

Article

Fed-CGIDS-UAV: Federated Causal Graph Learning for Cross-Domain Intrusion Detection in Cyber-Physical Drone Networks

Saleh Abdulrahman Alkhamis ¹, Abdalilah Alhalangy ², Galal Eldin Abbas Eltayeb ³ and Eman Abouelkheir ^{2,*}

¹ Department of Cybersecurity, College of Computer, Qassim University, Buraydah 52361, Saudi Arabia; khmiess@qu.edu.sa

² Department of Computer Engineering, College of Computer, Qassim University, Buraydah 52361, Saudi Arabia; a.alhalangy@qu.edu.sa

³ Department of Management Information Systems, College of Business and Economics, Qassim University, Buraydah 52571, Saudi Arabia; g.eltayeb@qu.edu.sa

* Correspondence: e.abouelkheir@qu.edu.sa

Abstract

Unmanned aerial vehicles (UAVs) have become essential cyber-physical platforms for applications such as surveillance, infrastructure inspection, emergency response, and intelligent transportation. However, their tight coupling among sensing, communication, control, actuation, and swarm coordination also exposes them to sophisticated cyber-physical attacks that are difficult to detect using conventional intrusion detection systems. Existing machine learning, deep learning, graph-based, and federated intrusion detection approaches generally rely on statistical feature representations or temporal patterns, providing limited capability to model causal dependencies among interacting UAV subsystems and to generalize across heterogeneous operating environments. To address these limitations, this paper proposes Fed-CGIDS-UAV, a federated causal graph learning framework for cross-domain intrusion detection in cyber-physical UAV networks. The proposed framework models each telemetry window as a typed causal graph in which nodes represent navigation, sensing, communication, control, actuation, and swarm states, while directed edges capture stable operational dependencies. Intrusions are detected by identifying violations of these learned causal relationships, and the framework provides interpretable node-edge explanations to support root-cause analysis. Furthermore, federated learning enables collaborative model training across distributed UAV clients without sharing raw telemetry, thereby preserving data privacy while improving robustness under heterogeneous operating conditions. The proposed framework was implemented and experimentally evaluated in a controlled simulation environment covering four UAV operating domains and six representative attack classes. All experiments were repeated over five independent runs using different random seeds, and the reported results correspond to the measured average performance. The proposed framework was implemented using **Python 3.12** (Python Software Foundation, Wilmington, DE, USA) and **PyTorch 2.3** (Meta Platforms, Menlo Park, CA, USA). UAV flight data were generated using **Microsoft AirSim 1.9.1** (Microsoft Corporation, Redmond, WA, USA), integrated with **PX4 Autopilot v1.14** (Dronecode Foundation, San Francisco, CA, USA) and **Gazebo Sim 11** (Open Source Robotics Foundation, Mountain View, CA, USA). Within this simulation-based evaluation, Fed-CGIDS-UAV achieved an accuracy of 0.968, an F1-score of 0.956, and an internal–external stability gap (IESG) of 0.028, outperforming conventional machine learning, deep learning, graph-based, and centralized causal baselines while maintaining competitive computational latency. Although these results



Academic Editor: Yanwei Zhou

Received: 12 June 2026

Revised: 22 July 2026

Accepted: 23 July 2026

Published: 29 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

demonstrate the effectiveness of the proposed framework under controlled simulation conditions, validation using real-flight UAV telemetry remains an important direction for future research. These results demonstrate that integrating causal graph learning with federated optimization provides an effective and interpretable solution for privacy-preserving intrusion detection in heterogeneous cyber-physical UAV environments.

Keywords: UAV security; drone intrusion detection; cyber-physical systems; explainable AI; cross-domain generalization; graph neural networks; edge computing

1. Introduction

UAV security is no longer a narrow communication problem. A drone can be attacked through its navigation stream, command channel, onboard sensors, flight controller, payload interface, or swarm coordination layer. A GPS spoofing event, for example, does not merely corrupt a position field; it can force a mismatch between expected motion, inertial readings, waypoint execution, and neighbor-relative behavior. A command injection attack may first look like a legitimate control action, but the physical response of the aircraft may become inconsistent with the normal relationship among command rate, motor current, velocity error, and attitude correction. This cyber-physical coupling makes UAV intrusion detection different from conventional network intrusion detection.

