Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

An Adaptive Hybrid Deep Learning Framework for Known and Zero- Day Cyberattack Detection in Next-Generation Smart Grids

H S Chethana
Kalpataru Institute of Technology
S Mallikarjunaswamy
JSS Academy of Technical Education, Bengaluru
N Sharmila
sharmilan@jssstuniv.in

JSS Science and Technology University
C Chandana
RNS Institute of Technology
D Prakyath
SJB Institute of Technology

Research Article

Keywords:

Posted Date: November 25th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11514914/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

An Adaptive Hybrid Deep Learning Framework for Known and Zero-Day Cyberattack Detection in Next-Generation Smart Grids

Chethana H S[1], Mallikarjunaswamy S[2], Sharmila N[3,*], Chandana C[4], Prakyath D[5]

[1]Department of Electronics and Communication Engineering, Kalpataru Institute of Technology, Tiptur, Karnataka, India
[2]Department of Electronics and Communication Engineering, JSS Academy of Technical Education, Bengaluru, Karnataka, India
[3]Department of Electrical and Electronics Engineering, JSS Science and Technology University, Mysuru, Karnataka, India
[4]Department of Computer Science and Engineering (AI&ML), RNS Institute of Technology, Bengaluru, Karnataka, India
[5]Department of Electrical and Electronics Engineering, SJB Institute of Technology, Bengaluru, Karnataka, India
[*]Corresponding author: Sharmila N, sharmilan@jssstuniv.in

Abstract Next-generation smart grids are exposed to false data injection, denial-of-service, replay, malware and previously unseen (zero-day) cyberattacks, yet most deep-learning intrusion detectors are trained on a closed set of known attacks and misclassify novel attacks silently. This paper proposes an adaptive physics-guided dual-branch Transformer–autoencoder intrusion detection (PG-ADTA-ID) framework. A physics-guided spatiotemporal Transformer with a breaker-aware graph embedding and Kirchhoff, Ohm and metering residuals classifies known attacks. A self-supervised denoising autoencoder models the known world through a reconstruction-error signature computed on channels that do not scale with the operating point. A physics-consistent open-set fusion turns classification uncertainty, reconstruction surprise and physics-consistency surprise into a calibrated novelty-risk score with a three-way decision, and physics-gated continual learning adapts the threshold and the models under concept drift. Formal results establish the operating-point invariance of the physics residuals, the visibility of power-flow-consistent injections, false-alarm control and the bounded influence of poisoning on adaptation. On a cyber-physical co-simulation of the IEEE 14-bus system with four known and five withheld attack families, PG-ADTA-ID achieved 99.30% accuracy and a 98.74% macro-F1-score on known attacks, detected 90.7% of the zero-day attack episodes and flagged 54.4% of the zero-day windows as unknown at a false-positive rate of 1.69%. Its area under the receiver operating characteristic curve for zero-day versus known traffic was 0.934, against at most 0.804 for eleven baselines. Under concept drift, continual learning reduced the false-positive rate from 100% to 11.0%, and a complete decision took 7.23 ms per window on a two-core processor.

Keywords Smart grid cybersecurity, Zero-day attack detection, Open-set recognition, Physics-guided deep learning, Transformer–autoencoder, Concept drift

Nomenclature

Abbreviations. ADASYN, adaptive synthetic oversampling; AMI, advanced metering infrastructure; AUPR, area under the precision–recall curve; AUROC, area under the receiver operating characteristic curve; BiGRU, bidirectional gated recurrent unit; BO, Bayesian optimisation; CDCL, concept-drift-aware continual learning; CNN, convolutional neural network; CP-DAP, cyber-physical data collection and adaptive preprocessing; DAE, denoising autoencoder; DoS, denial of service; FDIA, false data injection attack; FFN, feed-forward network; FPR, false-positive rate; GA, genetic algorithm; GW, gateway; IDS, intrusion detection system; KCL, Kirchhoff's current law; KS, Kolmogorov–Smirnov; LN, layer normalisation; LOCO, leave-one-known-class-out; LSTM, long short-term memory; MCC, Matthews correlation coefficient; MI, mutual information; MITM, man-in-the-middle; PC-OAF, physics-consistent open-set adaptive fusion; PDC, phasor data concentrator; PG-ADTA-ID, physics-guided adaptive dual-branch Transformer–autoencoder intrusion detection; PGST, physics-guided spatiotemporal Transformer; PMU, phasor measurement unit; PSO, particle swarm optimisation; PV, photovoltaic; RTU, remote terminal unit; SA-DAE, self-supervised adaptive denoising autoencoder; SCADA, supervisory control and data acquisition; SVM, support vector machine; TPE, tree-structured Parzen estimator; UNC, uncertain decision; ZD, suspected zero-day decision; ZDR, zero-day detection rate.

Symbols.

Symbol	Description	Symbol	Description
$\boldsymbol{x}_t$, $\boldsymbol{X}_t$	Fused feature vector at step t; window ending at t	$\boldsymbol{z}_t$	PGST window embedding
T, Δt	Window length; time step	$p_{t,c}$, T_c	Probability of class c; temperature
d	Number of fused features	$\mathcal{L}_{\text{cls}}$, $\mathcal{L}_{\text{DAE}}$	PGST and SA-DAE training losses
$\mathcal{K}$, C	Set and number of known classes (0 = normal)	σ_e, ρ_m, $\boldsymbol{M}_t$	Noise level; masking ratio; binary mask
$\mathcal{U}$	Set of unknown (zero-day) attack families	$e_{t,g}$, $\boldsymbol{r}_t$, G	Group log error; error signature; number of groups
$\hat{y}_t$	Decision for the window ending at t	u_t, a_t, c_t	Uncertainty, reconstruction surprise and physics surprise
ϕ_t, $\overline{\phi}_t$	Data freshness; its mean over the window	$D^2_{z,t}$, $D^2_{r,t}$, $D^2_{q,t}$	Mahalanobis distances of the three indicators
r^{KCL}, r^{Ohm}, r^{AMI}	Kirchhoff, Ohm's-law and AMI–SCADA residuals	κ_z, κ_r, κ_q	Scales of the three indicators
$\zeta_{t,j}$	Robustly standardised residual	λ_s	Covariance shrinkage factor

Symbol	Description	Symbol	Description
$\boldsymbol{q}_t$	Physics-consistency profile	α, β, γ	Fusion weights of u_t, a_t and c_t
$\boldsymbol{A}_t, \hat{\boldsymbol{A}}_t$	Breaker-aware adjacency; normalised adjacency	S_t	Fused novelty-risk score
d_m, L_e, h	Model width; encoder layers; attention heads	τ_0, τ_t	Calibrated and adapted novelty threshold
δ	Budget (quantile level of τ_0)	η	Minimum known-class confidence
$\mathcal{T}_t, \pi_T, c_{\text{trust}}$	Trusted set; its confidence and physics limits	$\mathcal{B}_t, \mathcal{R}$	Buffer of trusted scores; drift-test reference
$D_{\text{KS}}, D_{\min}, \alpha_{\text{KS}}$	KS distance; its floor; significance level	ρ, Δ_τ	Threshold adaptation rate; maximum step
$B_{\min}, B_{\text{ft}}, K_{\text{ft}}$	Buffer sizes for testing and fine-tuning; blocks between fine-tunings	$\mathcal{M}, \rho_c$	Replay memory; recalibration weight
J, λ_1–λ_4	Optimisation objective and its weights	$\varepsilon_{\text{FPR}}, T_{\text{norm}}$	FPR normaliser; normalised inference time
$\boldsymbol{\xi}, \boldsymbol{h}_{\text{pf}}$	Grid state (voltage angles and magnitudes); AC measurement function	$\boldsymbol{B}, \Delta\boldsymbol{P}$	DC bus susceptance matrix; change of active injections
m_v	Dimension of an evidence vector $\boldsymbol{v} \in \{\boldsymbol{z}_t, \boldsymbol{r}_t, \boldsymbol{q}_t\}$	ν, D_{clean}	Fraction of adversarial scores in $\mathcal{B}_t$; KS distance of its clean part

1 Introduction

Smart grids depend on a tight coupling of physical power flows and digital communication. This section explains why that coupling makes zero-day attacks hard to detect with existing intrusion detectors (Section 1.1), summarises the proposed framework and its contributions (Section 1.2) and outlines the rest of the paper (Section 1.3).

1.1 Background and motivation

Next-generation smart grids combine advanced metering infrastructure (AMI), supervisory control and data acquisition (SCADA) systems, phasor measurement units (PMUs), IEC 61850 digital substations, distributed energy resources and electric-vehicle (EV) charging infrastructure. These assets exchange data over bidirectional networks, which improves observability and control (Rouhani and Su 2026; Muhammed et al. 2026) but greatly enlarges the attack surface. An adversary who compromises a communication path, a gateway or a field device can inject false measurements, flood control channels, replay recorded telemetry, implant malware, issue unauthorised switching commands, spoof the GPS time reference of PMUs or map the network before a larger campaign (Mohammadpourfard et al. 2026; Verma et al. 2026; Ansari et al. 2026; Qiu et al. 2026). Such actions corrupt state estimation, mislead dispatch and protection, and ultimately threaten the reliability of the bulk power system (Ansari et al. 2026).

Intrusion detection systems (IDSs) are therefore an essential layer of defence. Support vector machines (SVMs), random forests and long short-term memory (LSTM) networks have been studied extensively for smart-grid attack detection (Mohammed et al. 2026; Singh and Sailaja Kumari 2026; Sheta et al. 2026), and recent hybrids reach high accuracy on public power-system benchmarks (Samin et al. 2026; Alamgir et al. 2026). Almost all of these detectors, however, make a *closed-world* assumption. A model trained on a fixed catalogue of attacks can only answer with a label it has seen, so a previously unseen (zero-day) attack is silently mapped to "normal" or to the most similar known class. The problem is growing. Physics-constrained generative models can synthesise covert false-data-injection attacks (FDIAs) that bypass bad-data detection (Mohammadpourfard et al. 2026), deep-learning-assisted load-redistribution attacks no longer require topology knowledge (Verma et al. 2026), and large language models can help adversaries assemble new attack procedures against industrial controllers (Cook et al. 2026).

Three characteristics of power systems complicate detection further. First, the data distribution drifts: load growth, renewable penetration, reconfiguration and sensor replacement shift the statistics of normal operation, so detectors with fixed thresholds produce rising false-alarm rates (Yuan et al. 2026; Zhou et al. 2026). Second, benign disturbances such as faults, protection operations and authorised switching can resemble attacks, and an operator-facing IDS must not raise alarms for them (Farsi et al. 2026; Hosseini Rostami et al. 2026). Third, PMU phasors, SCADA telemetry from remote terminal units (RTUs), AMI readings and communication traffic arrive at different rates and must be synchronised and fused before joint reasoning is possible (Cibin et al. 2026a; Wang et al. 2026). Power systems also offer something that most IT networks do not: physical laws. Kirchhoff's current law, Ohm's law on every branch and the balance between metered demand and supervisory injections hold at every operating point and provide label-free consistency evidence (Massaoudi and Ez Eddin 2026; Wu et al. 2026; Balahewa et al. 2026).

The literature has advanced on several fronts: hybrid convolutional, recurrent and Transformer architectures for known-attack classification (Samin et al. 2026; Alamgir et al. 2026; Tirulo et al. 2026), graph neural networks for topology-aware detection (Aboelmagd et al. 2026; Xi et al. 2026a), reconstruction-based and generative models for label-free anomaly detection (Goyel et al. 2026; Badakhshan and Zhang 2026; Lozano-Paredes et al. 2026), federated learning for privacy (Chen et al. 2026; Jeter et al. 2026; Bunko et al. 2026) and explainable AI for operator trust (Farsi et al. 2026; Nasir et al. 2026a). These advances have mostly been pursued in isolation. Supervised detectors name known attacks but cannot reject unknown ones, unsupervised detectors flag deviations but cannot name the attack, physical consistency is seldom part of the open-set decision itself, and few methods adapt their decision boundary safely, without being poisoned by the attacks they should detect, when the operating regime drifts.

1.2 Proposed approach and contributions

This paper addresses these gaps with the Physics-Guided Adaptive Dual-Branch Transformer–Autoencoder Intrusion Detection (PG-ADTA-ID) framework. A physics-guided spatiotemporal Transformer classifies known attacks, and a

self-supervised denoising autoencoder learns the "known world" of verified operation and catalogued attacks. A physics-consistent open-set fusion turns their evidence into a calibrated novelty-risk score and labels every observation as normal, a known attack, a suspected zero-day attack or uncertain. A drift-aware continual-learning mechanism keeps the detector calibrated, using physical consistency, which does not drift, to decide which observations may be trusted for adaptation. The main contributions are as follows.

1. **Physics-guided spatiotemporal representation.** A Transformer encoder (PGST) is designed whose token embedding fuses cyber-physical measurements with a susceptance-weighted, breaker-aware graph convolution and explicit Kirchhoff, Ohm's-law and AMI–SCADA residuals.
2. **Known-world self-supervised anomaly modelling.** A temporal convolutional denoising autoencoder (SA-DAE) is trained with an adaptive noise curriculum and random masking, and its group-wise error signature is scored on channels that do not scale with the operating point.
3. **Physics-consistent open-set decision.** A fusion (PC-OAF) of aleatoric and epistemic uncertainty with class-conditional reconstruction and physics-consistency surprise is formulated, with a distribution-free quantile threshold and a three-way decision rule.
4. **Safe adaptation under concept drift.** A continual-learning mechanism (CDCL) gates adaptation by physical consistency, detects drift with a two-sample test and an effect-size floor, bounds every threshold update and fine-tunes both branches with replay.
5. **Theoretical analysis and complete pseudocode.** Formal results are derived for the operating-point invariance of the physics residuals, the visibility of power-flow-consistent injections, the common scale of the fused indicators, false-alarm control and the bounded influence of poisoning on adaptation, and the framework is specified by module-level algorithms and a complete pseudocode.
6. **Leakage-free optimisation and rigorous evaluation.** Bayesian optimisation tunes the networks, and Bayesian optimisation, particle swarm optimisation and a genetic algorithm are compared for the decision stage on a leave-one-known-class-out pseudo-open-set, without zero-day data. A reproducible IEEE 14-bus cyber-physical co-simulation with four known and five withheld attack families, disjoint temporal splits and a drift period is used to compare PG-ADTA-ID with eleven baselines, including the CNN–Transformer–BiGRU system of Samin et al. (2026), and a test on the IEEE 118-bus system examines scalability.

1.3 Organisation of the paper

Section 2 reviews related work and Section 3 states the research gaps and objectives. Section 4 describes the existing CNN–Transformer–BiGRU system. Section 5 presents PG-ADTA-ID, its algorithms, its complete pseudocode and its theoretical properties. Section 6 describes the experimental set-up, Section 7 reports and discusses the results, and Section 8 concludes the paper.

2 Related Work

This section reviews 35 studies published in 2026 that are closest to the present work. They fall into five groups: deep-learning intrusion detection for smart-grid networks, data-driven detection and localisation of false data injection, physics-informed and model-based detection, zero-day and continual detection, and privacy-preserving or optimised detectors. Table 1 summarises the comparison.

2.1 Deep-learning intrusion detection in smart-grid networks

Samin et al. (2026) proposed a hybrid architecture for smart-grid SCADA intrusion detection. Convolutional blocks capture local patterns, a Transformer encoder models global dependencies across nodes and a bidirectional gated recurrent unit (BiGRU) produces the sequence representation. Preprocessing uses mutual-information (MI) feature reduction to 60 features, adaptive synthetic oversampling (ADASYN) and train-fitted z-scoring. On 14,415 held-out samples of the Mississippi State University power-system benchmark the model reached 88% accuracy, a macro-F1 of 0.86, a Matthews correlation coefficient (MCC) of 0.74 and a ROC-AUC of 0.94, with about 9% false negatives and 7% false alarms. Because it is a supervised closed-set classifier, attacks absent from training cannot be flagged as unknown. It is therefore adopted as the existing system in this paper (Section 4). Alamgir et al. (2026) coupled a CNN–BiLSTM feature extractor with an Extra-Trees decision head for real-time detection on high-frequency smart-meter data. On 72,073 labelled records from the same testbed, covering FDI, denial-of-service (DoS), replay and man-in-the-middle attacks, they obtained 92.17% accuracy, an F1-score of 85.66% and 12 ms latency. The authors themselves note a domain-shift risk. Sharma et al. (2026) optimised a CNN–BiLSTM for cyber-physical anomaly detection on merged AMI and SCADA data containing six synthetically generated attack types, regularising it with dropout and L2 penalties and validating it with five-fold cross-validation. Chandrakala et al. (2026) combined secure signal encoding with temporal feature extraction for multi-class intrusion detection in smart-grid environments. Singh and Sailaja Kumari (2026) trained LSTM and GRU models, and a GRU-based transfer-learning variant, on PMU data covering AC/DC FDIA, DoS and physical events on the IEEE 14-, 24-, 39- and 118-bus systems, reaching up to 99.85% accuracy. Every event class in that study was known at training time. Farsi et al. (2026) built a unified framework that distinguishes physical disturbances from cyber intrusions on time-synchronised measurements. It uses multi-strategy feature selection, Optuna-tuned ensemble learners and permutation-SHAP explanations, and exceeds 96% on the main metrics across 37 event scenarios, again in a closed-set setting.

Digital substations add protocol-specific traffic to the problem. Presekal et al. (2026) mapped IEC 61850 GOOSE traffic onto a three-dimensional similarity vector, combining a CNN payload score with a Fourier and Kolmogorov–Smirnov inter-arrival score, and clustered the result with a self-organising map and DBSCAN. On a hardware-in-the-loop substation the method achieved accuracy and F1 above 95%. The protocol-specific network, however, must be retrained for each new protocol. Goyel et al. (2026) trained a Transformer to reconstruct sliding windows of GOOSE features without supervision. They fitted statistical distributions to the reconstruction errors to set alarm thresholds and reported ROC-AUCs of 96.84% and 95.73% on two datasets. Their detector uses frame features only and performs no physics-consistency check on the measurements.

2.2 False data injection detection and localisation

Mohammed et al. (2026) proposed IG-APSO-DNN, which selects features with information gain and adaptive particle swarm optimisation before classifying power-system measurements with a deep neural network. Li et al. (2026) fused residual blocks, multi-scale feature fusion, convolutional block attention, multi-head attention and a BiLSTM (MSFF-AEN) to detect and localise FDIAs jointly. Pan et al. (2026) combined a Transformer with a CNN to localise FDIAs on the IEEE 14- and 39-bus systems. Qu et al. (2026) localised FDIAs with multi-level wavelet decomposition and spatio-temporal graph embedding. Dzafic et al. (2026) applied Graph Deviation Networks, which learn inter-sensor dependency graphs and score deviations from forecast behaviour, to anomaly detection in smart-grid time series. Zhao et al. (2026) modelled measurements as continuous functions of time using implicit neural representations whose parameters are generated by a Transformer hypernetwork, combined with structural time-series decomposition. The result is a fully unsupervised detector of stealthy FDIA on the IEEE 14- and 118-bus systems. Alwaisi and Soderi (2026) used ontology-driven semantic encoding with LSTM networks to catch statistically plausible but semantically inconsistent injections in 6G-enabled smart grids, evaluated on a custom smart-meter dataset.

Graph-based detectors are increasingly common. Wang et al. (2026) proposed a cross-layer graph-Transformer network that links communication-layer data points to the grid buses they affect through a sensitivity-based projection. They reported detection accuracy above 0.98 on the IEEE 118-bus and ACTIVSg2000 systems and flagged up to 90% of attacks before execution. Aboelmagd et al. (2026) introduced a simplicial recurrent graph neural network that models higher-order relations among buses and lines, improving detection by 9–39% on systems of up to 70,000 buses. Xi et al. (2026a) combined frequency-domain attention with a bidirectional spatiotemporal graph network to localise FDIAs on the IEEE 14-, 57- and 118-bus systems. Tirulo et al. (2026) targeted FDIAs that corrupt demand-response load forecasts. Their detector combines a variance-adaptive 1D-CNN, a temporal-attention BiLSTM, Bayesian optimisation and hybrid resampling, and reaches 99.26% accuracy with 1.2 ms inference on 377,000 household forecasts. Sheta et al. (2026) tuned a soft-voting ensemble of tree learners with the IceCube optimisation metaheuristic and ADASYN, obtaining 99% accuracy and a 95% F1-score on the Oak Ridge power-system dataset. All of these detectors are supervised on attack scenarios defined in advance and are evaluated on a fixed topology. Robustness to attack families unseen in training is not established.

2.3 Physics-informed and model-based detection

Massaoudi and Ez Eddin (2026) embedded power-system constraints in a constraint-aware physics-informed neural network for cyber-physical intrusion detection. Wu et al. (2026) gave state estimation a controllable uncertainty: a spatiotemporal LSTM–GNN with Monte Carlo dropout and physics-consistency constraints feeds a detector that couples uncertainty expansion with statistical thresholds, which avoids reliance on prior attack knowledge. Hu et al. (2026) proposed a physics-guided dynamic graph attention network with adaptive edge weights derived from power–voltage coupling and physics-constrained losses for FDIA detection and localisation on the IEEE 13- and 123-bus feeders. Balahewa et al. (2026) treated the natural variability of renewable generation as a watermark, allowing each inverter-based resource to detect attacks locally without attack labels. Rajalakshmi and Velmurugan (2026) combined physics guidance with contrastive temporal graph learning for anomaly detection and root-cause localisation in industrial control systems. These studies show that physical knowledge improves interpretability and label efficiency. None of them, however, fuses physical consistency with a learned classifier and an anomaly model inside an open-set decision that distinguishes known from unknown attacks.

2.4 Zero-day, open-set and continual detection

Badakhshan and Zhang (2026) combined a Wasserstein GAN with gradient penalty and LSTM layers to score anomalies in grid-connected integrated energy systems. On the IEEE 118-bus system the detector reached 95.4% accuracy, but held-out-attack results were not reported separately. Wilkie et al. (2026) designed a contrastive loss that learns the benign distribution from benign and known-malicious traffic. It generalises to zero-day classes and, through an open-set extension, improves OpenAUC on enterprise network flows. Park and Lim (2026) proposed a hybrid multi-stage framework that identifies both zero-day attacks and known threats in network traffic, which is the same known/unknown split targeted here. Zheng et al. (2026) detected cyberattacks on smart-grid substations without supervision by combining contrastive autoencoding, adversarial training and a risk-aware formulation. Yuan et al. (2026) placed conformal prediction on top of machine-learning anomaly detectors for industrial cyber-physical systems to control false alarms under distribution shift. Nasreen Fathima et al. (2026) proposed CLEAR, a continual-learning framework that combines anomaly-based zero-day detection with model updating. Mohammadpourfard et al. (2026) showed that physics-constrained autoencoders, variational autoencoders and conditional GANs can generate

covert FDIAs that bypass bad-data detection on the IEEE 14- and 57-bus systems. This result directly motivates detectors that do not depend on a fixed attack catalogue. Of these zero-day studies, the network-level ones ignore grid physics, while the grid-level ones mostly provide anomaly scores without naming known attack classes.

2.5 Federated, privacy-preserving and optimised detectors

Chen et al. (2026) proposed a three-tier cloud–edge–terminal federated IDS for power networks that combines a CNN–BiLSTM detector with homomorphic encryption, differential privacy and adaptive aggregation. Its F1-score stays within 0.7 points of centralised training and it tolerates 20% Byzantine clients. Jeter et al. (2026) trained Transformer-based anomaly detectors across institutions without sharing raw data, using four smart-grid datasets and testing robustness to a GAN-based anomaly injector. Banerjee et al. (2026) developed a federated multistage IDS for detecting zero-day and known threats in smart-home ecosystems. These works address privacy and distributed training. The open-set decision itself, its physical plausibility and its calibration under drift remain open.

2.6 Summary of the literature

Table 1 compares the reviewed studies along four axes that are central to this work: classification of known attacks, detection of unknown (zero-day) attacks, use of physical consistency, and adaptation to concept drift. Four observations follow. First, most high-accuracy detectors are supervised and closed-set (Samin et al. 2026; Alamgir et al. 2026; Singh and Sailaja Kumari 2026; Tirulo et al. 2026). Second, unsupervised and generative detectors flag anomalies but cannot name a known attack type (Goyel et al. 2026; Zhao et al. 2026; Badakhshan and Zhang 2026). Third, physics-informed methods (Massaoudi and Ez Eddin 2026; Wu et al. 2026; Hu et al. 2026) have not yet been combined with open-set recognition. Fourth, drift-aware and continual methods (Yuan et al. 2026; Nasreen Fathima et al. 2026) are rarely evaluated on cyber-physical power-system data. PG-ADTA-ID is designed to provide all four capabilities in a single framework.

