When Samples Look Normal but Transitions Do Not: Relational Anomaly Detection in Dynamical Logs

Joanna Piasek-Skupna[1][0000−0001−6621−8879]

Poznan University of Technology, Jacka Rychlewskiego 1, 61-131 Poznań, Poland
joanna.piasek@put.poznan.pl

Abstract. Logged sensorimotor datasets can contain transitions in which the individual states and controls are plausible, while their joint temporal relation is invalid. Such errors arise from sensor misalignment, delayed controls, or trajectory segments stitched together at similar states. Because the corrupted variables may preserve their per-channel distributions and local jump statistics, the anomaly is not necessarily pointwise; it lies in the transition relation $((x_t, u_t, x_{t+1}))$.

We formulate this setting as transition-level relational anomaly detection in dynamical logs. To evaluate it, we introduce the matched-stitch stress test: trajectory segments are joined at matched state, control, and step magnitude, making the corrupted tuples difficult for marginal, geometric, and nearest-neighbor detectors while breaking conditional dynamics. We score transition compatibility using one-step prediction residuals from learned dynamics models, including EDMDc, linear, MLP, and joint-Gaussian baselines.

Across four simulated systems and two real MuJoCo logs, we find that successful detection is largely model-agnostic. Conditional residual models expose matched-stitch inconsistencies, but EDMDc is not consistently superior: linear residuals and joint-Gaussian scores are often competitive, and a simple linear model frequently suffices. These results suggest that the central issue is not a particular Koopman representation, but whether anomaly detection is performed on isolated samples or on the transition relation.

Finally, we test a natural hypothesis: that Koopman eigenmodes provide a structured basis for classifying corruption types. Controlled ablations show no modal advantage. The apparent benefit of Koopman modal energy is explained by the modulus operation applied to residual coordinates; applying the same absolute-value transformation to linear, MLP, or lifted residuals matches or exceeds the modal representation. We report this as a negative result: residual magnitude patterns, not Koopman modal structure, explain corruption-type separability for the studied setting. We close by characterizing when transition residuals indicate data corruption and when they instead reflect model inadequacy.

Keywords: Relational anomaly detection · Dynamical systems · Koopman operator · Offline reinforcement learning · Log integrity

1 Introduction

Offline datasets of control and sensorimotor trajectories are used to fit dynamics models, train offline reinforcement learning and imitation policies, and perform system identification. These datasets are often assembled from heterogeneous sources: logs from several runs or machines, files merged by separate processes, and streams whose state and control channels are recorded asynchronously. Such data can contain corruptions that are relational rather than pointwise. The recorded values may each be plausible, while the tuple (x_t, u_t, x_{t+1}) violates the temporal relation imposed by the system dynamics.

Most anomaly detectors score individual samples, local distances, or marginal statistics. For logged dynamical data, the relevant question can instead be conditional:

$$(x_t, u_t, x_{t+1}) \overset{?}{\sim} p(x_{t+1} \mid x_t, u_t). \tag{1}$$

A transition may contain ordinary states and controls but still be inconsistent with the recorded input and the preceding state. Marginal detectors, jump-size tests, and nearest-neighbor scores can miss this case when the corruption preserves those quantities. For example, two pendulum swing-up episodes may pass through similar angles and angular velocities while following different controls or energy phases. Replacing the successor state of one transition with the successor from the other can preserve realistic states and a realistic step size, while producing a tuple that is incompatible with the recorded control. This paper studies anomalies at the transition level.

We evaluate transition consistency using the one-step residual of an extended dynamic mode decomposition with control (EDMDc) model. The residual measures dynamic inconsistency relative to the learned model. A large residual does not, by itself, identify data corruption: it may also indicate a real unmodeled event or a poor dynamics model. Our experiments therefore use injected corruptions with known ground truth. This allows us to evaluate detection under controlled conditions while keeping the deployment interpretation limited: in real logs, a large residual should be treated as evidence of inconsistency, not as automatic proof of corruption.