Recent work has shown that machine learning and deep learning can detect attacks in UAV communication settings, cellular-connected UAV networks, internet-of-drones environments, and drone controller networks [1–9]. These studies are important because they moved UAV security beyond purely rule-based monitoring. However, their dominant modeling assumption remains feature-centric: telemetry and network flows are represented as vectors or sequences, and the model learns statistical separation between benign and malicious samples. We do not dismiss that assumption. It works well when the training and test domains are similar. The difficulty appears when a detector trained on one drone type, simulator, mission profile, or communication condition is moved to another domain. In that setting, high internal accuracy can coexist with weak operational trust.

Recent UAV IDS literature reveals three linked gaps: limited cyber-physical explainability, insufficient real-time validation, and weak cross-domain generalization. These gaps are not separate problems. A detector that reports only a malicious label may still be difficult to trust during flight if it cannot indicate which cyber-physical dependency failed. Similarly, a model that performs well on a single simulator, drone type, or communication condition may not remain reliable when mission dynamics, sensing noise, swarm density, or channel delay change. This paper therefore treats UAV intrusion detection as a problem of monitoring whether cyber, sensing, control, actuation, and swarm subsystems continue to obey learned normal dependencies. A high GPS error alone may be explained by weather, urban canyon effects, or simulator noise; a high GPS error that breaks the learned relationship with IMU, yaw correction, communication state, and commanded trajectory is more informative.

Federated learning is included because UAV data are naturally distributed. Swarm members may be owned by different organizations, operate under privacy constraints, or communicate over bandwidth-limited links. Sending raw telemetry, video-derived state, or mission logs to a central server is often unrealistic. A federated IDS allows each UAV or ground station to train locally and share model updates instead of raw data [10–13]. Yet a naive federated classifier does not solve the causal and cross-domain problem. If local clients learn only feature-label correlations, the global model may still fail when the

physical and communication distributions shift. Fed-CGIDS-UAV addresses this issue by federating parameters associated with causal graph representation and anomaly scoring, while preserving local adaptation to domain-specific noise, delays, and mission profiles.

The main research question guiding the framework is: How can a UAV intrusion detection system detect attacks that break cyber-physical dependencies while preserving privacy and maintaining interpretability across heterogeneous UAV domains? We answer this question through a design-science framework and an evaluation protocol. The paper makes four contributions. First, it formalizes UAV IDS as causal dependency violation rather than feature classification only. Second, it defines a typed cyber-physical graph for UAV telemetry, where nodes represent GPS, IMU, command, actuation, communication, and swarm states. Third, it proposes a federated temporal graph learning architecture that can be trained across UAV clients without exchanging raw flight logs. Fourth, it defines a transparent evaluation plan covering attack-specific detection, ablation, cross-domain transfer, robustness, latency, communication overhead, and explanation quality.

2. Related Work

2.1. Machine Learning and Deep Learning-Based UAV Intrusion Detection

The increasing deployment of UAVs in civilian and military applications has motivated extensive research on intelligent intrusion detection systems. Recent surveys have comprehensively reviewed machine learning and deep learning techniques for UAV intrusion detection and highlighted the limitations of traditional signature-based security mechanisms in dynamic cyber-physical environments [1,2]. Explainable machine learning frameworks, such as DroneGuard, improve detection performance while providing interpretable security decisions [3]. Other studies have investigated machine learning-enabled intrusion detection for cellular-connected UAV networks [4], internet-of-drones environments [5], artificial neural network and genetic algorithm-based detection [6], and privacy-preserving intrusion detection and prevention of the internet of UAVs [7]. Explainable ensemble learning using SHAP has also been introduced for UAV controller-area networks [8], while ConvLSTM architectures have demonstrated strong temporal modeling capability for detecting attacks on small drones [9].