Table 1 Comparison of the 35 reviewed 2026 studies (K = known-attack classification, Z = unknown/zero-day detection, P = physics consistency, D = drift adaptation; ✓ = addressed, — = not addressed). Abbreviations: BO, Bayesian optimisation; DP, differential privacy; FA/FN, false alarms/false negatives; GAT, graph attention network; HE, homomorphic encryption; HIL, hardware-in-the-loop; KS, Kolmogorov–Smirnov; MC, Monte Carlo; MI, mutual information; PINN, physics-informed neural network; SE, state estimation; SOM, self-organising map

Ref.	Method	Data / test system	K	Z	P	D	Main limitation
Samin et al. (2026)	CNN–Transformer–BiGRU, MI + ADASYN	MSU power-system benchmark	✓	—	—	—	Closed-set; 9% FN, 7% FA
Alamgir et al. (2026)	CNN–BiLSTM + Extra-Trees	MSU smart-meter logs	✓	—	—	—	Closed-set; recall 81%
Sharma et al. (2026)	Optimised CNN–BiLSTM	AMI + SCADA, 6 synthetic attacks	✓	—	—	—	No unseen-attack test
Chandrakala et al. (2026)	Secure signal encoding + temporal features	Smart-grid IDS data	✓	—	—	—	Closed-set multi-class
Singh and Sailaja Kumari (2026)	LSTM/GRU + transfer learning	IEEE 14/24/39/118-bus PMU	✓	—	—	—	All events known a priori
Farsi et al. (2026)	Ensemble ML + SHAP	37 disturbance/attack scenarios	✓	—	—	—	Explanations, no novelty
Presekal et al. (2026)	CNN/KS similarity + SOM–DBSCAN	HIL digital substation	✓	✓	—	—	Protocol-specific retraining
Goyel et al. (2026)	Transformer reconstruction + KS fitting	GOOSE traffic datasets	—	✓	—	—	Frame features only
Mohammed et al. (2026)	IG-APSO feature selection + DNN	ICS power-system dataset	✓	—	—	—	FDIA focus, closed-set
Li et al. (2026)	Residual + attention + BiLSTM (MSFF-AEN)	FDIA measurement data	✓	—	—	—	Predefined attack patterns
Pan et al. (2026)	Transformer–CNN localisation	IEEE 14/39-bus	✓	—	—	—	Fixed topology
Qu et al. (2026)	Wavelet spatio-temporal graph embedding	Grid measurements	✓	—	—	—	Supervised localisation
Dzafic et al. (2026)	Graph Deviation Network	Smart-grid time series	—	✓	—	—	No attack-type output
Zhao et al. (2026)	Implicit neural representation	IEEE 14/118-bus	—	✓	—	—	Measurement attacks only
Alwaisi and Soderi (2026)	Semantic encoding + LSTM	6G smart-meter prototype	✓	—	✓	—	Single in-house dataset
Wang et al. (2026)	Cross-layer graph-Transformer	IEEE 118, ACTIVSg2000	✓	—	✓	—	Unseen attacks not studied
Aboelmagd et al. (2026)	Simplicial recurrent GNN	2,869–70,000-bus grids	✓	—	—	—	Synthetic telemetry attacks
Xi et al. (2026a)	Frequency attention + bidirectional ST-graph	IEEE 14/57/118-bus	✓	—	—	—	Supervised localiser
Tirulo et al. (2026)	Adaptive CNN + attention-BiLSTM + BO	290-household forecasts	✓	—	—	—	Labelled attacks required
Sheta et al. (2026)	Metaheuristic-tuned voting ensemble	ORNL power-system data	✓	—	—	—	Closed-set ensemble
Massaoudi and Ez Eddin (2026)	Constraint-aware PINN	Cyber-physical smart grid	✓	—	✓	—	Trained on labelled attacks
Wu et al. (2026)	Physics-informed SE + MC dropout	Standard test systems	—	✓	✓	—	No attack typing
Hu et al. (2026)	Physics-guided dynamic GAT	IEEE 13/123-bus feeders	✓	—	✓	—	Unseen attacks not reported
Balahewa et al. (2026)	Natural watermarking of IBRs	Microgrid, CHIL	—	✓	✓	—	Needs renewable variability
Rajalakshmi and Velmurugan (2026)	Physics-guided contrastive temporal graph	Industrial control systems	—	✓	✓	—	Not power-grid specific
Badakhshan and Zhang (2026)	WGAN-GP + LSTM	IEEE 118-bus IES (PSS/E)	—	✓	—	—	Held-out attacks unclear
Wilkie et al. (2026)	Contrastive loss, open-set extension	Lycos2017 flows	✓	✓	—	—	Enterprise IT traffic
Park and Lim (2026)	Hybrid multi-stage known + zero-day	Network traffic	✓	✓	—	—	No grid physics
Zheng et al. (2026)	Contrastive AE + adversarial training	Smart-grid substations	—	✓	—	—	No known-class output
Yuan et al. (2026)	Conformal anomaly detection	Industrial CPS	—	✓	—	✓	Generic CPS setting

Ref.	Method	Data / test system	K	Z	P	D	Main limitation
Nasreen Fathima et al. (2026)	Continual learning (CLEAR)	Intrusion detection data	—	✓	—	✓	No grid physics
Mohammadpourfard et al. (2026)	Generative covert-FDIA synthesis	IEEE 14/57-bus	—	—	✓	—	Attack generation only
Chen et al. (2026)	Federated CNN–BiLSTM + HE + DP	NSL-KDD, CICIDS2017, ICS	✓	—	—	—	Closed-set
Jeter et al. (2026)	Federated Transformer anomaly detection	Four smart-grid datasets	—	✓	—	—	Drift across clients open
Banerjee et al. (2026)	Federated multistage IDS	Smart-home IoT	✓	✓	—	—	No physics
This work	**PGST + SA-DAE + PC-OAF + CDCL**	**IEEE 14-bus co-simulation**	**✓**	**✓**	**✓**	**✓**	—

3 Research Gaps and Objectives

The review in Section 2 points to a small number of recurring limitations. This section states them as five research gaps (Section 3.1) and derives the three objectives that PG-ADTA-ID is designed to meet (Section 3.2).

3.1 Research gaps

Machine-learning-based intrusion detection has progressed clearly, but several gaps remain before next-generation smart grids can be protected against both known and unknown cyberattacks.

G1 – Closed-world detection. Conventional and hybrid supervised detectors (SVM, RF, LSTM, CNN–BiLSTM and CNN–Transformer–BiGRU) depend on representative labelled training data (Samin et al. 2026; Alamgir et al. 2026; Singh and Sailaja Kumari 2026). They have no principled mechanism for rejecting observations that belong to none of the training classes, so zero-day and evolving attacks are silently misclassified.

G2 – Incomplete use of cyber-physical structure. Most detectors process network traffic or electrical measurements in isolation and do not jointly capture temporal dependencies, grid topology and the coupling between communication events and physical quantities (Wang et al. 2026; Cibin et al. 2026a). Physical laws, which hold at every operating point, are rarely turned into explicit evidence for the open-set decision (Massaoudi and Ez Eddin 2026; Wu et al. 2026).

G3 – Unknown-attack scoring that is unaware of operating conditions. Reconstruction- and density-based anomaly detectors flag any departure from the training distribution, including legitimate load growth, renewable variability and authorised topology changes (Goyel et al. 2026; Badakhshan and Zhang 2026). This raises false alarms and erodes operator trust.

G4 – Unsafe adaptation under concept drift. Thresholds and models must follow the drifting operating regime without being poisoned by the attacks they are meant to detect (Yuan et al. 2026; Liu et al. 2026). Few smart-grid IDSs specify which observations may be trusted for adaptation, or how to retain knowledge of earlier attacks.

G5 – Evaluation protocols. Many studies report accuracy on random splits of a single dataset. They seldom hold out entire attack families, include benign disturbances, separate post-attack transition windows or report event-level detection delay, so their reported zero-day capability is hard to judge (Singh and Sailaja Kumari 2026; Badakhshan and Zhang 2026).

3.2 Research objectives

To close these gaps, this work pursues three objectives.

1. To develop an adaptive hybrid deep-learning model that detects and classifies known cyberattacks in next-generation smart grids by learning physics-guided spatiotemporal representations of fused cyber-physical data (addresses G1 and G2).
2. To design an intelligent anomaly-detection and open-set fusion mechanism that identifies previously unseen (zero-day) cyberattacks while remaining robust to legitimate operating changes and concept drift (addresses G1, G3 and G4).
3. To optimise the proposed model and evaluate it rigorously in terms of detection accuracy, precision, recall, F1-score, false-positive rate, zero-day detection rate, detection delay and computational efficiency, under a protocol that withholds attack families and includes a drift period (addresses G5).

4 Existing System: Hybrid CNN–Transformer–BiGRU Intrusion Detection

Fig. 1 shows the existing system, the hybrid CNN–Transformer–BiGRU architecture of Samin et al. (2026) for cyberattack detection in smart-grid SCADA networks. First, smart-grid measurements and network data are normalised. Mutual information (MI) then selects the most informative features, and adaptive synthetic minority oversampling (ADASYN) compensates for class imbalance. A convolutional neural network (CNN) extracts local features, a Transformer encoder captures contextual dependencies through multi-head attention, and a BiGRU learns temporal attack patterns in both directions. Finally, a softmax layer assigns each observation to normal operation or to one of the attack categories in the training set, such as DoS, FDIA, replay and malware. The system is effective for the classes it was trained on. However, it cannot recognise previously unseen attacks and it does not adapt to changing grid operating conditions, which motivates the framework proposed in Section 5.

4.1 Mathematical model of the existing system

The six processing stages of the existing framework are formulated as follows. These equations are a representative formulation of the architecture, not equations reproduced verbatim from Samin et al. (2026).

4.1.1 Mutual-information feature selection

MI measures the statistical dependence between a smart-grid feature X (e.g., voltage, current, frequency or a communication statistic) and the attack label Y, as defined in Eq. (1):

$$I(X;Y) = \sum_{x\in X}\sum_{y\in Y} p\,(x,y)\log\frac{p(x,y)}{p(x)\,p(y)}, \tag{1}$$

where $p(x,y)$ is the joint probability distribution and $p(x)$ and $p(y)$ are the marginal distributions. Features with the highest MI are retained, which reduces redundancy and computational load.

4.1.2 ADASYN oversampling

For each minority observation i, ADASYN allocates a number of synthetic samples proportional to its local learning difficulty, as given by Eq. (2):

$$G_i = G\,\frac{r_i}{\sum_{j=1}^{n_m} r_j}, \qquad r_i = \frac{\Delta_i}{K}, \tag{2}$$

where G is the total number of samples to be generated, Δ_i is the number of majority-class samples among the K nearest neighbours of observation i, and n_m is the number of minority observations.

4.1.3 CNN feature extraction

A one-dimensional convolution with filter f and kernel size k produces the feature map of Eq. (3):

$$H_t^{(f)} = \text{Swish}\left(\sum_{j=0}^{k-1}(\boldsymbol{W}_j^{(f)})^{\top}\boldsymbol{x}_{t+j} + b_f\right), \tag{3}$$

where $\boldsymbol{W}_j^{(f)}$ are the filter weights, $\boldsymbol{x}_{t+j}$ is the input feature vector at position $t+j$, b_f is the bias and $\text{Swish}(x) = x\,\sigma(x)$.

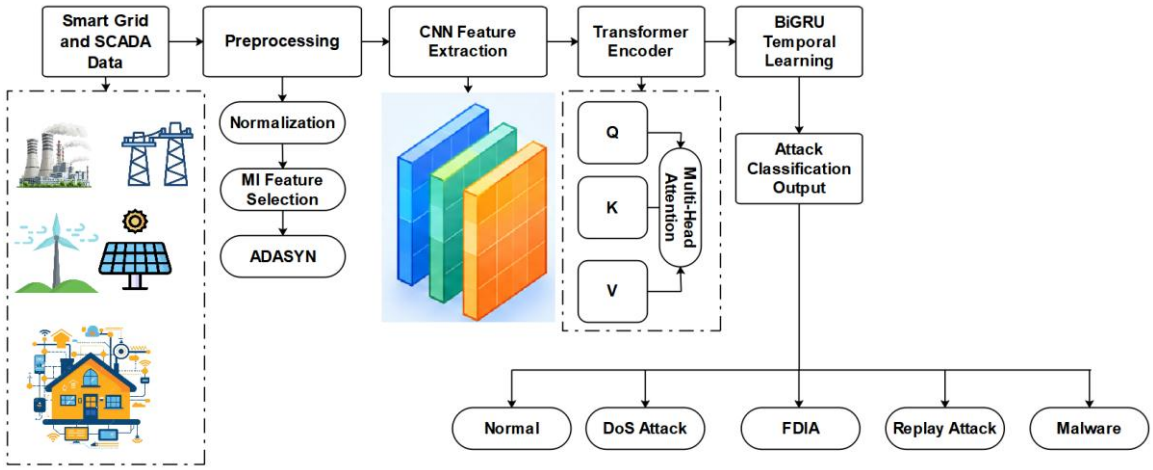

Fig. 1 Block diagram of the existing hybrid CNN–Transformer–BiGRU intrusion detection framework for smart-grid cybersecurity (Samin et al. 2026)

4.1.4 Transformer attention

Scaled dot-product attention, Eq. (4), relates all positions of the sequence to each other:

$$\text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\top}}{\sqrt{d_k}}\right)\boldsymbol{V}, \tag{4}$$

where $\boldsymbol{Q}$, $\boldsymbol{K}$ and $\boldsymbol{V}$ are the query, key and value matrices and d_k is the key dimension.

4.1.5 BiGRU temporal learning

The Transformer outputs $\boldsymbol{z}_t$ are processed in both temporal directions according to Eq. (5):

$$\boldsymbol{h}_t = \left[\text{GRU}_{\text{fw}}\left(\boldsymbol{z}_t, \boldsymbol{h}_{t-1}^{\text{fw}}\right);\ \text{GRU}_{\text{bw}}\left(\boldsymbol{z}_t, \boldsymbol{h}_{t+1}^{\text{bw}}\right)\right], \tag{5}$$

where fw and bw denote the forward and backward GRUs and the semicolon denotes concatenation of their hidden states.

4.1.6 Softmax classification

The class with the largest posterior probability is selected by Eq. (6):

$$\hat{y} = \arg\max_{c\in\{1,\dots,C\}} \frac{\exp(\boldsymbol{w}_c^\top \boldsymbol{h} + b_c)}{\sum_{l=1}^{C} \exp(\boldsymbol{w}_l^\top \boldsymbol{h} + b_l)}, \tag{6}$$

where $\boldsymbol{w}_c$ and b_c are the weight vector and bias of class c, and C is the number of predefined classes. Because Eq. (6) always returns one of the C training classes, the existing system cannot flag observations from unseen attack families. This limitation motivates the open-set formulation of PG-ADTA-ID.

5 Proposed PG-ADTA-ID Framework

This section presents the proposed framework. Section 5.1 formulates the open-set detection problem, Sections 5.2–5.5 describe the four processing stages and their algorithms, Section 5.6 describes the continual-learning mechanism and Section 5.7 the evaluation and optimisation stage. Section 5.8 gives the complete pseudocode of the framework, and Section 5.9 analyses its theoretical properties and computational cost.

5.1 Overview and problem formulation

Fig. 2 summarises the methodology of the proposed Physics-Guided Adaptive Dual-Branch Transformer–Autoencoder Intrusion Detection (PG-ADTA-ID) framework, and Fig. 3 shows its architecture. Stage 1 collects and synchronises smart-grid measurements, SCADA telemetry, AMI readings and network traffic, preprocesses them and fuses them into a cyber-physical feature matrix with explicit physics residuals. In Stage 2, a Physics-Guided Spatiotemporal Transformer (PGST) learns temporal and topology-dependent representations and classifies known attacks. In Stage 3, a Self-supervised Adaptive Denoising Autoencoder (SA-DAE) reconstructs the inputs and produces a group-wise reconstruction-error signature. Stage 4 is the Physics-Consistent Open-set Adaptive Fusion (PC-OAF), which combines classification uncertainty, reconstruction surprise and physics-consistency surprise into a novelty-risk score and decides between normal operation, a known attack, a suspected zero-day attack and an uncertain observation. Stage 5 evaluates the framework and optimises its hyperparameters and decision parameters. During operation, Concept-Drift-aware Continual Learning (CDCL) keeps Stages 2–4 calibrated as operating conditions change; in Fig. 3 it appears as the feedback block whose dashed arrows update the PGST, the SA-DAE and the decision threshold.

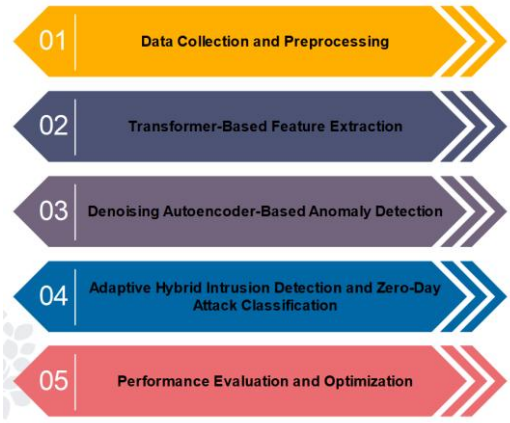

Fig. 2 Methodology of the proposed PG-ADTA-ID framework (Stages 1–5) for known and zero-day cyberattack detection in next-generation smart grids

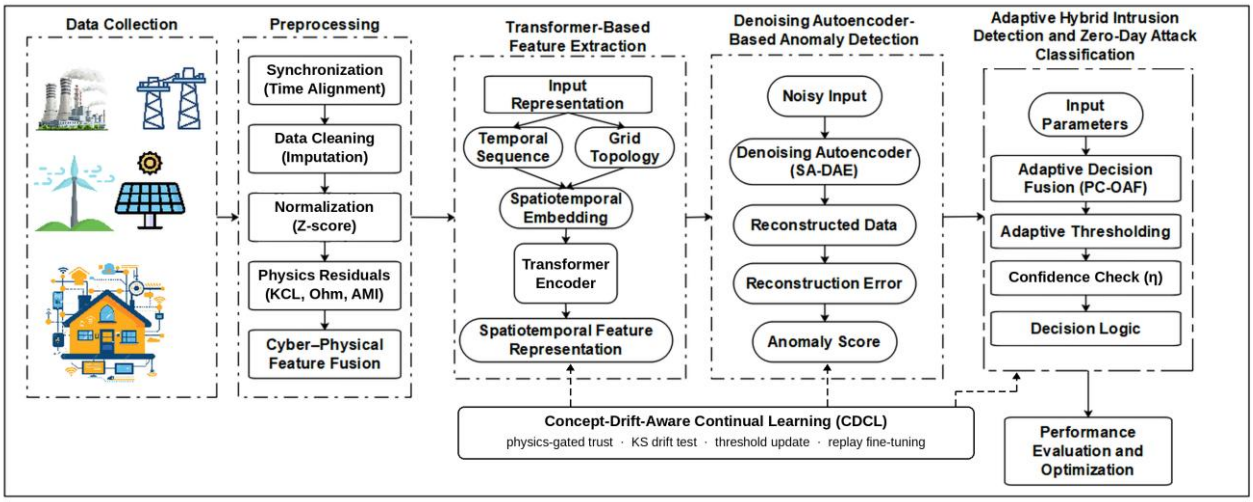

Fig. 3 Architecture of the proposed PG-ADTA-ID framework. The CDCL block (bottom) updates the PGST, the SA-DAE and the decision threshold during operation (dashed arrows)

Let $\boldsymbol{x}_t \in \mathbb{R}^d$ be the fused cyber-physical feature vector at time step t and let $\boldsymbol{X}_t = [\boldsymbol{x}_{t-T+1}, \dots, \boldsymbol{x}_t]^\top \in \mathbb{R}^{T\times d}$ be the sliding window that ends at t. The training data contain windows labelled with a set of *known* classes $\mathcal{K} = \{0,1,\dots,C-1\}$, where class 0 is normal operation, including benign disturbances. At deployment, windows may also come from *unknown* attack families $\mathcal{U}$ with $\mathcal{U} \cap \mathcal{K} = \emptyset$. Unlike a closed-set classifier, PG-ADTA-ID maps every window to a decision $\hat{y}_t \in \mathcal{K} \cup \{\text{ZD,UNC}\}$, where ZD denotes a suspected zero-day attack and UNC an uncertain observation referred to the operator. The framework must classify known attacks accurately, flag unknown attacks and keep false alarms rare. A user-specified budget δ sets the fraction of known-class calibration windows that may exceed the novelty threshold. The design goal is stated formally in Definition 1.

Definition 1 (Open-set detection with abstention) An open-set detector is a map $f: \boldsymbol{X}_t \mapsto \hat{y}_t \in \mathcal{K} \cup \{\text{ZD,UNC}\}$. For a budget $\delta \in (0,1)$, the design goal of PG-ADTA-ID is the constrained problem of Eq. (7),

$$\begin{aligned} &\max_f \Pr(f(\boldsymbol{X}_t) \in \{\text{ZD,UNC}\} \mid y_t \in \mathcal{U}) \\ &\text{subject to} \quad \Pr(f(\boldsymbol{X}_t) = \text{ZD} \mid y_t \in \mathcal{K}) \le \delta, \end{aligned} \tag{7}$$

together with accurate classification of the known classes, $\hat{y}_t = y_t$ for $y_t \in \mathcal{K}$.

PG-ADTA-ID enforces the constraint through a calibrated threshold on a novelty-risk score (Proposition 5 in Section 5.9.2) and pursues the objective through the design of that score and the optimisation of Stage 5.

5.2 Stage 1: Cyber-physical data collection and adaptive preprocessing (CP-DAP)

Fig. 4 illustrates the data collection and preprocessing stage. Cyber-physical information is collected from four sources: grid measurements from PMUs (voltage phasors and frequency), SCADA/RTU telemetry (bus injections, branch flows at both ends, generator output and breaker status), AMI (aggregated net demand at each load bus) and network-traffic logs (22 flow, protocol and timing statistics). As shown in Fig. 4, the sources are synchronised and time-aligned, cleaned (lost or corrupted values are imputed), normalised, and extended with physics residuals derived from the synchronised measurements, and all blocks are then fused into a single feature matrix (Sections 5.2.1–5.2.4).

5.2.1 Multisource synchronisation and cleaning

Sources reporting at different rates are aligned to the PMU time base by sample-and-hold resampling $\mathcal{I}_m$. A cleaning operator $\mathcal{C}_m$ then imputes lost packets by forward filling and records which entries were imputed, as expressed by Eqs. (8) and (9):

$$\tilde{\boldsymbol{x}}_t^{(m)} = \mathcal{C}_m\big(\mathcal{I}_m(D_m, t)\big), \tag{8}$$

$$\phi_t = 1 - \frac{1}{|\mathcal{P}|}\sum_{j\in\mathcal{P}} \mathbb{I}\left[x_{t,j} \text{ imputed}\right], \tag{9}$$

where D_m are the time-stamped observations of source m (grid, SCADA, AMI or network), $\mathcal{P}$ is the set of physical channels and $\phi_t \in [0,1]$ is the *data freshness*. Freshness falls when packets are dropped, for example during a flooding attack, and is used later to weight the physics evidence.

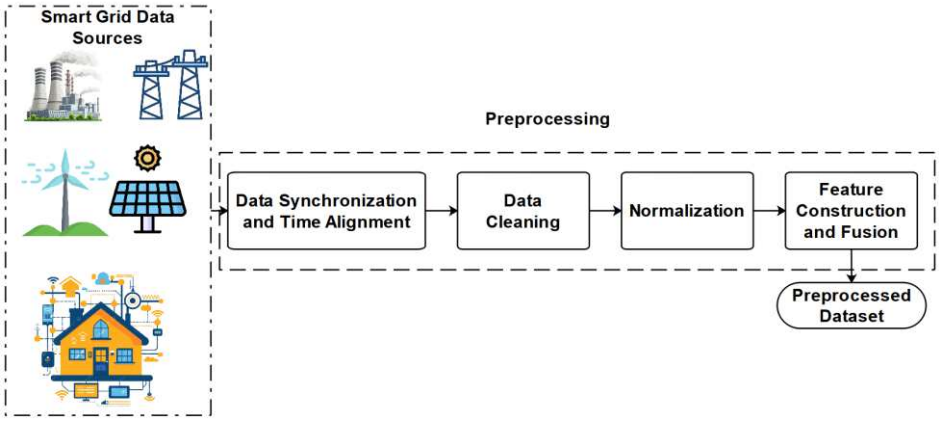

Fig. 4 Data collection and preprocessing stage (CP-DAP); feature construction adds the physics residuals before fusion

5.2.2 Physics residuals

Three families of residuals that hold at every operating point are computed from the synchronised measurements. Let $\tilde{S}_{t,i} = \tilde{P}_{t,i} + j\tilde{Q}_{t,i}$ be the measured complex injection at bus i, $\tilde{S}^{\mathrm{f}}_{t,k}$ and $\tilde{S}^{\mathrm{t}}_{t,k}$ the measured flows at the from and to ends of branch k, $\tilde{V}_{t,i} = |\tilde{V}_{t,i}|e^{j\tilde{\theta}_{t,i}}$ the PMU voltage phasor, $s_{t,k} \in \{0,1\}$ the breaker status, and y^{ff}_k, y^{ft}_k and y^{sh}_i the known branch and shunt admittances. The nodal Kirchhoff mismatch, the branch Ohm's-law mismatch between PMU phasors and SCADA flows, and the AMI–SCADA demand balance are given by Eqs. (10), (11) and (12), respectively:

$$r^{\mathrm{KCL}}_{t,i} = \tilde{S}_{t,i} - \sum_{k:\, f(k)=i} \tilde{S}^{\mathrm{f}}_{t,k} - \sum_{k:\, t(k)=i} \tilde{S}^{\mathrm{t}}_{t,k} - |\tilde{V}_{t,i}|^2\, \overline{y^{\mathrm{sh}}_i}, \tag{10}$$

$$r_{t,k}^{\mathrm{Ohm}} = \tilde{S}_{t,k}^{\mathrm{f}} - s_{t,k}\,\tilde{V}_{t,f(k)}\,\overline{\left(y_k^{\mathrm{ff}}\tilde{V}_{t,f(k)} + y_k^{\mathrm{ft}}\tilde{V}_{t,t(k)}\right)}, \tag{11}$$

$$r_{t,i}^{\mathrm{AMI}} = \tilde{P}_{t,i} - \left(P_{t,i}^{\mathrm{g}} - P_{t,i}^{\mathrm{AMI}}\right), \tag{12}$$

where $f(k)$ and $t(k)$ are the terminal buses of branch k, $P_{t,i}^{\mathrm{g}}$ is the generator telemetry at bus i and $P_{t,i}^{\mathrm{AMI}}$ is the aggregated metered net demand. The real and imaginary parts of the residuals are standardised robustly using the median and the median absolute deviation (MAD) of verified normal training data, $\zeta_{t,j} = \left(r_{t,j} - \mathrm{med}_j\right)/\left(1.4826\,\mathrm{MAD}_j\right)$. Each residual family $\mathcal{G} \in \{\mathrm{KCL},\mathrm{Ohm},\mathrm{AMI}\}$ is then summarised by a mean (diffuse) and a maximum (localised) inconsistency, Eqs. (13) and (14):

$$q_t^{\mathcal{G},\mathrm{mean}} = \log\left(1 + \frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}}\zeta_{t,j}^2\right), \tag{13}$$

$$q_t^{\mathcal{G},\mathrm{max}} = \log\left(1 + \max_{j\in\mathcal{G}}\zeta_{t,j}^2\right). \tag{14}$$

The six summaries form the physics-consistency profile $\boldsymbol{q}_t \in \mathbb{R}^6$. Stealthy, AC-consistent manipulations leave r^{KCL} and r^{Ohm} small but disturb r^{AMI}, because the independent metering path is not under the attacker's control. Spoofing of PMU time references disturbs r^{Ohm} alone.

5.2.3 Adaptive normalisation

All 22 traffic features are first compressed with $\log(1+x)$. Every channel except the breaker statuses, which enter as binary open indicators, is then standardised with training-set statistics of verified normal operation and clipped according to Eq. (15):

$$z_{t,j} = \mathrm{clip}\left(\frac{\tilde{x}_{t,j} - \mu_j}{\sigma_j + \epsilon},\, -10,\, 10\right), \tag{15}$$

where μ_j and σ_j are the training mean and standard deviation and ϵ prevents division by zero. Physics residuals are compressed with the signed logarithm $\mathrm{sign}(\zeta)\log(1+|\zeta|)$.

5.2.4 Cyber-physical feature fusion

The normalised blocks are concatenated into the fused feature matrix of Eq. (16):

$$\boldsymbol{X}_f = \left[\boldsymbol{Z}_g \,\|\, \boldsymbol{Z}_a \,\|\, \boldsymbol{Z}_b \,\|\, \boldsymbol{Z}_n \,\|\, \boldsymbol{\phi} \,\|\, \psi(\boldsymbol{R}) \,\|\, \boldsymbol{Q}\right] \in \mathbb{R}^{N\times d}, \tag{16}$$

where $\boldsymbol{Z}_g$, $\boldsymbol{Z}_a$, $\boldsymbol{Z}_b$ and $\boldsymbol{Z}_n$ are the grid, AMI, breaker-status and network blocks, $\boldsymbol{\phi}$ is the freshness channel, $\psi(\boldsymbol{R})$ are the compressed residuals of Eqs. (10)–(12), $\boldsymbol{Q}$ are the standardised physics summaries of Eqs. (13)–(14) and $\|$ denotes feature-wise concatenation. For the IEEE 14-bus system, $d = 280$. Algorithm 1 summarises the stage.

