Physics-Informed Defenses against Poisoning Attacks on Learning-based Anomaly Detection in Industrial Control Systems

Simeon Grossmann
simeon.grossmann@alumni.kit.edu
KASTEL Security Research Labs
Karlsruhe Institute of Technology
Karlsruhe, Germany

Alessandro Erba
alessandro.erba@kit.edu
KASTEL Security Research Labs
Karlsruhe Institute of Technology
Karlsruhe, Germany

Christian Wressnegger
christian.wressnegger@kit.edu
KASTEL Security Research Labs
Karlsruhe Institute of Technology
Karlsruhe, Germany

Abstract

Process-based anomaly detection has been widely studied in industrial control systems (ICS) as a means to identify deviations from expected physical behavior. However, machine-learning-based detectors are vulnerable to poisoning attacks, in which malicious samples are injected into the training data, degrading detection performance, so that the detector misses alarms at test time. In this work, we propose physics-informed defenses against model poisoning in ICS anomaly detection. We leverage continuous physical process equations to identify inconsistencies between sensor measurements and the expected physical evolution of the system in order to build two complementary defenses: (1) residual-based filtering, removing physically inconsistent samples from the training data, and (2) physics-informed regularization, discouraging the learning of poisoned behaviors. We evaluate the proposed defenses on the SWaT testbed and show that incorporating physical process knowledge can mitigate poisoning attacks while preserving the detection performance of the underlying anomaly detector.

CCS Concepts

• **Security and privacy** → **Intrusion detection systems**; **Distributed systems security**; • **Computing methodologies** → **Neural networks**.

Keywords

Anomaly Detection, Industrial Control Systems, Model Poisoning, Physics-Informed Machine Learning

ACM Reference Format:
Simeon Grossmann, Alessandro Erba, and Christian Wressnegger. 2026. Physics-Informed Defenses against Poisoning Attacks on Learning-based Anomaly Detection in Industrial Control Systems. In *8th Workshop on CPS&IoT Security and Privacy (CPSIoTSec '26), November 15–19, 2026, The Hague, Netherlands.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3847353.3847495

1 Introduction

Industrial Control Systems (ICS) are responsible for monitoring and controlling physical processes in critical infrastructures, such as water treatment, chemical plants, and power grids, forming the foundation of a modern society. Cyberattacks targeting these infrastructures have repeatedly demonstrated the potential to disrupt physical processes and cause large-scale societal impact [25, 27, 32].

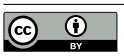
Process-based anomaly detection learns a model of the plant's normal operation to identify attacks that cause the system to deviate from its expected physical behavior [15]. Commonly, such systems assume training data collected under normal operating conditions; however, this may not hold in legacy or compromised ICS.

Operational constraints in ICS often require models to be trained on field data, giving rise to training-time poisoning attacks against process-based detectors that degrade detection performance and induce missed alarms at test time [20, 31]. The importance of defenses in this setting is further underlined by Article 15 of the EU AI Act, requiring AI-based methods deployed in critical infrastructures to be robust against exactly this type of attack. However, defenses against data poisoning in ICS anomaly detection remain scarce.

Existing poisoning attacks do not enforce physical consistency among related sensors [3, 20, 31], leaving open whether physical process knowledge can be used as a training-time prior to improve robustness against poisoning. In this paper, we propose a physics-informed methodology to mitigate poisoning attacks against ICS anomaly detectors, extending recent work that assumes clean training data [35, 37]. We build on the observation that poisoning attacks often introduce behaviors that are statistically useful for compromising the detector, but are physically inconsistent with the process.

One way to capture physical process knowledge is through process invariant rules [12, 34]. Such invariants are well-suited to describe boundaries of normal operation, for example, feasible tank-level ranges under which an inlet valve may open. However, we show that invariant rules do not provide a continuous consistency signal to counter poisoning attempts while the process remains within its admissible operating range. Instead, we leverage continuous physical equations as a measure of physical consistency, which inherently provides a residual signal at every timestep.

We relax the clean-training-data assumption[1] and propose two physics-informed defenses to mitigate poisoning success: (1) removing physically inconsistent timesteps from the training data, acting as a hard constraint, and (2) training-time regularization as leveraged by physics-informed neural networks [26], acting as a soft constraint for learning. We evaluate both on the SWaT testbed, showing that our defenses systematically reject poisoning that violates the modeled process equations while preserving the original detection performance. Our approach identifies poisoning attempts that would otherwise compromise the detector and lead to missed



This work is licensed under a Creative Commons Attribution 4.0 International License.
CPSIoTSec '26, The Hague, Netherlands
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-3032-0/2026/11
https://doi.org/10.1145/3847353.3847495

[1] We only require a small, manually verifiable held-out dataset to parameterize the physical equations, while the bulk of the training data may be poisoned.

alarms at test time. For synthetic ramp attacks on a level sensor, 30% training-data poisoning reduces the Data-Only recall from 90.8% to 14.2% on unseen attacks, whereas the filtering and physics-informed training retain high recall at 89.0% and 76.9%, respectively.

In summary, this paper makes the following contributions:

- **Physics-informed poisoning defenses.** We introduce two defenses against training-time poisoning of process-based anomaly detectors: a residual-based filter that removes physically inconsistent samples, and a training objective that penalizes deviations from the governing physical equations. We further show that invariant-based detection [12, 34] offers only coarse-grained consistency checks, and is therefore insufficient to counter poisoning.
- **Instantiation on a realistic testbed.** We apply the proposed defenses to the SWaT testbed [16], modeling the water flow across its six stages through continuous equations.
- **Robustness under poisoning.** We experimentally evaluate the proposed defenses under training-time poisoning attacks and show that they mitigate poisoning success while preserving the accuracy of the underlying anomaly detector. We investigate adaptive attacker strategies, showing that bypassing the proposed defenses requires stronger capabilities—namely coordinated multi-point process spoofing—and discuss the broader implications, assumptions, and limitations of the methodology beyond the considered empirical setup.

To foster open science and reproducibility, our artifacts are available at: `https://intellisec.de/research/physICS`.

2 Related Work

Below, we discuss directly related strands of research, ranging from learning-based anomaly detection for ICS, to poisoning attacks against anomaly detection and physics-informed machine learning. We close with a discussion of the research gap of prior work.

Learning-Based Anomaly Detection. Detecting cyber-attacks in legacy constrained systems like ICS is a long-standing research problem. Anomaly detection approaches enable attack detection and system recovery before the attacks physically damage the systems. Anomaly detectors are characterized by the following design principle, model/learn normal operations of the plant to identify process deviations [15]. A window of recent process observations is fed to a model trained to either reconstruct the current timestep or to predict the next timestep. During inference, the reconstruction or prediction error is assumed to be small for known operation modes. If the error of a timestep exceeds a threshold fitted on normal operation, the timestep is considered an anomaly.

Multiple approaches were applied to the problem, from process models, such as linear time-invariant models [30] and process invariants [12], to machine-learning models, such as support vector machines [18] and deep-learning approaches [7, 17, 18, 21, 23, 28, 29].

Poisoning of Anomaly Detection. Prior work largely assumed that the data used to train anomaly detection systems consists of clean samples not manipulated by an attacker. In contrast to test-time evasion attacks—which manipulate targeted process variables to conceal malicious activity from a detector trained on normal behavior [10]—data poisoning attacks compromise the training data so that the classifier learns undesired process behaviors. Kravchik et al. [20] demonstrated interpolation- and back-gradient-based poisoning attacks against deep-learning detectors in ICS by spoofing continuous sensor measurements. For the purpose of this work we focus on the interpolation-based poisoning attack, which gradually poisons the training data to implant a desired behavior into the learned detector. We deem the back-gradient poisoning attack too strong for the considered ICS setting, as it requires the attacker to have access to the attack data that describe the dynamical system behavior under attack to poison the training data [2, 9]. Bhattacharjee et al. [3] poisoned power-grid observations, either randomly or through gradient-based methods, to influence the resulting anomaly thresholds. Similarly, Walita et al. [31] demonstrated training-time backdoor attacks that leverage actuator states.

Despite these demonstrated training-time threats, defenses remain limited. Bhattacharjee et al. [3] explored the use of different loss functions to mitigate the impact of FGSM- and random-based poisoning. We observe that current poisoning approaches primarily focus on successfully compromising targeted sensors and generally do not enforce physical consistency during poisoning. We argue that poisoning continuous sensor measurements in a physically consistent manner, as considered for evasion attacks [22], constrains attack deployment by requiring coordinated spoofing of coupled sensors, increasing attacker effort. For this reason, we believe that leveraging physical constraints represents a valuable path towards training-time defenses against poisoning of ICS anomaly detectors.

Physics-Informed Learning. Physics-informed neural networks (PINNs) add a loss to the training objective to introduce physics equations as a training prior. PINNs were originally introduced by Raissi et al. [26] to solve and identify partial differential equations over continuous spatio-temporal domains from sparse observations, where the physics residual is typically computed via automatic differentiation of the network output and serves as a central part of the learning objective. In this setting, the formulation is motivated by sparse observations, known or partially known governing equations, and the need to interpolate a physically consistent solution between them. How the physics term is formulated and implemented depends on the data regime and the goal it serves [6].

Beyond the original setting, physics-informed learning has been applied to anomaly detection. Zideh et al. [36, 37] use physics-informed learning to improve anomaly detection of power distribution grids. Next to encoding physics information via additional loss terms, there are further approaches incorporating the prior knowledge by adapting the architecture of the anomaly detector. Wu et al. [35] use prior knowledge about the piping and sensor placement in a water-treatment plant to align the architecture of a learned graph neural network, improving the detection performance and interpretability. These approaches assume clean training data and focus on improving anomaly detection by guiding the learned model toward more physically consistent behavior.

Research Gap. Existing poisoning research on ICS anomaly detection demonstrates training-time threats but offers few defenses, and the proposed poisoning attacks generally do not enforce physical consistency [3, 20, 31]. Physics-informed anomaly detection, in turn, has seen little attention in the ICS domain, and the few existing works study it only on clean data [35, 37]. To our knowledge, no

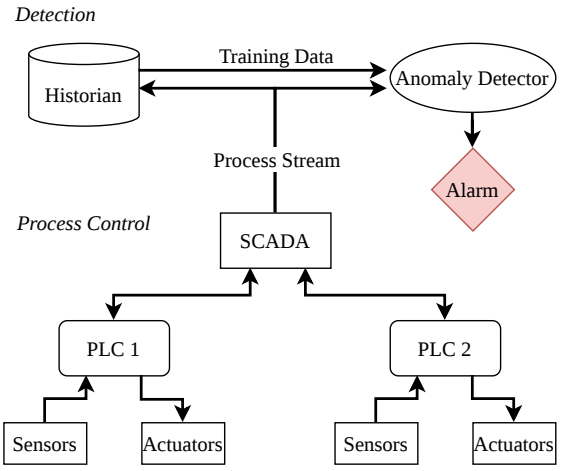


Figure 1: System model of the considered industrial control system, following the Purdue architecture.