2.2. Federated Learning-Based Intrusion Detection

Federated learning has recently emerged as an effective solution for privacy-preserving distributed intrusion detection. Federated IDS frameworks have been proposed to enhance UAV privacy and security [10], while blockchain-assisted collaborative federated learning improves secure model aggregation across distributed UAV clients [11]. Privacy-preserving federated intrusion detection has also been investigated for distributed cyber-defense [12], and comprehensive surveys have summarized the concepts, challenges, and future research directions of federated learning for intrusion detection [13].

2.3. Graph Learning, Causal Learning and Explainable Security

Graph-based learning has become increasingly important because UAV cyber-physical systems naturally consist of interconnected sensing, communication, and control components. Comprehensive surveys of graph neural networks for intrusion detection have demonstrated the advantages of graph representations over conventional feature-based learning [14]. Dynamic graph modeling has further improved intrusion detection in IoT environments [15], while causality-guided counterfactual learning has enhanced anomaly detection for cyber-physical systems [16]. Causal deep learning has also improved explainability for anomaly detection in intelligent edge networks [17], and graph neural networks have been successfully applied to anomaly detection in human cyber-physical

systems [18]. In addition, explainable machine learning frameworks have shown that interpretable intrusion detection significantly improves the trustworthiness of automated security systems [19].

2.4. Advanced Federated Security Frameworks

More recent studies have investigated blockchain-assisted privacy protection for UAV-MEC systems [20], federated intrusion detection for underwater drone networks [21], decentralized federated anomaly detection in autonomous vehicular networks [22], and adaptive federated learning for UAV anomaly detection under non-IID data distributions [23]. Explainable artificial intelligence techniques, including SHAP, have further improved the interpretability of machine learning predictions in cybersecurity applications [24].

2.5. Additional UAV Security Studies

Beyond the above approaches, several recent studies have further strengthened UAV security using alternative learning paradigms. Comprehensive reviews have summarized machine learning approaches for UAV intrusion detection [25], while video stream-based frameworks have been proposed for detecting invasive FPV drones [26]. Belief rule-based reasoning has also been applied to UAV intrusion detection [27]; hybrid ICMetric-based security systems have strengthened UAV authentication [28]; malware detection techniques have been developed for internet of UAV environments [29].

2.6. Foundations of Federated Learning, Graph Learning and Causal Inference

The proposed framework is built upon the theoretical foundations of federated learning introduced by FedAvg [30] and subsequent advances in distributed optimization [31,32]. Graph representation learning is supported by graph convolutional networks [33], graph attention networks [34], GraphSAGE [35], and comprehensive graph neural network surveys [36]. Finally, the causal reasoning methodology adopted in this work follows Pearl's structural causal framework [37] and the causal inference principles established by Peters et al. [38]. Recent federated graph neural network (FedGNN) studies have further demonstrated the benefits of combining graph representation learning with federated optimization for distributed intrusion detection [39–41]. However, these methods primarily learn statistical graph associations and generally do not model explicit causal dependencies or provide subsystem-level causal explanations.

To contextualize the contributions of Fed-CGIDS-UAV, Table 1 provides a systematic comparison of representative UAV intrusion detection system (IDS) paradigms across key evaluation dimensions.

Table 1. Comparison of representative UAV intrusion detection and cyber-physical security approaches.

Category	Representative References	Core Technique	Federated Learning	Graph Learning	Causal Learning	Explainability	Cross-Domain Support	Main Limitation
Machine Learning/Deep Learning UAV IDS	[1–9]	ML, DL, CNN, LSTM, ANN, ConvLSTM	✗	✗	✗	Partial	✗	Centralized learning, limited robustness, weak cross-domain generalization
Federated Learning IDS	[10–13]	Federated optimization, privacy-preserving IDS	✓	✗	✗	Partial	Partial	Feature-vector learning, no causal reasoning
Graph-based IDS	[14,15]	GNN, GCN, GAT, GraphSAGE	✗	✓	✗	Limited	Partial	Learns statistical graph relations only

Table 1. Cont.