Algorithm 1 Cyber-physical data collection and adaptive preprocessing (CP-DAP)

Require: grid/PMU data D_g, SCADA data D_s, AMI data D_a, network traffic D_n; branch admittances; training-day index set $\mathcal{T}_{\mathrm{tr}}$
Ensure: fused matrix $\boldsymbol{X}_f \in \mathbb{R}^{N\times d}$, physics profile $\boldsymbol{Q} \in \mathbb{R}^{N\times 6}$, freshness $\boldsymbol{\phi}$

1: Align all sources to the PMU time base by sample-and-hold resampling $\mathcal{I}_m$
2: Forward-fill lost records; set the imputation flags and ϕ_t ▷ Eqs. (8)–(9)
3: **for** each time step t **do**
4: Compute the residuals $r_{t,i}^{\mathrm{KCL}}$, $r_{t,k}^{\mathrm{Ohm}}$ and $r_{t,i}^{\mathrm{AMI}}$ ▷ Eqs. (10)–(12)
5: **end for**
6: Standardise each residual to $\zeta_{t,j}$ with its median and MAD on verified normal steps of $\mathcal{T}_{\mathrm{tr}}$
7: Build the physics profile $\boldsymbol{q}_t$ from the mean and maximum summaries ▷ Eqs. (13)–(14)
8: Apply $\log(1+x)$ to the traffic counts; standardise and clip all channels ▷ Eq. (15)
9: Concatenate all blocks into the fused matrix $\boldsymbol{X}_f$ ▷ Eq. (16)
10: **return** $\boldsymbol{X}_f$, $\boldsymbol{Q}$, $\boldsymbol{\phi}$

5.3 Stage 2: Physics-guided spatiotemporal Transformer (PGST)

Fig. 5 shows the PGST module, which extracts discriminative spatiotemporal features and classifies known attacks. Each time step of the window becomes a token. Its embedding combines a linear projection of the fused features, a sinusoidal positional encoding and a graph embedding that injects the electrical topology of the grid.

5.3.1 Topology-aware graph embedding

For each bus i and time step τ, a node vector $\boldsymbol{v}_{\tau,i} \in \mathbb{R}^8$ collects the standardised voltage magnitude and angle, the active and reactive injections, the two KCL residual components, the AMI demand and the AMI residual. The electrical adjacency is weighted by branch susceptance and masked by the current breaker status, so that switching operations change the graph, as expressed by Eqs. (17) and (18):

$$\boldsymbol{A}_t = \sum_k s_{t,k}\,\left|y_k^{\mathrm{ft}}\right|\left(\boldsymbol{e}_{f(k)}\boldsymbol{e}_{t(k)}^{\mathsf{T}} + \boldsymbol{e}_{t(k)}\boldsymbol{e}_{f(k)}^{\mathsf{T}}\right), \tag{17}$$

$$\widehat{\boldsymbol{A}}_t = \boldsymbol{D}_t^{-1/2}(\boldsymbol{A}_t + \boldsymbol{I})\boldsymbol{D}_t^{-1/2}, \tag{18}$$

where $\boldsymbol{e}_i$ is the i-th unit vector and $\boldsymbol{D}_t$ is the degree matrix of $\boldsymbol{A}_t + \boldsymbol{I}$. A two-layer graph convolution followed by mean–max read-out yields the graph embedding of each token through Eqs. (19), (20) and (21):

$$\boldsymbol{H}_\tau^{(1)} = \mathrm{GELU}\big(\widehat{\boldsymbol{A}}_t \boldsymbol{V}_\tau \boldsymbol{W}_1\big), \tag{19}$$

$$\boldsymbol{H}_\tau^{(2)} = \mathrm{GELU}\big(\widehat{\boldsymbol{A}}_t \boldsymbol{H}_\tau^{(1)} \boldsymbol{W}_2\big), \tag{20}$$

$$\boldsymbol{E}_\tau^{\mathrm{graph}} = \boldsymbol{W}_g\big[\mathrm{mean}_i\, \boldsymbol{H}_\tau^{(2)} \,\|\, \max_i \boldsymbol{H}_\tau^{(2)}\big]. \tag{21}$$

5.3.2 Spatiotemporal embedding

The initial token representation of the window is given by Eq. (22):

$$\boldsymbol{H}_t^{(0)} = \boldsymbol{X}_{t-T+1:t}\boldsymbol{W}_x + \boldsymbol{E}^{\mathrm{time}} + \boldsymbol{E}_{t-T+1:t}^{\mathrm{graph}}, \tag{22}$$

where $\boldsymbol{W}_x \in \mathbb{R}^{d \times d_m}$ is a learnable projection, $\boldsymbol{E}^{\mathrm{time}}$ is the sinusoidal positional encoding and d_m is the model width.

5.3.3 Multi-head self-attention and encoder

Each of the h attention heads relates every pair of tokens, and the heads are concatenated, as in Eqs. (23) and (24):

$$\mathrm{head}_i = \mathrm{softmax}\left(\frac{\boldsymbol{Q}_i \boldsymbol{K}_i^{\top}}{\sqrt{d_k}}\right)\boldsymbol{V}_i, \tag{23}$$

$$\mathrm{MHSA}(\boldsymbol{H}) = \mathrm{Concat}(\mathrm{head}_1, \ldots, \mathrm{head}_h)\,\boldsymbol{W}_O, \tag{24}$$

with $\boldsymbol{Q}_i = \boldsymbol{H}\boldsymbol{W}_i^Q$, $\boldsymbol{K}_i = \boldsymbol{H}\boldsymbol{W}_i^K$ and $\boldsymbol{V}_i = \boldsymbol{H}\boldsymbol{W}_i^V$. Each of the L_e encoder layers applies residual connections, layer normalisation (LN) and a position-wise feed-forward network (FFN), as in Eqs. (25) and (26):

$$\widetilde{\boldsymbol{H}}^{(l)} = \mathrm{LN}\Big(\boldsymbol{H}^{(l-1)} + \mathrm{MHSA}\big(\boldsymbol{H}^{(l-1)}\big)\Big), \tag{25}$$

$$\boldsymbol{H}^{(l)} = \mathrm{LN}\Big(\widetilde{\boldsymbol{H}}^{(l)} + \mathrm{FFN}\big(\widetilde{\boldsymbol{H}}^{(l)}\big)\Big). \tag{26}$$

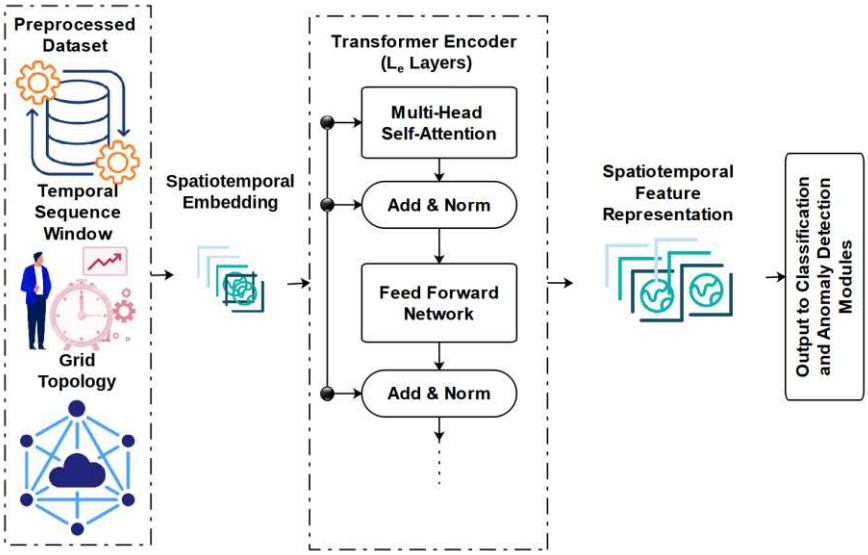

Fig. 5 Physics-guided spatiotemporal Transformer (PGST): spatiotemporal embedding of the preprocessed window and the grid topology, followed by L_e encoder layers

5.3.4 Pooling and known-attack classification

The window representation of Eq. (27) concatenates the last token, which carries the most recent evidence, with the mean token, which carries context, and Eq. (28) gives the temperature-calibrated class probabilities:

$$\boldsymbol{z}_t = \mathrm{LN}\left(\left[\boldsymbol{h}_t^{(L_e)} \,\|\, \frac{1}{T}\sum_{\tau} \boldsymbol{h}_\tau^{(L_e)}\right]\right), \tag{27}$$

$$p_{t,c} = \frac{\exp(g_c(\boldsymbol{z}_t)/T_c)}{\sum_{j \in \mathcal{K}} \exp\big(g_j(\boldsymbol{z}_t)/T_c\big)}, \tag{28}$$

where $g(\cdot)$ is a two-layer perceptron head and T_c is a temperature fitted by minimising the negative log-likelihood on the calibration split. PGST is trained with the class-weighted, label-smoothed cross-entropy loss of Eq. (29):

$$\mathcal{L}_{\text{cls}} = -\frac{1}{B}\sum_{b=1}^{B}\sum_{c\in\mathcal{K}} w_c \left[(1-\varepsilon)\,\mathbb{I}[y_b = c] + \frac{\varepsilon}{C}\right] \log p_{b,c}, \tag{29}$$

where $w_c = \left(N/(C\,N_c)\right)^{1/2}$ is the weight of class c, N_c is the number of training windows of that class and $\varepsilon = 0.05$ is the label-smoothing factor. Algorithm 2 gives the procedure.

Algorithm 2 Physics-guided spatiotemporal Transformer (PGST): training and inference

Require: training windows $\{(\boldsymbol{X}_b, \boldsymbol{V}_b, \boldsymbol{s}_b, y_b)\}$ with $y_b \in \mathcal{K}$; validation windows; epochs E; patience P
Ensure: trained PGST parameters Θ_c; calibration temperature T_c
Initialise Θ_c and compute the class weights w_c ▷ Eq. (29)
for epoch $e = 1, \ldots, E$ **do**
  for each mini-batch **do**
    Build $\widehat{\boldsymbol{A}}_t$ from the breaker status ▷ Eqs. (17)–(18)
    Compute the graph embedding $\boldsymbol{E}^{\text{graph}}$ ▷ Eqs. (19)–(21)
    Form the token embeddings $\boldsymbol{H}^{(0)}$ ▷ Eq. (22)
    Apply the L_e encoder layers ▷ Eqs. (23)–(26)
    Pool to $\boldsymbol{z}_t$; compute the logits and $\mathcal{L}_{\text{cls}}$ ▷ Eqs. (27)–(29)
    Update Θ_c with AdamW, gradient clipping and a cosine learning-rate schedule
  end for
  Evaluate the validation macro-F1-score and keep the best parameters
  if there has been no improvement for P epochs **then**
    break
  end if
end for
Fit T_c on the calibration split by minimising the negative log-likelihood
return Θ_c and T_c; at inference, output $\boldsymbol{p}_t$ and $\boldsymbol{z}_t$ for each window

5.4 Stage 3: Self-supervised adaptive denoising autoencoder (SA-DAE)

Fig. 6 shows the SA-DAE branch: each window is corrupted by noise injection and masking, encoded, decoded and compared with the clean input. The branch models the *known world*, that is, verified normal operation together with the catalogued attacks. An autoencoder trained only on normal data reconstructs every attack poorly, known or unknown, so its raw error cannot separate the two. Trained on the known world, a large error signals *unfamiliar* rather than merely *abnormal* behaviour, and PC-OAF further compares the error signature with the statistics of the predicted class (Section 5.5.2). Training is self-supervised and needs no attack labels.

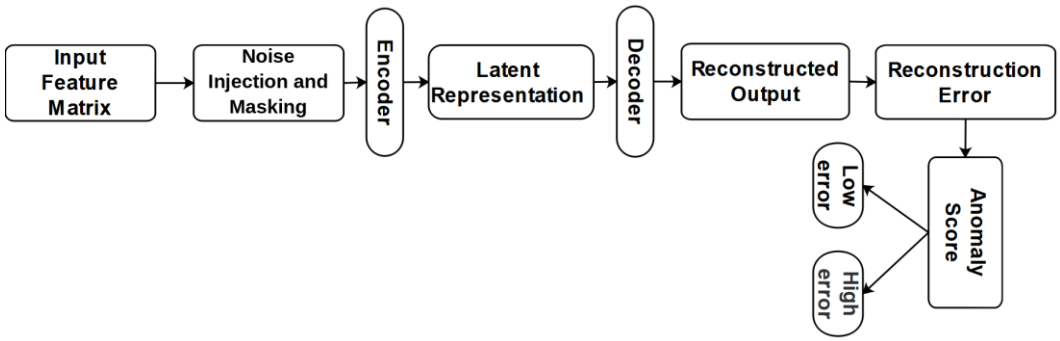

Fig. 6 Self-supervised adaptive denoising autoencoder (SA-DAE); the group-wise reconstruction error is the anomaly evidence passed to PC-OAF

5.4.1 Adaptive corruption

During training, each window is corrupted with Gaussian noise whose level follows an adaptive curriculum and with random feature masking, as in Eqs. (30) and (31):

$$\widetilde{\boldsymbol{X}}_t = (\boldsymbol{X}_t + \sigma_e\,\boldsymbol{\varepsilon}_t) \odot (\boldsymbol{1} - \boldsymbol{M}_t), \tag{30}$$

$$\sigma_e = \sigma_{\min} + (\sigma_{\max} - \sigma_{\min})\min\left(1, \frac{e}{E_w}\right), \tag{31}$$

where $\boldsymbol{\varepsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, the entries of the binary mask $\boldsymbol{M}_t$ are 1 with probability ρ_m (the masking ratio), e is the epoch index and E_w is the warm-up length. The curriculum lets the network first learn coarse structure and then progressively harder denoising.

5.4.2 Encoder–decoder reconstruction

A temporal convolutional encoder E_θ compresses the corrupted window into a latent code, and a transposed-convolution decoder D_ϕ reconstructs the clean window, Eq. (32):

$$\boldsymbol{c}_t = E_\theta(\widetilde{\boldsymbol{X}}_t), \qquad \widehat{\boldsymbol{X}}_t = D_\phi(\boldsymbol{c}_t). \tag{32}$$

5.4.3 Training objective

The loss of Eq. (33) combines full reconstruction with a masked-prediction term that forces the model to infer hidden channels from their cyber-physical context:

$$\mathcal{L}_{\mathrm{DAE}} = \frac{1}{Td} \| \boldsymbol{X}_t - \widehat{\boldsymbol{X}}_t \|_F^2 + \lambda_m \frac{\| \boldsymbol{M}_t \odot \left(\boldsymbol{X}_t - \widehat{\boldsymbol{X}}_t \right) \|_F^2}{\| \boldsymbol{M}_t \|_1}. \tag{33}$$

5.4.4 Error signature on operating-point-independent channels

Raw voltages, injections and flows vary legitimately with load, renewable output and topology. To make the novelty score less sensitive to such changes, it is computed only on channels that do not scale directly with the operating point: breaker context, four groups of traffic statistics (volume, connection, command and quality), freshness, and the KCL, Ohm, AMI and summary residual groups. For each group $\mathcal{G}_g$ the log mean-squared error is computed with Eq. (34):

$$e_{t,g} = \log\left(\frac{1}{T|\mathcal{G}_g|} \sum_{\tau=t-T+1}^{t} \sum_{j \in \mathcal{G}_g} (x_{\tau,j} - \hat{x}_{\tau,j})^2 \right), \tag{34}$$

with $G = 10$. The vector $\boldsymbol{r}_t = \left[e_{t,1}, \dots, e_{t,G} \right]^\top \in \mathbb{R}^G$ is the reconstruction-error signature of the window. Algorithm 3 summarises SA-DAE.

Algorithm 3 Self-supervised adaptive denoising autoencoder (SA-DAE)

Require: verified known-world training windows $\{\boldsymbol{X}_b\}$ (no labels); validation windows; $\sigma_{\min}$, $\sigma_{\max}$, E_w, ρ_m, λ_m
Ensure: encoder–decoder parameters (θ, ϕ); error-signature function $\boldsymbol{r}(\cdot)$
1: Initialise (θ, ϕ)
2: **for** epoch $e = 1, \dots, E$ **do**
3: Set the curriculum noise level σ_e ▷ Eq. (31)
4: **for** each mini-batch **do**
5: Sample $\boldsymbol{\varepsilon}$ and $\boldsymbol{M}$; corrupt $\boldsymbol{X} \to \widetilde{\boldsymbol{X}}$ ▷ Eq. (30)
6: Encode, decode, compute $\mathcal{L}_{\mathrm{DAE}}$ and update (θ, ϕ) ▷ Eqs. (32)–(33)
7: **end for**
8: Keep the parameters with the lowest clean validation reconstruction error
9: **end for**
10: At inference, compute the group error signature $\boldsymbol{r}_t$ ▷ Eq. (34)
11: **return** (θ, ϕ) and $\boldsymbol{r}(\cdot)$

5.5 Stage 4: Physics-consistent open-set adaptive fusion (PC-OAF)

Fig. 7 shows the PC-OAF mechanism. It fuses three complementary indicators, the classification uncertainty, the reconstruction anomaly score (reconstruction surprise) and the physics-consistency score, into a single novelty-risk score, and its adaptive decision mechanism then assigns normal operation, a known attack, a suspected zero-day attack or an uncertain observation.

5.5.1 Classification uncertainty

Softmax confidence reflects aleatoric ambiguity between known classes but is known to be over-confident far from the training data. PC-OAF therefore complements it with an epistemic term, the Mahalanobis distance of the PGST embedding from the Gaussian of the predicted class, which gives the uncertainty indicator of Eqs. (35) and (36):

$$u_t = \max\{1 - \max_c p_{t,c},\ 1 - \exp\left(-D_{z,t}^2 / \kappa_z\right)\}, \tag{35}$$

$$D_{z,t}^2 = (\boldsymbol{z}_t - \boldsymbol{\mu}_{\hat{c}}^z)^\top \boldsymbol{\Sigma}_z^{-1} (\boldsymbol{z}_t - \boldsymbol{\mu}_{\hat{c}}^z), \tag{36}$$

where $\hat{c} = \mathrm{argmax}_c p_{t,c}$ and $\boldsymbol{\mu}_c^z$ and $\boldsymbol{\Sigma}_z$ are the class means and the shared covariance of correctly classified calibration windows.

5.5.2 Class-conditional reconstruction surprise

Different known classes have different typical reconstruction signatures; a DoS window, for example, naturally reconstructs worse than a normal one. The SA-DAE signature is therefore compared with the signature distribution of the *predicted* class through Eqs. (37) and (38):

$$a_t = 1 - \exp\left(-D_{r,t}^2 / \kappa_r\right), \tag{37}$$

$$D_{r,t}^2 = (\boldsymbol{r}_t - \boldsymbol{\mu}_{\hat{c}}^r)^\top (\widehat{\boldsymbol{\Sigma}}_{\hat{c}}^r)^{-1} (\boldsymbol{r}_t - \boldsymbol{\mu}_{\hat{c}}^r). \tag{38}$$

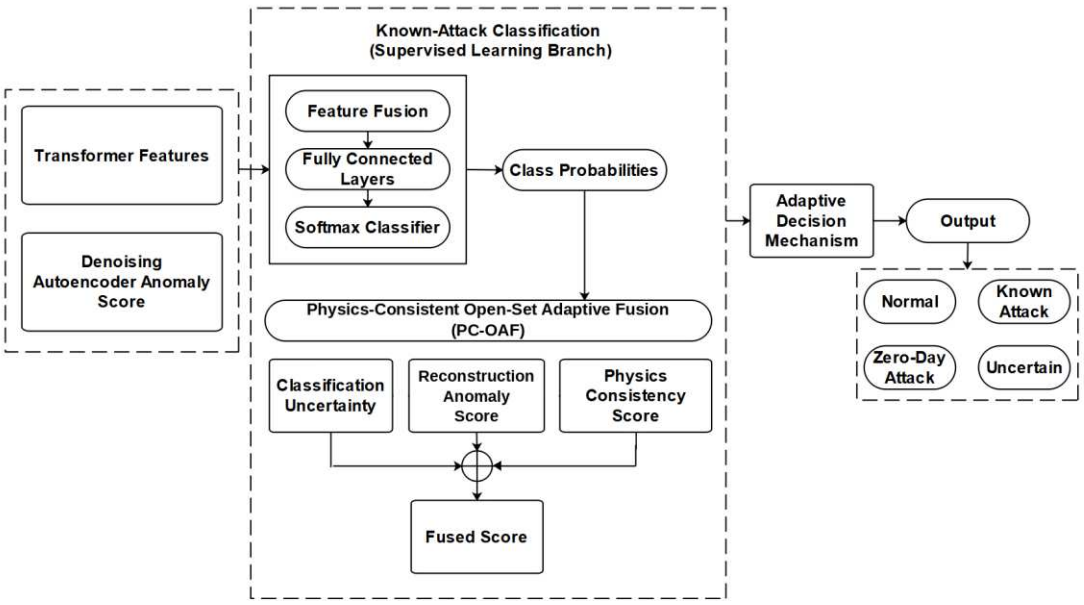

Fig. 7 Physics-consistent open-set adaptive fusion (PC-OAF): the indicators u_t, a_t and c_t are fused into the score S_t, from which the adaptive decision mechanism derives the output

5.5.3 Class-conditional physics-consistency surprise

In the same way, the physics profile $\boldsymbol{q}_t$ (Eqs. (13)–(14)), averaged over the last three steps of the window, is compared with the profile expected for the predicted class through Eqs. (39) and (40):

$$c_t = 1 - \exp\left(-D^2_{q,t}/\kappa_q\right), \tag{39}$$

$$D^2_{q,t} = \left(\boldsymbol{q}_t - \boldsymbol{\mu}^q_{\hat{c}}\right)^{\top} \left(\widehat{\boldsymbol{\Sigma}}^q_{\hat{c}}\right)^{-1} \left(\boldsymbol{q}_t - \boldsymbol{\mu}^q_{\hat{c}}\right). \tag{40}$$

A window classified as normal but showing an AMI–SCADA imbalance, or classified as FDIA but showing *no* Kirchhoff inconsistency, is therefore physically surprising. Covariances are shrunk towards their diagonal, $\widehat{\boldsymbol{\Sigma}} = (1-\lambda_s)\boldsymbol{\Sigma} + \lambda_s \mathrm{diag}(\boldsymbol{\Sigma}) + \epsilon \boldsymbol{I}$ ($\lambda_s = 0.2$, and 0.1 for the shared latent covariance $\boldsymbol{\Sigma}_z$), and each scale $\kappa = \chi^2_{0.99}(\mathrm{dim})/\ln 2$ is chosen so that a score of 0.5 corresponds to the 99th percentile of the reference distribution (Proposition 3).

5.5.4 Adaptive fusion

The three indicators are fused with non-negative weights in Eq. (41). The physics weight is modulated by the data freshness ϕ_t through Eq. (42), which reduces its influence when packets are lost and the residuals are computed from stale measurements:

$$S_t = \alpha_t u_t + \beta_t a_t + \gamma_t c_t, \tag{41}$$

$$\alpha_t = \frac{\alpha}{Z_t}, \quad \beta_t = \frac{\beta}{Z_t}, \quad \gamma_t = \frac{\gamma\,\overline{\phi}_t}{Z_t}, \quad Z_t = \alpha + \beta + \gamma\,\overline{\phi}_t, \tag{42}$$

so that $\alpha_t + \beta_t + \gamma_t = 1$ at every step, where $\overline{\phi}_t$ is the mean freshness over the window.

5.5.5 Calibrated threshold and three-way decision

The threshold is the $(1-\delta)$ empirical quantile of the fused scores of the known-class windows in a held-out calibration split $\mathcal{D}_{\mathrm{cal}}$, Eq. (43):

$$\tau_0 = Q_{1-\delta}\left(\{S_j : j \in \mathcal{D}_{\mathrm{cal}}\}\right). \tag{43}$$

Under a split-conformal protocol, this threshold bounds the probability that a known-class window is flagged as novel by δ (Proposition 5). In the main experiments, the temperature, the class statistics and τ_0 are estimated on the same two calibration days, so the quantile is computed in-sample, δ acts as a nominal budget and the realised FPR is higher; Section 7.8.2 compares this protocol with a strict split. The final decision mechanism is the three-way rule of Eq. (44):

$$\hat{y}_t = \begin{cases} \mathrm{ZD}, & S_t > \tau_t, \\ \operatorname{argmax}_{c \in \mathcal{K}} p_{t,c}, & S_t \le \tau_t \text{ and } \max_c p_{t,c} \ge \eta, \\ \mathrm{UNC}, & \text{otherwise}, \end{cases} \tag{44}$$

where η is the minimum acceptable known-class confidence. Both suspected zero-day (ZD) and uncertain (UNC) outputs raise an alert for operator review, which gives a human-in-the-loop path for previously unseen threats. Algorithm 4 summarises PC-OAF.

Algorithm 4 Physics-consistent open-set adaptive fusion (PC-OAF)

Require: calibration windows $\mathcal{D}_{\text{cal}}$ (known classes); trained PGST and SA-DAE; weights (α, β, γ); budget δ; confidence η
Ensure: decisions $\hat{y}_t \in \mathcal{K} \cup \{\text{ZD,UNC}\}$ and novelty-risk scores S_t
Calibration (offline)
Compute $\boldsymbol{p}_j$, $\boldsymbol{z}_j$, $\boldsymbol{r}_j$, $\boldsymbol{q}_j$ and $\overline{\boldsymbol{\phi}}_j$ for all $j \in \mathcal{D}_{\text{cal}}$; fit T_c
for each class $c \in \mathcal{K}$ **do**
  Fit $\boldsymbol{\mu}_c^z$, $(\boldsymbol{\mu}_c^r, \hat{\boldsymbol{\Sigma}}_c^r)$ and $(\boldsymbol{\mu}_c^q, \hat{\boldsymbol{\Sigma}}_c^q)$ on correctly classified windows
end for
Fit $\boldsymbol{\Sigma}_z$; set each scale $\kappa \leftarrow \chi^2_{0.99}(m_v)/\ln 2$ ▷ Proposition 3
Compute S_j and set $\tau_0 \leftarrow Q_{1-\delta}(\{S_j\})$ ▷ Eqs. (35)–(43)
Online inference
for each incoming window $\boldsymbol{X}_t$ **do**
  Obtain $\boldsymbol{p}_t$ and $\boldsymbol{z}_t$ from PGST, $\boldsymbol{r}_t$ from SA-DAE, and $\boldsymbol{q}_t$ and $\overline{\boldsymbol{\phi}}_t$ from CP-DAP
  Compute the indicators u_t, a_t and c_t ▷ Eqs. (35)–(40)
  Fuse them into the novelty-risk score S_t ▷ Eqs. (41)–(42)
  Assign $\hat{y}_t$ with the current threshold τ_t; alert if $\hat{y}_t \neq 0$ ▷ Eq. (44)
end for

5.6 Concept-drift-aware continual learning (CDCL)

CDCL keeps Stages 2–4 calibrated during operation. It answers three questions in turn: which observations may be trusted for adaptation (Section 5.6.1), when the operating regime has changed (Section 5.6.2), and how the threshold and the models are then updated (Sections 5.6.3 and 5.6.4).