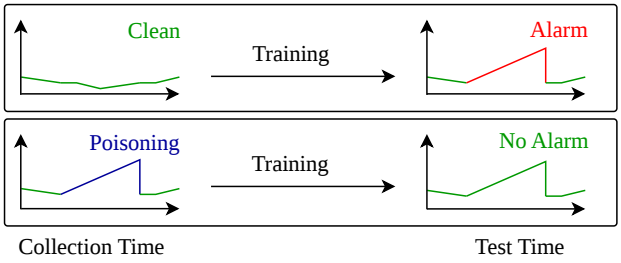


Figure 2: Attacker goal: poisoning the anomaly detector during data collection to evade test-time detection.

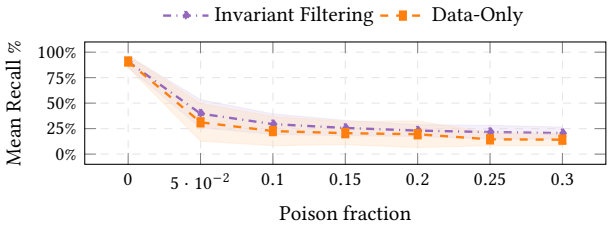


Figure 3: Motivating Example: Detection recall under synthetic ramp poisoning of the SWaT PLC1 level sensor. Models are evaluated on synthetic ramp attacks in the same setting. Data-Only is the conventional forecaster; Invariant Filtering filters training data using GeCos rules [34].

prior work uses physical process equations as a training-time prior to defend anomaly detection against data poisoning. We address this gap by using physical consistency among related sensors as a signal for identifying poisoned samples that are not explicitly designed to preserve such consistency. We leverage this physics consistency in two ways: removing the violating samples through a filter or penalizing them via a physics-informed loss.

3 Threat Model

System Model. We assume an industrial control system that is operating an industrial plant adopting the classical Purdue architecture [33], see Figure 1. At the operational level, sensors and actuators interface the physical process, and multiple programmable logic controllers (PLCs) implement the control logic to actuate the system. The supervisory control and data acquisition (SCADA) aggregates the streams coming from the PLCs. For safety and security reasons, the plant operator wants to enable plant and process monitoring at the historian level, aggregating the multivariate process measurements from sensor values and actuator states.

Due to the widespread ICS legacy requirements, the operator does not have guarantees that no attacker is present during data collection, which poses the risk of training data poisoning [20]. Despite that, the defender aims to enable attack detection by applying a learning algorithm over the historian database. Given the high amount of data collected to train the anomaly detector, the defender is assumed not to have the workforce or expertise to manually filter the bad data. Finally, for threshold calibration purposes, the defender is assumed to possess a clean set of observations. These observations contain a limited number (e.g., a few hours) of manually inspected sensor readings of the process.

Attacker Capability. We assume an attacker who can spoof sensor measurements by exploiting common ICS insecure channels or misconfigurations. We abstract from the specifics of how the data is poisoned, but we envision two possibilities. i) The attacker spoofs the network data on the fly, by exploiting unauthenticated or misconfigured industrial protocols [8, 9] or ii) compromises the data collected in the historian [3, 20, 31]. The attacker does not have direct access to the anomaly detection model parameters or the detection pipeline and cannot actively alter its learned parameters.

Attacker Goal. The attacker is present during data collection, and wants to embed unintended behavior into the data collection used to train the anomaly detection system. This will result in the implant of a poisoned behavior in the learned anomaly detector [3, 20, 31].

At test time, the attacker aims to cause unsafe behavior [1, 32] while remaining unnoticed by the security countermeasures in place. Due to the embedded poisoned behavior in the training data, the learned classifier is misled into treating the data as normal, so the attacker's test-time manipulations evade detection (Figure 2).

Example: Invariant-Based Filtering

We apply a model poisoning attack from prior work [20] to assess whether invariant-rule-based detection is sufficient to remove poisoned samples. Invariant rules encode logical relationships between process variables [12]. They capture operational constraints and boundary conditions, but typically abstract the continuous evolution of process signals between those boundaries. This abstraction limits their effectiveness against training-time poisoning. An attacker can craft poisoned samples that stay within the admissible

region defined by the invariants, while still perturbing the temporal evolution of the process to bias the learned detector.

To evaluate this, we use the expert-crafted invariant rules for SWaT PLC1 from Wolsing et al. [34] (Appendix Table 1). We remove an input window from the training set whenever at least one timestep in the window violates one of the rules. Then we train the anomaly detector on the remaining data samples. The considered rules identify 20% of the injected poisoned samples, while the remaining poisoned samples pass the filtering step undetected, leading to a successful poisoning attack. As shown in Figure 3, invariant-rule-based filtering provides limited protection compared to training on all the data points. This example shows that invariant-based filtering is insufficient against poisoning attacks that preserve coarse logical consistency, motivating the use of continuous physical equations to provide a consistency signal.

4 Methodology

In this section, we present our proposed methodology for countering data poisoning attempts in process-based anomaly detection, as depicted in Figure 4. We propose enhancing anomaly detection training with physically grounded information. We start the section by providing the preliminaries about the underlying anomaly detection in Section 4.1. Then, in Section 4.2, we present the physics-based process modeling. Finally, we present the two proposed data poisoning defenses, the first a hard-constrained physics-informed dataset filtering technique (Section 4.3), the second a soft-constrained physics-informed learning technique (Section 4.4).

4.1 Forecasting-Based Anomaly Detection

Our goal is to safeguard the training of anomaly detectors against data poisoning. These architectures are well established in prior work [13, 21, 28]. The process is captured through a multivariate time series $\mathbf{x}_t \in \mathbb{R}^d$, where each component is a sensor or actuator value and d is the number of process variables. A forecasting model $f_\phi : \mathbb{R}^{W\times d} \to \mathbb{R}^d$ maps a window of W past states to a prediction $\hat{\mathbf{x}}_{t+1}$ of the next process state, trained on normal operation data.

4.1.1 Training Objective. The training objective for f_ϕ is the mean squared prediction error against the observed next state [21].

$$\ell_{\text{data},t} = \frac{1}{d}\sum_{i=1}^{d}\left(\hat{x}_{t+1,i} - x_{t+1,i}\right)^2.$$

We refer to a forecaster trained with only this objective as the *Data-Only* baseline. Such models capture statistical regularities in the training data but are not constrained to respect physical relations. When the training data is contaminated by poisoning samples, the forecaster learns these as normal behavior [20].

4.1.2 Residual-Based Detection. At inference (Figure 4, right), the forecaster predicts the next state, and for each predicted variable we compute the squared residual $e_{t,i} = (\hat{x}_{t,i} - x_{t,i})^2$ between prediction and observation. For detection, we consider two strategies.

(1) The *MSE detector* aggregates these residuals into a single score and fits one threshold on it,

$$\tau = \text{Quantile}_q\left(\left\{\tfrac{1}{d}\sum_{i=1}^{d} e_{t,i} \;\middle|\; t \in \mathcal{T}\right\}\right),$$

and raises an alarm when $\frac{1}{d}\sum_{i=1}^{d} e_{t,i} > \tau$, where $\mathcal{T}$ is the trusted clean threshold-fitting set.

(2) The *per-sensor detector* instead keeps the residuals separate and fits one threshold per variable,

$$\tau_i = \text{Quantile}_q\left(\{e_{t,i} \mid t \in \mathcal{T}\}\right),$$

raising an alarm when the threshold is exceeded, $e_{t,i} > \tau_i$.

The quantile q is an operator-chosen hyperparameter balancing detection sensitivity and False-Positive Rate (FPR).

4.1.3 Model Selection. To prevent overfitting towards attack patterns that may not match real-world conditions, we follow established pipelines [21, 28] and train forecasters across a range of hyperparameter settings. For each setup, we keep the model with the highest next-step prediction accuracy on a clean held-out validation set. This approach assumes that models that generalize well to normal behavior will also produce residuals that help separate normal and anomalous behavior at inference time, allowing model selection without relying on attack-specific validation data.

4.2 Physics-Based Process Modeling

To apply physical relations to every timestep, we express the process dynamics in continuous process equations. Examples in ICS are conservation laws, such as mass, energy, pressure-flow, or electrical balance. We capture the relevant relation between two consecutive process states as $g(\mathbf{x}_t, \hat{\mathbf{x}}_{t+1}, \mathbf{u}_t, \boldsymbol{\theta}) = 0$, where $\hat{\mathbf{x}}_{t+1}$ is the forecaster's prediction, $\mathbf{u}_t$ represents exogenous inputs such as actuator states or operating-mode indicators, and $\boldsymbol{\theta}$ holds physical parameters of the modeled process. Conservation laws are stated in continuous time and relate the accumulation of a quantity to its inflow, outflow, generation, and loss.

The continuous-time relation is discretized with the sampling interval Δt of the historian data. As an example, we consider a tank with level h_t, inflow q_t^{in}, outflow q_t^{out}, and cross-sectional area A (the physical plant in Figure 4). For this process, we instantiate the generic relation $g(\cdot)$ as a concrete equation. Without leakage or unmodeled inputs, the level change must match the difference between inflow and outflow, giving the continuous-time mass balance

$$\frac{dh(t)}{dt} = \frac{1}{A}\left(q^{\text{in}}(t) - q^{\text{out}}(t)\right), \tag{1}$$

which is discretized using backward (implicit) Euler, evaluating the right-hand side at the new timestep,

$$h_{t+1} = h_t + \frac{\Delta t}{A}\left(q_{t+1}^{\text{in}} - q_{t+1}^{\text{out}}\right). \tag{2}$$

We use the backward form of Euler since it evaluates the flow terms at the new timestep $t+1$ rather than at t, so the forecaster's predicted flows enter the equation, and the physics loss provides a gradient on them (Section 4.4.1).