Prediction residuals are not the novelty here; they are the simplest way to test whether a recorded transition is compatible with a learned dynamics model. The benchmark is what makes the test nontrivial. In a state-matched stitch, two trajectory segments are joined only when the boundary states, controls, and one-step displacement magnitudes are close, so the corrupted transition has realistic endpoints and a realistic jump size and detectors based on marginal statistics, Euclidean jumps, or nearest neighbors have little signal to use. A detector must instead evaluate whether the observed successor follows from the recorded predecessor and control. We evaluate EDMDc because Koopman lifting gives a structured residual representation, and a reasonable hypothesis was that this structure might aid detection or corruption typing. The experiments do not support that hypothesis: across systems, linear, MLP, EDMDc, and joint-Gaussian models trade places, a linear residual is often enough, and the apparent advantage of Koopman modal energy disappears once the same absolute-value transformation is applied to baseline residuals. The useful object is the transition residual itself, not Koopman modal structure.

Real-world relevance. Relational corruptions arise in logged cyber-physical data when streams are recorded or merged independently. A state logger and a control logger may have a timestamp offset; actuator commands may be buffered and written later than the transition they caused; packets may be dropped or duplicated; and trajectory files from multiple experimental runs may be concatenated with an erroneous boundary. Similar inconsistencies can appear in replay or log-tampering attacks, where inserted values are copied from real trajectories and remain plausible under marginal checks. In these cases, the invalid object is the transition tuple rather than an individual recorded value.

Contributions. We (i) formulate logged-trajectory integrity as a conditional dynamics-consistency test for tuples (x_t, u_t, x_{t+1}), targeting relational corruptions that preserve state and control marginals (Sect. 3); (ii) construct a marginal-preserving corruption benchmark across controlled dynamical systems (swaps, control delays, sensor lags, and trajectory stitches) with a strict-split, label-free calibration protocol that sets per-regime thresholds on clean episodes (Sects. 5, 7); (iii) identify matched-stitch detection as the hardest stress test, where matching the seam on state, control, and transition magnitude drives geometric, marginal, and nearest-neighbor baselines to chance, while conditional dynamics models remain informative (Sect. 6); and (iv) show that detection is model-agnostic—linear, MLP, EDMDc, and joint-Gaussian models trade places, and that Koopman modal residuals give no corruption-typing advantage once the absolute-value transformation on residual coordinates is controlled for (Sect. 8).

2 Related Work

Prediction-residual anomaly detection. Detecting anomalies from one-step prediction error is mature, spanning autoregressive, state-space, recurrent, and autoencoder models. Hierarchical residual cascades such as the stacked interpretable-component model of [6] remove linearly predictable structure first and detect anomalies in the remainder. Our setting is a *contextual* anomaly in the sense of [10,11]: the context is (x_t, u_t) and the behavioral attribute is the successor x_{t+1}, so a transition can be anomalous in context while every value is marginally ordinary. Our contribution is not the residual principle but its use for relational, marginal-preserving corruption and the structured analysis the Koopman form enables.

Koopman models for forecasting and detection. EDMD [1] and its control extension EDMDc [2] fit a linear operator in a lifted observable space. Koopa [4] disentangles time-invariant and time-varying dynamics in stacked residual blocks for forecasting; KoopAGRU [5] transfers this decomposition to anomaly detection from reconstruction error. Gottwald and Gugole [7] interpret DMD reconstruction error as evidence that the current dynamics no longer fit a low-dimensional regime. Deep Koopman embeddings [3] learn the observable dictionary; we use a fixed dictionary for a stable, reproducible operator (Sect. 10).

Model-based fault detection. Residual generation against a nominal model is the classical basis of fault detection and isolation (FDI), where structured residuals and statistical decision rules like parity-space generators [13] and generalized-likelihood-ratio tests [12] map faults to residual directions. FDI assumes a known nominal model and an online physical system; we target offline *log* integrity, where the physical system may have behaved normally, but the stored tuple is invalid due to logging, synchronization, or merging errors, and the model is learned, not given. We connect to this lineage in Sect. 8; a direct comparison to parity-space and likelihood-ratio detectors is left to future work.