5.6.1 Physics-gated trust

A fixed threshold calibrated on historical data produces more and more false alarms as load grows, renewable penetration increases or the topology is reconfigured. Naive online adaptation, however, can be steered by an attacker who slowly shifts the data. CDCL resolves this tension with *physics-gated trust*. Physical laws do not drift, so only windows that the classifier labels as normal with high confidence *and* that are physically consistent form the trusted set of Eq. (45), which alone is used for adaptation:

$$\mathcal{T}_t = \{ j \le t:\ \hat{c}_j = 0,\ \ p_{j,0} \ge \pi_T,\ \ c_j \le c_{\text{trust}} \}, \tag{45}$$

where $\pi_T = 0.9$ and c_{trust} is the 95th percentile of c over the normal calibration windows.

5.6.2 Drift detection

A buffer $\mathcal{B}_t$ holds the fused scores of the 360 most recent trusted windows. Drift is declared when a two-sample Kolmogorov–Smirnov test, based on the maximum distance between the empirical cumulative distributions $F_{\mathcal{B}_t}$ and $F_{\mathcal{R}}$, rejects the hypothesis that the buffer and the reference $\mathcal{R}$ come from the same distribution. The reference holds the scores of the calibration windows that pass the same trust gate, so the test compares like with like and responds to a change of regime rather than to the selection made by the gate. Buffers of hundreds of windows would also reject the ordinary differences between operating days, so the KS distance of Eq. (46) must also exceed a floor $D_{\min}$, the largest distance observed on the drift-free validation day:

$$D_{\text{KS}} = \max_s \left| F_{\mathcal{B}_t}(s) - F_{\mathcal{R}}(s) \right|, \tag{46}$$

and drift is declared when the p-value of D_{KS} falls below α_{KS} and $D_{\text{KS}} > D_{\min}$.

5.6.3 Threshold adaptation

While drift is declared, the threshold follows the trusted scores through the bounded exponential moving quantile of Eq. (47), which prevents abrupt manipulation:

$$\tau_t = \tau_{t-1} + \text{clip}(\rho[Q_{1-\delta}(\mathcal{B}_t) - \tau_{t-1}],\ -\Delta_\tau,\ \Delta_\tau), \tag{47}$$

which is the bounded form of $\tau_t = (1-\rho)\tau_{t-1} + \rho\, Q_{1-\delta}(\mathcal{B}_t)$; $\rho \in (0,1)$ is the adaptation rate and Δ_τ the maximum step.

5.6.4 Replay-based fine-tuning and recalibration

When drift is declared, the buffer holds at least B_{ft} windows and no fine-tuning has taken place in the last K_{ft} blocks, both branches are fine-tuned for 40 gradient steps on the 240 most recent trusted windows mixed with a stratified replay memory $\mathcal{M}$ of training exemplars. The replay memory retains knowledge of earlier attacks and prevents catastrophic forgetting. The SA-DAE and the PGST are fine-tuned with Eqs. (48) and (49), respectively:

$$\min_{\theta,\phi} \sum_{j \in \mathcal{T}_t \cup \mathcal{M}} \| \boldsymbol{X}_j - g_\phi\left(f_\theta(\tilde{\boldsymbol{X}}_j)\right) \|_F^2, \tag{48}$$

$$\min_{\Theta_c} \sum_{j \in \mathcal{T}_t} \mathcal{L}_{\text{cls}}\left(\boldsymbol{X}_j, 0\right) + \sum_{j \in \mathcal{M}} \mathcal{L}_{\text{cls}}\left(\boldsymbol{X}_j, y_j\right), \tag{49}$$

where $\widetilde{\boldsymbol{X}}_j$ is the window corrupted with a fixed noise level ($\sigma = 0.1$) and the same random masking as in pre-training ($\rho_m = 0.15$), and f_θ and g_ϕ are the SA-DAE encoder and decoder. Finally, PC-OAF is recalibrated. Known-attack statistics are re-estimated on the calibration split with the updated networks, and the normal-class statistics blend the calibration reference with the recent trusted regime according to Eqs. (50) and (51):

$$\boldsymbol{\mu}_0 \leftarrow (1-\rho_c)\,\boldsymbol{\mu}_{0,\text{cal}} + \rho_c\,\boldsymbol{\mu}_{0,\mathcal{T}}, \tag{50}$$

$$\boldsymbol{\Sigma}_0 \leftarrow (1-\rho_c)\,\boldsymbol{\Sigma}_{0,\text{cal}} + \rho_c\,\boldsymbol{\Sigma}_{0,\mathcal{T}}, \tag{51}$$

applied to the reconstruction and physics statistics of the normal class (means and covariances) and to the mean of its latent statistics. The reference $\mathcal{R}$ is then replaced by the recalibrated scores of the recent trusted windows, so that later tests measure drift relative to the adapted regime, and the threshold is re-anchored to the larger of the $(1-\delta)$ quantiles of these scores and of the recalibrated calibration scores. The dashed arrows in Fig. 3 correspond to these three actions: threshold adaptation for the decision stage, and fine-tuning and recalibration for the PGST and the SA-DAE. Algorithm 5 summarises CDCL.

Algorithm 5 Concept-drift-aware continual learning (CDCL)

Require: stream of windows in blocks of B steps; PGST, SA-DAE and PC-OAF; replay memory $\mathcal{M}$; ρ, Δ_τ, α_{KS}, π_T, ρ_c, $B_{\min}$, B_{ft}, K_{ft}
Ensure: decisions $\hat{y}_t$, adapted threshold τ_t and adapted models
$\tau \leftarrow \tau_0$; $\mathcal{B} \leftarrow \emptyset$; $\mathcal{R} \leftarrow$ scores of the trusted normal calibration windows
Set $D_{\min}$ to the largest KS distance observed on the drift-free validation day
for each block **do**
  Score the windows and decide with PC-OAF using the current τ ▷ Algorithm 4
  Append the trusted scores to $\mathcal{B}$; keep the 360 most recent ▷ Eq. (45)
  drift ← **false**
  if $|\mathcal{B}| \geq B_{\min}$ **then**
    drift ← $(p_{\text{KS}}(\mathcal{B},\mathcal{R}) < \alpha_{\text{KS}}) \wedge (D_{\text{KS}}(\mathcal{B},\mathcal{R}) > D_{\min})$ ▷ Eq. (46)
    if *drift* **then**
      Update τ with the bounded moving quantile ▷ Eq. (47)
    end if
  end if
  if *drift* **and** $|\mathcal{B}| \geq B_{\text{ft}}$ **and** no fine-tuning in the last K_{ft} blocks **then**
    Fine-tune SA-DAE and PGST on trusted windows and $\mathcal{M}$ ▷ Eqs. (48)–(49)
    Recalibrate the PC-OAF statistics ▷ Eqs. (50)–(51)
    Re-anchor $\mathcal{R}$ and τ on the recalibrated trusted and calibration scores
  end if
end for

5.7 Stage 5: Performance evaluation and multi-objective optimisation

Stage 5 defines how PG-ADTA-ID is evaluated (Section 5.7.1) and how its network hyperparameters and decision parameters are optimised without access to zero-day data (Section 5.7.2).

5.7.1 Performance metrics

Fig. 8 shows the evaluation and optimisation stage. Classification of known attacks is measured with accuracy, macro-averaged precision, recall and F1-score, MCC and Cohen's κ. The macro-F1-score of Eq. (52) gives minority attack classes the same weight as the majority class:

$$F_{1,\text{macro}} = \frac{1}{C}\sum_{c\in\mathcal{K}} \frac{2P_cR_c}{P_c + R_c + \epsilon}, \tag{52}$$

where $P_c = TP_c/(TP_c + FP_c)$ and $R_c = TP_c/(TP_c + FN_c)$ are the precision and recall of class c.

Open-set performance is measured by the zero-day detection rate (ZDR) and the false-positive rate (FPR) of Eq. (53):

$$\text{ZDR} = \frac{TP_Z}{TP_Z + FN_Z}, \qquad \text{FPR} = \frac{FP_N}{FP_N + TN_N}, \tag{53}$$

both reported in percent, where TP_Z counts windows of withheld attack families flagged as ZD or UNC, FN_Z counts those accepted as normal or as a known class, FP_N counts normal windows that raise any alert and TN_N counts normal windows accepted as normal. The *zero-day alarm rate* (the fraction of unseen-attack windows that raise any alert), event-level detection rates, detection delay, the area under the ROC curve (AUROC) for separating known from unknown windows and the inference latency are also reported.

5.7.2 Two-stage multi-objective optimisation

The framework is optimised in two stages with the multi-objective criterion of Eq. (54), which balances the competing goals:

$$J(\boldsymbol{\omega}) = \lambda_1 F_1 + \lambda_2\,\text{ZDR}_n - \lambda_3\frac{\text{FPR}_n}{\varepsilon_{\text{FPR}}} - \lambda_4 T_{\text{norm}}, \tag{54}$$

which is maximised over the parameters $\boldsymbol{\omega}$ of each stage, where $\lambda_1, \ldots, \lambda_4 \geq 0$ with $\sum_i \lambda_i = 1$, ZDR_n and FPR_n are expressed between 0 and 1, $\varepsilon_{\text{FPR}} = 0.02$ normalises the FPR to the target budget and T_{norm} is the inference time per

window in milliseconds, capped at 1. *Stage A* tunes the network hyperparameters (model width, numbers of encoder layers and attention heads, learning rate and dropout) by Bayesian optimisation (BO) with a tree-structured Parzen estimator (TPE) on the validation split. Only known classes exist there, so J reduces to $J_A = F_{1,\text{macro}} - \lambda_T T_{\text{norm}}$ with $\lambda_T = 0.02$. *Stage B* tunes the decision parameters $\boldsymbol{\theta} = (\alpha, \beta, \gamma, \delta, \eta)$ on a *leave-one-known-class-out* (LOCO) pseudo-open-set: for each known attack class k, PGST and SA-DAE are retrained without k, which then plays the role of an unseen attack on the calibration split. Here F_1 is the open-set macro-F1-score over the remaining known classes and the pseudo-unknown class, $(\lambda_1, \lambda_2, \lambda_3, \lambda_4) = (0.40, 0.40, 0.15, 0.05)$, and the latency term is constant because the decision parameters do not change the inference cost. Bayesian optimisation, particle swarm optimisation (PSO) and a genetic algorithm (GA) are compared under an identical evaluation budget, and the best configuration is retained. Neither stage uses the zero-day test families. Algorithm 6 summarises the procedure.

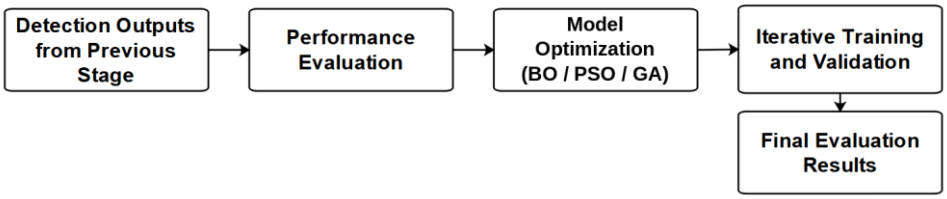

Fig. 8 Performance evaluation and optimisation stage of PG-ADTA-ID

Algorithm 6 Two-stage multi-objective optimisation of PG-ADTA-ID

Require: training, validation and calibration splits with known classes $\mathcal{K}$; budgets N_A and N_B; weights $\lambda_1, \ldots, \lambda_4$
Ensure: network hyperparameters $\boldsymbol{\omega}_{\text{net}}^{\text{opt}}$; decision parameters $\boldsymbol{\theta}^{\text{opt}}$
Stage A: network hyperparameters (BO with TPE)
for $n = 1, \ldots, N_A$ **do**
  Propose $\boldsymbol{\omega}_n$ with the TPE surrogate and train PGST briefly
  Record the validation $F_{1,\text{macro}}$ and the latency; update the surrogate
end for
$\boldsymbol{\omega}_{\text{net}}^{\text{opt}} \leftarrow$ the configuration with the best $J_A = F_{1,\text{macro}} - \lambda_T T_{\text{norm}}$
Stage B: decision parameters on a LOCO pseudo-open-set
for each known attack class $k = 1, \ldots, C-1$ **do**
  Retrain PGST and SA-DAE without class k
  Score the calibration split with class k treated as unknown
end for
for each optimiser in {BO,PSO,GA} **do**
  Maximise the fold-averaged $J(\boldsymbol{\theta})$ with N_B evaluations ▷ Eq. (54)
end for
return $\boldsymbol{\omega}_{\text{net}}^{\text{opt}}$ and the best $\boldsymbol{\theta}^{\text{opt}}$ found by any optimiser

5.8 Complete pseudocode of PG-ADTA-ID

Algorithm 7 combines the module algorithms into the complete pseudocode of the proposed framework. The offline procedure prepares the data, optimises and trains the two branches, tunes the decision stage and calibrates PC-OAF. The online procedure scores each new window, takes the three-way decision, raises an alert together with the evidence that caused it, and adapts the detector when drift is declared. Every step refers to the equation or algorithm that defines it.

Algorithm 7 PG-ADTA-ID: complete pseudocode of the proposed framework

Require: historical PMU, SCADA, AMI and traffic records with labels $y \in \mathcal{K}$; live data streams; T, π_T, ρ, Δ_τ, α_{KS}, $B_{\min}$, B_{ft}, K_{ft}
Ensure: decisions $\hat{y}_t \in \mathcal{K} \cup \{\mathrm{ZD}, \mathrm{UNC}\}$ with alerts and their evidence (u_t, a_t, c_t)

procedure OFFLINE(historical records)
  $(\boldsymbol{X}_f, \boldsymbol{Q}, \boldsymbol{\phi}) \leftarrow$ CP-DAP of the historical records ▷ Algorithm 1
  Form the windows $\boldsymbol{X}_t$; split them chronologically into $\mathcal{D}_{\mathrm{tr}}$, $\mathcal{D}_{\mathrm{va}}$ and $\mathcal{D}_{\mathrm{cal}}$
  $\boldsymbol{\omega}_{\mathrm{net}}^{\mathrm{opt}} \leftarrow$ Stage A on $\mathcal{D}_{\mathrm{va}}$ ▷ Algorithm 6
  Train PGST on $\mathcal{D}_{\mathrm{tr}}$ with early stopping on $\mathcal{D}_{\mathrm{va}}$ ▷ Algorithm 2
  Train SA-DAE on the known-world windows of $\mathcal{D}_{\mathrm{tr}}$ ▷ Algorithm 3
  $(\alpha, \beta, \gamma, \delta, \eta) \leftarrow$ Stage B on the LOCO folds ▷ Algorithm 6
  Fit T_c, the class statistics and the scales κ on $\mathcal{D}_{\mathrm{cal}}$ ▷ Algorithm 4
  $\tau_0 \leftarrow Q_{1-\delta}\big(\{S_j\}_{j \in \mathcal{D}_{\mathrm{cal}}}\big)$ ▷ Eq. (43)
  $c_{\mathrm{trust}} \leftarrow Q_{0.95}(c)$ on the normal calibration windows ▷ Eq. (45)
  $\mathcal{R} \leftarrow$ trusted calibration scores; set $D_{\min}$ on $\mathcal{D}_{\mathrm{va}}$; $\mathcal{M} \leftarrow$ replay exemplars of $\mathcal{D}_{\mathrm{tr}}$
end procedure
procedure ONLINE(data streams)
  $\tau \leftarrow \tau_0$; $\mathcal{B} \leftarrow \emptyset$
  for each new time step t **do**
    $\boldsymbol{x}_t \leftarrow$ CP-DAP of the new records; $\boldsymbol{X}_t \leftarrow [\boldsymbol{x}_{t-T+1}, \dots, \boldsymbol{x}_t]$
    $(\boldsymbol{p}_t, \boldsymbol{z}_t) \leftarrow \mathrm{PGST}\big(\boldsymbol{X}_t, \widehat{\boldsymbol{A}}_t\big)$; $\boldsymbol{r}_t \leftarrow$ SA-DAE$(\boldsymbol{X}_t)$ ▷ Algorithms 2 and 3
    $\boldsymbol{q}_t$, $\overline{\phi}_t \leftarrow$ physics profile and mean freshness of the window
    Compute u_t, a_t and c_t ▷ Eqs. (35)–(40)
    $S_t \leftarrow \alpha_t u_t + \beta_t a_t + \gamma_t c_t$ ▷ Eqs. (41)–(42)
    $\hat{y}_t \leftarrow \mathrm{argmax}_c p_{t,c}$ ▷ Eq. (44)
    if $S_t > \tau$ **then**
      $\hat{y}_t \leftarrow$ ZD
    else if $\max_c p_{t,c} < \eta$ **then**
      $\hat{y}_t \leftarrow$ UNC
    end if
    Raise an alert with $\hat{y}_t$ and its evidence (u_t, a_t, c_t) whenever $\hat{y}_t \neq 0$
    if $\hat{c}_t = 0$ **and** $p_{t,0} \geq \pi_T$ **and** $c_t \leq c_{\mathrm{trust}}$ **then**
      Append S_t to $\mathcal{B}$; keep the 360 most recent scores ▷ Eq. (45)
    end if
    if the current block ends **and** $|\mathcal{B}| \geq B_{\min}$ **then**
      if $p_{\mathrm{KS}}(\mathcal{B}, \mathcal{R}) < \alpha_{\mathrm{KS}}$ **and** $D_{\mathrm{KS}}(\mathcal{B}, \mathcal{R}) > D_{\min}$ **then**
        $\tau \leftarrow \tau + \mathrm{clip}(\rho\,[Q_{1-\delta}(\mathcal{B}) - \tau],\ -\Delta_\tau,\ \Delta_\tau)$ ▷ Eq. (47)
        if $|\mathcal{B}| \geq B_{\mathrm{ft}}$ **and** no fine-tuning in the last K_{ft} blocks **then**
          Fine-tune both branches on trusted windows and $\mathcal{M}$ ▷ Eqs. (48)–(49)
          Recalibrate the PC-OAF statistics ▷ Eqs. (50)–(51)
          $\mathcal{R} \leftarrow$ recalibrated trusted scores; $\mathcal{S}_{\mathrm{cal}} \leftarrow$ recalibrated calibration scores
          $\tau \leftarrow \max\{Q_{1-\delta}(\mathcal{R}),\ Q_{1-\delta}(\mathcal{S}_{\mathrm{cal}})\}$
        end if
      end if
    end if
  end for
end procedure

5.9 Theoretical analysis

The design choices of PG-ADTA-ID rest on a few properties that can be stated formally. This section establishes them under idealised assumptions (exact physics, exchangeable scores and Gaussian class-conditional statistics) and points to the experiments that test how far they carry over to the co-simulated data. Section 5.9.1 concerns the physics evidence, Section 5.9.2 the fusion and calibration of PC-OAF, Section 5.9.3 the adaptation of CDCL and Section 5.9.4 the computational cost.

5.9.1 Physics evidence

Proposition 1 (Operating-point invariance of the physics residuals) *Let the PMU phasors, SCADA injections and flows, breaker statuses and AMI demands at step t be exact and consistent with an AC power-flow solution of the current topology. Then $r_{t,i}^{\mathrm{KCL}} = 0$, $r_{t,k}^{\mathrm{Ohm}} = 0$ and $r_{t,i}^{\mathrm{AMI}} = 0$ for every bus i and branch k, whatever the loading, the renewable output and the generator dispatch.*

Proof Let I_k^{f} and I_k^{t} be the currents that enter branch k at its from and to ends, so that $S_{t,k}^{\mathrm{f}} = V_{t,f(k)}\overline{I_k^{\mathrm{f}}}$ and $S_{t,k}^{\mathrm{t}} = V_{t,t(k)}\overline{I_k^{\mathrm{t}}}$. Kirchhoff's current law states that the current injected at bus i equals the sum of the branch currents leaving the bus plus the shunt current, $I_{t,i} = \sum_{k:f(k)=i} I_k^{\mathrm{f}} + \sum_{k:t(k)=i} I_k^{\mathrm{t}} + y_i^{\mathrm{sh}} V_{t,i}$. Multiplying the conjugate of this identity by $V_{t,i}$ gives $S_{t,i} = \sum_{k:f(k)=i} S_{t,k}^{\mathrm{f}} + \sum_{k:t(k)=i} S_{t,k}^{\mathrm{t}} + |V_{t,i}|^2\,\overline{y_i^{\mathrm{sh}}}$, so $r_{t,i}^{\mathrm{KCL}} = 0$ by Eq. (10). For an in-service branch, the π-model gives $I_k^{\mathrm{f}} = y_k^{\mathrm{ff}} V_{t,f(k)} + y_k^{\mathrm{ft}} V_{t,t(k)}$, so $r_{t,k}^{\mathrm{Ohm}} = 0$ by Eq. (11); for an open branch, both $S_{t,k}^{\mathrm{f}}$ and $s_{t,k}$ vanish. Finally, the net active injection of a bus is its generation minus its demand, $P_{t,i} = P_{t,i}^{\mathrm{g}} - P_{t,i}^{\mathrm{d}}$, and exact metering gives $P_{t,i}^{\mathrm{AMI}} = P_{t,i}^{\mathrm{d}}$, so $r_{t,i}^{\mathrm{AMI}} = 0$ by Eq. (12). None of these identities involves the loading or the dispatch. □

With measurement noise, the residuals are therefore driven by the measurement errors and not by the operating point itself, and the robust standardisation of Section 5.2.2 brings them to a common scale. A residual whose mean departs from zero signals inconsistent data, such as falsified measurements, wrong breaker information or desynchronised phasors, rather than a new operating point. This is why the residuals serve both as features and, through the physics surprise, as the trust criterion of CDCL (Eq. (45)).

Proposition 2 (Visibility of power-flow-consistent injections) *Let an attacker replace the PMU and SCADA measurements by* $\boldsymbol{h}_{\mathrm{pf}}(\boldsymbol{\xi}+\Delta\boldsymbol{\xi})$*, the values that the AC power-flow model of the current topology produces for a falsified state* $\boldsymbol{\xi}+\Delta\boldsymbol{\xi}$*, and leave the AMI readings and the generator telemetry untouched. Then (i) the Kirchhoff and Ohm's-law residuals are unaffected; (ii) at every metered bus* i*, the AMI–SCADA residual becomes the change of the active injection, as given by Eq. (55),*

$$r_{t,i}^{\mathrm{AMI}} = P_{t,i}(\boldsymbol{\xi}+\Delta\boldsymbol{\xi}) - P_{t,i}(\boldsymbol{\xi}) = \Delta P_{t,i}; \tag{55}$$

and (iii) in the DC approximation, $\Delta\boldsymbol{P} = \boldsymbol{B}\,\Delta\boldsymbol{\theta}$*, and if the falsified angles are confined to a set* $\mathcal{A}$ *of metered buses that does not contain every bus, then* $\Delta\boldsymbol{P}_{\mathcal{A}} = \boldsymbol{B}_{\mathcal{A}\mathcal{A}}\Delta\boldsymbol{\theta}_{\mathcal{A}} \neq \boldsymbol{0}$ *whenever* $\Delta\boldsymbol{\theta}_{\mathcal{A}} \neq \boldsymbol{0}$*.*

Proof (i) The falsified measurements satisfy the AC power-flow equations for the state $\boldsymbol{\xi}+\Delta\boldsymbol{\xi}$, so Proposition 1 applies to them. (ii) The falsified injection is $\tilde{P}_{t,i} = P_{t,i}(\boldsymbol{\xi}+\Delta\boldsymbol{\xi})$, whereas the untouched generator telemetry and metering satisfy $P_{t,i}^{\mathrm{g}} - P_{t,i}^{\mathrm{AMI}} = P_{t,i}(\boldsymbol{\xi})$; substituting both into Eq. (12) gives Eq. (55). (iii) In the DC model, $\boldsymbol{P} = \boldsymbol{B}\boldsymbol{\theta}$, where $\boldsymbol{B}$ is a weighted Laplacian of the network graph with positive branch susceptances. If $\Delta\boldsymbol{\theta}$ vanishes outside $\mathcal{A}$, then $\Delta\boldsymbol{P}_{\mathcal{A}} = \boldsymbol{B}_{\mathcal{A}\mathcal{A}}\Delta\boldsymbol{\theta}_{\mathcal{A}}$. In a connected network, every connected component of the subgraph induced by $\mathcal{A}$ has at least one branch to a bus outside $\mathcal{A}$, so $\boldsymbol{B}_{\mathcal{A}\mathcal{A}}$ is irreducibly diagonally dominant in each component, hence positive definite, and $\boldsymbol{B}_{\mathcal{A}\mathcal{A}}\Delta\boldsymbol{\theta}_{\mathcal{A}} \neq \boldsymbol{0}$ for $\Delta\boldsymbol{\theta}_{\mathcal{A}} \neq \boldsymbol{0}$. □

Proposition 2 explains why the stealthy FDIA of the benchmark, which falsifies the state at one to three load buses, leaves the Kirchhoff and Ohm's-law residuals intact but not the AMI–SCADA balance, and why it is detected through the physics evidence (Section 7.3). It also states the price of evasion: to remain invisible to the residuals, an attacker must compromise, in addition to the PMU and SCADA paths, the independently operated metering of every bus whose injection the attack changes.

5.9.2 Fusion and calibration

Proposition 3 (Common scale of the indicators) *For a predicted class* $\hat{c}$*, let* $\boldsymbol{v} \in \mathbb{R}^{m_v}$ *be one of the evidence vectors* $\boldsymbol{z}_t$*,* $\boldsymbol{r}_t$ *or* $\boldsymbol{q}_t$*, and suppose that* $\boldsymbol{v} \mid \hat{c} \sim \mathcal{N}(\boldsymbol{\mu}_{\hat{c}}, \boldsymbol{\Sigma}_{\hat{c}})$ *with known parameters. Then* $D^2 = (\boldsymbol{v}-\boldsymbol{\mu}_{\hat{c}})^{\top}\boldsymbol{\Sigma}_{\hat{c}}^{-1}(\boldsymbol{v}-\boldsymbol{\mu}_{\hat{c}})$ *follows a* $\chi^2_{m_v}$ *distribution, and the indicator* $I = 1-\exp(-D^2/\kappa)$ *with* $\kappa = \chi^2_{0.99}(m_v)/\ln 2$ *has the tail probability of Eq. (56),*

$$\Pr(I > s) = 1 - F_{\chi^2_{m_v}}\big(-\kappa\ln(1-s)\big), \quad s \in [0,1), \tag{56}$$

where $F_{\chi^2_{m_v}}$ *is the* $\chi^2_{m_v}$ *distribution function. In particular,* $\Pr(I > 1/2) = 0.01$ *for every dimension* m_v*.*

Proof Whitening by $\boldsymbol{\Sigma}_{\hat{c}}^{-1/2}$ turns $\boldsymbol{v}-\boldsymbol{\mu}_{\hat{c}}$ into a standard normal vector of dimension m_v, whose squared norm follows the $\chi^2_{m_v}$ distribution. The map $D^2 \mapsto 1-\exp(-D^2/\kappa)$ is strictly increasing, so $I > s$ if and only if $D^2 > -\kappa\ln(1-s)$, which gives Eq. (56). For $s = 1/2$, $-\kappa\ln(1/2) = \kappa\ln 2 = \chi^2_{0.99}(m_v)$, whose exceedance probability is 0.01. □

The three indicators therefore share one probabilistic scale, although the latent embedding, the error signature and the physics profile have different dimensions (m_v = 64, 10 and 6 here), which makes the fusion weights of Eq. (41) comparable. Shrinkage and non-Gaussian tails make this calibration approximate, and the empirical threshold of Eq. (43) absorbs the remaining mismatch.