The residual of the relation measures how strongly a given set of process values violates it. For the tank, it is

$$r_t^{\text{tank}} = h_{t+1} - h_t - \frac{\Delta t}{A}\left(q_{t+1}^{\text{in}} - q_{t+1}^{\text{out}}\right) - b, \tag{3}$$

which is near zero on physically consistent samples and grows with the violation. We use a static bias term b, fitted once on the clean reference set and held fixed, which absorbs static sensor

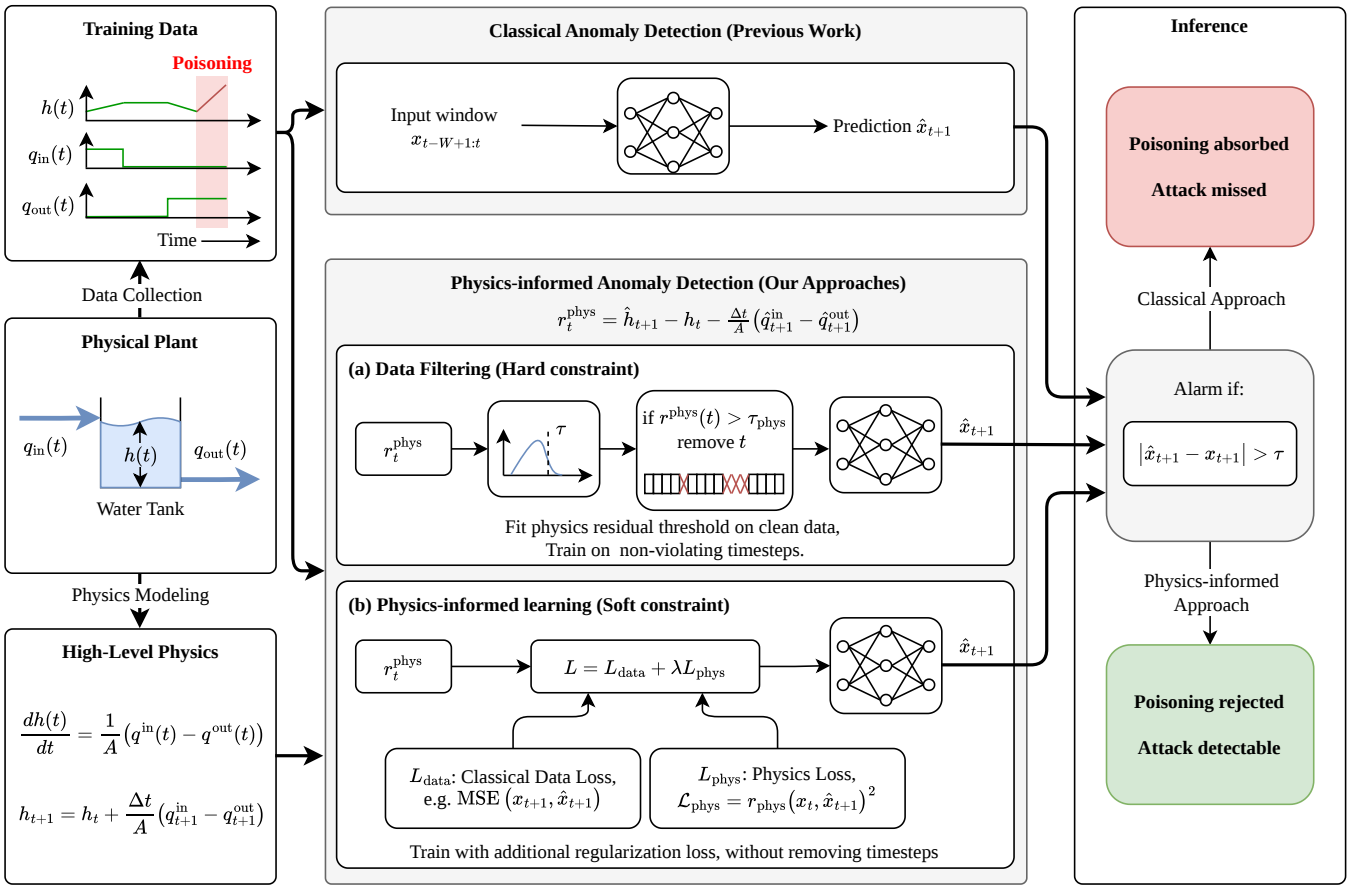


Figure 4: Overview of the proposed poisoning-aware, physics-informed anomaly detection in ICS.

offsets. Mode-dependent contributions, such as filter flushing, can be captured by additional bias terms gated on $\mathbf{u}_t$.

4.3 Poisoning Data Filtering

As a straightforward approach to exploit the physics-informed equations, we propose a Filtering technique based on the physics prior knowledge to filter poisoned samples from the dataset (Figure 4, a). We evaluate the residual (Eq. 3) on a clean held-out set and compute the corresponding violation. The distribution of these residuals is then used to define a filter threshold, chosen as the quantile q of the observed residual values:

$$\tau = \text{Quantile}_q\left(\{|r_t| : t \in D_{\text{heldout}}\}\right). \tag{4}$$

This threshold is then applied to the training data, dropping all timesteps from the training corpus that contain physical residuals exceeding the threshold:

$$D_{\text{filter}} = \{t \in D_{\text{train}} : |r_t| \leq \tau\}. \tag{5}$$

For the continuity of the training set, we not only drop the violating timestep, but all input windows containing it. After filtering, we assume the training set to be clean and train the forecaster. This Filtering approach raises two practical considerations.

Multivariate Data Dropping. The filter operates on the entire training window, even when only a subset of process variables enters the modeled equation. A single variable violating the equation removes the full window from training, including the values of all other variables that may have been unaffected. Thus, safeguarding against the violation costs data points for unaffected variables.

Threshold Sharpness. The physics-based filtering needs a binary decision between dropping and keeping. This is affected by the quantile q of the threshold and depends on the representativeness of the clean held-out data. If the resulting threshold τ is too high, poisoning samples will be kept; if it is too low, benign transients are lost. Appendix C shows that this sensitivity is sharp: a small shift in the quantile can cause an abrupt drop in detection recall.

The soft-constraint formulation (Section 4.4) instead keeps every window in training, regularizing only the equation-involved variables and without a tuned threshold.

4.4 Physics-Informed Training

Unlike the original formulation of PINNs (Section 2), our setting differs in three ways, shifting the role of the physics loss. First, the *data regime* differs. ICS historians provide dense, regularly-sampled

measurements, while limited modeling expertise restricts physical knowledge to high-level equations. Observations are therefore abundant relative to the equations, so the physics loss does not interpolate but acts as a *soft constraint* on top of the empirical loss, which remains the primary learning signal.

Second, the *goal* is different. We do not solve a PDE or identify parameters, but detect anomalies when the training data may be poisoned. The physics loss therefore acts as a *robustness inductive bias*, penalizing predictions that violate the modeled relations, for example, single point poisoning data.

Third, the *temporal structure* differs. Historian data is sampled at discrete, regular intervals rather than on a continuous field, so the residual is stated directly in discrete time, avoiding the automatic differentiation over a continuous solution used in the original PINN.

4.4.1 Physics-Informed Training Objective. The physics-informed objective evaluates the residual (e.g., Eq. 3) on the forecaster's predictions, using the predicted next-step quantities together with the previous observed state, and penalizes its square,

$$\ell_{\text{phys},t} = r_t^2. \tag{6}$$

When the training data is consistent with the modeled relation, the empirical loss already drives the forecaster toward physically valid predictions, and the physics loss only reinforces this signal. Intuitively, when training data violates the modeled relation, e.g. under single point poisoning, the physics loss provides a competing gradient toward more physically consistent predictions. The physical process is described by K equations, each with residual $r_{t,k}$, so the physics loss extends to a weighted sum,

$$\ell_{\text{phys},t} = \sum_{k=1}^{K} \alpha_k \left(\frac{r_{t,k}}{s_k} \right)^2, \tag{7}$$

where α_k prioritizes specific equations and s_k rescales each residual.

Different equations produce different magnitudes of residuals. Without rescaling, an equation could dominate the loss simply by being numerically larger. We set each s_k to the root-mean-square residual of equation k on consecutive clean samples, which normalizes the residuals to a comparable scale. An equation the forecaster fits worse than this clean-data baseline then produces $|r_{t,k}/s_k| > 1$ and attracts more regularization. More sophisticated dynamic weighting exists [5, 19]; however, the static scaling suffices.

4.4.2 Multi-Objective Training. The forecaster is trained by combining the empirical and physical losses into a single objective,

$$\ell_{\text{total}} = \ell_{\text{data}} + \lambda_{\text{phys}}\, \ell_{\text{phys}}, \tag{8}$$

where $\lambda_{\text{phys}} \geq 0$ is a fixed regularization weight.

To make λ_{phys} comparable across unit scales, we evaluate both losses on a few batches of clean data before training and rescale the physics loss so that $\lambda_{\text{phys}} = 1$ gives equal contributions. This loss balancing is distinct from the per-equation scaling s_k in Eq. 7, which balances the physics equations against one another. Setting $\lambda_{\text{phys}} = 0$ disables the physics loss (Data-Only baseline).

5 Evaluation Setup

In this section, we present the experimental setup. In Section 5.1, we provide the details of the SWaT testbed and its process. Then, we present the evaluation scenarios for data poisoning in Section 5.2.

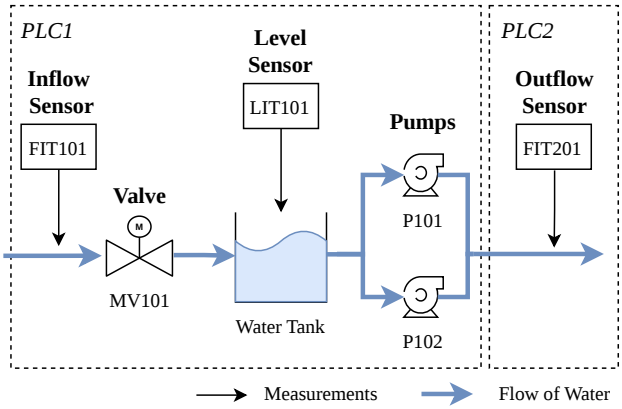


Figure 5: Sensors, actuators, and process flow of the first SWaT tank system (PLC1), including PLC2's flow sensor.

Finally, we discuss the data preprocessing (Section 5.3), the deep architectures (Section 5.4), and the evaluation metrics (Section 5.5).

5.1 SWaT Dataset

We use the SWaT water-treatment testbed dataset [16], providing seven days of normal operation and four days of recorded attacks on a six-stage plant with 51 process variables. Consistent with prior work [4, 17], we focus on the first stage (Figure 5), a tank coupling a level sensor with an inflow and an outflow sensor.

We instantiate the discrete tank dynamics of Eq. 3 with level LIT101, inflow FIT101, outflow FIT201, and tank area A. The valve and pump states are included as additional inputs to the forecaster.

Of the original SWaT attacks, we consider those that target at least one variable of the first stage. Four attacks violate the mass-balance relation[2], and six remain consistent with it, reported in Section 6. Following Fung et al. [14], we use the refined attack labels, since the original labels do not always align with the attack data.

Extension to the Full SWaT Plant. To test whether the PLC1 results generalize to larger plants, we extend the setup to the full SWaT dataset, using all process variables and modeling the additional tank and flow-conservation relations of the downstream stages (Appendix D). We poison as in the single point level poisoning setting, spoofing $1.943\,\text{mm}\,\text{s}^{-1}$ ramps, but now target LIT101, LIT301, and LIT401 sequentially in both training and testing. For the full plant, the forecaster predicts the state change added to the last observation, and model selection and threshold fitting share one 24 h holdout, as the 12 h split is too small to calibrate representative thresholds covering all occuring operation modes.