Offline-data quality and practical use cases. Filtering offline RL or imitation data is usually driven by return or behavior-policy density. Dynamics consistency is orthogonal to return: a high-return trajectory may contain corrupted transitions and a low-return one may be perfectly consistent. We therefore target a quality dimension distinct from task performance, and the detector is naturally a pre-training audit of logged (x_t, u_t, x_{t+1}) tuples. Offline RL learns from static, often heterogeneous transition datasets [14]; recent work analyzes offline model-based learning when the recorded *actions* are corrupted [16]—exactly the case where a logged action is incompatible with the observed successor. Real-robot datasets aggregate thousands of trajectories across tasks [17], where local logging errors corrupt tuples without making any single value implausible. In industrial cyber-physical systems, temporal inconsistency between sensing, computation, and actuation is a known safety concern [18], and water-process testbeds such as SWaT [19] and WADI [20] log sensor and actuator histories for anomaly detection, so settings for which our quadruple-tank is a simplified controlled analogue.

3 Problem Formulation and Method

Let $\mathcal{D} = \{(x_t, u_t, x_{t+1})\}_{t=1}^{N}$ be a logged dataset with states $x_t \in \mathbb{R}^n$ and controls $u_t \in \mathbb{R}^m$. Under the null hypothesis a transition is generated by the system dynamics,

$$H_0: \; x_{t+1} \sim p(\cdot \mid x_t, u_t), \qquad H_1: \; (x_t, u_t, x_{t+1}) \text{ is dynamically inconsistent.} \tag{2}$$

We learn observables $\psi : \mathbb{R}^n \to \mathbb{R}^p$ and operators $K \in \mathbb{R}^{p \times p}$, $L \in \mathbb{R}^{p \times m}$ via EDMDc—a polynomial dictionary with $[K\, L]$ fit by least squares on the lifted data—such that $\psi(x_{t+1}) \approx K\psi(x_t) + Lu_t$, and define the lifted residual $e_t = \psi(x_{t+1}) - K\psi(x_t) - Lu_t$. The integrity score restricts e_t to the state-recovery coordinates of ψ (the *state read-out*),

$$s_t = \|P_x\, e_t\|_2 = \|P_x(\psi(x_{t+1}) - K\psi(x_t) - Lu_t)\|_2\,, \tag{3}$$

where P_x selects the lifted coordinates equal to the raw state. This makes the score directly comparable to a state-space forward-model error and avoids inflating it with dictionary-feature error; a large score signals that x_{t+1} is anomalous *given* x_t and u_t (a conditional, not a marginal, statement).

Regime normalization. Residual scale varies across operating regimes. We assign each transition a regime by physical definition where available (e.g. pendulum energy bands, tank level bands) or by clustering source states, and convert the raw residual to a robust per-regime z-score using the median and MAD of clean reference residuals. This removes the confound by which a naturally high-energy regime looks anomalous.

Strict split and label-free thresholds. Episodes are partitioned into disjoint train, clean-calibration, and test sets. The dynamics model is fit on clean train episodes; per-regime detection thresholds are set on clean calibration residuals at a target false-positive rate α; corruptions are injected only into test. No transition is scored by a model trained on it, and threshold setting uses no corruption labels.

Modal residual (tested in Sect. 8). Eigendecomposing $K = V \Lambda V^{-1}$, the residual in eigencoordinates $\tilde{e}_t = V^{-1} e_t$ assigns the violation energy to individual Koopman modes. The vector of per-mode magnitudes is the *mode signature*, a candidate structured representation for typing corruptions; Sect. 8 tests whether it offers any advantage over simpler residual representations and finds that it does not.

Robust fit. To resist training-set contamination we optionally refit after trimming the largest-residual fraction of training rows.

4 Experimental Setup

Systems. We choose four controlled systems spanning a spectrum of how well a fixed polynomial dictionary spans the true dynamics: (i) a *switching-linear* system alternating between a stable node and spiral under persistent excitation (trivially spanned); (ii) a forced double-well *Duffing* oscillator (exactly spanned by a degree-3 dictionary); (iii) *pendulum* swing-up with continuous torque and energy-based regimes (trigonometric; approximately spanned); and (iv) a *quadruple-tank* process [8] with $\sqrt{h}$ outflow (not spanned by any polynomial dictionary, and sparsely sampled). We avoid contact-rich locomotion, where footstrike discontinuities a global model cannot fit would masquerade as corruption.