Proposition 4 (Range, monotonicity and single-indicator triggering) *For every window,* $u_t, a_t, c_t \in [0,1)$*, and the weights of Eq. (42) are non-negative with* $\alpha_t+\beta_t+\gamma_t = 1$*, so* $S_t \in [0,1)$*. The score* S_t *is non-decreasing in* $1-\max_c p_{t,c}$ *and in each of* $D^2_{z,t}$*,* $D^2_{r,t}$ *and* $D^2_{q,t}$*. Moreover, an indicator* I *with weight* w *can cause a suspected zero-day decision irrespective of the other indicators if and only if* $w > \tau_t$*, and then* $I > \tau_t/w$ *suffices.*

Proof Each indicator has the form $1-\exp(-x)$ with $x \geq 0$ or, for u_t, is the maximum of such a term and $1-\max_c p_{t,c} \in [0,1-1/C]$; it therefore lies in $[0,1)$ and is non-decreasing in its argument. By Eq. (42), the weights are non-negative, do not depend on the indicators and sum to one, so S_t is a convex combination of numbers in $[0,1)$ and inherits the monotonicity. For the last claim, $S_t \geq wI$, with equality when the other indicators are zero. If $w > \tau_t$, then $\tau_t/w < 1$, and $I > \tau_t/w$ implies $S_t > \tau_t$. If $w \leq \tau_t$, then $wI < w \leq \tau_t$, so a window whose other indicators are zero is never flagged. □

With the selected weights (α,β,γ) = (0.23, 0.63, 0.14) and the seed-0 threshold τ_0 = 0.609, only the reconstruction surprise can raise a zero-day decision on its own ($\beta > \tau_0$). Physics-only manipulations need supporting evidence. With the mean indicators of AMI-tampering windows, (u_t, a_t, c_t) = (0.29, 0.49, 0.64), the fused score is about 0.47, below τ_0, which anticipates the weak window-level detection of that family (Sections 7.3 and 7.10).

Proposition 5 (False-alarm control of the calibrated threshold) *Let* $S_1,\dots,S_n$ *be the fused scores of* n *known-class calibration windows, computed with PC-OAF statistics fitted on other data, and let* S_{n+1} *be the score of a new known-class window such that* $(S_1,\dots,S_{n+1})$ *are exchangeable. If* τ_0 *is the* $\lceil (n+1)(1-\delta)\rceil$*-th smallest calibration score, then the false-alarm bound of Eq. (57) holds:*

$$\Pr(S_{n+1} > \tau_0) \le \delta. \tag{57}$$

Proof By exchangeability, the rank R of S_{n+1} among the $n+1$ scores is uniformly distributed on $\{1, \dots, n+1\}$, with ties broken at random. The event $S_{n+1} > \tau_0$ requires S_{n+1} to exceed at least $\lceil (n+1)(1-\delta) \rceil$ calibration scores, that is, $R \ge \lceil (n+1)(1-\delta) \rceil + 1$, which has probability $(n+1-\lceil (n+1)(1-\delta) \rceil)/(n+1) \le \delta$ (Yuan et al. 2026; Sun et al. 2026). □

Eq. (57) is the guarantee behind the constraint of Eq. (7). The empirical quantile of Eq. (43) differs from the conformal order statistic by at most one rank, which changes the level by less than 10^{-4} for the roughly 1.7×10^4 calibration windows used here. The bound requires, however, that the PC-OAF statistics be fitted on data other than the calibration scores and that test windows be exchangeable with calibration windows. The in-sample protocol of the main experiments violates the first condition and rare topologies, such as the outage of line 1–2 on a test day, violate the second, which is why the realised FPR exceeds δ (Sections 7.3 and 7.8.2). Because the quantile pools all known classes, it bounds the fraction of known-class windows flagged as zero-day rather than the FPR of the normal class alone; a per-class (Mondrian) quantile would bound the latter.

5.9.3 Adaptation

Proposition 6 (Bounded and poisoning-resistant adaptation) *Let the threshold follow Eq. (47). (i) Every update satisfies* $|\tau_t - \tau_{t-1}| \le \Delta_\tau$*, so* m *consecutive updates move the threshold by at most* $m\Delta_\tau$*. (ii) Let an adversary control a fraction* ν *of the scores in the trusted buffer* $\mathcal{B}_t$*, and let* D_{clean} *be the KS distance between the empirical distribution of the remaining scores and the reference* $\mathcal{R}$*. Then the KS distance of Eq. (46) obeys the bound of Eq. (58),*

$$D_{\text{KS}} \le (1-\nu)\, D_{\text{clean}} + \nu, \tag{58}$$

so no drift can be declared, and the threshold cannot move, unless $\nu > (D_{\text{min}} - D_{\text{clean}})/(1 - D_{\text{clean}})$*. (iii) If* $\nu < \delta$*, the buffer quantile* $Q_{1-\delta}(\mathcal{B}_t)$ *used in Eq. (47) lies between the quantiles of the clean scores at the levels* $1 - \delta/(1-\nu)$ *and* $1 - (\delta - \nu)/(1-\nu)$*.*

Proof (i) follows from the clipping in Eq. (47). (ii) The empirical distribution function of the buffer is the mixture $F_{\mathcal{B}_t} = (1-\nu)F_{\text{clean}} + \nu F_{\text{adv}}$, so $F_{\mathcal{B}_t} - F_{\mathcal{R}} = (1-\nu)(F_{\text{clean}} - F_{\mathcal{R}}) + \nu(F_{\text{adv}} - F_{\mathcal{R}})$. Taking the supremum and using $|F_{\text{adv}} - F_{\mathcal{R}}| \le 1$ gives Eq. (58). Drift requires $D_{\text{KS}} > D_{\text{min}}$ (Section 5.6.2), which by Eq. (58) is possible only if $\nu\,(1 - D_{\text{clean}}) > D_{\text{min}} - D_{\text{clean}}$, and τ changes only while drift is declared. (iii) Let $N = |\mathcal{B}_t|$ and $q = Q_{1-\delta}(\mathcal{B}_t)$. At most δN buffer scores exceed q, hence at most a fraction $\delta/(1-\nu)$ of the clean scores, which gives the lower bound. At least δN buffer scores are not below q, and at most νN of them are adversarial, so at least a fraction $(\delta - \nu)/(1-\nu)$ of the clean scores is not below q, which gives the upper bound. □

Proposition 6 separates the roles of the three safeguards of CDCL. The trust gate of Eq. (45) decides which windows may enter the buffer: an adversarial window must be classified as normal with a confidence of at least π_T and must be physically consistent. The effect-size floor turns this into a quantitative barrier. With $D_{\text{min}} = 0.58$, an adversary acting on a drift-free day whose clean buffer lies at a KS distance of 0.3 from the reference must control more than 40% of the trusted windows before the threshold can move at all. The step bound then limits any change to $\Delta_\tau = 0.05$ per 10-min block. The same mechanism also shows the main risk of adaptation: a genuine but benign regime change, such as a rare line outage, satisfies the drift test and moves the threshold upwards, and without re-anchoring the threshold would not come back (Section 7.6.2).

5.9.4 Computational complexity

For a window of T steps with d fused features, n buses, model width d_m, feed-forward width d_{ff} and L_e encoder layers, PGST costs $\mathcal{O}\left(Td\,d_m + Tn^2 + L_e\left(T^2 d_m + T d_m d_{ff}\right)\right)$ per window. The SA-DAE costs $\mathcal{O}(Td\,c_h k)$ for c_h hidden channels and kernel size k, and PC-OAF costs $\mathcal{O}(d_z^2 + G^2 + 36)$. The overall cost is linear in the number of measured channels and quadratic only in the short window length T, which keeps inference within milliseconds (Section 7.9).

6 Experimental Setup

This section describes the cyber-physical co-simulation benchmark on which PG-ADTA-ID is evaluated (Section 6.1), the evaluation protocol (Section 6.2), the implementation (Section 6.3) and the eleven baseline detectors (Section 6.4).

6.1 Cyber-physical co-simulation benchmark

No public smart-grid dataset provides synchronised PMU, SCADA, AMI and communication-traffic channels together with withheld attack families and a documented drift period. Public datasets such as the Mississippi State University/ORNL power-system set (Samin et al. 2026; Alamgir et al. 2026) and recent digital-substation corpora (Cibin et al. 2026a) are valuable, but each covers only part of this evidence. Following common practice in power-system security research (Singh and Sailaja Kumari 2026; Zhao et al. 2026; Mohammadpourfard et al. 2026; Xi et al. 2026a), and in the spirit of open cyber ranges such as PowerRange (Bader et al. 2026), a reproducible cyber-physical co-simulation of the IEEE 14-bus system was therefore built (Fig. 9). The test system has 14 buses, 20 branches (15 lines and 5 transformers), 5 generators or synchronous condensers and 11 load buses carrying 259 MW of base load.

Network data are taken from the standard case file and the steady state is solved with a full Newton–Raphson AC power flow every $\Delta t = 10$ s, giving 103,680 time steps over 12 simulated days.

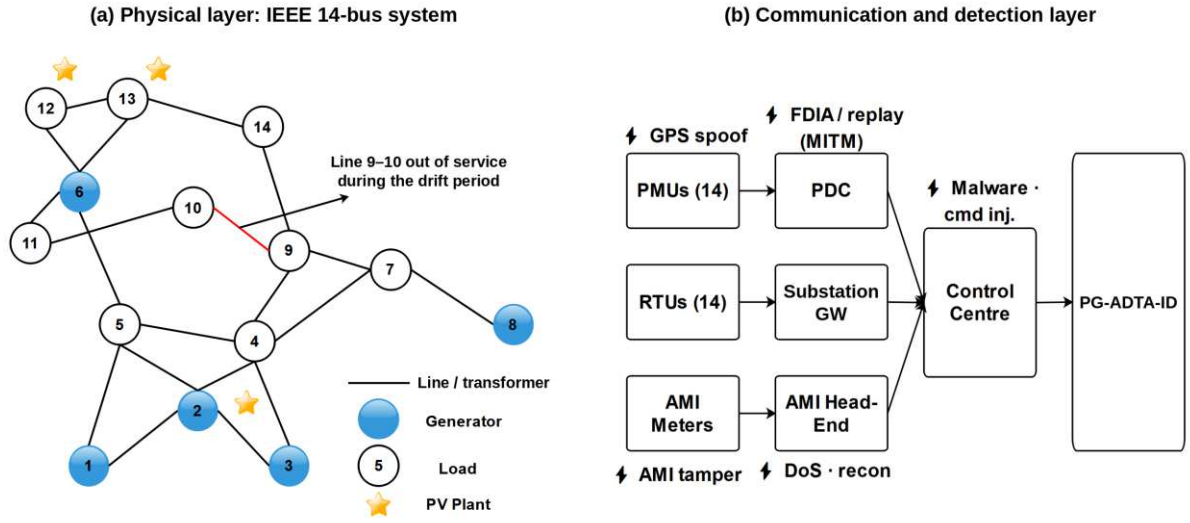

Fig. 9 Cyber-physical co-simulation testbed: (a) IEEE 14-bus physical layer with three PV plants (line 9–10, red, is out of service in the drift period); (b) communication and detection layer (PDC, phasor data concentrator; GW, gateway). Lightning symbols mark where the attack families act

6.1.1 Physical layer

Each load bus follows its own mixture of residential, commercial and industrial daily profiles. The profiles include day-to-day variation, a weekend reduction, first-order autoregressive fluctuations (10-min correlation, 2% standard deviation) and fast noise. Three PV plants, at buses 2, 12 and 13 (Fig. 9a), supply about 13% of the peak load. Their output follows a clear-sky profile modulated by a two-state Markov cloud process with fast transients. Generator dispatch follows the net load through participation factors, and the slack bus absorbs the losses. The *normal* class deliberately includes benign disturbances that make detection realistic: authorised switching of lines (1–2 per day, 10–60 min), faults followed by protection tripping and automatic reclosing (1–2 per day), load steps (3–6 per day), generator redispatch and nightly data backups that generate heavy traffic.

6.1.2 Measurement and cyber layers

PMUs report voltage magnitudes and angles and the system frequency at every step (noise standard deviation 0.002 p.u., 0.002 rad and 2 mHz). RTUs report bus injections, branch flows at both ends, generator output and breaker status every 20 s (noise $0.01|x| + 0.003$ p.u. for injections and flows and $0.005|x| + 0.002$ p.u. for generator output). The AMI head-end reports aggregated net demand per load bus every 60 s (noise $0.01|x| + 0.001$ p.u.), and 0.3% of measurement packets are lost at random. As shown in Fig. 9b, PMU data reach the control centre through the PDC, RTU data through the substation gateway and AMI data through the head-end. The communication layer is described by 22 traffic features: packet and byte rates, mean packet size, the mean and standard deviation of inter-arrival times, the numbers of distinct sources and destination ports, SYN and RST ratios, Modbus/DNP3 read and write commands, unsolicited responses, round-trip time, jitter, loss, retransmission, sequence anomalies, timestamp skew, authentication failures, port entropy, outbound byte ratio and the new-connection rate. These features follow diurnal operator activity, 15-min AMI collection bursts and the nightly backups.

6.1.3 Attack catalogue

Nine attack families were implemented (Table 2); the lightning symbols in Fig. 9b mark where each family acts. Four *known* families appear in all splits: DoS flooding with stale RTU data, non-stealthy FDIA, replay of the previous day's telemetry, and malware with command-and-control beaconing and lateral movement. Five *zero-day* families are withheld from training, validation and calibration and appear only in the test and drift periods: an AC-consistent stealthy FDIA built from a falsified state, $\boldsymbol{z}_a = \boldsymbol{h}_{\mathrm{pf}}(\boldsymbol{\xi} + \Delta\boldsymbol{\xi})$ (Mohammadpourfard et al. 2026; Zhao et al. 2026); a GPS time-synchronisation attack that ramps PMU phase angles (Qiu et al. 2026); unauthorised command injection that really trips a line (Acharya and Bhalja 2026); reconnaissance scanning; and AMI meter tampering. Episodes last 4–16 min. The FDIA, the stealthy FDIA and AMI tampering ramp in over 5–20 steps to evade simple change detection, the GPS attack shifts the phase angles by 0.0005–0.003 rad per step, and the other families start abruptly.

Table 2 Attack catalogue of the co-simulation benchmark (✓ = present in the split)

Family	Type	Physical effect	Cyber effect	Train/val/calib	Test/drift
DoS	Known	PMU and RTU channels of 3–6 buses lose 30–90% of packets (stale data)	1.8–8× packet rate, SYN, RTT, loss, retransmissions	✓	✓
FDIA	Known	Bias on 2–6 random measurements (V: 1–5%, θ: 0.01–0.06 rad, P/Q/flows: 5–30% plus 0.02–0.1 p.u.)	Small man-in-the-middle (MITM) latency, sequence noise	✓	✓
Replay	Known	PMU, RTU and breaker telemetry replaced by records from 24 h earlier (2–24 h on the first day)	Duplicate sequence numbers, timestamp skew	✓	✓
Malware	Known	None	Outbound ratio, ports, entropy, periodic beacons, authentication failures	✓	✓
Stealthy FDIA	Zero-day	AC-consistent falsified state at 1–3 load buses ($\Delta\theta$ 0.0015–0.007 rad)	None	—	✓
Time-sync (GPS)	Zero-day	PMU angle ramp at 1–3 buses (up to 0.02–0.15 rad)	None	—	✓
Command injection	Zero-day	Real unauthorised line trip	Rogue host, write burst, authentication failures	—	✓
Reconnaissance	Zero-day	None	20–200 extra ports, SYN/RST surge, probing reads	—	✓
AMI tampering	Zero-day	AMI demand of 1–3 buses scaled by 0.6–0.9	None	—	✓

Fig. 10 shows one representative episode of each family through two aggregate indicators. The first is a physics-inconsistency indicator, the largest of the Kirchhoff, Ohm's-law and AMI–SCADA residual maxima of Eq. (14). The second is a cyber-deviation indicator, the largest normalised traffic feature. Both are standardised on normal training data. The figure shows why both views are needed. DoS, malware and reconnaissance are visible almost only in the traffic. FDIA, time-synchronisation attacks and AMI tampering are visible only in the physics residuals. Replay affects both. The stealthy FDIA is consistent with the AC power-flow equations within the PMU and SCADA measurements, so it would pass a residual-based bad-data detector, but, as Proposition 2 predicts, it contradicts the untouched AMI demand and generator telemetry. Time-synchronisation attacks build up slowly, and command injection leaves only a short burst of cyber evidence around the unauthorised trip.

6.1.4 Splits and drift

The 12 days are split *chronologically* into disjoint sets (Table 3). Days 1–5 are used for training, day 6 for validation (early stopping and Stage-A tuning), days 7–8 for calibration (temperature, PC-OAF statistics and thresholds), days 9–10 for testing and days 11–12 as the drift period. During the drift period the load grows by 18%, the PV capacity doubles, line 9–10 is permanently out of service (Fig. 9a), PMU noise increases by 50% and the communication baseline shifts (+30% traffic, four new hosts and +3 ms round-trip time). Windows of $T = 20$ steps (200 s) are labelled by their last step. Training windows use a stride of two, and all other splits use a stride of one.

Table 3 Chronological splits and window counts (classes 0–4 known; 5–9 zero-day)

Split	Days	Windows	Normal	DoS	FDIA	Replay	Malware	Zero-day (5 families)
Training	1–5	21,591	16,887	1,135	1,093	1,271	1,205	0
Validation	6	8,621	6,738	460	421	415	587	0
Calibration	7–8	17,261	13,414	975	967	945	960	0
Test	9–10	17,261	11,383	719	762	624	911	2,862
Drift	11–12	17,261	11,477	766	537	794	603	3,084

6.2 Evaluation protocol

Window-level metrics follow Eqs. (52)–(53). Closed-set metrics (accuracy, macro-precision, macro-recall, macro-F1, MCC and κ) are computed on known-class windows. Open-set metrics (ZDR, zero-day alarm rate, FPR and the open-set macro-F1 over the $C + 1$ classes, in which all withheld families are merged into an "unknown" class) are computed on all remaining test windows. A window labelled normal but whose 200-s span still overlaps an attack that has just ended is ambiguous. These *transition* windows (1,862 in the test split) are therefore excluded from all window-level metrics, as is common practice for windowed detectors. At the *event* level, an attack episode counts as detected if at least one window ending inside it raises the correct alert. Detection delay is the time from the start of the episode to that first alert, and false-alarm events are counted as clusters of consecutive false alerts on normal windows. PG-ADTA-ID and six baselines (CNN–Transformer–BiGRU, Transformer, CNN–BiLSTM, LSTM, XGBoost and random forest) were trained with three random seeds, and the mean ± standard deviation over seeds is reported (population standard deviation, $n = 3$). The other five baselines were trained with one seed, and the single-model analyses (ablation, drift and interpretability) use the seed-0 model.

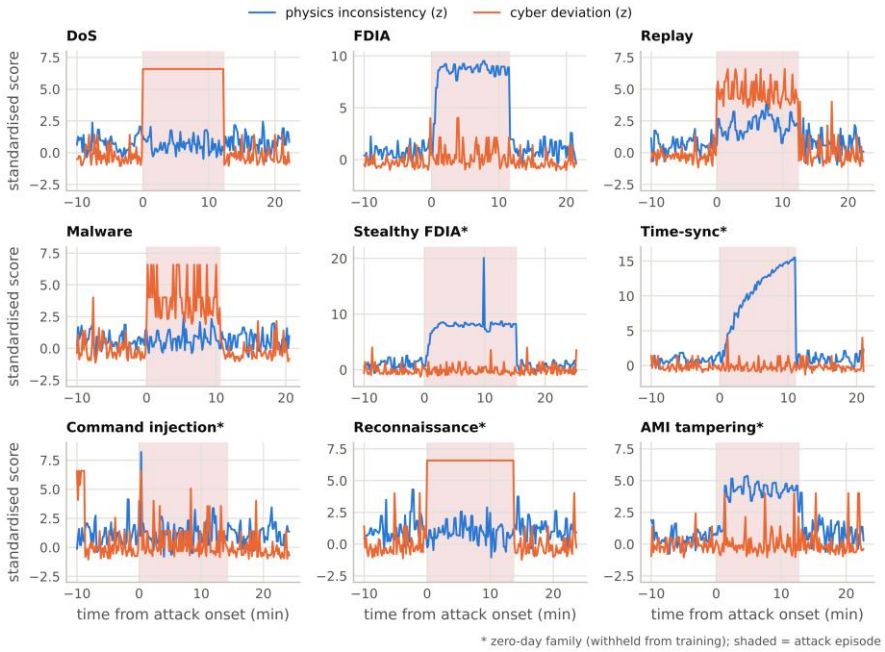

Fig. 10 Physical and cyber signatures of one representative episode of each attack family (shaded). Physics inconsistency is the largest physics-residual summary and cyber deviation the largest normalised traffic feature, both standardised on normal training data; * marks withheld zero-day families

6.3 Implementation details

All models were implemented in PyTorch and trained on a two-core CPU with 7 GB of RAM and no GPU, a deliberately modest platform close to substation-edge hardware. Table 4 lists the hyperparameters. The PGST configuration was selected by Stage-A Bayesian optimisation (Section 7.7.1). The decision parameters were selected in Stage B on the LOCO pseudo-open-set and fixed before the test data were evaluated.

Table 4 Hyperparameters of PG-ADTA-ID

Module	Hyperparameter	Value
CP-DAP	Window length T / step Δt	20 steps / 10 s
CP-DAP	Fused features d (14-bus)	280 (141 grid, 11 AMI, 20 breaker, 22 network, 1 freshness, 79 residuals, 6 summaries)
PGST	Width d_m / layers L_e / heads h / FFN d_{ff}	32 / 3 / 4 / 64
PGST	Graph convolution	8 → 16 → 16 channels, mean–max read-out
PGST	Optimiser / learning rate / dropout	AdamW (weight decay 10^{-4}) / 1.28×10^{-3} / 0.26
PGST	Batch / max epochs / patience / label smoothing	128 / 20 / 6 / 0.05
SA-DAE	Encoder channels / latent size / kernel	128–64 / 32 / 3
SA-DAE	Noise curriculum $\sigma_{\min} \rightarrow \sigma_{\max}$ / warm-up E_w	0.02 → 0.15 / 10 epochs
SA-DAE	Mask ratio ρ_m / masked-loss weight λ_m	0.15 / 1.0
PC-OAF	Weights (α, β, γ)	(0.23, 0.63, 0.14)
PC-OAF	Budget δ / confidence η / shrinkage λ_s (latent $\boldsymbol{\Sigma}_z$)	0.0035 / 0.50 / 0.2 (0.1)
Optimisation	Stage A: J_A weight λ_T / Stage B: $(\lambda_1, \lambda_2, \lambda_3, \lambda_4)$	0.02 / (0.40, 0.40, 0.15, 0.05)
CDCL	Block / buffer / $B_{\min}$ / ρ / Δ_τ / α_{KS}	60 / 360 / 120 windows / 0.3 / 0.05 / 0.01
CDCL	π_T / replay memory / fine-tuning steps / K_{ft} / B_{ft} / ρ_c	0.9 / 2,000 exemplars / 40 / 6 blocks / 180 windows / 0.5

6.4 Baselines

PG-ADTA-ID is compared with eleven baselines that represent the main families reviewed in Section 2. The *classical* baselines are an SVM with an RBF kernel ($C = 10$, trained on a stratified subset of 7,628 windows), and a random forest (300 trees, balanced class weights) and XGBoost (300 trees, depth 6), which represent tree-ensemble detectors (Sheta et al. 2026). These operate on window summaries (last value, mean and standard deviation of each channel). The *sequential deep-learning* baselines are a two-layer LSTM (Singh and Sailaja Kumari 2026), a two-layer GRU (Singh and Sailaja Kumari 2026), a CNN–LSTM and a CNN–BiLSTM (Alamgir et al. 2026; Sharma et al. 2026; Chen et al. 2026), a vanilla Transformer encoder (Pan et al. 2026; Goyel et al. 2026) and the existing CNN–Transformer–BiGRU system with MI feature selection (top 60% of features) and ADASYN oversampling in window space (Samin et al. 2026). The *unsupervised* baselines are an isolation forest and a normal-only denoising autoencoder with the same convolutional architecture as SA-DAE (Lozano-Paredes et al. 2026; Tripathy et al. 2026). As in their original designs, all baselines use the raw cyber-physical channels without the physics residuals. The supervised baselines are given an open-set option: they reject an input as unknown when its maximum class probability (temperature-scaled for the neural networks) falls below a threshold set at the same budget δ on the same calibration windows as PG-ADTA-ID. The unsupervised baselines model normal operation, so they raise an alert when their anomaly score exceeds the $(1 - \delta)$

quantile of the normal calibration windows. Because they cannot name attacks, all of their alerts count as suspected zero-day alerts.

7 Results and Discussion

This section reports results on the test days (days 9–10) unless stated otherwise. All decision parameters were fixed in Stage B before the test results were evaluated. Values for PG-ADTA-ID are means over three training seeds, with standard deviations where shown; baselines are reported likewise where several seeds were run.

7.1 Training behaviour

Fig. 11 shows the training curves of the seed-0 model. The PGST reached its best validation macro-F1-score of 98.62% at epoch 5, and early stopping ended training at epoch 11. The validation loss lies below the training loss because class weighting, label smoothing and dropout act only in training. The SA-DAE loss fell steadily even while the curriculum raised the noise level from 0.02 to 0.15, and its clean validation error fell from 0.471 to 0.358 (standardised units). Training both branches took 6.7 min on the two-core CPU. The PGST has only 38,469 parameters and the SA-DAE 314,616.

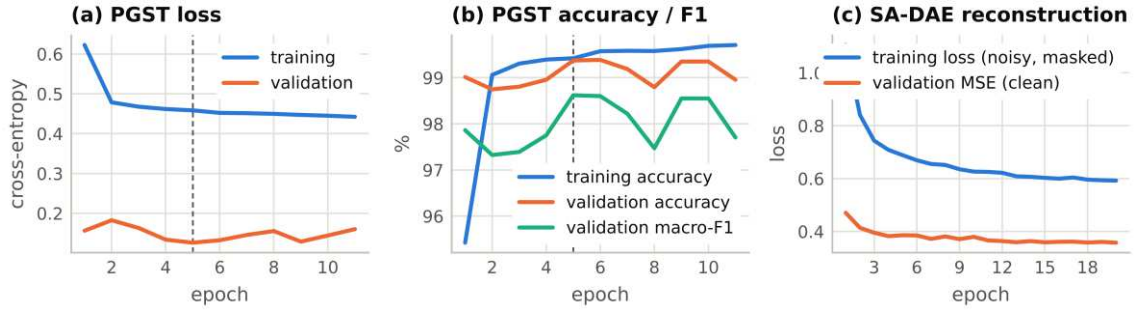

Fig. 11 Training behaviour of PG-ADTA-ID (seed 0): (a) PGST cross-entropy; (b) PGST training accuracy and validation accuracy and macro-F1-score; (c) SA-DAE training loss under the noise curriculum and clean validation reconstruction error. Dashed line: epoch with the best validation macro-F1-score, whose parameters were kept

7.2 Classification of known attacks

Table 5 reports the closed-set classification of the known classes. PG-ADTA-ID reached an accuracy of 99.30 ± 0.19%, a macro-precision of 99.65%, a macro-recall of 97.91% and a macro-F1-score of 98.74 ± 0.30% (MCC = 0.983, κ = 0.983). DoS and replay were recognised almost perfectly because they combine strong cyber signatures (packet floods, duplicate sequence numbers) with physical ones (stale or day-old telemetry). FDIA was the hardest known class (F1 = 95.22%). In the seed-0 model, 65 FDIA and 48 malware windows were taken for normal operation, against only 8 normal windows classified as an attack. Most misses are windows at the start of an episode: a window is labelled by its last step, so its earlier steps still show normal operation, and FDIA biases also ramp in over 5–20 steps. Whole episodes were rarely missed: the correct label was raised in 100.0% of the DoS, 100.0% of the FDIA, 100.0% of the replay and 91.7% of the malware episodes (Table 6).