5.2 Poisoning Scenarios

We first establish a clean-training baseline by training models on the unmodified normal SWaT training set and evaluating them on the original SWaT attack set. Building on this, the main poisoning experiments assess whether physics-informed training improves robustness when manipulated records are included in the training data before model retraining. Inspired by the poisoning attacks from

[2]Violating attacks: IDs 3, 21, 33, 36; Consistent attacks: IDs 1, 2, 26, 30, 34, 35

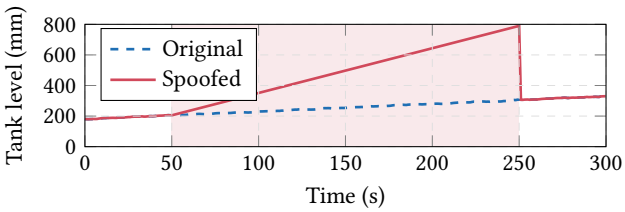


Figure 6: Exemplary ramp overwriting the original level.

Kravchik et al. [20], we consider two poisoning settings: single point poisoning, where the manipulation alters a single sensor reading, and physically coordinated poisoning to preserve physical relations.

Ramp Poisoning Model. The attacker aims to move a manipulated level or flow measurement toward a desired target state. Poisoning the anomaly detector to accept such behavior as normal establishes a behavior that the attacker can exploit later. Examples are spoofing a low tank level to cause a tank overflow, or spoofing a full tank to provoke downstream pump damage [16]. As demonstrated in prior work [20], model poisoning through ramps reduces detectability. The spoofed value is changed gradually over consecutive timesteps instead of being shifted abruptly. The ramp overwrites the targeted process variable by multiple episodes of linear ramp segments (Figure 6). Each episode lasts between 60 s and 375 s, increasing or decreasing the measurement, after which the value returns to the real measurement. We use fixed ramp rates of $1.943\,\mathrm{mm\,s^{-1}}$ for LIT101 and $1.085 \times 10^{-4}\,\mathrm{m^3\,s^{-2}}$ for FIT101, corresponding to half of the maximum clean-data increment. The amount of poisoned training data is varied between 0% and 30%.

We deliberately use ramps rather than optimization-based poisoning [20], since the ramp rate is a controllable axis along which we vary the induced physics violation and locate where the defenses fail. As optimization-based poisoning does not explicitly constrain physical consistency, it remains subject to the same residual-based defenses unless it is additionally coordinated.

Single Point Poisoning. The attacker overwrites either LIT101 or FIT101 while leaving physically related variables unchanged, breaking the modeled mass-balance relation. The poisoned samples are treated as normal during training and evaluated against synthetic attacks of a related type. We further test whether the effect is specific to the injected ramp pattern by evaluating the test-time generalization across sensors, attack trajectories (staircase, sigmoid, and noisy ramps), and ramp rates (i.e., slopes).

Physically Coordinated Poisoning. Here, the attacker adapts related variables such that the mass-balance relation remains approximately satisfied. For the level poisoning experiment, a sinusoidal offset is added to LIT101, starting at zero, reaching a target value, and returning to zero, avoiding discontinuities at the episode boundaries. The flow variables are adapted consistently with this trajectory by increasing inflow during rising phases and increasing outflow during falling phases. For the flow poisoning experiment, a linear ramp that first rises to the target offset and then decreases back to zero is added to both the attacked inflow and the related outflow variable, preserving the modeled inflow–outflow balance.

5.3 Preprocessing and Data Splits

Normalization. We normalize the process variables to zero mean and unit standard deviation, using the training data. Predictions are transformed back to SI units before the physics residual is computed, so the modeled equation, the filter threshold, and the physics loss remain independent of these statistics.

Sampling Rate and Windowing. The historian records data at one-second intervals. Following recommendations from prior work [7], we downsample the dataset by averaging ten consecutive samples per block to suppress noise. In Appendix Section A, we provide a systematic methodology to match sampling rate to the physics residual. Training samples use a sliding window of 50 raw seconds [7, 14].

Data Splits. We partition the SWaT normal recording into four non-overlapping sets. We exclude the first six hours from all experiments since they contain non-steady state process observations [7, 16]. The remainder splits into a training set (96 h) for learning the forecaster, and three 12 h holdouts: a validation set for model selection, a threshold-fitting set for the detector, and a clean evaluation set for generating synthetic poisoning attacks. Poisoning is applied only to the training set. The original SWaT attack recording forms the evaluation set for experiments on recorded attacks, excluding attacks not targeting the considered process stage.

5.4 Forecasting Models and Training

Architectures. We evaluate the following deep-learning architectures: multi-layer perceptrons (MLP), one-dimensional convolutional networks (CNN1d), long short-term memory networks (LSTM), and transformers. The four architectures cover different inductive biases and have approximately 50 k trainable parameters.

Model Variants. We train three variants: the classical Data-Only without any poisoning robustness measures, the PINN variant, adding the physics-loss to the training, and the Filtering approach that filters the training set before training a Data-Only model. We fix the filter quantile at $q = 0.99$ across all experiments, treating the 1% of clean samples with the largest residuals as outliers. This follows residual-threshold calibration on clean SWaT data [11], and the same quantile has been applied to outlier-based approaches [24, 28].

Training and Hyperparameter Sweep. Following the approach described in Section 4.1.3, for each architecture and each random seed, we train the Data-Only ($\lambda_{\text{phys}} = 0.0$), the Filtering ($\lambda_{\text{phys}} = 0.0$), and PINN variants ($\lambda_{\text{phys}} = 1.0$). To allow reliable convergence, we sweep over four learning rates, $\{2\cdot10^{-4}, 5\cdot10^{-4}, 7\cdot10^{-4}, 10^{-3}\}$, with remaining training parameters fixed. We train each model with 10 random seeds and select the best run across learning rates for each architecture-seed pair by next-step prediction accuracy on the validation set (Section 4.1.3). The variation across seeds and architectures is reported as uncertainty bands in Section 6. We discuss the sensitivity to λ_{phys} and its selection in Appendix B.

5.5 Detector Instantiation and Metrics

Detectors. We instantiate both the MSE detector and the per-sensor detector of Section 4.1.2. For most reported results, we use the per-sensor detector, which avoids the residual dilution that

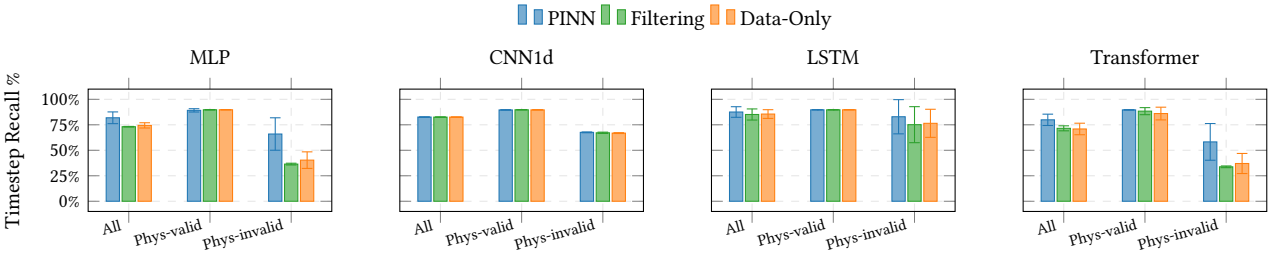


Figure 7: Detection recall of PINN, Filtering, and Data-Only forecasters trained on the clean SWaT training set and evaluated on the original SWaT attacks. Results are grouped by architecture and by attack properties (preserving or violating mass-balance).

plagues the MSE detector in case of localized manipulations. The detection quantile q is set as $q = 0.99$ for the MSE detector, which corresponds to a 1 % false-positive budget on clean threshold data, and $q = 0.999$ for the per-sensor detector, to compensate for the added sensitivity from fusing per-sensor thresholds.

Recall and False-Positive Rate. An alarmed timestep is counted as a true positive if it lies within a labeled attack window and as a false positive otherwise. For attack k with TP_k flagged and FN_k missed timesteps, the per-attack recall is $\text{Recall}_k = TP_k/(TP_k + FN_k)$. We average uniformly over the K attacks, $\overline{\text{Recall}} = \frac{1}{K}\sum_{k=1}^{K} \text{Recall}_k$, so that each attack contributes equally regardless of its duration. For comparability with prior work, we report the classical timestep-wise recall for the results on the original SWaT attacks. We report the FPR on the clean evaluation set as $\text{FPR} = FP/(FP + TN)$.

6 Experimental Results

We present the empirical results for the proposed physics-informed anomaly detection. Our goal is to assess whether the proposed methods enable the detection of the process anomalies (original SWaT attacks) while defending against poisoning attempts. We assess our method by answering five research questions.

(RQ1) How does physics-informing affect detection performance when the model is trained on unpoisoned data?
(RQ2) How does physics-informing affect detection performance when the model is trained on poisoned data and evaluated on the original attacks?
(RQ3) How does physics-informing induce robustness against poisoning attacks?
(RQ4) How does the poisoning robustness generalize across sensors and poisoning patterns?
(RQ5) Is the proposed physics-informed method robust against an adaptive attacker who poisons data consistently with the underlying physics?

As we cover different poisoning scenarios (Section 5.2), the training and test data are specified in each experiment. Unless stated otherwise, results are reported as the mean over the 4 deep architectures and 10 random seeds (see Section 5.4).

6.1 RQ1: Clean Training Performance

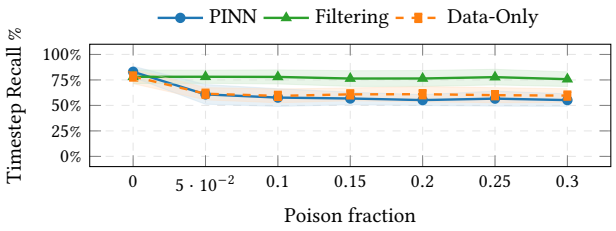
We first ask how the physics-informed approaches affect detection when the training data is unpoisoned, training the models on the clean SWaT training set and testing on the original SWaT attacks. We consider the first process stage (PLC1) and split the original attacks into physically valid and physically invalid attacks (Section 5.1). Figure 7 shows the resulting timestep-wise recall per architecture and attack group for PINN, Filtering, and Data-Only. For physically valid attacks, physics-informing does not differ significantly from Data-Only. For attacks that break the physical consistency, the impact of physics-informing differs between the PINN and the Filtering approach. PINN improves the detection of physically invalid attacks for all architectures except the CNN1d, which performs equally for all variants. A Data-Only model may also converge to a consistent predictor on clean data (e.g., LSTM), but this is not guaranteed and varies across architectures. PINN, by contrast, penalizes inconsistent predictions during training, regularizing toward consistent convergence. Filtering instead slightly decreases detection performance for most architectures, since dropping benign samples above the threshold hurts generalization, and acting before training cannot steer the model toward consistent predictions. Across architectures, the FPR on the clean test data remains at approx. 0.3% for all variants. We confirm this behavior for the full swat that is used in the following sections. Appendix Table 2 reports timestep-wise metrics for this experiment (for both PLC1 and full SWAT). Further data ablations are reported in Appendix F.