Corruptions. All headline corruptions are built from real measured values so that per-channel marginals are preserved by construction (verified by a two-sample Kolmogorov–Smirnov test). Each models a realistic logging failure in which every value stays in-distribution and only the transition relation is violated: temporal next-state swap and control mismatch (dropped/duplicated packets); timestamp misalignment as a one-step control delay (async logging); joint sensor lag on the $(\cos\theta, \sin\theta)$ channels, lagged together to preserve $\cos^2 + \sin^2 = 1$ (sensor write delay); trajectory stitch (multi-run merge); and a state-matched stitch (replay/log tampering). A large additive spike is included only as a positive control detectable by any method.

Baselines. Per-channel z-score and Isolation Forest [9] (marginal); Mahalanobis distance on the transition vector (a joint-Gaussian one-step model); a linear state-space model $x_{t+1} = Ax_t + Bu_t$; a nearest-transition kNN predictor; and an MLP forward model. The linear and kNN baselines are the pointed tests: if either matches EDMDc, the gain is not Koopman-specific.

Metrics. Because corruptions are rare we report the area under the precision–recall curve (AUPRC), the expected calibration error of the realized FPR (ECE-FPR) at a target α, and macro-F1 for corruption-type attribution.

Protocol details. We study controlled simulated systems precisely so that ground truth is known and corruption is marginal-preserving by construction; in uncontrolled logs a large residual would conflate corruption, unmodeled events, and model inadequacy. All splits are episode-disjoint (train / clean-calibration / test); the calibration set is entirely clean and corruptions are injected only into test. EDMDc uses a fixed polynomial dictionary of degree 2 for the switching-linear, pendulum, and quadruple-tank systems (6, 10, and 15 observables for the 2-, 3-, and 4-dimensional state spaces) and degree 3 for Duffing so that the cubic restoring term is represented exactly (10 observables on its 2-dimensional state), fit by ordinary least squares, with an optional trimmed refit (discarding the top $\max(r, 0.05)$ residual fraction at contamination rate r). The MLP forward model has two hidden layers of 64 units with early stopping on a clean validation split; the kNN baseline uses k=5 over standardized (x, u). Regimes are physical where available (four pendulum energy/angle bands; three quadruple-tank level bands by tertiles) and k-means otherwise; the target FPR is α=0.05. Matched stitches are formed by nearest-neighbor search over standardized features: state, optionally control, and $\|x_{t+1} - x_t\|$, taking the first candidate from a different episode (same regime at the regime level). Stitch results are mean±std over seeds (Table 2); other tables report a single representative seed. The injected corruption rate is 9% for mixed-detection tables and 5% for stitch tasks; each simulated dataset has $\sim$19–24k transitions (60–120 episodes), and the real datasets have 15k (Reacher) and 60k (Swimmer). Regime normalization and per-regime calibration are applied identically to every residual method in every table. Attribution macro-F1 is from a multinomial logistic-regression classifier with 5-fold cross-validation at matched representation dimensionality. All tables are produced by one consolidated driver, so every reported number is reproducible from a single run.

5 Detection and Marginal Invisibility

Table 1 reports detection AUPRC under the strict split for a mixture of marginal-preserving corruptions. Marginal and density detectors sit near chance on all

Table 1. Detection AUPRC (strict split, mixed marginal-preserving corruption). Marginal/density methods sit at chance (≈ 0.09, the 9% base rate); transition models succeed; EDMDc improves with dictionary adequacy, reaching exact detection on Duffing and tying simpler models where the dictionary is linear or inadequate.