Table 5 Closed-set classification of the known classes on the test days (PG-ADTA-ID, mean ± s.d. over three seeds)

Class	Precision (%)	Recall (%)	F1-score (%)	Test windows
Normal	99.15 ± 0.23	99.93 ± 0.02	99.54 ± 0.12	9,521
DoS	100.00	100.00	100.00	719
FDIA	99.19 ± 0.18	91.56 ± 0.12	95.22 ± 0.15	762
Replay	99.89 ± 0.15	100.00	99.95 ± 0.08	624
Malware	100.00	98.06 ± 2.37	99.01 ± 1.22	911
Macro average	99.65 ± 0.09	97.91 ± 0.49	98.74 ± 0.30	12,537

Open-set operation has a small price for the known classes. After the PC-OAF decision, 97.45% of all known-class windows kept their correct label, and 3.46% of the known-attack windows were flagged as unknown instead of being named; they still raise an alert, but a less specific one. The complete decision matrix is discussed in Section 7.3.

7.3 Detection of zero-day attacks

Table 6 breaks the results down by attack family, and Fig. 12 shows the ROC and precision–recall curves and the distribution of the novelty-risk score. At the operating point selected in Stage B, PG-ADTA-ID flagged 54.4% of the zero-day windows as unknown and raised an alert on 65.8% of them, at an FPR of 1.69%. Its score separated zero-day from known-class windows with an AUROC of 0.934 (area under the precision–recall curve, AUPR, 0.816). At the episode level, 90.7% of the zero-day episodes were flagged as unknown at least once, with a median delay of 70 s.

Table 6 Detection per attack family on the test days (mean ± s.d. over three seeds; delays pooled over seeds; 12 episodes per known and 10 per zero-day family). AUROC: windows of the family versus known-class windows

Family	Windows correctly handled (%)[a]	Windows alarmed (%)	Episodes detected (%)	Median delay (s)	AUROC vs known
DoS	99.1 ± 1.2	100.0	100.0	0	–
FDIA	90.5 ± 0.1	92.8 ± 0.1	100.0	60	–
Replay	97.8 ± 0.2	100.0	100.0	0	–
Malware	92.7 ± 0.1	100.0	91.7	0	–
Stealthy FDIA*	84.1 ± 0.6	84.2 ± 0.5	100.0	90	0.963
Time-sync (GPS)*	28.8 ± 7.5	83.4 ± 0.5	100.0	110	0.903
Command injection*	36.3 ± 9.6	36.7 ± 9.7	93.3 ± 9.4	20	0.963
Reconnaissance*	95.2 ± 2.3	100.0	100.0	20	0.988
AMI tampering*	25.7 ± 2.6	26.0 ± 2.7	60.0 ± 8.2	200	0.855

[a]Known families: windows given the correct class; zero-day families (*): windows flagged as unknown (ZD or UNC)

The withheld families behave very differently. *Reconnaissance* (95.2% of windows flagged) and the *stealthy FDIA* (84.1%) were detected reliably. Reconnaissance produces traffic unlike that of any known class, and the stealthy FDIA, although consistent with the AC power-flow equations within the PMU and SCADA data, breaks the balance with the independent AMI measurements, as Proposition 2 predicts. *Time-synchronisation* attacks raised an alert on 83.4% of their windows, but 54.6% of the windows were labelled as FDIA rather than as unknown. A ramp on PMU phase angles is in effect a false-data injection on phasors, so this label is physically reasonable and calls for the same response. *Command injection* was flagged on only 36.3% of its windows, yet 93.3% of its episodes were detected, with a median delay of 20 s. The unauthorised command leaves only a short burst of cyber evidence, after which the grid is in a legitimate post-trip state that resembles authorised switching; its high AUROC (0.963) shows that its windows still score above most known-class windows, only not above the threshold. *AMI tampering* was the hardest family (25.7% of windows and 60.0% of episodes). It lowers the metered demand of one to three buses by 10–40% and ramps in slowly, so its evidence is mainly physical, and Proposition 4 shows that with the selected physics weight such evidence cannot cross the threshold on its own. Together with the time-synchronisation attack, these two families explain most of the gap between window-level and episode-level rates.

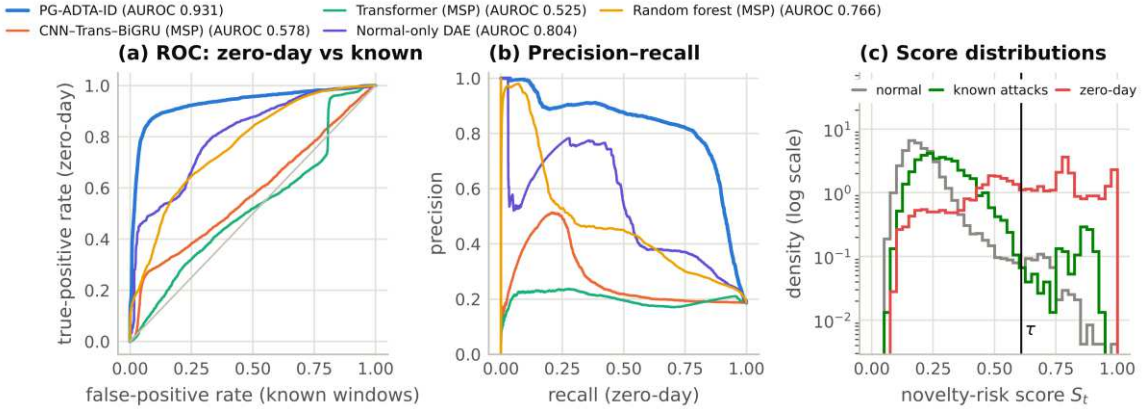

Fig. 12 Zero-day detection on the test days (seed 0): (a) ROC curves for separating zero-day windows from known-class windows, (b) precision–recall curves and (c) distribution of the novelty-risk score S_t of PG-ADTA-ID for normal, known-attack and zero-day windows, with the calibrated threshold τ

Fig. 13a shows the complete decision matrix, and Fig. 13b the detection delay of every episode. Known attacks were usually labelled correctly in the first window (median 0 s), whereas zero-day attacks needed a median of 70 s, because the stealthy FDIA, the time-synchronisation attack and AMI tampering build up over several minutes.

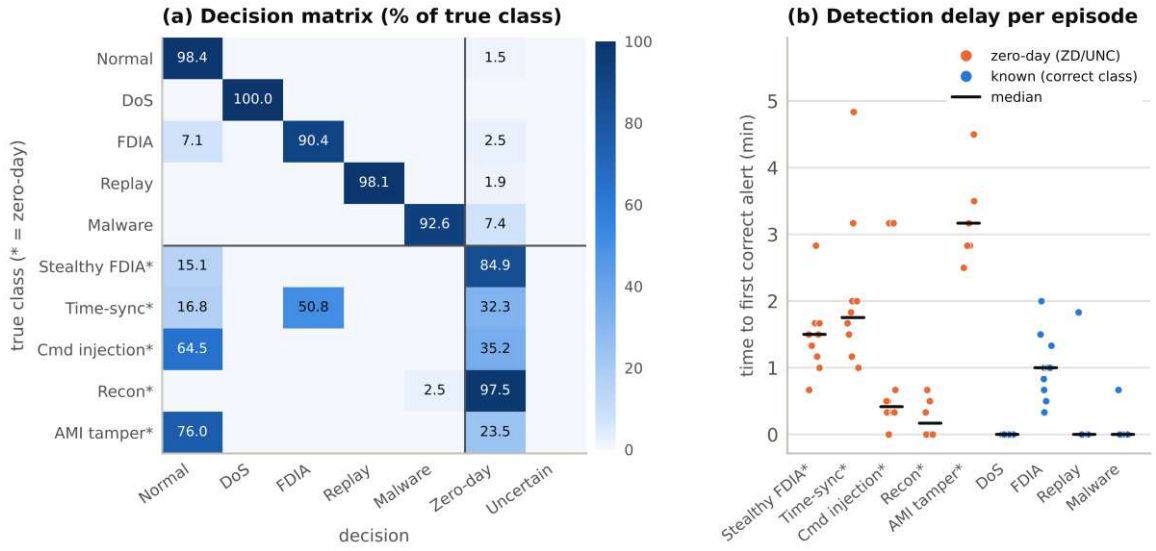

Fig. 13 (a) Decision matrix of PG-ADTA-ID on the test days (seed 0): rows are true classes (* = withheld zero-day families), columns are decisions including suspected zero-day (ZD) and uncertain (UNC), and entries are percentages of each true class. (b) Detection delay of each episode: time from attack onset to the first correct alert (zero-day: first ZD or UNC decision; known: first correct class label); bars mark the medians

The FPR exceeds the nominal budget $\delta = 0.0035$ for two reasons, which correspond to the two conditions of Proposition 5. The first is the in-sample calibration discussed in Section 7.8.2. The second is informative. Of the 151 false-positive windows of the seed-0 model, 140 (93%) were attributed to benign topology changes. Most of them (107) fell within a 53-min authorised outage of line 1–2 at midday. Line 1–2 is the most heavily loaded line of the network and had been switched out only once in the training days, for 15 min at night; the rest occurred during protection operations on lines 4–5 and 12–13. Such rare topologies are barely represented in the known world and violate the exchangeability of calibration and test windows. Almost all false positives were suspected zero-day alerts (1.52 of the 1.59 percentage points of the seed-0 FPR), which a control centre could clear automatically by correlating them with the switching schedule and protection logs of its energy management system.

7.4 Comparison with baseline detectors

Tables 7 and 8 compare PG-ADTA-ID with the eleven baselines on known and on unseen attacks, and Fig. 14 summarises the main metrics.

7.4.1 Classification of known attacks

PG-ADTA-ID achieved the highest macro-F1-score, 4.4 percentage points above the best baseline, XGBoost (94.35%), while the existing CNN–Transformer–BiGRU system reached 86.21%. All baselines process the raw channels, as in their original designs, and their main weakness is the FDIA class. Their FDIA F1-scores (seed 0) ranged from 29.2% (SVM) to 74.3% (Transformer), against 95.22% for PG-ADTA-ID, because many FDIA windows were taken for normal operation. Biases of a few percent on a handful of measurements are hard to see in raw telemetry that varies with the load, but they stand out in the Kirchhoff, Ohm's-law and AMI residuals (Section 7.5.1).

Table 7 Closed-set classification of the known classes on the test days (mean ± s.d. over three seeds; values without s.d.: one seed)

Method	Accuracy (%)	Precision (%)	Recall (%)	Macro-F1 (%)	MCC
PG-ADTA-ID (proposed)	99.30 ± 0.19	99.65 ± 0.09	97.91 ± 0.49	98.74 ± 0.30	0.983 ± 0.005
CNN–Transformer–BiGRU (Samin et al. 2026)	94.87 ± 0.27	96.72 ± 0.77	83.80 ± 1.08	86.21 ± 1.18	0.872 ± 0.007
Transformer	96.44 ± 0.49	98.13 ± 0.37	89.12 ± 1.94	92.23 ± 1.66	0.912 ± 0.012
CNN–BiLSTM	96.14 ± 0.06	98.29 ± 0.09	87.60 ± 0.20	90.54 ± 0.25	0.905 ± 0.002
CNN–LSTM	96.25	98.75	87.81	90.85	0.907
LSTM	96.15 ± 0.27	97.80 ± 0.79	88.09 ± 0.66	91.25 ± 0.43	0.904 ± 0.007
GRU	96.49	98.38	88.70	91.52	0.913
XGBoost	97.44 ± 0.13	99.27 ± 0.03	91.66 ± 0.44	94.35 ± 0.37	0.937 ± 0.003
Random forest	95.32 ± 0.16	98.83 ± 0.03	84.59 ± 0.53	87.06 ± 0.64	0.884 ± 0.004
SVM (RBF)	94.23	97.86	81.28	83.99	0.856

7.4.2 Detection of unseen attacks

The decisive differences appear on the withheld families. With softmax rejection, the supervised baselines flagged at most 8.3% of the zero-day windows as unknown (random forest), and their confidence separated zero-day from known windows poorly (AUROC 0.468–0.770). Two deep baselines even fell below 0.5, that is, they were *more* confident on the unseen attacks than on the known classes. Closed-set classifiers are often confident on inputs unlike their training data, so the zero-day windows were assigned confidently to normal operation or to a known attack. The alerts that these baselines did raise on them (7.7–28.3% of the windows) came mostly from such misclassifications. The existing CNN–Transformer–BiGRU ranked zero-day windows little better than chance (AUROC 0.601) and flagged 0.3% of the zero-day windows and 8.0% of the zero-day episodes. The normal-only DAE ranked them best among the baselines

(AUROC 0.804). Benign switching and faults, however, produce anomaly scores as large as those of subtle attacks, so at the same false-positive budget it flagged only 8.1% of the zero-day windows (isolation forest: 12.3%). Neither unsupervised detector can name an attack. PG-ADTA-ID raised the AUROC to 0.934 and flagged 54.4% of the zero-day windows and 90.7% of the zero-day episodes, while keeping the closed-set accuracy of a supervised classifier. Its open-set macro-F1-score (88.7%) was 16.1 points above the best baseline. These gains come at a higher FPR (1.69%) than that of the supervised baselines, which almost never reject anything. The ROC curves (Fig. 12a) show, however, that PG-ADTA-ID detects more zero-day windows at every false-positive level. When 1% of the known-class windows are flagged, for example, it detects 41% of the zero-day windows, against at most 15% for the baselines (random forest).

Table 8 Open-set detection on the test days (mean ± s.d. over three seeds; values without s.d.: one seed). ZDR: zero-day windows flagged as unknown; ZD alarm: zero-day windows raising any alert; AUROC/AUPR: zero-day versus known-class windows; ZD episodes: zero-day episodes flagged at least once

Method	ZDR (%)	ZD alarm (%)	FPR (%)	AUROC	AUPR	Open-set F1 (%)	ZD episodes (%)
PG-ADTA-ID (proposed)	54.4 ± 0.6	65.8 ± 1.6	1.69 ± 0.22	0.934 ± 0.004	0.816 ± 0.016	88.7 ± 0.4	90.7 ± 2.5
CNN–Transformer–BiGRU (Samin et al. 2026)	0.3 ± 0.1	22.0 ± 0.7	0.28 ± 0.18	0.601 ± 0.060	0.263 ± 0.035	65.7 ± 0.9	8.0 ± 1.6
Transformer	0.2 ± 0.2	27.3 ± 1.4	0.30 ± 0.18	0.530 ± 0.009	0.217 ± 0.013	69.6 ± 1.0	8.0 ± 6.5
CNN–BiLSTM	0.2 ± 0.1	24.8 ± 0.7	0.18 ± 0.03	0.468 ± 0.009	0.181 ± 0.001	68.7 ± 0.1	5.3 ± 1.9
CNN–LSTM	0.3	26.6	0.15	0.469	0.183	68.5	6.0
LSTM	0.6 ± 0.5	28.3 ± 0.8	0.26 ± 0.16	0.576 ± 0.021	0.272 ± 0.015	69.0 ± 0.5	4.7 ± 3.8
GRU	0.3	25.3	0.13	0.555	0.225	69.4	6.0
XGBoost	0.1 ± 0.1	20.0 ± 0.1	0.05 ± 0.01	0.671 ± 0.006	0.329 ± 0.019	72.6 ± 0.3	5.3 ± 4.7
Random forest	8.3 ± 0.9	19.7 ± 0.1	0.01 ± 0.00	0.770 ± 0.004	0.466 ± 0.006	70.7 ± 0.7	13.3 ± 0.9
SVM (RBF)	0.2	7.7	0.05	0.770	0.452	67.1	10.0
Normal-only DAE	8.1	8.1	1.45	0.804	0.530	15.1	10.0
Isolation forest	12.3	12.3	2.17	0.700	0.353	16.1	30.0

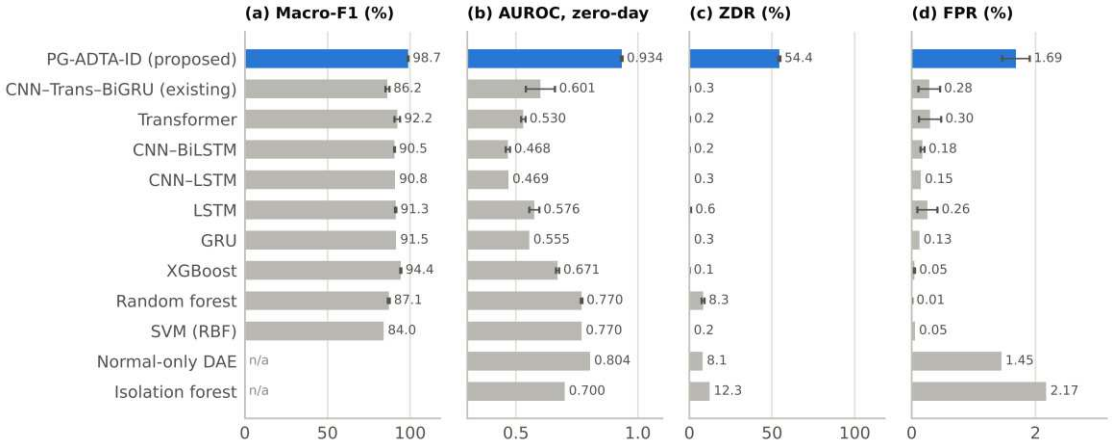

Fig. 14 Comparison with the eleven baselines on the test days: (a) known-attack macro-F1-score, (b) zero-day AUROC, (c) zero-day detection rate and (d) false-positive rate (error bars: s.d. over seeds; n/a: anomaly detectors cannot classify known attacks)

7.5 Ablation study

Table 9 isolates the contribution of each component of the seed-0 model. The first nine rows (the full model and eight variants) share the trained networks and change only the decision stage, so their differences are free of training noise. The variants marked † retrain the networks without a component. The AUROC is the threshold-free comparison. The window-level rates also reflect how each variant moves the calibrated threshold.

Table 9 Ablation study on the test days (seed 0)

Variant	Macro-F1 (%)	AUROC	ZDR (%)	ZD alarm (%)	FPR (%)	Open-set F1 (%)	ZD episodes (%)
Full PG-ADTA-ID	98.33	0.931	55.1	65.2	1.59	89.2	94.0
w/o epistemic (latent) uncertainty	98.33	0.927	57.3	69.4	1.94	88.8	90.0
w/o uncertainty term ($\alpha = 0$)	98.33	0.926	57.0	69.3	1.93	88.7	90.0
w/o reconstruction surprise ($\beta = 0$)	98.33	0.898	21.7	50.2	0.21	80.4	66.0
w/o physics surprise ($\gamma = 0$)	98.33	0.919	56.1	66.0	1.87	89.3	90.0
w/o freshness weighting	98.33	0.932	55.1	65.3	1.59	89.2	94.0
w/o class-conditional statistics	98.33	0.895	21.3	46.9	0.08	80.8	48.0
All 15 error groups (incl. operating-point-dependent)	98.33	0.938	63.0	73.3	2.08	90.0	96.0
PGST + softmax rejection only	98.33	0.586	3.2	34.2	0.18	74.6	34.0
w/o physics guidance (residuals, graph, c_t)[a]	92.69	0.736	32.5	39.7	2.27	80.3	38.0
w/o graph embedding[a]	98.73	0.939	61.7	69.4	1.81	90.6	92.0
SA-DAE trained on normal data only[a]	98.33	0.937	57.2	65.9	1.12	89.5	92.0
Plain DAE (no masking, fixed noise)[a]	98.33	0.935	59.7	67.2	1.97	89.9	94.0

[a]Networks retrained without the component

7.5.1 Physics guidance

Removing the physics residuals and summaries from the input, together with the graph embedding and the physics term, had the largest effect of all. The known-attack macro-F1-score fell from 98.33% to 92.69%, mainly because FDIA

windows were again taken for normal operation, and the zero-day AUROC fell from 0.931 to 0.736. The physical zero-day families became almost invisible. Only 1.0% of the stealthy-FDIA, 0.0% of the time-synchronisation and 0.0% of the AMI-tampering windows were flagged, and the episode detection rate fell from 94.0% to 38.0%. This variant flagged more command-injection windows (63.6%), because its threshold was then set by the cyber evidence alone, but this did not offset the losses elsewhere.

7.5.2 Evidence and calibration in PC-OAF

Among the three indicators, the reconstruction surprise contributes most. Without it ($\beta = 0$), the AUROC fell to 0.898 and the episode detection rate to 66.0%. Class-conditional statistics are just as important. Pooling the statistics of all known classes reduced the AUROC to 0.895 and the episode detection rate to 48.0%. A DoS window naturally reconstructs worse than a normal one, and only class-conditional statistics keep such expected deviations from masking unexpected ones. Removing the uncertainty term or the explicit physics term lowered the AUROC less (to 0.926 and 0.919). Each, however, cost 4 points of episode detection and raised the FPR slightly. Removing only the epistemic part of the uncertainty had almost the same effect as removing the whole term. The freshness weighting had no measurable effect, because packet loss outside DoS attacks is rare (0.3%) and the classifier recognises DoS windows reliably. Softmax confidence alone, the rejection rule of the baselines, reached an AUROC of only 0.586 with the same PGST. The advantage over the baselines therefore comes from the fusion, not from the classifier alone. Scoring all fifteen error groups, including the operating-point-dependent ones, raised the AUROC slightly on the stationary test days (0.938), at a higher FPR (2.08% against 1.59%). The restriction to groups that do not scale with the operating point is thus a design choice for robustness whose benefit this benchmark does not show. In the drift period these groups still shifted by up to 4.3 standard deviations (Section 7.6.1), because that period also changed the topology, the traffic baseline and the sensor noise.

7.5.3 Network design

The other retrained variants changed little. Without the graph embedding, with an SA-DAE trained on normal data only, and with an SA-DAE trained without masking and noise curriculum, the AUROC was 0.939, 0.937 and 0.935. These values lie within the range spanned by the three training seeds of the full model (0.931–0.939), so this benchmark shows no measurable gain from these three design choices. For the autoencoder the reason is clear. Once PC-OAF compares each error signature with the statistics of the predicted class, even a normal-only autoencoder yields class-specific signatures, which removes the drawback that known-world training was meant to address. The graph embedding adds little to a 14-bus network whose fused features already contain every nodal and branch residual. Whether it helps on larger networks remains to be tested.

7.6 Robustness to concept drift

The drift period (days 11–12) combines five abrupt changes: an 18% load increase, twice the PV capacity, a permanent outage of line 9–10, 50% more PMU noise and a shifted communication baseline with four new hosts. Table 10 compares three operating modes over this period and over the stationary test days. The *static* detector keeps the models and the threshold fixed after calibration. The *adaptive-threshold* mode updates only τ_t (Eq. (47)). *Full CDCL* also fine-tunes both branches with replay and recalibrates PC-OAF when drift is detected (Algorithm 5). Fig. 15 shows how the scores, the threshold and the FPR evolve.

7.6.1 Behaviour during the drift period

Without adaptation the detector became unusable after the regime change. The static detector flagged every normal window of the drift period (FPR 100.00%). The permanent line outage, the new traffic baseline and the noisier PMUs moved even the operating-point-independent error groups by up to 4.3 standard deviations of their calibration distribution (breaker context 4.0, traffic volume 4.3, Ohm's-law residuals 4.1). Adapting the threshold alone lowered the FPR to 36.94%. With stale models, however, the detector flickered between alarm and no alarm (492.0 false-alarm events per day) and named only 77.9% of the known-attack windows correctly. Full CDCL restored a usable detector. Drift was declared in 24 of the 288 blocks and triggered 7 short fine-tuning rounds. The first eight hours were an adaptation phase, because the noisier PMU measurements raised the physics surprise of most normal windows above the trust limit and trusted evidence accumulated slowly. Only 13% of the normal windows passed the gate before the first fine-tuning, after 6.7 h, and 21% over the first eight hours, against 95% on the calibration days and 91% after the adaptation phase. Afterwards the FPR stayed at or below 1% in 6 of the remaining 10 4-h periods (maximum 12.5%), and over the whole drift period it was 11.01%. CDCL also kept its sensitivity to unseen attacks: it flagged 57.0% of the zero-day windows and 84.0% of the zero-day episodes. The price of adaptation is that known attacks were often reported as unknown: only 20.2% of the known-attack windows were named correctly, although 96.1% raised an alert. CDCL recalibrates only the normal-class statistics from physically trusted data, whereas the known-attack statistics can only be updated from attacks that analysts have confirmed in the new regime.

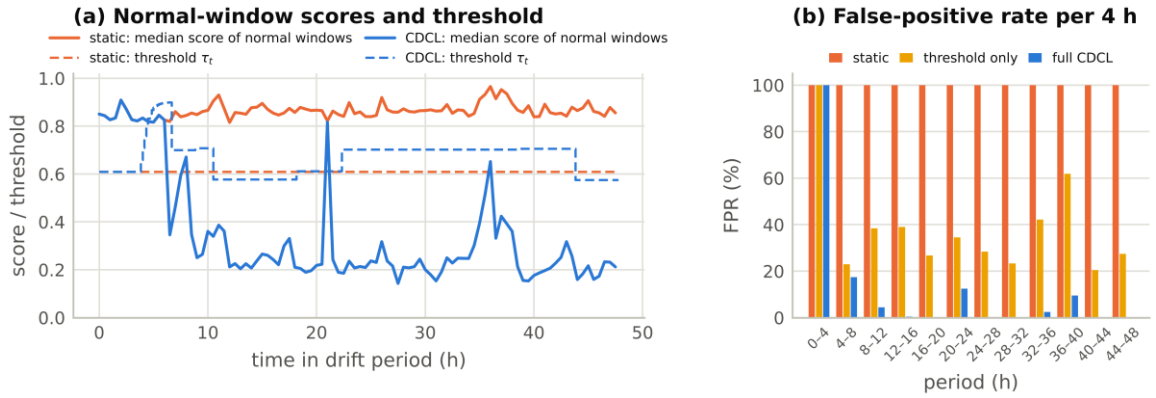

Fig. 15 Concept drift (days 11–12): (a) median novelty-risk score of normal windows (30-min bins) and the threshold τ_t for the static detector and for CDCL; (b) false-positive rate in consecutive 4-h periods for the three operating modes

7.6.2 Behaviour on the stationary test days

On the stationary test days CDCL should ideally change nothing. It declared drift only once, at the end of the 53-min outage of line 1–2, but the single fine-tuning that followed cost some zero-day sensitivity. The window-level ZDR fell from 55.1% to 50.3%, the AUROC from 0.931 to 0.912 and the share of zero-day episodes detected from 94.0% to 84.0%, at an almost unchanged FPR (1.48%). Such spurious adaptations after rare benign events are the main cost of CDCL. Checking drift declarations against the switching schedule would have prevented this one. Threshold adaptation alone is riskier. It declared drift at the end of the outages of lines 1–2 and 6–12, raised the threshold from 0.609 to 0.823 and never lowered it again, which is the risk identified after Proposition 6. Its ZDR on the stationary days therefore fell to 30.5%. Full CDCL avoids this ratchet because it re-anchors the threshold after every adaptation. Both details of the drift test in Section 5.6.2 proved essential. Without the trust-gated reference, the test fires even without drift, and without the effect-size floor ($D_{\min} = 0.58$) it fires on ordinary day-to-day differences.