RQ1 Takeaway. Overall, the proposed physics-informed methods do not impact the clean accuracy of the anomaly detection. Specifically, PINN improves detection in physically invalid regimes, whereas Filtering can slightly reduce clean-data detection. Based on this result, we are now motivated to investigate the robustness of the proposed models against data poisoning.

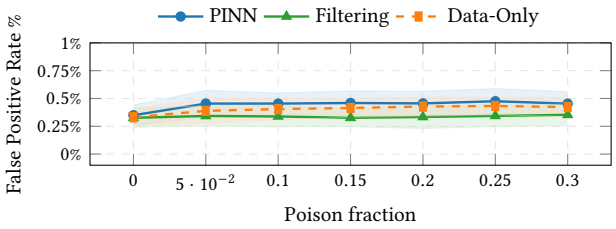
6.2 RQ2: Poisoned Training Performance

To answer RQ2, we poison the training data with level-ramp attacks (Section 5.2), which overwrite the level sensor while leaving related flow variables unchanged, and evaluate how well the resulting models still detect the original SWaT attacks.

6.2.1 PLC1. Having evaluated detection under clean training (Section 6.1), we now evaluate the same attacks after training-time poisoning. As shown in Figure 8, the Data-Only and PINN variants lose recall, while the Filtering variant maintains its original detection performance. The linear ramp steers the level toward a target value and violates the modeled physical consistency. Because the ramps are deployed without regard for the process logic, they can

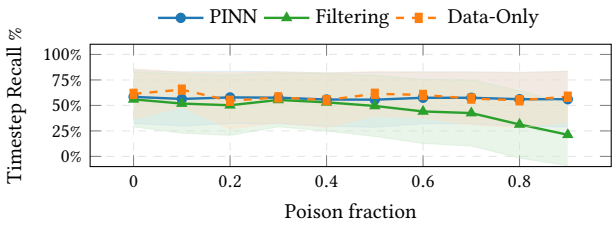


(a) Timestep-wise detection recall.

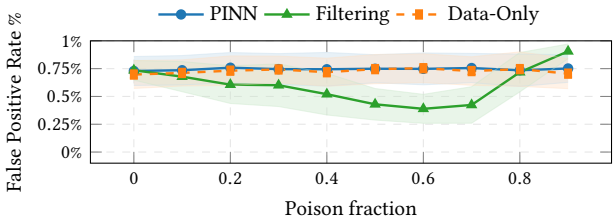


(b) False Positive Rate on clean evaluation set.

Figure 8: Detection recall and FPR of models trained under single point level poisoning, evaluated on the SWaT attacks.



(a) Timestep-wise detection recall (MSE detector).



(b) False Positive Rate on clean evaluation set (MSE detector).

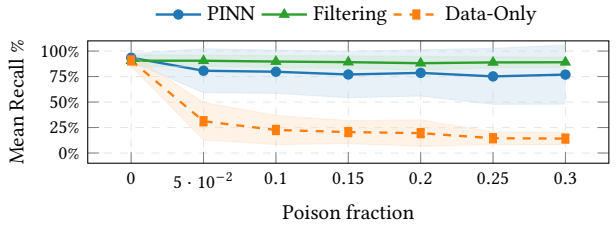
Figure 9: Detection recall and FPR of models trained on full SWaT plant under multi-tank level poisoning, evaluated by the MSE detector on the SWaT attacks.

drive the level above the normally observed range or empty the tank, depending on the starting point and attack duration. The original SWaT attacks on PLC1 mainly target this process logic (e.g., opening the input valve while the tank is full), thus they remain undetected. Filtering removes the complete timesteps that violate the equation, so the model no longer sees the out-of-range tank levels the poison ramps introduce as normal. The PINN regularizes only the physics violation and keeps these previously unseen levels in the training data, losing detection of such PLC-logic violations.

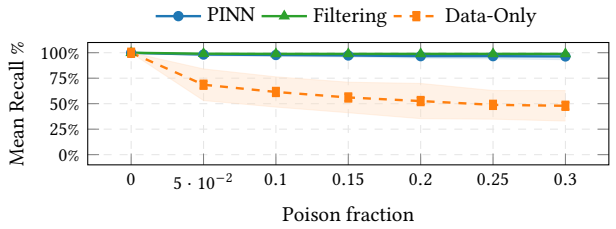
6.2.2 Full SWaT Plant. We now extend the scope of the plant to the full SWaT setting for this experiment, to analyze the impact of plant complexity and additional unmodeled process variables. Figure 9 shows the detection recall and the FPR when evaluating the models trained on single point level poisoning targeting all three levels of the plant and evaluating on original SWaT attacks.

In contrast to PLC1, Data-Only and PINN retain their recall and FPR across poison fractions, while Filtering degrades at severe fractions, with its FPR first dropping below the others before rising slightly above them. Poisoning with 70% or more leads to a significant loss of training data, and the recall reduces while the FPR increases. Although dropping the complete timestep helped to safeguard against secondary effects on PLC1 (Section 6.2.1), it implies a loss of training signal for unrelated variables. The PINN approach, on the other hand, does not lose information for unrelated sensors and keeps a learning signal across the full poison range. We discuss the trade-offs between the two physics-informed approaches in Section 7.1.

RQ2 Takeaway. Under training data poisoning at practical poison fractions, the proposed methods do not negatively impact the detectability of the original SWaT attacks. This provides us with evidence that the proposed method can be applied to detect the original anomalies.



(a) Level poisoning (LIT101).



(b) Flow poisoning (FIT101).

Figure 10: Detection recall, single point poisoning. A fraction of the training data is overwritten with synthetic ramps while physically related variables remain unchanged.

6.3 RQ3: Poisoning Robustness

In the previous sections, we demonstrated that the physics-informed detectors maintain strong performance on the original SWaT anomalies under both clean data and training-data poisoning conditions. We now evaluate how physics-informing improves robustness against adversarial patterns at test time—that is, cases where a poisoning pattern is exploited to evade detection (Section 5.2).

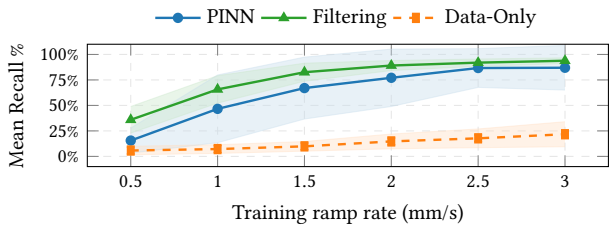


Figure 11: Detection recall, single point level poisoning across ramp rates, using the same ramp rate for training and testing.

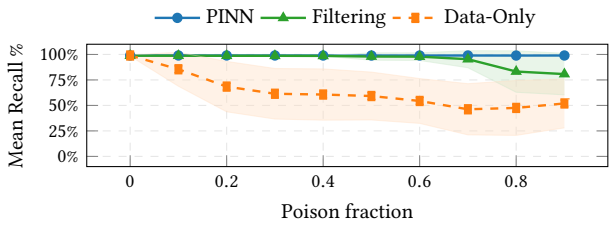


Figure 12: Detection recall, single point level poisoning on the full SWaT plant. Training and test attacks target one tank among T101, T301, and T401, ramp rate of $1.943\,\mathrm{mm\,s^{-1}}$.

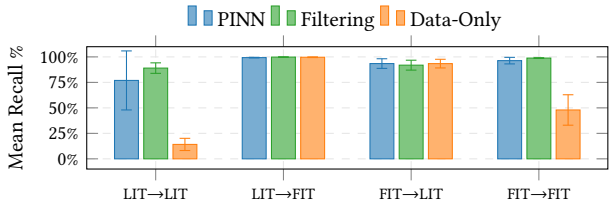


Figure 13: Cross-sensor transfer, single point poisoning at the highest poison fraction. Models are trained with ramp poisoning on the source sensor (left of the arrow) and evaluated on the ramp over the target sensor (right of the arrow).

6.3.1 Level Poisoning. We start with poisoning on the level sensor. Data-Only models lose recall strongly as the poison fraction increases, whereas both physics-informed approaches continue to detect the attacks across all four architectures (Figure 10a). The physics loss thus prevents the model from absorbing physically inconsistent poisoned behavior as normal, acting as a protective inductive bias against poisoning that violates the modeled process dynamics. Filtering performs comparably, slightly stronger, but within PINN's uncertainty, as its hard quantile threshold removes the violating samples more strictly.

6.3.2 Flow Poisoning. Repeating the poisoning experiment on the input flow sensor, both physics-informed approaches again outperform the Data-Only baseline and perform comparably to each other, but the attack is generally easier to detect (Figure 10b). The input flow is close to constant under normal operation, unlike the ramp-like tank level, so the poisoned ramps stand out from the normal regime, which already limits the poisoning's impact.

6.3.3 Sensitivity to Ramp Rate. To test the sensitivity of the regularization against different ramp rates, we retrain the single point level-poisoning experiment with rates from 0.5 to 3 mm/s, testing each model with the same rate it has seen during training. Lower rates make the attack stealthier, and higher rates reduce the number of spoofed timesteps required to reach the target.

Figure 11 compares PINN, Filtering, and Data-Only after poisoning 30% of the training data. While the detection performance of the Data-Only remains low across ramp rates, the robustness of the physics-informed approaches increases with larger ramps. We attribute this to the larger physics violation at higher rates. A steeper ramp produces a larger residual, which gives PINN more regularization signal and also makes the poisoned samples easier for the Filtering to remove. On PLC1, Filtering achieves the highest recall, showing that dropping poisoned windows is an effective defense for simple processes. The FPRs remain low for all approaches.

6.3.4 Full SWaT Plant. We now ask whether the robustness against single point level poisoning transfers to the full SWaT plant, repeating the single point level experiment (Section 6.3.1) with all sensors and actuators, where different poisoning episodes target LIT101, LIT301, or LIT401 (Section 5.2).

The results follow the previous observations, showing the detection performance of the classical Data-Only variant dropping significantly, while both physics-informed variants maintain a robust detection (Figure 12). The gap against Data-Only persists despite the many additional process variables not covered by the modeled equations, confirming that the regularization transfers to larger systems. In contrast to the PLC1 stage, Filtering no longer outperforms PINN here, with both physics-informed variants performing comparably up to 70% poisoning, beyond which Filtering's recall degrades while PINN remains stable, matching the training-data loss observed in Section 6.2.2.

RQ3 Takeaway. The proposed physics-informed countermeasures effectively mitigate the poisoning attempts as they do not learn the poisoning patterns. As a consequence—in contrast to Data-Only models—they achieve high recall on the poisoned samples. The robustness generalizes across ramp rates and applies to different sensor sources and to both SWaT PLC 1 and the full plant.