Method	Synthetic	Duffing	Pendulum	Quadruple-tank
z-score (marginal)	0.10	0.10	0.10	$\approx$ chance
Isolation Forest (density)	0.10	0.10	0.27	$\approx$ chance
Mahalanobis (joint-Gaussian)	0.98	0.78	0.80	0.63
Linear $Ax + Bu$	0.95	0.80	0.64	0.61
MLP forward	0.95	0.88	0.79	0.57
EDMDc	0.95	**1.00**	**0.95**	0.66

systems, confirming the corruptions are invisible to methods that ignore the transition. Methods that model the transition succeed in proportion to how well they capture the dynamics: on the near-linear synthetic system the linear model, MLP, EDMDc, and the joint-Gaussian Mahalanobis detector tie; on the nonlinear pendulum the linear model falls behind and EDMDc is strongest. Any Koopman advantage in detection thus scales with nonlinearity, and the strong Mahalanobis result reinforces the key distinction—conditional transition modeling versus marginal scoring, not Koopman versus classical anomaly detection: when a joint one-step Gaussian approximation is adequate it is itself a competitive detector.

5.1 Asynchronous-Logging Severity

Timestamp misalignment is a common fault in real logged control data, arising when state and control streams are joined with a clock offset or one sensor channel is written with stale values. Sweeping the offset on the pendulum, the z-score stays at chance (≈ 0.07) at every offset, since each mispaired value is itself real, while the EDMDc residual detects the inconsistency and its detectability grows monotonically with the offset—from 0.83 to 0.91 for control delays of 1–3 steps and from 0.92 to 0.96 for sensor lags—as the mispaired control or stale sample decorrelates further from the correct one. The same trend holds on the quadruple-tank for sensor lag; there, control delay is weakly detectable because the per-step actuator effect is small relative to the autonomous dynamics, an instance of the excitation precondition discussed in Sect. 10.

6 Dynamically Inconsistent Stitch Detection

Our lead task is detecting trajectory *stitches*: a recorded successor x_{t+1} is replaced by the successor x_{s+1} of a state $x_s \approx x_t$ from a different episode. This models a multi-run file merge (logs concatenated or sorted on an incorrect key) and, when the substituted values are copied from valid trajectories, a replay or log-tampering attack. Genuine episode *resets* produce a large state jump (7–100×

Table 2. Matched-stitch detection, AUPRC mean ± std over seeds at the state+control level. EDMDc leads with non-overlapping margins; on the polynomial Duffing system, where the dictionary is exact, it is perfect.

System	jump	z-score	kNN	linear	MLP	EDMDc
Pendulum	0.08±0.01	0.05±0.00	0.10±0.02	0.27±0.02	0.36±0.02	**0.57±0.04**
Duffing	0.26±0.02	0.05±0.00	0.14±0.02	0.52±0.02	0.80±0.01	**1.00±0.00**

the within-episode step) that a trivial $\|x_{t+1} - x_t\|$ detector solves, so they are not the contribution; the interesting case is *matched stitches*, where candidates are matched on transition magnitude as well as state, so $\|x_{t+1} - x_t\|$ carries no cue.

We define a severity hierarchy by what the match must preserve: random → state → state+control → state+control+regime. On the pendulum, once the seam is magnitude-matched the jump and z-score baselines fall to chance (≤ 0.10). The nearest-transition kNN baseline is *also* at chance (≈ 0.05): the substituted successor is a genuine successor of a nearby state, so nearest-neighbor lookup cannot distinguish it. Detection therefore requires a model that *generalizes* the dynamics, and AUPRC rises strictly with model power, kNN < linear < MLP < EDMDc, from ≈ 0.05 to ≈ 0.60 at the matched levels. Across seeds (Table 2) EDMDc leads the MLP with non-overlapping intervals, so the Koopman advantage is seed-supported on this clean, nonlinear system.

Dictionary adequacy spectrum. The two clean-stitch systems mainly differ in dictionary adequacy. On Duffing the degree-3 dictionary spans the cubic dynamics exactly, the clean residual is near machine precision, and matched-stitch detection is perfect (1.00); the pendulum's trigonometric dynamics are only approximately spanned, so detection is strong but bounded (0.57); the quadruple-tank's $\sqrt{h}$ outflow is not spanned by any polynomial dictionary *and* its state is sampled too sparsely relative to the per-step motion ($\Delta h \approx 0.25$), so the nearest matchable state is still ≈ 2 units away, the jump ratio stays 7–9×, and the seam remains geometrically visible. The tank is therefore not a failed third example but the informative low end of the spectrum: subtle stitch detection requires both dense state coverage and a dictionary that spans the dynamics. We present the Duffing perfect score as a controlled upper bound, not a claim of universal perfection.

