.564
	0.5	**0.608**
	0.7	0.549
λ_m	0.02	0.573
	0.1	**0.608**
	0.2	0.591
n	20	0.565
	30	**0.608**
	50	0.602

Table 21: Scoring level, window and steps (Recall@3).

Parameter	Value	R@3
k'	5	0.543
	10	**0.608**
	15	0.559
	25	0.581
L	25	0.578
	50	**0.608**
	100	0.523
T	5	0.500
	25	0.520
	50	**0.608**

E.9 Aggregation and sparsity penalty

Table 22 compares aggregation strategies on SWaT. Magnitude alone, whether summarized by the mean, the 90th percentile or their combination, misses sensors that are nominated in many windows but never dominate any single one; voting alone recovers those but discards magnitude information entirely. The combination outperforms both, which is why equation 15 keeps them. The same routine is applied to every baseline in the matched comparison of Section 5.3, so this is a property of the readout and not of SPARCE.

Table 23 tests whether channel-level sparsity should be imposed explicitly. A group-sparse $\ell_{2,1}$ penalty over sensors is the natural alternative to the entrywise ℓ_1 of equation 11. It performs worse on both metrics, so the entrywise penalty of equation 11 is not improved by imposing channel-level sparsity explicitly.

Table 22: Aggregation strategies on SWaT.

Strategy	R@3	R@5
Mean + max	0.544	0.583
Mean + p90	0.569	0.618
Voting only	0.534	0.628
p90 + votes	0.579	0.628
Full equation 15	**0.608**	**0.656**

Table 23: Sparsity penalty on SWaT.

Penalty	R@3	CW-RCS@3
$\ell_{2,1}$ (group-sparse)	0.565	0.381
ℓ_1 (entrywise, ours)	**0.608**	**0.394**

E.10 Self-supervised score as a normality signal

Table 24: S_θ as a normality proxy on held-out test windows.

Dataset	ROC-AUC	Spearman ρ with f
SWaT	0.942	0.887
WADI	0.915	0.842
MSDS	0.928	0.855
SMD	0.896	0.864

Table 24 evaluates S_θ on held-out test windows: its ROC-AUC for separating normal from anomalous windows, and its rank correlation with the external detector. Both are high on all four datasets, which

supports using S_θ as the guidance regularizer. It does not establish that the two criteria are equivalent, and Table 18 measures the difference that remains.

E.11 QUALITATIVE EXAMPLES

Figure 3 shows SWaT attack 10, where FIT401 is the only annotated sensor. The attribution map concentrates on FIT401 across all 50 timesteps with near-zero attribution on the remaining 50 sensors, and the restored window corrects that channel toward normal behaviour while leaving the correlated channels almost unchanged, which is the behaviour the sparsity term is meant to produce.

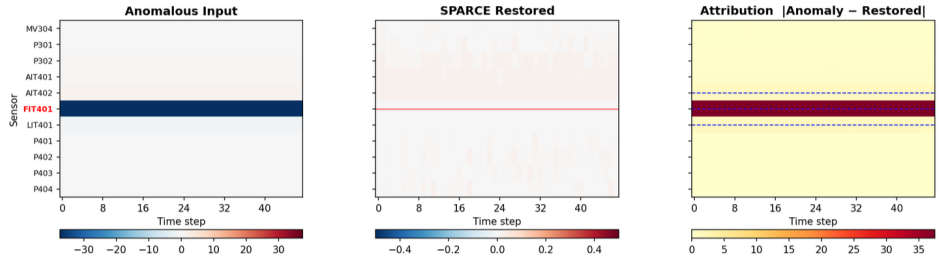


Figure 3: SWaT attack 10 with a single annotated sensor (FIT401, highlighted). Left: anomalous input. Centre: restored window. Right: attribution map $|\boldsymbol{W}_{\text{anom}} - \boldsymbol{W}^\star|$. Colour bars differ because the restored window has a smaller amplitude range than the input.

Figure 4 shows SWaT attack 38, where AIT402 and AIT502 are jointly anomalous. Both annotated sensors are recovered with high attribution across all 50 timesteps while the remaining 49 receive near-zero attribution, in a single guided pass and without per-sensor enumeration, so the joint optimization extends to multi-sensor events rather than resolving one sensor at a time.