6.4 RQ4: Robustness Generalization

We now ask how the poisoning affects generalization across sensors and ramp rates at test time. We test the generalization to unseen attack trajectories in Appendix G.

6.4.1 Cross-Sensor Generalization. We train with poisoning on either the level sensor or the flow sensor (as in Section 6.3) and test on attacks targeting the same or the other sensor. The results in Figure 13 show that the poisoning effect is largely sensor-specific. Recall decreases mainly when training and test attacks target the same sensor, where both PINN and Filtering remain substantially more robust than Data-Only models. In contrast, cross-sensor evaluation shows little degradation for any approach. This indicates that the reduced robustness of the Data-Only approach observed in Section 6.3 stems from learning poisoned behavior as normal for the corrupted sensor, rather than from a general degradation of detection. Since the physics-informed approaches act on the

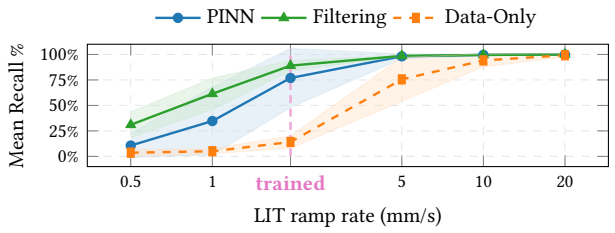
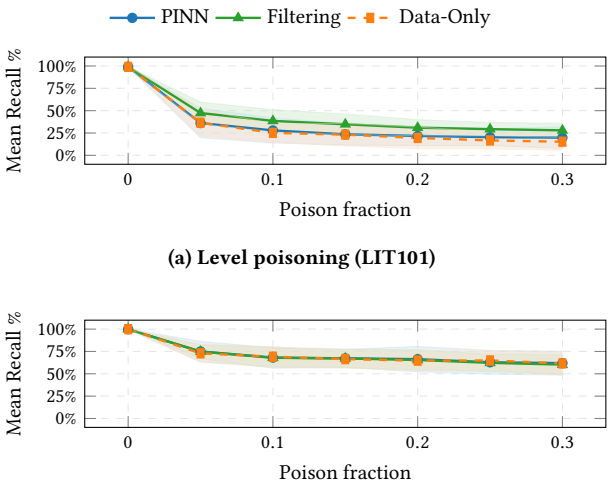


Figure 14: Cross-rate transfer under level poisoning (LIT101). Models are poisoned at a ramp rate of 1.943 mm/s (vertical line) and evaluated across different ramp rates.



(a) Level poisoning (LIT101)

(b) Flow poisoning (FIT101)

Figure 15: Physically coordinated poisoning. (a) The level trajectory is modified while the related flow variables are adapted to preserve the modeled mass-balance relation. (b) The inflow ramp is paired with an adapted outflow trajectory so that the modeled mass-balance relation remains satisfied. In both cases the models are evaluated on synthetic the same physically coordinated attacks.

physics-violating patterns seen during training, cross-sensor attacks still produce residuals exceeding detection thresholds.

6.4.2 Cross-Rate Generalization. We further train on level-ramp poisoning at a fixed ramp rate and evaluate across different ramp rates. The physics-informed models keep their advantage across rates (Figure 14), with the gap narrowing at very low rates, where the attack is less pronounced, and at large rates, where all variants detect it reliably. At low rates, Filtering stays modestly stronger than PINN. As in the cross-trajectory experiment, this indicates physics-informing does not only learn to suppress a specific attack pattern but also benefits from the underlying physical consistency, so the improved robustness transfers across attack strengths.

RQ4 Takeaway. Our results demonstrate that attacks on one sensor do not affect the detectability of poisoning on other sensors. Moreover, we show that the robustness generalizes across different ramp rates and poisoning patterns (Appendix G).

6.5 RQ5: Robustness to Adaptive Attacks

Next, we ask how the robustness improvement of the physics-informed approaches changes when an adaptive attacker poisons with physically coordinated attacks, by leveraging equation knowledge and spoofing the dependent variables so the mass-balance relation holds (Section 5.2). Such samples remain plausible under the modeled dynamics, and the attacker uses them in both training and testing to circumvent the physics regularization.

The robustness advantage of physics-informing decreases in this setting (Level: Figure 15a, Flow: Figure 15b). Under flow poisoning (easier to detect, Section 6.3.2), PINN, Filtering, and Data-Only now perform comparably, while under level poisoning, Filtering retains a small improvement whose uncertainty band overlaps with PINN.

RQ5 Takeaway. Under an adaptive attacker, the proposed physics-informed methodology demonstrates limited robustness. The gap between physics-informed and Data-Only approaches, evident under single point poisoning, disappears for coordinated attacks. This supports the intuition that the physics prior corrects training data violating the modeled physical relationships. When the poisoning is consistent with the physical manifold, residuals stay near zero, affecting the physics-informed approaches.

7 Discussion

In this section, we critically discuss the findings, compare the two proposed approaches, and discuss the defense limitations.

7.1 Filtering vs. PINN

Shared Properties. Both defenses work on the same physics residuals and use the violation of this equation to identify poisoned samples. They successfully regularize against single point poisoning (Section 6.3). Single point poisoning leaves a non-zero residual, which the filter uses to remove the violating samples and the physics loss uses as a corrective gradient, with the effect growing with the ramp rate (Section 6.3.3). This robustness generalizes to other single point poisoning attacks across shapes and rates (Section 6.4), and it also extends to multiple modeled equations and transfers to a larger plant (Section 6.3.4). Coordinated poisoning (Section 6.5) instead stays on the physically valid manifold, where the residual remains near zero, so neither approach contributes meaningful regularization against it. However, such poisoning is more demanding for the attacker than the single point ones, since it requires system knowledge of the modeled relations and, whenever the attack acts on the live process, coordinated, simultaneous multi-point spoofing. On unpoisoned data, the physics signal and the data observations agree, so neither approach reduces detection performance relative to the Data-Only baseline (Section 6.1). Physics-informing is therefore composable but bounded: a complete physical model is not required, as a limited set of equations lets us learn the remaining process information through the data loss. At the same time, this only protects what the physical equations capture.

Key Differences. Our proposed physics-informed approaches differ in how they incorporate the physics prior knowledge, either as a hard constraint (Filtering) or as a soft constraint (PINN). This leads to the following differences: (a) PINN training needs no threshold fitted on clean reference data. (b) PINN keeps the poisoning data, whereas Filtering discards whole windows. The lost training signal hurts more on complex plants and at high poison fractions, while PINN retains a signal even when the entire set is poisoned (Section 6.2.2). (c) The same window-dropping lets Filtering also remove secondary effects, such as the out-of-range tank levels the poisoning introduces, so the model never learns them as normal. PINN constrains only the modeled equation and keeps these levels, losing detection of the PLC-logic attacks that exploit them (Section 6.2.1). (d) PINN makes the forecaster more physically consistent on clean training data by emphasizing the modeled relation, improving detection of physically invalid attacks (Section 6.1). Filtering does not improve detection on clean data, and it might induce a lower performance by dropping benign samples.

7.2 Performance Implications

Adding the physics loss does not increase the computational training cost significantly, with the difference lying well within the run-to-run variation. When training on clean data using an NVIDIA RTX PRO 6000 Blackwell GPU, the per-epoch compute is unchanged (0.26 s/epoch), with PINNs converging after 130 epochs on average, approx. 7% more than the Data-Only baseline. Filtering instead adds a one-time preprocessing pass over the training set, whose runtime is negligible compared to a single training run. Additionally, the clean-data results show that physics-informed models perform at least as well as the Data-Only baseline (Section 6.1). Thus, physics-informing provides a low-cost countermeasure against poisoning.

7.3 Limitations

Trusted Held-Out Set. Although the large training set may be poisoned, threshold-based detection still requires a small held-out set that is clean and representative. This requirement is inherent to such methods, which identify anomalies based on deviations from normal behavior. Regardless of the forecasting or reconstruction model used, if the threshold-calibration set contains attack samples, their prediction or reconstruction errors may be treated as normal. Our proposed approach, therefore, still assumes access to a trusted clean held-out set; however, this set can be small enough to be manually inspected by a human operator, making the process far less costly than verifying the entire training set.

Evading the Physics Signal. We identify that an attacker who keeps physical violations sufficiently small can evade the regularization, revealing three main implications. (1) An adaptive attacker with knowledge of the modeled process can spoof sensor readings to satisfy the equations, as in the coordinated poisoning of Section 6.5. At the same time, while this was practical for the small PLC1 stage of SWaT, we argue that the effort increases with plant size, as consistency must be preserved across coupled downstream stages requiring a strong attack capability. (2) Knowing which relations are modeled also reveals which variables are not covered by the physical model. Since the physics loss only constrains the modeled variables, an attacker can target the un-modeled ones and avoid the influence of physics-based regularization. (3) A stealthy attacker can keep the per-timestep residual small and extend the manipulation over time. All considered variants lose detection performance at low ramp rates, where the timestep violation stays within the normal residual range. Violations that stay small at each step are hard to separate from sensor noise. This is also a known property of stateless detectors [30] and not exclusive to physics-informing. Detectors that aggregate violations over time, e.g., stateful cumulative approaches may recover this signal [30].

Expert Modeling. In this work, we model the high-level governing equations of the SWaT testbed based on publicly available information. In practice, physics-informed anomaly detection incurs an additional modeling cost and requires access to plant-specific process information. We assume that plant operators possess such knowledge, for example, the cross-sectional area of a tank. Thus, physics-informed methods are most applicable in settings where the relevant process information is available. A promising direction for future work is to extend the incorporated physical knowledge beyond the considered conservation laws, for instance, by including PLC logic to capture secondary effects in the system dynamics.

Evaluation Scope. We evaluated a single testbed (SWaT), both on the PLC1 stage and on the full plant. Although we observed that the regularization capability transferred to the larger plant, we did not evaluate other domains, such as smart grids. However, since prior work in smart grids already incorporates Kirchhoff's laws [37], evaluating it in electrical plants is a natural next step.

8 Conclusion

In this work, we present a practical defense against physics-violating data poisoning attacks in ICS anomaly detection. We propose two complementary methods—physics-informed filtering and physics-informed neural networks—to safeguard trained anomaly detection models against poisoning attempts. We empirically demonstrate the effectiveness of the proposed techniques on the SWaT dataset, evaluating both Stage 1 and the full process. Our results show that leveraging knowledge of the underlying physical process preserves clean accuracy while preventing the model from learning poisoned samples (keeping a high recall of 89%). Furthermore, we demonstrate that a sophisticated adaptive attacker may still evade detection by physically coordinating the attack. Nevertheless, our defenses substantially raise the bar for a successful attack, forcing the adversary into more complex and coordinated manipulations.

Acknowledgments

The authors gratefully acknowledge funding from the Helmholtz Association (HGF) within the topic "46.23 Engineering Secure Systems." Alessandro Erba acknowledges support as a Fellow of the Young Investigator Group Preparation Program funded jointly via the University of Excellence strategic fund at the Karlsruhe Institute of Technology (administered by the federal government of Germany) and the Ministry of Science, Research and Arts of Baden-Württemberg (Germany).