7 Regime-Aware Label-Free Calibration

In a realistic data-cleaning setting, corruption labels are unavailable, so a deployable detector must choose its threshold from clean data alone. We set per-regime thresholds at $\alpha = 0.05$ using clean calibration episodes and measure the realized false-positive rate per regime on clean test episodes, reporting ECE-FPR as the mean absolute deviation of these rates from α. Per-regime thresholding helps when residual scales differ across regimes: on the pendulum, whose energy regimes have very different scales, it reduces ECE-FPR by 4–6 times across methods (e.g.

EDMDc $0.023 \rightarrow 0.006$, linear $0.059 \rightarrow 0.010$), with a smaller improvement on Duffing ($0.051 \rightarrow 0.027$). The gain is not universal—on the synthetic system, and for EDMDc on the quadruple-tank, it gives no improvement and can slightly increase ECE-FPR. Two conclusions follow: regime-aware calibration is useful mainly under regime-scale heterogeneity, and it is a property of the scoring protocol, not of the Koopman model—any scalar forward-model residual can be calibrated per regime identically. Neither calibration nor corruption typing (Sect. 8) is specific to the Koopman form; EDMDc is one conditional transition model among several.

8 When Does Koopman Modal Structure Help?

A natural hypothesis is that the *structure* of the EDMDc residual, not only its size, should help characterize a corruption. Koopman eigenmodes are coherent dynamical components with distinct timescales, so a corruption violating a slow, persistent mode and one violating a fast, oscillatory mode are dynamically different events; projecting the residual onto the eigenbasis and reading per-mode energy therefore looks like a principled, timescale-indexed representation for typing corruptions. We test this and find it holds only under a condition that simpler representations already meet, so the modal representation provides no advantage on the systems studied.

The hypothesis has a precise failure condition. The modal coordinates are $z = V^{-1}\rho$, an *invertible linear* map of the residual ρ. A linear decision rule is invariant to invertible linear maps, so the eigenbasis rotation *cannot* create separability that is not already present in ρ: any modal advantage must come from a *nonlinear* readout of the modal coordinates—in the standard signature, the per-mode energy $|z_j|$. The modulus is the operative ingredient, not the basis. This is directly testable: apply the same modulus to the *raw* residual coordinates and to other models' residuals (Table 3). Across four systems, the signed residual is near-chance for typing corruption; applying $|\cdot|$ to *any* residual—raw lifted, linear, MLP, or Koopman-modal—raises macro-F1 to a comparable band, and the Koopman modal energy is never the best representation. A plain linear-model residual magnitude ties or beats it on every system (e.g. 0.95 vs. 0.88 on Swimmer). The eigenbasis rotation, applied without the modulus, ties the raw residual, as the invariance argument predicts.

The takeaway is a scoping result, not a method. Corruption *type* is recoverable from the *magnitude* pattern of a one-step residual, but this is model-agnostic and not Koopman-specific; given channel-structured corruptions it is close to expected, since different faults leave residual energy on different coordinates. We report it as a practical diagnostic rather than a contribution. Whether modal structure would genuinely help—under nonlinear readouts, or for systems whose corruption signatures align with eigenmode timescales rather than raw coordinates—is left open.

Table 3. Corruption-type macro-F1 at matched dimensionality. The signed residual is near-chance; the gain comes from the modulus $|\cdot|$ applied to *any* residual, not from the Koopman eigenbasis—the modal energy is never the column-best. (–: not run.)