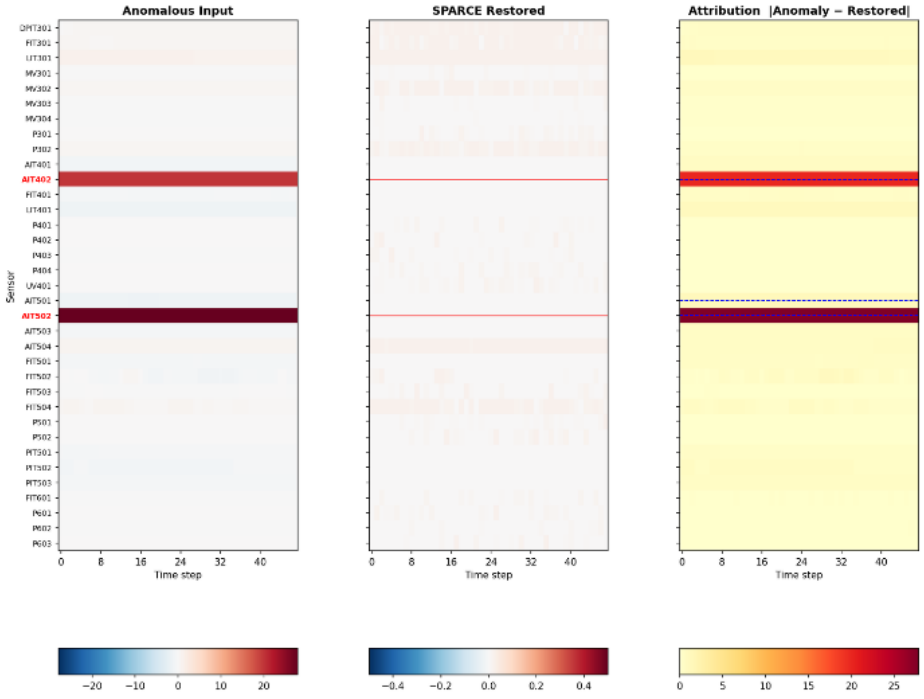


Figure 4: SWaT attack 38 with two annotated sensors (AIT402 and AIT502, highlighted). Both are recovered in one guided pass, with near-zero attribution on the remaining sensors.

F STRUCTURAL-VAR BENCHMARK

Why a synthetic benchmark. On the real datasets the annotations mark every sensor behaving anomalously during an event, so recovery of the initiating fault cannot be measured: a method that returns the full observable support scores perfectly whether or not it distinguishes the root. This benchmark removes that ambiguity by construction. Faults are injected at known nodes, only those nodes are labelled, and attributing an affected descendant counts as an error.

Generator. We simulate a linear structural VAR over $K = 20$ sensors with Gaussian innovations, partitioned into 3 roots, 13 nodes forming propagation chains from the roots, and 4 distractors independent of them. Every node carries a lag-1 self-term with coefficient 0.8, shared across nodes, and every cross-sensor edge has a lag drawn from $\{1, 2, 3\}$, so the graph contains no contemporaneous edges. This extends the propagation setting of Proposition A.4: an intervention at a root propagates along its chain, so the observable deviation support is strictly larger than the intervention support and the two can be scored separately. The graph is drawn once and held fixed; only the innovations and the faults are resampled per window.

Faults and labels. A fault is a positive mean shift of 4 standard deviations of the affected sensor, applied to a root for the whole window. We generate 200 single-root and 200 two-root windows, and label only the injected roots. Normal training windows come from the same generator with no intervention, and the diffusion model is retrained on them from scratch.

Protocol. Every method is evaluated per window on this benchmark: attribution is the per-window readout of equation 6, without the event-level aggregation of equation 15, since each window is an independent fault. SPARCE uses the hyperparameters of Table 10 unchanged, and nothing was tuned here. The comparison is deviation ranking, which scores sensor j by $L^{-1} \sum_{\ell} |(\boldsymbol{W}_{\mathrm{anom}})_{j,\ell} - \mu_j| / \sigma_j$ with the training mean and standard deviation, the natural baseline when no restoration is performed.

Metrics. Root Recall@r is computed against the injected root set, at $r = 1$ for single-root windows and $r = 2$ for two-root windows, so the budget never exceeds the number of injected roots. Separation rate is the fraction of (window, root) pairs in which the root receives a strictly higher attribution than all of its direct children; distractors are excluded, ties count as failures, and roots without direct children are skipped.

Table 25: Structural-VAR benchmark. Labels are injected roots only; descendants count as errors. Mean and standard deviation over model-training seeds on one generator draw.

Metric	SPARCE	Deviation ranking
Root Recall@1, single root	$\mathbf{0.865 \pm 0.007}$	0.417
Separation rate	$\mathbf{0.768 \pm 0.015}$	0.368
Root Recall@2, two roots	$\mathbf{0.728 \pm 0.020}$	0.080

Results. SPARCE recovers the injected root in 86.5% of single-root windows at a budget of one, against 41.7% for deviation ranking and 5% for a random choice among the 20 candidates, and it ranks a root above its direct children in roughly three quarters of the pairs, more than twice the baseline rate. With two injected roots and a budget of two the gap widens: 0.728 against 0.080, the latter below the 0.10 expected from a random top-2, so deviation ranking systematically places descendants above the roots that caused them. The two scores answer different questions. Deviation magnitude ranks whatever moved most, which under propagation need not be the intervened node; the guided restoration ranks whatever had to be edited for the window to become normal again, and in this generator that recovers the intervention targets. Joint restoration uses learned cross-sensor dependencies, whereas deviation ranking scores each sensor in isolation. No method receives any information identifying the root nodes: all 20 sensors are candidates, and the budget never exceeds the number of injected roots, so nominating the three structural roots in every window cannot succeed.

Scope. These results establish that the readout recovers intervention targets under propagation, in a setting where the ground truth is known by construction and the baseline fails. They do not

transfer automatically to the real benchmarks, whose generating process is unknown and whose annotations cannot separate roots from effects. The generator is linear with Gaussian innovations, a single fixed graph and lagged edges only, and the reported deviations are over model-training seeds on one generator draw.