References

[1] Marshall Abrams. 2008. Malicious control system cyber security attack case study–Maroochy Water Services, Australia. (2008).

[2] Daniel Arp, Erwin Quiring, Feargus Pendlebury, Alexander Warnecke, Fabio Pierazzi, Christian Wressnegger, Lorenzo Cavallaro, and Konrad Rieck. 2022. Dos and Don'ts of Machine Learning in Computer Security. In *Proc. of the USENIX Security Symposium*.

[3] Shameek Bhattacharjee, Mohammad Jaminur Islam, and Sahar Abedzadeh. 2022. Robust anomaly based attack detection in smart grids under data poisoning attacks. In *Proc. of the Cyber-Physical System Security Workshop (CPSS)*.

[4] John H Castellanos, Mohamed Maghenem, Alvaro A Cárdenas, Ricardo G Sanfelice, and Jianying Zhou. 2023. Provable Adversarial Safety in Cyber-Physical Systems. In *Proc. of the IEEE European Symposium on Security and Privacy (EuroS&P)*.

[5] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. 2018. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *Proc. of the International Conference on Machine Learning (ICML)*.

[6] Salvatore Cuomo, Vincenzo Schiano Di Cola, Fabio Giampaolo, Gianluigi Rozza, Maziar Raissi, and Francesco Piccialli. 2022. Scientific machine learning through physics–informed neural networks: Where we are and what's next. *Journal of Scientific Computing* 92, 3 (2022).

[7] Ailin Deng and Bryan Hooi. 2021. Graph neural network-based anomaly detection in multivariate time series. In *Proc. of the Annual AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 35.

[8] Alessandro Erba, Anne Müller, and Nils Ole Tippenhauer. 2022. Security analysis of vendor implementations of the OPC UA protocol for industrial control systems. In *Proc. of the ACM Workshop on CPS & IoT Security and Privacy (CPS&IoTSEC)*.

[9] Alessandro Erba, Andres F. Murillo, Riccardo Taormina, Stefano Galelli, and Nils Ole Tippenhauer. 2024. On Practical Realization of Evasion Attacks for Industrial Control Systems. In *Proc. of the International Workshop on Re-design Industrial Control Systems with Security (RICSS)*.

[10] Alessandro Erba, Riccardo Taormina, Stefano Galelli, Marcello Pogliani, Michele Carminati, Stefano Zanero, and Nils Ole Tippenhauer. 2020. Constrained concealment attacks against reconstruction-based anomaly detectors in industrial control systems. In *Proc. of the Annual Computer Security Applications Conference (ACSAC)*.

[11] Daniel Fährmann, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. 2022. Lightweight long short-term memory variational auto-encoder for multivariate time series anomaly detection in industrial control systems. *Sensors* 22, 8 (2022).

[12] Cheng Feng, Venkata Reddy Palleti, Aditya Mathur, and Deeph Chana. 2019. A Systematic Framework to Generate Invariants for Anomaly Detection in Industrial Control Systems. In *Proc. of the Network and Distributed System Security Symposium (NDSS)*.

[13] Clement Fung, Shreya Srinarasi, Keane Lucas, Hay Bryan Phee, and Lujo Bauer. 2022. Perspectives from a comprehensive evaluation of reconstruction-based anomaly detection in industrial control systems. In *Proc. of the European Symposium on Research in Computer Security (ESORICS)*.

[14] Clement Fung, Eric Zeng, and Lujo Bauer. 2024. Attributions for ML-based ICS Anomaly Detection: From Theory to Practice. In *Proc. of the Network and Distributed System Security Symposium (NDSS)*.

[15] Jairo Giraldo, David Urbina, Alvaro Cardenas, Junia Valente, Mustafa Faisal, Justin Ruths, Nils Ole Tippenhauer, Henrik Sandberg, and Richard Candell. 2018. A survey of physics-based attack detection in cyber-physical systems. *ACM Computing Surveys (CSUR)* 51, 4 (2018).

[16] Jonathan Goh, Sridhar Adepu, Khurum Nazir Junejo, and Aditya Mathur. 2016. A dataset to support research in the design of secure water treatment systems. In *International conference on critical information infrastructures security*.

[17] Jonathan Goh, Sridhar Adepu, Marcus Tan, and Zi Shan Lee. 2017. Anomaly detection in cyber physical systems using recurrent neural networks. In *Proc. of the international symposium on high assurance systems engineering (HASE)*.

[18] Jun Inoue, Yoriyuki Yamagata, Yuqi Chen, Christopher M Poskitt, and Jun Sun. 2017. Anomaly detection for a water treatment system using unsupervised machine learning. In *Proc. of the IEEE international conference on data mining workshops (ICDMW)*.

[19] Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[20] Moshe Kravchik, Luca Demetrio, Battista Biggio, and Asaf Shabtai. 2022. Practical evaluation of poisoning attacks on online anomaly detectors in industrial control systems. *Computers & Security* 122 (2022).

[21] Moshe Kravchik and Asaf Shabtai. 2018. Detecting cyber attacks in industrial control systems using convolutional neural networks. In *Proc. of the Workshop on Cyber-Physical Systems Security and Privacy (CPS-SPC)*.

[22] Jiangnan Li, Yingyuan Yang, Jinyuan Stella Sun, Kevin Tomsovic, and Hairong Qi. 2021. Conaml: Constrained adversarial machine learning for cyber-physical systems. In *Proc. of the ACM Asia Conference on Computer and Communications Security (ASIA CCS)*.

[23] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, Puneet Agarwal, et al. 2015. Long short term memory networks for anomaly detection in time series. In *The European Symposium on Artificial Neural Networks*.

[24] Andrea Paudice, Luis Muñoz-González, Andras Gyorgy, and Emil C Lupu. 2018. Detection of adversarial training examples in poisoning attacks through anomaly detection. *arXiv preprint arXiv:1802.03041* (2018).

[25] CERT Polska. 2026. Energy Sector Incident Report—29 December 2025. *CERT Polska: Warsaw, Poland* (2026).

[26] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics* 378 (2019).

[27] Luis Salazar, Sebastián R Castro, Juan Lozano, Keerthi Koneru, Emmanuele Zambon, Bing Huang, Ross Baldick, Marina Krotofil, Alonso Rojas, and Alvaro A Cardenas. 2024. A tale of two Industroyers: It was the season of darkness. In *Proc. of the IEEE Symposium on Security and Privacy (S&P)*.

[28] Riccardo Taormina and Stefano Galelli. 2018. Deep-learning approach to the detection and localization of cyber-physical attacks on water distribution systems. *Journal of Water Resources Planning and Management* 144, 10 (2018).

[29] Shreshth Tuli, Giuliano Casale, and Nicholas R. Jennings. 2022. TranAD: deep transformer networks for anomaly detection in multivariate time series data. *Proc. VLDB Endow.* 15, 6 (Feb. 2022). doi:10.14778/3514061.3514067

[30] David I Urbina, Jairo A Giraldo, Alvaro A Cardenas, Nils Ole Tippenhauer, Junia Valente, Mustafa Faisal, Justin Ruths, Richard Candell, and Henrik Sandberg. 2016. Limiting the impact of stealthy attacks on industrial control systems. In *Proc. of the ACM Conference on Computer and Communications Security (CCS)*.

[31] Tim Walita, Alessandro Erba, John H Castellanos, and Nils Ole Tippenhauer. 2023. Blind concealment from reconstruction-based attack detectors for industrial control systems via backdoor attacks. In *Proc. of the Cyber-Physical System Security Workshop (CPSS)*.

[32] Sharon Weinberger. 2011. Computer security: Is this the start of cyberwarfare? *Nature* 174 (June 2011).

[33] Theodore Joseph Williams. 1992. *The Purdue Enterprise Reference Architecture, A Technical Guide for CIM Planning and Implementation*.

[34] Konrad Wolsing, Eric Wagner, Luisa Lux, Klaus Wehrle, and Martin Henze. 2025. GeCos replacing experts: Generalizable and comprehensible industrial intrusion detection. In *Proc. of the USENIX Security Symposium*.

[35] Weiqiang Wu, Chunyue Song, Jun Zhao, and Zuhua Xu. 2023. Physics-informed gated recurrent graph attention unit network for anomaly detection in industrial cyber-physical systems. *Information Sciences* 629 (2023).

[36] Mehdi Jabbari Zideh, Paroma Chatterjee, and Anurag K Srivastava. 2023. Physics-informed machine learning for data anomaly detection, classification, localization, and mitigation: A review, challenges, and path forward. *IEEE Access* 12 (2023).

[37] Mehdi Jabbari Zideh and Sarika Khushalani Solanki. 2024. Physics-informed convolutional autoencoder for cyber anomaly detection in power distribution grids. In *2024 IEEE Power & Energy Society General Meeting (PESGM)*.

A Sampling-Rate Diagnostic

Sensor measurements are noisy, while the underlying physical state evolves continuously and often slowly. Figure 16 shows an exemplary tank at a fast sampling rate and at a slower downsampled one. When sampling too fast, the noise exceeds the real physical change between two timesteps, so the physical equation motivates fitting noise, while sampling too slow grows the discretization error. We thus select the fastest sampling rate without noise domination, using a residual-based diagnostic (Γ) on the clean reference data.

Table 1: Rules from Wolsing et al. [34] used for invariant-based filtering. A timestep is flagged if any rule is violated.

Rule	Condition
R_1	$\text{LIT101} < 490 \Rightarrow \text{MV101} = 2 \wedge \text{FIT101} > 0.1$
R_2	$\text{LIT101} > 810 \Rightarrow \text{MV101} = 1 \wedge \text{FIT101} < 1$
R_3	$\text{MV101} = 2 \Rightarrow \text{FIT101} > 0.1$
R_4	$\text{LIT101} < 490 \Rightarrow \text{FIT101} > 0$

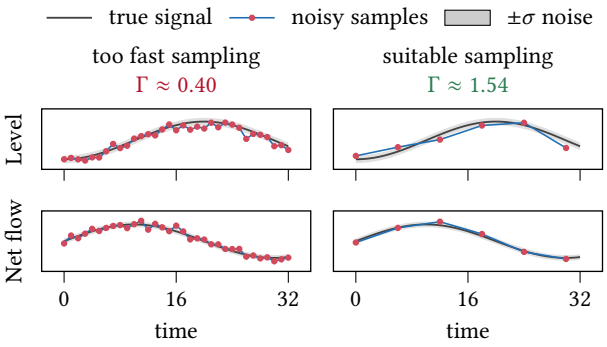


Figure 16: Sampling-rate selection, tank example. Too fast: noise dominates over the real physical change ($\Gamma < 1$). At a suitable rate the physical change exceeds the noise ($\Gamma > 1$).