Representation	Pendulum	Duffing	Reacher	Swimmer
signed residual	0.24	0.28	0.45	0.53
|lifted residual|	0.49	**0.71**	**0.82**	0.80
|linear residual|	**0.54**	0.64	–	**0.95**
|MLP residual|	0.43	0.52	–	0.83
Koopman modal energy	0.53	0.46	0.78	0.88

9 Transfer to Real Logged Datasets

We repeat the benchmark on two real logged-trajectory datasets from Minari [15]—MuJoCo Reacher (a two-link arm, 6-D dynamical state) and Swimmer (a three-link body in viscous fluid, 8-D)—injecting the same labeled corruptions into held-out episodes while preserving marginals (KS $p \approx 1$). We choose these non-contact systems deliberately: contact-rich locomotion would reintroduce footstrike discontinuities that can mimic corruption.

The two main findings transfer to real logs. Marginal and density detectors remain near chance (z-score and Isolation Forest AUPRC $\leq$0.24), and the matched stitch is again subtle (seam/step ratio 1.0 on Swimmer, 1.6 on Reacher), so jump, z-score, and kNN scores stay at chance and a generalizing conditional model is required to expose the seam. Consistent with simulation, no single conditional model dominates: on Reacher the linear model leads the matched-stitch task (0.78 vs. EDMDc 0.71) and a joint-Gaussian baseline leads detection, while on Swimmer the MLP leads (0.33 vs. EDMDc 0.08). Neither system is preferentially spanned by a low-degree polynomial dictionary, so EDMDc has no advantage. The real-data takeaway is the same as in simulation but sharper: the relational problem is robust and the appropriate detector is model-agnostic.

10 Limitations and Future Work

The residual measures *model-relative* inconsistency, which can arise from data corruption, a real unmodeled event (a collision, a mode switch), or model inadequacy. Our injected-label benchmark is what allows attributing residuals to corruption; field deployment cannot. Concretely: (i) slow sensor drift is absorbed into the model and is not detected, what we report as an explicit negative result; (ii) stitch detection is informative only when state coverage is dense relative to the step size, as the quadruple-tank shows; (iii) a fixed polynomial dictionary underfits strongly nonlinear dynamics such as $\sqrt{h}$ outflow, capping detection there. We use a fixed dictionary deliberately, for a stable and reproducible operator; a learned (deep) Koopman embedding would improve fit at the cost of a latent eigenbasis that varies across training runs. Training-set contamination is a further concern: cross-fitting alone does not prevent a model from learning corruption present throughout training, but refitting after trimming the largest-residual fraction restores robustness (on the pendulum the plain EDMDc detector degrades $0.95 \rightarrow 0.61$ at 30% contamination while the trimmed fit holds at 0.87, with the same ordering on all four systems). A further extension is to control-affine and bilinear Koopman embeddings: systems such as the unicycle, whose exact finite-dimensional closure on $\{x, y, \cos\theta, \sin\theta\}$ is bilinear in the control, are not represented by the additive $K\psi + Lu$ form used here, and integrity checking with a bilinear operator is a natural next step toward mobile-robot odometry–command logs.

Several comparisons are deliberately left open. Our score is a *one-step* residual; multi-step rollouts or state-estimator residuals (Kalman-filter innovations, fixed-interval smoothers) could strengthen detection in low-excitation regimes such as the quadruple-tank, where a single step barely moves the state. We compare against deterministic forward models and a joint-Gaussian score, but not against probabilistic conditional density models (normalizing flows or conditional VAEs that estimate $p(x_{t+1} \mid x_t, u_t)$ directly), nor against classical model-based fault detection with structured residuals and statistical decision rules (parity-space generators, generalized-likelihood-ratio tests) [12,13]. These are the natural strong baselines for a future study and we do not claim superiority over them. Finally, our real-data evaluation injects *labeled* corruption into real clean logs to retain ground truth; it therefore does not measure detection of *uncontrolled*, naturally occurring anomalies, which bounds external validity and is the main open question for deployment.