Table 10 Static detection, adaptive threshold only and full CDCL on the drift period (days 11–12) and on the stationary test days (days 9–10) (seed 0). Known correct: known-class windows (normal and known attacks) given their correct label; false alarms/day: false-alarm events per day; FT runs: number of fine-tuning runs

Period	Mode	Known correct (%)	ZDR (%)	ZD alarm (%)	FPR (%)	AUROC	Open-set F1 (%)	False alarms/day	FT runs
Drift	Static	2.15	99.8	100.0	100.00	0.665	16.4	49.6	0
	Threshold only	66.30	56.7	83.8	36.94	0.665	66.8	492.0	0
	Full CDCL	73.90	57.0	62.9	11.01	0.768	41.7	46.1	7
Stationary	Static	97.58	55.1	65.2	1.59	0.931	89.2	10.0	0
	Threshold only	97.81	30.5	55.3	1.43	0.931	82.1	9.0	0
	Full CDCL	97.65	50.3	61.4	1.48	0.912	88.3	10.5	1

7.7 Optimisation results

This section reports the outcome of the two optimisation stages of Section 5.7.2: the network hyperparameters selected in Stage A and the decision parameters selected in Stage B.

7.7.1 Stage A: network hyperparameters

Bayesian optimisation evaluated 14 PGST configurations in 13 min (Fig. 16b), each trained briefly to keep the search affordable. The validation macro-F1-score varied little between configurations (98.36–98.60%), so model size and latency decided the choice. The selected configuration ($d_m = 32$, $L_e = 3$, $h = 4$) reached a validation macro-F1-score of 98.56% in these short runs (98.62% after full training, Section 7.1), practically the same as the default configuration ($d_m = 64$, $L_e = 2$; 98.60%), with 38,469 instead of 96,293 parameters and 74 instead of 104 µs of batched inference time per window.

7.7.2 Stage B: decision parameters

Table 11 compares the three optimisers on the LOCO pseudo-open-set, each with a budget of 120 evaluations and five independent runs, and Fig. 16a shows their convergence. BO achieved the highest mean objective (J = 0.6018 ± 0.0011, against 0.5989 ± 0.0010 for PSO and 0.5928 ± 0.0045 for GA) and the best single run. PSO converged quickly but to a lower plateau, and GA was the slowest and the most variable. The tree-structured surrogate of BO suits this problem, which has only five parameters but a noisy, plateau-like objective. The selected parameters, (α, β, γ) = (0.23, 0.63, 0.14), $\delta = 0.0035$ and $\eta = 0.50$, raised the objective from 0.5134 for equal weights with $\delta = 0.01$ to 0.6033. Averaged over the four folds, the held-out class was flagged in 82.9% of its windows (equal weights: 77.0%) at an FPR of 1.19% (equal weights: 1.76%). Replay was the hardest pseudo-novel class (53.7%), because replayed telemetry is physically consistent and only its sequence and timing anomalies are unusual.

Table 11 Stage-B optimisation of the decision parameters on the LOCO pseudo-open-set (five runs of 120 evaluations per optimiser)

Optimiser	Objective J (mean ± s.d.)	Best J	Evaluations to 99%	Time per run (s)	Best $(\alpha, \beta, \gamma, \delta, \eta)$
BO	0.6018 ± 0.0011	0.6033	35	2	(0.23, 0.63, 0.14, 0.0035, 0.50)
PSO	0.5989 ± 0.0010	0.6008	27	2	(0.23, 0.63, 0.13, 0.0032, 0.78)
GA	0.5928 ± 0.0045	0.5987	68	2	(0.16, 0.60, 0.24, 0.0048, 0.69)

Two features of the selected parameters matter for the interpretation of the test results. First, the objective penalises each 2% of FPR as heavily as a 37.5-point loss of ZDR (λ_3/λ_2 = 0.375), so Stage B chose a conservative budget δ. Operators who accept more alarms can move along the trade-off curve of Section 7.8.1 without retraining. Second, the SA-DAE received the largest weight and the explicit physics term the smallest. This does not make the physics evidence unimportant. The error signature of the SA-DAE already contains the Kirchhoff, Ohm's-law and AMI residual groups, so much of the physical evidence reaches the score through a_t. The explicit term c_t adds most for manipulations that are visible *only* through the AMI–SCADA balance. None of the four known classes is of this kind, so the pseudo-open-set could not reward a larger γ. AMI tampering, the weakest test family, is exactly such a case (Section 7.10).

7.8 Sensitivity analysis

The sensitivity of PG-ADTA-ID to its decision parameters, to its calibration protocol and to the window length was assessed on the seed-0 model, with all other settings fixed at their selected values.

7.8.1 FPR budget

Fig. 16c shows how the budget δ trades zero-day detection against false alarms when all other parameters are fixed (seed 0). Raising δ from the selected 0.0035 to 0.02 lifted the window-level ZDR from 55.1% to 78.0% and the open-set macro-F1-score to its maximum of 91.4%. The FPR rose to 2.97%, 7.9% of the known-attack windows were flagged as unknown, and false-alarm events rose to 22 per day. The episode detection rate reached 96.0% at δ = 0.005 and stayed there up to δ = 0.08 (98.0% at δ = 0.12), so larger budgets mainly add coverage within episodes that are already detected. At δ = 0.12 the ZDR reached 89.7%, but the detector raised 128 false-alarm events per day and flagged 34.9% of the known attacks as unknown. The realised FPR grew much more slowly than δ, from 1.17% to 8.31% over a 60-fold range, because most of it comes from the few rare topology events identified in Section 7.3. An operator can therefore choose the operating point after deployment, without retraining. δ between 0.005 and 0.02 is a reasonable range for this benchmark.

7.8.2 Calibration protocol

In the main experiments, the temperature, the class statistics and the threshold are all estimated on the two calibration days, so τ_0 is an in-sample quantile. Only 0.36% of the calibration windows exceed it, close to δ, whereas 2.07% of the known-class test windows do (mean of three seeds). A strict split removes this optimism and satisfies the first condition of Proposition 5. With the statistics fitted on day 7 and τ_0 set on day 8, the realised FPR stayed below δ in 22 of 24 combinations of seed and budget (δ = 0.0035–0.05). At the selected δ, however, the threshold rose, and the FPR fell to 0.16% and the ZDR to 15.8% (38% of the zero-day episodes; AUROC 0.919, as the statistics then come from one day only). With δ = 0.03, the strict split reproduced the reported operating point (FPR 1.48%, ZDR 54.7%, 90% of the episodes). Both protocols therefore lie on the same trade-off curve. The selected δ is best read as a setting of the in-sample protocol, and operators who want the split-conformal bound should use the strict split with a correspondingly larger budget, about 0.03 here.

7.8.3 Confidence threshold

The parameter η had almost no effect. The temperature-scaled PGST is rarely uncertain between known classes, so even with η = 0.9 only 0.40% of the windows were declared uncertain, and the ZDR rose only from 55.1% to 56.6%. In this benchmark the UNC branch acts as a safety net for rare ambiguous windows, and almost all novelty decisions come from the fused score.

7.8.4 Fusion weights

Fig. 16d maps the zero-day AUROC over the simplex of fusion weights. No single indicator is competitive: uncertainty alone reached 0.836, reconstruction surprise alone 0.907 and physics surprise alone 0.762. Mixtures of all three form a broad plateau, on which 15 of the 36 grid points with all three weights of at least 0.1 reached an AUROC of at least 0.93. The highest AUROC (0.942) was obtained with a larger physics weight, (α, β, γ) = (0.1, 0.4, 0.5). At the same budget, however, this weighting flagged only 34.3% of the zero-day windows. It raised the share of AMI-tampering windows flagged to 41.4% but cut that of reconnaissance to 4.1% and of command injection to 4.6%. A high AUROC alone is therefore not a sufficient criterion, which supports the operating-point objective used in Stage B.

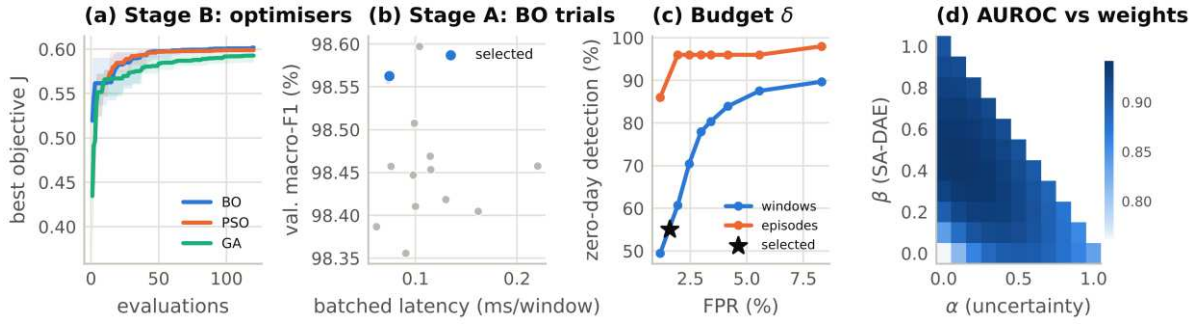

Fig. 16 Optimisation and sensitivity: (a) Stage-B convergence of BO, PSO and GA on the LOCO pseudo-open-set (mean of five runs; shaded: range); (b) Stage-A BO trials, validation macro-F1-score against batched latency; (c) window-level ZDR and episode detection rate against the test FPR as the budget δ varies (star: selected δ); (d) zero-day AUROC over the weight simplex ($\gamma = 1 - \alpha - \beta$)

7.8.5 Window length

Windows of 10, 20 and 30 steps gave similar known-attack macro-F1-scores (98.75%, 98.33% and 98.95%) and zero-day AUROCs (0.936, 0.931 and 0.929). Shorter windows react faster, with a mean zero-day delay of 81 s for $T = 10$ against 109 s for $T = 30$, but they produce more false positives (FPR 2.14% against 1.59% for $T = 20$). Longer windows delay the alarm and lowered the episode detection rate to 80.0%. $T = 20$ (200 s) is a good compromise.

7.9 Computational efficiency and scalability

Two practical questions are addressed here: whether PG-ADTA-ID is fast enough for real-time operation on modest hardware (Section 7.9.1), and whether it scales to a larger network (Section 7.9.2).

7.9.1 Computational efficiency

Table 12 lists the model size, training time and inference cost on the two-core CPU. PG-ADTA-ID is not the lightest model. Its two networks have 353,085 parameters, most of them in the SA-DAE (314,616), whose input and output layers grow with the number of fused features. The PGST alone (38,469) is smaller than every deep baseline. Training both branches took 403 s, about half the training time of the CNN–Transformer–BiGRU (782 s). Scoring one window with both networks took 4.30 ms. The complete decision, which also assembles the fused window, computes the three PC-OAF indicators and takes the three-way decision, took 7.23 ms, against 0.86–3.33 ms for the single-network baselines. A new window arrives only every 10 s, so this cost is negligible. In batches of 256 windows, the complete pipeline processed 3,134 windows per second.

Table 12 Model size, training time and inference cost on the IEEE 14-bus benchmark (last row: IEEE 118-bus system; two-core CPU, no GPU). Latency: one window at a time; throughput: batches of 256 windows

Model	Parameters	Training time (s)	Latency, batch 1 (ms)	Throughput (windows/s)
PG-ADTA-ID (PGST + SA-DAE)	353,085	403	4.30	3,884
PG-ADTA-ID (end-to-end decision)	353,085	–	7.23	3,134
CNN–Transformer–BiGRU	106,565	782	3.33	9,213
Transformer	88,325	151	1.52	9,391
CNN–BiLSTM	104,709	40	1.34	10,835
CNN–LSTM	71,109	50	0.98	16,540
LSTM	100,421	63	0.86	13,033
GRU	75,397	83	1.68	18,332
Normal-only DAE	249,251	237	1.26	10,034
PG-ADTA-ID, 118-bus system (end-to-end)	1,963,896	1,495	8.82	525

7.9.2 Scalability to the IEEE 118-bus system

To test whether the framework scales to a larger network, an 8-day benchmark was generated on the IEEE 118-bus system (186 branches, 54 generators, 99 metered load buses and 15 PV plants) with the same measurement layers, benign disturbances and attack families. Days 1–4 were used for training, day 5 for validation, day 6 for calibration and days 7–8 for testing, again with the five zero-day families withheld. CP-DAP produces $d = 2{,}291$ fused features per time step (versus 280 for the 14-bus system). PG-ADTA-ID was trained once with 12 epochs per branch in 25 min, the decision parameters selected on the 14-bus system were reused unchanged, and the baselines were retrained with the same protocol (single seed). The PGST fitted the training days almost perfectly (99.9% training accuracy) but reached a validation macro-F1-score of only 92.2%, so generalising across days is harder on the larger network, which also has one training day fewer.

Table 13 shows that known-attack classification transferred well. PG-ADTA-ID reached a macro-F1-score of 95.31%, against 91.27% for the best baseline (CNN–Transformer–BiGRU). Zero-day detection, by contrast, was much weaker than on the 14-bus system. The AUROC fell to 0.780, and PG-ADTA-ID flagged 18.4% of the zero-day windows and 42.0% of the zero-day episodes, at an FPR of 0.30%. It still ranked zero-day windows better than every baseline (best: normal-only DAE, 0.713), and no baseline detected more than 20% of the zero-day episodes. The per-family rates show where the sensitivity was lost. Reconnaissance (46.7% of windows flagged) and the time-synchronisation attack (36.0%) were still detected, but the stealthy FDIA (12.8%), command injection (0.1%) and AMI tampering (0.0%) were rarely or never flagged. Each of these attacks touches one line or one to three buses, whatever the size of the network, so on the 118-bus system its traces are diluted in the system-wide group statistics of the SA-DAE and the physics profile. Scoring each grid zone separately, re-running Stage B on the larger system and training on more days are the natural remedies. The computational cost, in contrast, scaled gracefully. Although d grew eightfold, a complete decision took 8.82 ms (Table 12).

Table 13 Results on the IEEE 118-bus system (test days 7–8, single seed). The decision parameters were transferred from the 14-bus system without re-tuning

Method	Accuracy (%)	Macro-F1 (%)	AUROC	ZDR (%)	ZD alarm (%)	FPR (%)	ZD episodes (%)
PG-ADTA-ID	97.64	95.31	0.780	18.4	35.4	0.30	42.0
CNN–Transformer–BiGRU (Samin et al. 2026)	95.49	91.27	0.505	0.5	24.0	2.41	20.0
Transformer	94.13	87.86	0.490	0.3	25.2	2.61	10.0
Random forest	87.72	63.22	0.622	5.8	8.6	0.00	10.0
Normal-only DAE	–	–	0.713	8.7	8.7	0.67	14.0

7.10 Interpretability of the decisions

Fig. 17a projects the PGST embeddings of the test windows with t-SNE. The known classes form compact and well-separated clusters, which explains the high closed-set accuracy. Reconnaissance forms a cluster of its own, and the time-synchronisation windows lie next to the FDIA cluster, which matches their frequent FDIA label. Command-injection and AMI-tampering windows, by contrast, fall inside the normal cluster. The classifier has no representation that separates them from normal operation, so their detection must come from the reconstruction and physics evidence.

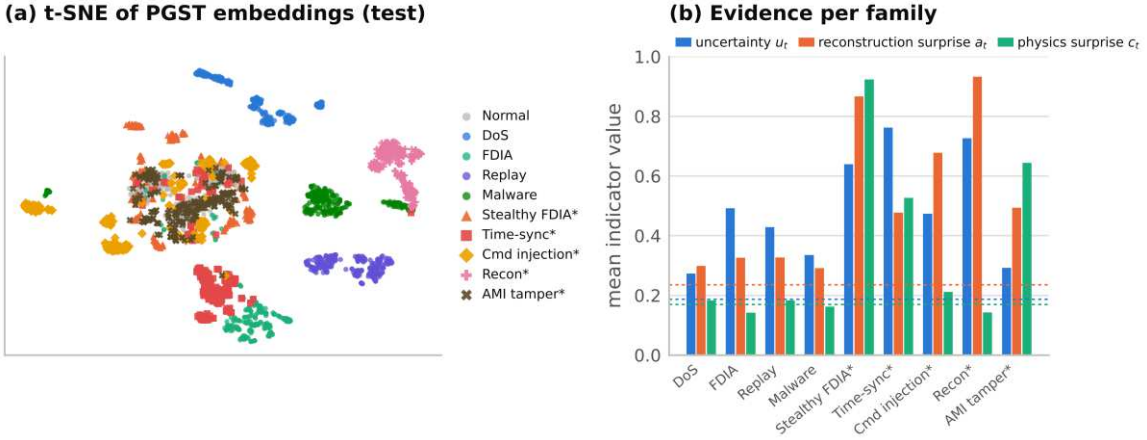

Fig. 17 Interpretability of PG-ADTA-ID (seed 0, test days): (a) t-SNE projection of the PGST embeddings (* = withheld zero-day families); (b) mean value of the three PC-OAF indicators for each attack family, with dashed lines giving their mean over normal windows

Fig. 17b shows which evidence drives the decision for each family. Over normal windows the three indicators average 0.19 (u_t), 0.24 (a_t) and 0.17 (c_t), and the known attacks raise them only moderately because they are scored against the statistics of their own class. The zero-day families leave distinct fingerprints. The stealthy FDIA raises all three indicators, and its physics surprise is the highest of all families (0.92). Reconnaissance raises the reconstruction surprise (0.93) and the uncertainty (0.73) but not the physics surprise (0.14), as expected for a purely cyber attack. The time-synchronisation attack mainly raises the uncertainty (0.76). AMI tampering is the opposite case: its physics surprise is high (0.64), whereas its uncertainty (0.29) stays close to normal and its reconstruction surprise (0.49) is only moderately raised. With the small physics weight selected in Stage B ($\gamma = 0.14$), the fused score of most AMI-tampering windows therefore stays below the threshold although their physical evidence is clear, exactly as Proposition 4 predicts. A larger physics weight would flag more AMI tampering but lower the sensitivity to the cyber-only families (Section 7.8.4); without prior knowledge of which families will appear, the weighting found in Stage B is a reasonable compromise. This per-indicator view also gives operators a direct explanation of every alert, because it shows whether an alert was caused by unusual traffic, an unfamiliar reconstruction or a physical inconsistency.

7.11 Statistical significance

Across the three training seeds, PG-ADTA-ID varied little. The standard deviations were 0.30 points for the known-attack macro-F1-score, 0.004 for the AUROC, 0.6 points for the ZDR and 0.22 points for the FPR, all small compared with the differences to the baselines. Consecutive windows overlap and are strongly autocorrelated, so tests that treat windows as independent would overstate significance. A paired block bootstrap was therefore used. The 15,399 non-transition test windows were cut into 43 non-overlapping blocks of 360 consecutive windows (about one hour each), and 2,000 replicates were drawn by resampling whole blocks with replacement, using the same blocks for PG-ADTA-ID and each baseline. Table 14 reports the differences with their 95% percentile intervals. Against every baseline, the gains in AUROC, ZDR and open-set macro-F1-score were significant, and so was the gain in known-attack macro-F1-score over every supervised baseline: none of these intervals includes zero. The FPR is the exception. PG-ADTA-ID raised significantly more false positives than six of the nine supervised baselines, by 1.4–1.6 percentage points. The difference was not significant against the CNN–Transformer–BiGRU, Transformer and LSTM or the normal-only DAE, and the isolation forest had a higher FPR than PG-ADTA-ID, although not significantly. This cost is the price of rejecting unknown inputs at all. The supervised baselines achieve their low FPR by almost never declaring anything unknown. The seed-level results lead to the same conclusions. For the six baselines trained with three seeds, Welch's t-tests on the per-seed metrics found every gain in macro-F1-score, AUROC, ZDR and open-set macro-F1-score significant ($p \leq 0.028$), and also the higher FPR of PG-ADTA-ID ($p \leq 0.009$).

Table 14 Paired block-bootstrap comparison with each baseline on the test days (seed 0): difference PG-ADTA-ID minus baseline with its 95% percentile interval (pp = percentage points)

Baseline	ΔMacro-F1 (pp)	ΔAUROC	ΔZDR (pp)	ΔFPR (pp)	ΔOpen-set F1 (pp)
CNN–Transformer–BiGRU	+11.6 [8.4, 14.8]*	+0.354 [0.287, 0.426]*	+54.7 [44.9, 63.7]*	+1.13 [−0.31, 3.00]	+22.9 [17.9, 27.5]*
Transformer	+4.6 [2.3, 7.5]*	+0.406 [0.341, 0.471]*	+54.9 [45.0, 64.0]*	+1.18 [−0.20, 2.89]	+18.7 [13.9, 23.7]*
CNN–BiLSTM	+7.6 [2.3, 14.2]*	+0.454 [0.374, 0.530]*	+54.9 [45.1, 64.1]*	+1.44 [0.01, 3.21][a]	+20.4 [14.1, 26.6]*
CNN–LSTM	+7.5 [2.1, 14.2]*	+0.462 [0.398, 0.526]*	+54.8 [44.9, 63.9]*	+1.44 [0.04, 3.23][a]	+20.7 [14.3, 26.9]*
LSTM	+6.5 [2.3, 11.1]*	+0.353 [0.280, 0.421]*	+55.0 [45.1, 64.2]*	+1.34 [−0.11, 3.15]	+19.8 [13.8, 25.3]*
GRU	+6.8 [1.6, 13.0]*	+0.377 [0.305, 0.449]*	+54.8 [44.8, 63.9]*	+1.46 [0.06, 3.24][a]	+19.8 [13.0, 26.2]*
XGBoost	+4.0 [1.7, 6.3]*	+0.252 [0.191, 0.312]*	+54.9 [45.1, 64.0]*	+1.55 [0.18, 3.29][a]	+16.6 [11.6, 21.3]*
Random forest	+10.9 [5.6, 16.9]*	+0.166 [0.120, 0.214]*	+45.5 [36.0, 55.7]*	+1.58 [0.19, 3.32][a]	+17.5 [11.1, 23.7]*
SVM (RBF)	+14.3 [7.2, 19.9]*	+0.162 [0.113, 0.212]*	+54.9 [45.1, 64.1]*	+1.53 [0.16, 3.29][a]	+22.1 [16.0, 26.7]*
Normal-only DAE	–	+0.127 [0.088, 0.175]*	+47.0 [35.0, 57.4]*	+0.14 [−0.51, 0.78]	+74.1 [69.6, 77.5]*
Isolation forest	–	+0.231 [0.176, 0.288]*	+42.8 [30.2, 55.2]*	−0.59 [−1.35, 0.05]	+73.1 [68.7, 76.7]*

*The interval excludes zero in favour of PG-ADTA-ID

[a]The interval shows a significantly higher FPR for PG-ADTA-ID

7.12 Discussion

This section places the results in the context of the literature and of grid operation. It compares PG-ADTA-ID with reported detectors (Section 7.12.1), relates it to model-based detection (Section 7.12.2), outlines how its outputs can drive a response (Section 7.12.3) and closes with the main threats to validity (Section 7.12.4).

7.12.1 Comparison with the literature

High closed-set accuracies, often above 97%, are reported for smart-grid and industrial-IoT intrusion detectors such as non-local attention networks for SCADA traffic (Yilmaz et al. 2026), graph–Transformer edge models for topology attacks in microgrids (Gowtham Chakravarthy et al. 2026) and EV-charging detectors based on transfer learning or wavelet scalograms (Thiruppathy Kesavan et al. 2026; Abu Nassar and Morsi 2026). Multi-task and scalable deep architectures continue to raise detection performance (Yang et al. 2026a; Vasanthi et al. 2026). The known-attack performance of PG-ADTA-ID is in the same range, but Section 7.4.2 shows that a high closed-set score says little about how a detector behaves when the attack is new. In the present benchmark, the existing CNN–Transformer–BiGRU reached a macro-F1-score of 86.21%, close to the 0.86 reported by its authors on the MSU dataset (Samin et al. 2026), yet it flagged only 8.0% of the zero-day episodes. Zero-day detectors developed for IT networks, such as VAE-AttNet with its dynamic 99th-percentile threshold (Yang et al. 2026b) and similarity-learning detectors evaluated across datasets (Hossain et al. 2026), rely on deviations in a learned representation alone, whereas unsupervised spatiotemporal graph models of grid communications (Su et al. 2026) and the fusion of process, supervisory and communication data in water networks (Behrooz and Ilbeigi 2026) point in the same direction as PG-ADTA-ID: the evidence of an unknown attack is spread across the physical and cyber layers, and the ablations show that both layers are needed.

7.12.2 Relation to model-based detection

Model-based FDIA detectors offer strong guarantees for the attacks they model. Examples include observer-based detection with conformal thresholds (Sun et al. 2026), linear detectors with robustness certificates for nonlinear grids (Guo et al. 2026), numerically stable Kalman-filter state estimation with geometric inconsistency tests (Xu et al. 2026) and Gaussian belief propagation for unbalanced feeders (Wei et al. 2026). Graph neural networks (Kfouri et al. 2026) and optimised extreme learning machines (Xi et al. 2026b) also identify or localise the corrupted measurements. These methods complement PG-ADTA-ID, which applies the distribution-free quantile principle of conformal prediction (Sun et al. 2026) (Proposition 5) to a learned score that also covers attacks leaving the physics untouched, such as reconnaissance and malware. Proactive moving-target defence (Yan et al. 2026) would add a deliberately induced source of physical inconsistency and so strengthen the evidence against stealthy FDIAs, which in the present benchmark came from the independent AMI path.

7.12.3 From detection to response

The three-way output of PG-ADTA-ID is designed to feed mitigation. A known-attack label can trigger a specific countermeasure. A ZD or UNC alert can switch controllers to conservative, prediction-based modes such as the H∞ fallback used in wind energy systems (Kumar et al. 2026), activate secure learning-based control in microgrids (Sepehrzad et al. 2026; Tasnim et al. 2026), or start trust-aware restoration of distributed state estimation (Nasiri et al. 2026). Once an analyst has confirmed a new attack family, class-incremental learning with diversity-aware replay (Khaldi et al. 2026) or few-shot learning (Abdulganiyu et al. 2026; Martinez-Lopez et al. 2026) can turn it into a known class from a few labelled episodes. CDCL deliberately stops short of this step: it adapts only to physically trusted normal behaviour, which limits poisoning but keeps an analyst in the loop. Protocol-level evidence such as IEC 61850 GOOSE and Sampled Value monitoring (Thao et al. 2026; Cibin et al. 2026b) would sharpen the detection of command injection, whose cyber evidence in the flow features used here lasts only a few steps. Clustering-guided federated learning (García-Sáez et al. 2026) offers a route to deployment across substations with heterogeneous traffic. Generative models are already being combined with intrusion detection in cyber-physical power systems (Aygül et al. 2026) and can synthesise covert, physically consistent attacks (Mohammadpourfard et al. 2026), so they could supply realistic scenarios for pre-deployment testing.