For integrative conservation laws, such as the tank balance Eq. 2, the net flows are accumulated over Δt. With approximately constant measurement noise, increasing Δt improves the signal-to-noise ratio. To find the smallest viable Δt, we compare two quantities, the physical level change given by the net flow over the interval, $S_{\text{phys}} = \frac{\Delta t}{A}(\text{inflow}_t - \text{outflow}_t)$, which acts as the signal we want the equation to capture. The second is the mass-balance residual evaluated on the true measurements, $N_{\text{resid}} = \text{level}_t + \frac{\Delta t}{A}(\text{inflow}_{t+1} - \text{outflow}_{t+1}) - \text{level}_{t+1}$, which on physically consistent data reduces to the measurement noise. Both quantities are evaluated over all timesteps and compared through their root-mean-square values. We want the noise floor to stay below the physical signal, quantified by their ratio

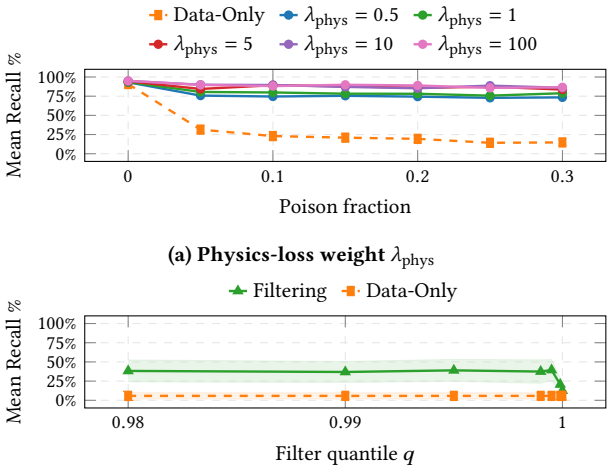
$$\Gamma(\Delta t) = \frac{S_{\text{phys}}(\Delta t)}{N_{\text{resid}}(\Delta t)}. \tag{9}$$

$\Gamma < 1$ means the physical signal S_{phys} is dominated by the noise floor N_{resid}. $\Gamma > 1$ means the equation prefers the true dynamics. We select the smallest Δt with $\Gamma > 1$. This criterion identifies the noise-limited lower bound, but is not guaranteed to find an upper bound. We avoid a too slow sampling by selecting the fastest possible sampling rate for which all equations yield a $\Gamma > 1$.

This criterion only applies to equations that integrate over Δt. Equations without such a term (e.g. Eq. 12) accumulate nothing between timesteps, so $\Gamma = 1$ for every rate and the criterion cannot select one. However, their signal-to-noise ratio still benefits from downsampling, since averaging consecutive measurements suppresses the per-timestep noise.

B Sensitivity to the Physics-Loss Weight

The physics weight λ_{phys} controls the regularization strength. Figure 17a shows that higher physics weights improve robustness against the single point level attack, while lower weights reduce it. However, tuning λ_{phys} on attack performance would assume attack data unavailable at training time, while the available clean-data criterion, validation forecasting accuracy, tends to select a weaker regularization. We therefore retain $\lambda_{\text{phys}} = 1.0$, the operating point at which data and physics losses contribute equally on clean data by construction (Section 4.4.2), as an attack-agnostic setting. This is



(a) Physics-loss weight λ_{phys}

(b) Filtering threshold q

Figure 17: Sensitivity of the two hyperparameters that cannot be tuned on attack data, under single point level poisoning (LIT101). (a) Detection recall across physics weights, trained and evaluated at a $1.943\,\text{mm}\,\text{s}^{-1}$ ramp rate. (b) Detection recall across filter quantiles, trained and evaluated on low-rate $0.5\,\text{mm}\,\text{s}^{-1}$ ramps.

conservative, as higher weights would have yielded stronger robustness. Across all evaluated weights, both the FPR and the detection of the original SWaT attacks stay within uncertainty.

C Sensitivity to the Filtering Threshold

The effectiveness of Filtering depends on how clearly poisoned residuals separate from the upper tail of clean residuals, and on placing the quantile threshold between them (Section 4.3). Figure 17b shows this for single point level poisoning (LIT101) at a low $0.5\,\text{mm}\,\text{s}^{-1}$ ramp rate, a stress test inducing smaller violations than the default ramp. Filtering stays clearly above Data-Only across most quantiles, but degrades abruptly once the threshold becomes slightly too permissive. As with λ_{phys}, the threshold cannot be tuned on attack data at training time, so this sharp transition is a practical liability of the hard constraint, which PINN's soft constraint avoids.

D Full SWaT Physical Equations

For the full-plant evaluation of Section 5.2, we extend the modeled relations beyond the PLC1 tank by two further tank equations,

$$\dot{LIT}_{301} = \frac{1}{A}\left(FIT_{201} - FIT_{301}\right) + b, \tag{10}$$

$$\dot{LIT}_{401} = \frac{1}{A}\left(FIT_{301} - FIT_{401}\right) + b + b_{mode}, \tag{11}$$

where b_{mode} models the water loss during filter flushing and is gated by a mode mask, active only when a drain valve is open, and a tank-3 pump is on.

We model the flow conservation through process stages 4 and 5

$$FIT_{501} = FIT_{401} + b, \tag{12}$$

$$FIT_{501} = FIT_{502} + FIT_{503} - FIT_{504} + b. \tag{13}$$

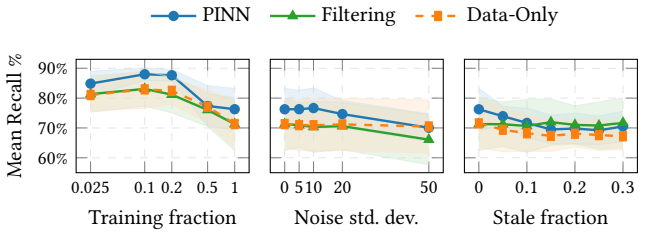
Table 2: Timestep-wise detection metrics under clean training, mean ± std over 40 runs. Prevalence: fraction of timesteps under attack.

(a) PLC1, per-sensor detector, 1.6% prevalence.

Variant	Recall	Precision	F1	Mean Rec.
Data-Only	0.78 ± 0.07	0.32 ± 0.02	0.45 ± 0.03	0.72 ± 0.08
Filtering	0.78 ± 0.07	0.32 ± 0.02	0.45 ± 0.02	0.71 ± 0.08
PINN	0.83 ± 0.05	0.31 ± 0.03	0.45 ± 0.03	0.77 ± 0.07

(b) Full plant, MSE detector, 11.5% prevalence.

Variant	Recall	Precision	F1	Mean Rec.
Data-Only	0.61 ± 0.25	0.59 ± 0.21	0.59 ± 0.23	0.29 ± 0.06
Filtering	0.56 ± 0.27	0.61 ± 0.25	0.58 ± 0.27	0.24 ± 0.06
PINN	0.58 ± 0.26	0.57 ± 0.22	0.57 ± 0.24	0.29 ± 0.06



(a) Scarce data **(b) Sensor noise** **(c) Stale episodes**

Figure 18: Detection recall under degraded training, evaluated on the original SWaT attacks. (a) Reduced clean-training data: models are trained on different fractions of the clean SWaT training set. (b) Additive Gaussian noise on the level sensor LIT101 during training, expressed as multiples of the standard deviation of the level change during normal operation. (c) Stale level-sensor episodes on LIT101 during training, with a per-episode duration $\in [50, 300]$ s.

Tracking the full water flow from stage 1 to 6 yields

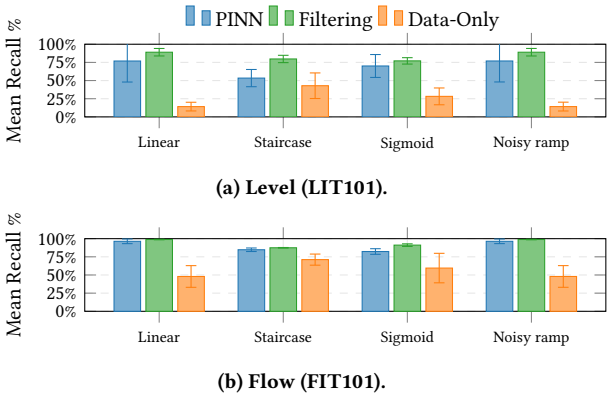
$$A\left(\dot{LIT}_{101} + \dot{LIT}_{301} + \dot{LIT}_{401}\right) = FIT_{101} - FIT_{502} - FIT_{503} + FIT_{504} + b + b_{\text{mode}}. \quad (14)$$

E Timestep-Wise Detection Metrics

Table 2 reports timestep-wise metrics under clean training. On PLC1, the low prevalence means even a low FPR yields many false positives relative to the true positives, limiting precision. On the full plant, attack 28 covers 66% of the attacked timesteps, so detecting it dominates the scores and explains their large uncertainty, while the mean recall is more stable as each attack contributes equally.

F Detection Under Degraded Training

Detection Under Scarce Clean Training Data. Training on different fractions of the clean training set, PINN and Data-Only perform similarly in low-data regimes (Figure 18a), and Filtering comparably to Data-Only, as it removes little data during clean operation.



(a) Level (LIT101).

(b) Flow (FIT101).

Figure 19: Cross-trajectory transfer under single point poisoning at the 30% poison rate. Models are trained with linear-ramp poisoning and evaluated on the shown trajectories.

Reducing the clean-data amount thus does not change the detection problem for the PLC1 attacks, so the poisoning results are unlikely to stem solely from reduced data availability.

Robustness to Noisy Level-Sensor Measurements. We simulate a degraded sensor by adding Gaussian noise to LIT101 during training, scaled as multiples of the standard deviation of the level change during normal operation (Figure 18b). The physics-informed variants remain stable up to ten times this standard deviation and degrade only under more extreme noise, where the physics loss receives a less reliable signal. As in Appendix A, the physics term depends on the quality of the variables entering the equation, but only at noise levels well beyond those expected from a functioning sensor.

Impact of Stale Level-Sensor Episodes. We inject episodes in which LIT101 remains fixed (Figure 18c). The PINN slightly loses its initial improvement as the stale fraction increases, while Filtering and Data-Only stay comparable across fractions. A stale sensor produces a residual equal only to the tank evolution missed between two consecutive timesteps, which is small during steady operation and yields weak regularization.

G Cross-Trajectory Generalization

We test whether physics-informed approaches exclusively suppress a specific poisoning pattern. We train over linear ramp poisoning and evaluate over unseen attacks with different shapes (staircase, sigmoid, and noisy ramp attacks). The new attack shapes only change the trajectory in the interpolation but still reach the same final spoofed value at the same time as the linear shape.

The advantage of PINN and Filtering remains visible across the attack trajectories, indicating that the improvement is not limited to the linear ramp patterns used during training (Level: see Figure 19a, Flow: see Figure 19b). Instead, the results suggest that physics-informed models capture the underlying physical inconsistency introduced by the attack rather than the specific training pattern.