11 Conclusion

We framed logged-trajectory integrity as a transition-level *relational* anomaly problem, in which the anomaly is not visible in x_t, u_t, or x_{t+1} separately but in the relation between them. The clearest example is the matched stitch, where two segments are joined so that the seam matches state, control, and step magnitude, removing the geometric cue and making marginal or nearest-neighbor scoring insufficient. Across four simulated systems and two real logged datasets, the best detector is not tied to one model class: EDMDc, linear, MLP, and joint-Gaussian residuals trade places depending on how well the representation captures the local dynamics, and a simple linear residual is often sufficient. The main distinction is conditional transition modeling versus marginal scoring, not Koopman versus classical anomaly detection. We also tested whether a Koopman residual's structure aids corruption typing and found it does not—the useful signal comes from a residual-magnitude nonlinearity applicable to any conditional model, not from the eigenbasis. The contribution is therefore a reframing of the anomaly-mining object: for logged dynamical data, integrity analysis should operate on the transition relation rather than on individual samples, after which the choice of dynamics model matters less than whether the detector scores the right object.

References

1. Williams, M.O., Kevrekidis, I.G., Rowley, C.W.: A data-driven approximation of the Koopman operator: extending dynamic mode decomposition. J. Nonlinear Sci. **25**(6), 1307–1346 (2015)
2. Proctor, J.L., Brunton, S.L., Kutz, J.N.: Dynamic mode decomposition with control. SIAM J. Appl. Dyn. Syst. **15**(1), 142–161 (2016)
3. Lusch, B., Kutz, J.N., Brunton, S.L.: Deep learning for universal linear embeddings of nonlinear dynamics. Nat. Commun. **9**, 4950 (2018)
4. Liu, Y., Li, C., Wang, J., Long, M.: Koopa: learning non-stationary time series dynamics with Koopman predictors. In: NeurIPS (2023)
5. Ait Yahia, I., Berrada, I.: KoopAGRU: a Koopman-based anomaly detection in time-series using gated recurrent units. arXiv:2501.17976 (2025)
6. Zancato, L., Achille, A., Paolini, G., Chiuso, A., Soatto, S.: Stacked residuals of dynamic layers for time series anomaly detection. arXiv:2202.12457 (2022)
7. Gottwald, G.A., Gugole, F.: Detecting regime transitions in time series using dynamic mode decomposition. J. Stat. Phys. **179**, 1028–1045 (2020)
8. Johansson, K.H.: The quadruple-tank process: a multivariable laboratory process with an adjustable zero. IEEE Trans. Control Syst. Technol. **8**(3), 456–465 (2000)
9. Liu, F.T., Ting, K.M., Zhou, Z.-H.: Isolation forest. In: ICDM, pp. 413–422 (2008)
10. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: a survey. ACM Comput. Surv. **41**(3), 1–58 (2009)
11. Song, X., Wu, M., Jermaine, C., Ranka, S.: Conditional anomaly detection. IEEE Trans. Knowl. Data Eng. **19**(5), 631–645 (2007)
12. Willsky, A.S., Jones, H.L.: A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. IEEE Trans. Autom. Control **21**(1), 108–112 (1976)
13. Gertler, J.: Fault Detection and Diagnosis in Engineering Systems. Marcel Dekker (1998)
14. Fu, J., Kumar, A., Nachum, O., Tucker, G., Levine, S.: D4RL: datasets for deep data-driven reinforcement learning. arXiv:2004.07219 (2020)
15. Younis, O.G., et al.: Minari: a standard format for offline reinforcement learning datasets. Zenodo (2024). `https://doi.org/10.5281/zenodo.13767625`
16. Li, H., Gangwani, T., Ying, L.: An analysis of offline model-based learning with action noise. J. Sci. Comput. **103**(2), 51 (2025)
17. Zhou, G., Ke, L., Srinivasa, S., Gupta, A., Rajeswaran, A., Kumar, V.: Real world offline reinforcement learning with realistic data source. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 7176–7183 (2023)
18. Kopetz, H., Steiner, W.: Temporal consistency of data and information in cyber-physical systems. arXiv:2409.19309 (2024)
19. Mathur, A.P., Tippenhauer, N.O.: SWaT: a water treatment testbed for research and training on ICS security. In: CySWater (2016)
20. Ahmed, C.M., Palleti, V.R., Mathur, A.P.: WADI: a water distribution testbed for research in the design of secure cyber-physical systems. In: CySWATER, pp. 25–28 (2017)