7.12.4 Threats to validity

Five limitations qualify the results. First, all results come from a co-simulation whose AC power flow, noise, packet loss, natural disturbances and drift are realistic but cannot reproduce every artefact of field data, and the gap between simulation and practice remains a main obstacle to adoption (Nasir et al. 2026b). Second, the zero-day families were designed by the authors. To limit optimistic bias, their parameters were fixed before any model was trained, they were never used for tuning, and the stealthy FDIA was made consistent with the AC power-flow equations. Third, window-level metrics depend on the labelling convention (a window is labelled by its last step, and transition windows are excluded), which is why episode-level results are also reported, and the nominal budget δ depends on the calibration protocol (Section 7.8.2). Fourth, most experiments use the 14-bus system and three seeds. The 118-bus experiment shows that the computation scales and that known-attack classification transfers, but also that the zero-day sensitivity to localised attacks drops on a larger network unless the decision stage is adapted to it. Fifth, the drift test of CDCL was revised once. Its first version compared the trusted buffer with all normal calibration windows and fired even on the stationary days. The revised test (Section 5.6.2) uses a trust-gated reference and an effect-size floor set on the validation day, so it uses no test data, but the revision was prompted by behaviour observed on the test days.

8 Conclusion and Future Work

This paper presented PG-ADTA-ID, an adaptive hybrid deep-learning framework that classifies known cyberattacks and flags previously unseen (zero-day) attacks in next-generation smart grids. Five components work together: a physics-guided spatiotemporal Transformer that learns from fused PMU, SCADA, AMI and network-traffic evidence enriched with Kirchhoff, Ohm's-law and metering residuals and a breaker-aware graph embedding; a self-supervised denoising autoencoder that models the known world on channels that do not scale with the operating point; a physics-consistent open-set fusion with a calibrated three-way decision; a physics-gated, drift-aware continual-learning mechanism; and a two-stage multi-objective optimisation that never touches zero-day data. The theoretical analysis showed that the physics residuals are invariant to the operating point, that power-flow-consistent injections remain visible through independent metering, that the fused indicators share a common probabilistic scale, that the calibrated threshold controls false alarms under exchangeability, and that the influence of poisoning on adaptation is bounded by the trust gate, the effect-size floor and the step limit.

On a cyber-physical co-simulation of the IEEE 14-bus system with four known and five withheld attack families, natural disturbances and chronologically disjoint splits, PG-ADTA-ID achieved 99.30% accuracy, 99.65% precision, 97.91% recall and a 98.74% macro-F1-score on known attacks. It detected 90.7% of the zero-day attack episodes with a median delay of 70 s, flagged 54.4% of the zero-day windows as unknown and raised an alarm on 65.8% of them, at a false-positive rate of 1.69% and with a zero-day AUROC of 0.934, above all eleven baselines (best: 0.804). The CNN–Transformer–BiGRU system reached only 0.601 with softmax-based rejection and flagged 8.0% of the zero-day episodes. Physics guidance, the reconstruction evidence and class-conditional calibration contributed most to these results, whereas the graph embedding and training the autoencoder on known attacks brought no measurable gain on this benchmark. Under abrupt concept drift, CDCL reduced the false-positive rate from 100% to 11.0% while keeping the zero-day detection rate at 57.0%, although on stationary days a single spurious adaptation after a benign line outage lowered the share of zero-day episodes detected from 94.0% to 84.0%. A complete decision took 7.23 ms per window on a two-core CPU. On the IEEE 118-bus system (2,291 fused features), PG-ADTA-ID kept a macro-F1-score of 95.31% on known attacks at 8.82 ms per decision, but its zero-day AUROC fell to 0.780, because attacks on one line or a few buses are diluted in system-wide statistics.

Some limitations point to future work. The results come from a co-simulation, so validation on hardware-in-the-loop testbeds and field data from digital substations is the natural next step (Cibin et al. 2026a; Bader et al. 2026). Zero-day attacks that closely resemble a known class, such as GPS time-synchronisation spoofing, are often reported as that class; command injection is detected at its onset rather than throughout; weak AMI tampering often stays below the threshold; and most false alarms occurred during rare benign topology changes, which suggests correlating alerts with the switching schedule of the energy management system. On the 118-bus system, attacks on one line or a few buses were diluted in system-wide statistics, so the decision stage must become zone-aware before it is used on large networks. Future work will (i) compute the PC-OAF indicators per grid zone, add host- and protocol-level provenance features (Zhang and Li 2026) and attack-localisation outputs (Aboelmagd et al. 2026; Xi et al. 2026a), (ii) extend CDCL to federated, privacy-preserving deployment across substations and harden it further against poisoning (Chen et al. 2026; De Santis et al. 2026; Liu et al. 2026), (iii) study robustness to adaptive and adversarial attackers (Mohammad et al. 2026; Mohammadpourfard et al. 2026), and (iv) explore few-shot incorporation of newly confirmed attack families (Lu et al. 2026; Yin et al. 2026) and natural-language explanations for operators.

Statements and Declarations

Funding. No funding was received for conducting this study.

Competing interests. The authors have no competing interests to declare that are relevant to the content of this article.

Ethics approval and consent to participate. Not applicable.

Consent for publication. Not applicable.

Data availability. The simulated datasets generated and analysed during the current study are available from the corresponding author on reasonable request.

Materials availability. Not applicable.

Code availability. The co-simulation generator, the preprocessing (CP-DAP), the PG-ADTA-ID implementation and the evaluation scripts are available from the corresponding author on reasonable request.

Author contribution. Conceptualization: Sharmila N, Mallikarjunaswamy S; Methodology: Sharmila N, Mallikarjunaswamy S; Formal analysis: Sharmila N, Mallikarjunaswamy S; Software: Sharmila N, Chethana H S; Investigation: Chethana H S, Sharmila N; Data curation: Chandana C, Prakyath D; Validation: Chethana H S, Chandana C, Prakyath D; Visualization: Chandana C, Prakyath D; Writing – original draft: Sharmila N, Mallikarjunaswamy S; Writing – review and editing: Chethana H S, Mallikarjunaswamy S, Sharmila N, Chandana C, Prakyath D; Supervision and project administration: Mallikarjunaswamy S. All authors read and approved the final manuscript.

References

Abdulganiyu OH, Fadi O, Moukafih Y, et al (2026) Explainable attention based few shot LSTM for intrusion detection in imbalanced cyber physical system networks. Sci Rep 16(1):7217. https://doi.org/10.1038/s41598-026-38668-4

Aboelmagd S, Atat R, Takiddin A (2026) Simplicial graph-based detection and localization of cyber attacks against large-scale smart grids. IEEE Trans Smart Grid 17(5):4681–4696. https://doi.org/10.1109/tsg.2026.3682950

Abu Nassar A, Morsi WG (2026) An interpretable deep learning approach for early detection, classification and mitigation of cyberattacks in cyber-physical microgrids. Sustain Energy Grids Netw 46:102266. https://doi.org/10.1016/j.segan.2026.102266

Acharya A, Bhalja BR (2026) Novel deep learning-based approach for detection, classification, and mitigation of cyber attacks in power distribution systems. IEEE Trans Ind Appl https://doi.org/10.1109/tia.2026.3678596

Alamgir FM, Das S, Akand AR, et al (2026) Hybrid CNN BiLSTM architecture for smart grid cyberattack detection using smart meter data. Sci Rep 16(1):25923. https://doi.org/10.1038/s41598-026-55727-y

Alwaisi Z, Soderi S (2026) Semantic communication-based detection of false data injection attacks in 6G-enabled smart grids. Int J Electr Power Energy Syst 175:111649. https://doi.org/10.1016/j.ijepes.2026.111649

Ansari M, Ghafouri M, Ameli A, et al (2026) Identification of cyberattacks targeting poorly-damped modes in wind-integrated grids. IEEE Trans Smart Grid 17(3):2463–2479. https://doi.org/10.1109/tsg.2026.3652933

Aygül K, Srivastava AK, Mohammadpourfard M, et al (2026) A generative AI-enhanced framework for intrusion detection and resilient transient stability prediction in networked cyber–physical power systems. Comput Electr Eng 139:111359. https://doi.org/10.1016/j.compeleceng.2026.111359

Badakhshan S, Zhang J (2026) Generative AI-enhanced real-time anomaly detection in integrated energy systems. IEEE Trans Smart Grid 17(2):1549–1560. https://doi.org/10.1109/tsg.2025.3630562

Bader L, Wagner E, Serror M (2026) PowerRange: an immersive cyber range for power grid operators. Int J Crit Infrastruct Prot 53:100845. https://doi.org/10.1016/j.ijcip.2026.100845

Balahewa I, Bjorndal L, Mi C, et al (2026) Cyber attack detection in renewable-rich IBR-dominated microgrids: A natural watermarking approach. Electr Power Syst Res 260:113148. https://doi.org/10.1016/j.epsr.2026.113148

Banerjee R, Goswami A, Chowdhury R, et al (2026) An enhanced federated learning-based multistage IDS for detecting zero-day and known threats in smart home ecosystem. Cluster Comput 29(7):457. https://doi.org/10.1007/s10586-026-06279-3

Behrooz H, Ilbeigi M (2026) Unsupervised cyber-physical intrusion detection and localization in water distribution systems via multivariate fusion of process, supervisory, and communication data. Reliab Eng Syst Saf 275:112770. https://doi.org/10.1016/j.ress.2026.112770

Bunko T, Johnstone MN, Yang W, et al (2026) A survey of privacy-preserving federated learning for intrusion detection systems. Artif Intell Rev 59(5):125. https://doi.org/10.1007/s10462-026-11519-4

Chandrakala D, Kanaga Suba Raja S, Sudha S (2026) Secure signal encoding and temporal feature extraction for multi class intrusion detection in smart grid environments. Sci Rep https://doi.org/10.1038/s41598-026-75689-5

Chen L, Tang Z, Yang Y, et al (2026) Federated deep learning for distributed intrusion detection and privacy preservation in power networks. Sci Rep https://doi.org/10.1038/s41598-026-55878-y

Cibin N, Kabbara N, Presekal A, et al (2026a) Cyber–physical power system dataset for cyber security of digital substations. IEEE Access 14:83420–83433. https://doi.org/10.1109/access.2026.3699153

Cibin N, Mulder B, Carstens H, et al (2026b) Cyber attacks detection, prevention, and source localization in digital substation communication using hybrid statistical-deep learning. IEEE Open J Ind Electron Soc 7:1622–1637. https://doi.org/10.1109/ojies.2026.3740198

Cook M, Stoica A, Shah A, et al (2026) With power comes responsibility: Attack synthesis for industrial control systems using large language models. ACM Trans Priv Secur 29(3):1–31. https://doi.org/10.1145/3815116

De Santis M, Esposito C, Seo JT (2026) A secure model-lifecycle layer for federated and gossip learning in smart grid attack detection. Comput Netw 289:112735. https://doi.org/10.1016/j.comnet.2026.112735

Dzafic D, Džafić I, Akagic A (2026) Anomaly detection in smart grid time-series data using Graph Deviation Networks. Eng Appl Artif Intell 165:113512. https://doi.org/10.1016/j.engappai.2025.113512

Farsi M, Alwateer M, Alsaedi SA, et al (2026) Detection of disturbances and cyber-attacks in smart grids using explainable machine learning. Sci Rep 16(1):9834. https://doi.org/10.1038/s41598-026-35449-x

García-Sáez LM, Ruiz-Villafranca S, Roldán-Gómez J, et al (2026) Hybrid clustering-guided federated learning for robust intrusion detection in highly heterogeneous IoT environments. Comput Netw 281:112205. https://doi.org/10.1016/j.comnet.2026.112205

Gowtham Chakravarthy D, Gopi R, Bhaskar KB, et al (2026) An edge-AI framework with graph transformer learning for resilient microgrid topology attack identification. Sci Rep 16(1):31721. https://doi.org/10.1038/s41598-026-61150-0

Goyel H, Presekal A, Palensky P, et al (2026) Anomaly detection in digital substation communication using Transformer-based distribution fitting. IEEE Trans Smart Grid 17(4):3405–3416. https://doi.org/10.1109/tsg.2026.3652798

Guo S, Pan K, Xu W (2026) Tractable attack detection with robustness certificate for nonlinear power systems under stochastic disturbances. IEEE Trans Smart Grid 17(5):4658–4680. https://doi.org/10.1109/tsg.2026.3681977

Hossain MM, Turja SD, Tasnim S, et al (2026) A cross-dataset based zero-day intrusion detection system by integrating siamese network and reinforcement learning. ICT Express 12(3):752–757. https://doi.org/10.1016/j.icte.2026.05.001

Hosseini Rostami SM, Pourgholi M, Asharioun H (2026) Cyber attack and fault detection in DC microgrids by designing an event-triggered based-robust algorithm. Sci Rep 16(1):1701. https://doi.org/10.1038/s41598-025-31202-y

Hu X, Qi Q, Wu J, et al (2026) A physics-guided dynamic graph attention network for false data injection attacks detection and localization in smart grids. IEEE Trans Smart Grid https://doi.org/10.1109/tsg.2026.3706685

Jeter TR, Alharbi R, Seo JT, et al (2026) Federated anomaly detection in smart grid systems. Digit Commun Netw 12(3):451–461. https://doi.org/10.1016/j.dcan.2026.02.001

Kfouri R, Jabr RA, Džafić I (2026) Bad data detection and identification based on graph neural network for power system state estimation. J Mod Power Syst Clean Energy 14(2):760–772. https://doi.org/10.35833/mpce.2025.000178

Khaldi Y, Azzaoui H, Benzaoui A, et al (2026) DARE-IDS: diversity-aware replay exemplars for intrusion detection systems. Cluster Comput 29(15):860. https://doi.org/10.1007/s10586-026-06696-4

Kumar P, Paul S, Chatterjee K (2026) Detection and mitigation of cyber attacks in PMSG based wind energy system using H-infinity controller. Int J Crit Infrastruct Prot 53:100842. https://doi.org/10.1016/j.ijcip.2026.100842

Li J, Lu H, Su Q (2026) Detection and localization of false data injection attacks based on multi-scale feature fusion and attention enhancement network in smart grid. Eng Appl Artif Intell 163:112787. https://doi.org/10.1016/j.engappai.2025.112787

Liu Z, Liu M, Xu J (2026) Poisoned data detection in power systems based on frequency-domain clustering and key feature selection. Int J Electr Power Energy Syst 180:112065. https://doi.org/10.1016/j.ijepes.2026.112065

Lozano-Paredes D, Bote-Curiel L, Feijoo-Martínez JR, et al (2026) Explainable autoencoder-based anomaly detection in IEC 61850 GOOSE networks. IEEE Access 14:71029–71042. https://doi.org/10.1109/access.2026.3690842

Lu Y, Lu Y, Wang Z, et al (2026) D-MAML: a few-shot learning algorithm for intrusion detection based on meta-learning and diffusion models. IEEE Internet Things J 13(11):24519–24531. https://doi.org/10.1109/jiot.2026.3674767

Martinez-Lopez F, Santana L, Rahouti M, et al (2026) Learning in multiple spaces: Prototypical few-shot learning with metric fusion for next-generation network security. IEEE Trans Netw Serv Manag 23:3156–3165. https://doi.org/10.1109/tnsm.2026.3665647

Massaoudi M, Ez Eddin M (2026) A constraint-aware physics-informed neural network for enhanced intrusion detection in cyber-physical smart grids. Sustain Energy Grids Netw 47:102406. https://doi.org/10.1016/j.segan.2026.102406

Mohammad UG, Afzal A, Alghamdi S, et al (2026) An optimized graph neural network approach for robust and explainable IoT intrusion detection against adversarial attacks. Sci Rep 16(1):21453. https://doi.org/10.1038/s41598-026-48715-9

Mohammadpourfard M, Ghanaatpishe F, Weng Y, et al (2026) Optimizing and evaluating deep learning techniques for stealthy false data injection attacks on smart grids. Energy Rep 15:108816. https://doi.org/10.1016/j.egyr.2025.11.075

Mohammed SH, Singh MSJ, Al-Jumaily A, et al (2026) IG-APSO-DNN: deep learning intrusion detection model to detect false data injection attacks in smart grids. Ad Hoc Netw 180:104053. https://doi.org/10.1016/j.adhoc.2025.104053

Muhammed AO, El Moursi MS, Hatziargyriou N (2026) Review of deep learning techniques applications in modern power systems stability and cyber security. Appl Energy 402:126927. https://doi.org/10.1016/j.apenergy.2025.126927

Nasir K, Badri SK, Alghazzawi D, et al (2026a) HED-ID: an edge-deployable and explainable intrusion detection system optimized via metaheuristic learning. Sci Rep 16(1):2313. https://doi.org/10.1038/s41598-025-32183-8

Nasir N, Lutui R, Sarkar NI, et al (2026b) Adaptive machine learning based cyber threat intelligence in industrial control systems: A systematic literature review. Int J Crit Infrastruct Prot 54:100874. https://doi.org/10.1016/j.ijcip.2026.100874

Nasiri S, Seifi H, Delkhosh H (2026) A trust-aware consensus mechanism for post-attack restoration of power system distributed state estimation. IEEE Trans Smart Grid 17(1):562–575. https://doi.org/10.1109/tsg.2025.3609891

Nasreen Fathima AH, Khraisat A, Syed Ibrahim SP, et al (2026) CLEAR: a continual learning framework for zero-day and anomaly-based intrusion detection. Knowl-Based Syst 352:116878. https://doi.org/10.1016/j.knosys.2026.116878

Pan H, Yang H, Na C, et al (2026) A transformer and CNN-based hybrid model for localization detection of false data injection attacks in smart grids. Sustain Energy Grids Netw 45:102150. https://doi.org/10.1016/j.segan.2026.102150

Park YS, Lim YS (2026) Hybrid multi-stage framework for identifying zero-day attacks and known threats in network traffic. Comput Netw 275:111875. https://doi.org/10.1016/j.comnet.2025.111875

Presekal A, Semertzis I, Goyel H, et al (2026) Intrusion detection system for digital substations using semi-supervised learning and traffic distance similarity clustering. IEEE Trans Smart Grid 17(1):576–589. https://doi.org/10.1109/tsg.2025.3611345

Qiu J, Gao W, Li Y, et al (2026) State estimation in smart grids with gated recurrent unit-Kalman filter network for Global Positioning System spoofing attack detection and correction. Reliab Eng Syst Saf 276:112935. https://doi.org/10.1016/j.ress.2026.112935

Qu Z, Liang F, Qu N, et al (2026) Intelligent localization of FDIA in smart grids: A multi-level wavelet spatio-temporal graph embedding approach. Expert Syst Appl 298:129620. https://doi.org/10.1016/j.eswa.2025.129620

Rajalakshmi M, Velmurugan T (2026) Physics-guided contrastive temporal graph learning for anomaly detection and root-cause localization in industrial control systems. Sci Rep 16(1):16487. https://doi.org/10.1038/s41598-026-45677-w

Rouhani SH, Su CL (2026) Future cyber-resilient renewable and sustainable smart grids: A critical review from power system researchers' perspective on emerging threats and solutions. Renew Sustain Energy Rev 226:116299. https://doi.org/10.1016/j.rser.2025.116299

Samin A, Zahin F, Duti NI, et al (2026) Hybrid deep learning for smart grid SCADA intrusion detection with mutual information feature selection for smart meter data analysis. Sci Rep https://doi.org/10.1038/s41598-026-70577-4

Sepehrzad R, Moafi M, Khosravi N, et al (2026) Secure FDRL for safe decision-making and policy embedding validation in hybrid microgrids under hybrid cyberattacks. Int J Electr Power Energy Syst 176:111646. https://doi.org/10.1016/j.ijepes.2026.111646

Sharma A, Kansal G, Tiwari R (2026) Cyber-physical anomaly detection in smart grid using hybrid deep learning framework. Array 31:101214. https://doi.org/10.1016/j.array.2026.101214

Sheta AN, Osman SF, Eladl AA, et al (2026) FalsEye: proactive detection of false data injection attacks in smart grids using IceCube-optimised ensemble learning. Sci Rep 16(1):9093. https://doi.org/10.1038/s41598-026-38723-0

Singh K, Sailaja Kumari M (2026) Detection of cyber and physical disturbances in power systems using data-driven deep learning methods. Int J Crit Infrastruct Prot 53:100844. https://doi.org/10.1016/j.ijcip.2026.100844

Su Y, Zhang T, Zhang Z, et al (2026) Unsupervised representation learning for anomaly detection in power grid communications via spatio-temporal graph neural networks. Neurocomputing 699:134408. https://doi.org/10.1016/j.neucom.2026.134408

Sun H, Zuo X, Sun Q, et al (2026) Collaborative H–/H_∞ observer and conformal prediction for false data injection attack detection in multiarea power systems. IEEE Syst J 20(3):906–917. https://doi.org/10.1109/jsyst.2026.3710557

Tasnim S, Das M, Billah S, et al (2026) A Bayesian learning-based distributed security framework for cyber-resilient frequency restoration in isolated microgrids. Int J Electr Power Energy Syst 181:112090. https://doi.org/10.1016/j.ijepes.2026.112090

Thao TLV, Hong J, Park K, et al (2026) AS-SRDT: semi-supervised anomaly detection for IEC 61850 GOOSE communication using anomaly-score supervision with restricted distribution transformation. IEEE Access 14:146228–146249. https://doi.org/10.1109/access.2026.3735745

Thiruppathy Kesavan V, Gopi R, Hossen MJ, et al (2026) Early detection of cyberattacks in renewable-integrated EV charging systems using transfer learning. Sci Rep https://doi.org/10.1038/s41598-026-71041-z

Tirulo A, Chauhan S, Issac B, et al (2026) Bayesian-optimized deep learning for adaptive real-time FDIA detection in smart grid demand response systems. IEEE Trans Smart Grid 17(5):4602–4613. https://doi.org/10.1109/tsg.2026.3668905

Tripathy SS, Behera B, Kollie DSJr., et al (2026) Enhancing anomaly-based zero-day attack detection framework using CNN-driven feature extraction and OC-SVM. Sci Rep 16(1):27164. https://doi.org/10.1038/s41598-026-55170-z

Vasanthi R, Adarsh MJ, Santhosh Kumar M, et al (2026) An advanced deep learning framework for scalable smart grid intrusion detection. Electr Power Syst Res 265:114060. https://doi.org/10.1016/j.epsr.2026.114060

Verma P, Dasgupta P, Chakraborty C (2026) DeepLR: deep learning assisted load redistribution attack on cyber-physical power system. IEEE Trans Smart Grid 17(1):551–561. https://doi.org/10.1109/tsg.2025.3605526

Wang L, Zhang K, Wang Y, et al (2026) Joint cyber-physical data-driven enhanced cyber-attack detection for power systems. IEEE Trans Smart Grid 17(5):4772–4789. https://doi.org/10.1109/tsg.2026.3692152

Wei S, Wu Z, Shahidehpour M, et al (2026) A Gaussian belief propagation-based cyber-attack detection approach for state estimation model in unbalanced distribution networks. IEEE Trans Smart Grid 17(4):3431–3444. https://doi.org/10.1109/tsg.2026.3654234

Wilkie J, Hindy H, Michie CW, et al (2026) A novel contrastive loss for zero-day network intrusion detection. IEEE Trans Netw Serv Manag 23:2064–2076. https://doi.org/10.1109/tnsm.2026.3652529

Wu S, Wang Q, Zhang Z, et al (2026) Physics-informed state estimation model with controllable uncertainty: Enhancing resilience against false data injection attacks in power systems. IEEE Trans Ind Inform 22(8):6871–6882. https://doi.org/10.1109/tii.2026.3681361

Xi L, Li Z, Chen C, et al (2026a) False data injection attack localization detection based on bidirectional spatiotemporal graph network enhanced by frequency domain perception. IEEE Trans Reliab 75:2499–2512. https://doi.org/10.1109/tr.2026.3708844

Xi L, Li Z, Wang Y, et al (2026b) Localizing detection of false data injection attack in CPPS based on CFSSA improved multi-layer extreme learning machine. Prot Control Mod Power Syst 11(4):112–124. https://doi.org/10.23919/pcmp.2025.000396

Xu D, Wu Z, Hu Q, et al (2026) A unified framework for numerically-stable state estimation and false data injection attack detection in distribution networks. IEEE Trans Ind Inform 22(2):1173–1181. https://doi.org/10.1109/tii.2025.3624288

Yan J, Sun K, Li Z, et al (2026) BT-MTD: bus traversal-based moving target defense for smart grid. IEEE Syst J 20(3):1177–1188. https://doi.org/10.1109/jsyst.2026.3717717

Yang K, Cai L, Wu J (2026a) An intrusion detection model for electric vehicle supply equipment based on multi-task learning and probability transfer mechanism. Sci Rep 16(1):12983. https://doi.org/10.1038/s41598-026-42331-3

Yang X, Zhao Z, Wang L, et al (2026b) A variational autoencoder attention fusion model for zero-day exploit behavior detection VAE-AttNet. J King Saud Univ Comput Inf Sci 38(7):852. https://doi.org/10.1007/s44443-026-01245-w

Yilmaz MT, Polat O, Algül E, et al (2026) Non-local attention enhanced deep learning for robust cyberattack detection in industrial IoT-based SCADA systems. Sci Rep 16(1):7857. https://doi.org/10.1038/s41598-026-37146-1

Yin Z, Wang W, Cheng Y, et al (2026) Efficient intrusion detection for edge network via multi-stage few-shot class-incremental learning. IEEE Trans Netw Sci Eng 13:4689–4706. https://doi.org/10.1109/tnse.2025.3644385

Yuan S, Li J, Wang C, et al (2026) Conformal machine learning for reliable anomaly detection in industrial cyber-physical systems. Reliab Eng Syst Saf 274:112417. https://doi.org/10.1016/j.ress.2026.112417

Zhang F, Li H (2026) GridPRISM: provenance-aware real-time intrusion detection via prior-guided subgraph routing and auditable budgeted semantic masking. Int J Crit Infrastruct Prot 53:100849. https://doi.org/10.1016/j.ijcip.2026.100849

Zhao S, Luo W, Shu Q, et al (2026) Unifying temporal continuity and structural decomposition for stealthy FDIA detection in smart grids via implicit neural representations. Int J Electr Power Energy Syst 174:111507. https://doi.org/10.1016/j.ijepes.2025.111507

Zheng H, Li X, Li F (2026) Risk-aware unsupervised cyberattack detection via contrastive autoencoding and adversarial training in smart grid substations. Reliab Eng Syst Saf 272:112529. https://doi.org/10.1016/j.ress.2026.112529

Zhou Y, Deng R, Jiang Q, et al (2026) IDAT: an impact-driven adaptive threshold for detecting false data injection attacks on wind turbines. IEEE Trans Smart Grid 17(4):3417–3430. https://doi.org/10.1109/tsg.2026.3653113

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- ESM1.pdf