Category	Representative References	Core Technique	Federated Learning	Graph Learning	Causal Learning	Explainability	Cross-Domain Support	Main Limitation
Causal and Explainable IDS	[16–19]	Counterfactual learning, causal inference, SHAP, XAI	✗	Partial	✓	✓	Partial	Limited distributed learning and scalability
Advanced Federated Security Frameworks	[20–24]	Blockchain, decentralized federated, non-IID adaptation, SHAP	✓	✗	✗	Partial	Partial	Focuses primarily on data privacy and non-IID shifts without causal tracking
Additional UAV Security Paradigms	[25–29]	Video stream, belief rules, ICMetrics, Fourier malware detection	✗	✗	✗	Partial	✗	Domain-specific application, lacks distributed cross-domain adaptability
Foundations & Federated Graph Learning (FedGNN)	[30–41]	Federated GNNs, foundational optimization & causal principles	✓	✓	✗	Limited	Partial	No explicit causal dependency modeling
Proposed Fed-CGIDS-UAV	This work	Federated causal graph learning	✓	✓	✓	✓	✓	Lightweight edge computation with interpretable causal tracing

Note: ✓ = fully supported; ✗ = not supported; Partial = partially supported or available only under specific conditions; Limited = supported to a limited extent but not as a core capability.

Early centralized machine learning and deep learning solutions (e.g., CNN, LSTM, ConvLSTM) pioneered automated anomaly detection in UAV communications [1–9]. However, these approaches suffer from single-point-of-failure risks, poor cross-domain generalization, and limited interpretability under dynamic cyber-physical attacks [1–9]. Federated Learning (FL) frameworks address data privacy by decentralizing model training across edge nodes [10–13], yet traditional vector-based FL models fail to capture structural network interactions and lack causal reasoning capabilities [10–13].

Graph Neural Network (GNN) architectures excel at modeling complex network topologies [14,15], but they primarily learn statistical correlations, making them susceptible to spurious features and adversarial perturbations [14,15]. While recent research has leveraged explainable AI (XAI) and counterfactual learning to improve model transparency [16–19], these methods typically lack scalable, distributed edge implementations [16–19].

Furthermore, advanced federated security frameworks incorporate blockchain and decentralized consensus mechanisms [20–24], and specialized paradigms address domain-specific threats such as video streams or malware signatures [25–29]. However, these solutions remain focused on data privacy or narrow domain applications without providing causal tracking [20–29]. Finally, foundational federated graph learning (FedGNN) models combine topology learning with optimization principles [30–41], yet they still struggle to model explicit causal dependencies across non-IID cross-domain environments [30–41].

As highlighted in Table 1, Fed-CGIDS-UAV bridges these critical gaps by integrating causal graph learning within a federated edge framework. This enables robust cross-domain adaptability, privacy preservation, and interpretable root-cause event tracing beyond standard statistical correlations.

3. System Model

The UAV cyber-physical network is modeled as a distributed set of clients $K = \{1, \dots, k\}$, where each client represents either an individual UAV, a ground-control node, or a swarm sub-cluster. Consistent with prior UAV-IDS architectures [1,3,14], each client observes multi-domain telemetry that reflects the coupling between cyber, sensing, control, actuation, and swarm coordination processes.

3.1. Telemetry Representation

For each client k and time window t , the raw cyber-physical observation is defined as $X_t^k = \{x_{\text{gps}}, x_{\text{imu}}, x_{\text{cmd}}, x_{\text{act}}, x_{\text{comm}}, x_{\text{swarm}}\}_t^k$, where each component corresponds to a subsystem described as follow: x_{gps} navigation GPS error, heading deviation, way-point deviation, estimated velocity, x_{imu} inertial sensing IMU acceleration, gyroscope readings, vibration, attitude error, x_{cmd} command rate, mode changes, waypoint updates, motor/control-surface commands, x_{act} actuation motor current, thrust proxy, ESC temperature, battery voltage, x_{comm} communication packet loss, latency, RSSI, retransmission rate, command-channel delay, swarm neighbor disagreement, relative-position consistency, consensus score, and identity-change rate [3,7,25].

3.2. Typed Causal Graph Construction

Each telemetry window is mapped to a typed-directed graph $G_t^k = (V_t^k, E_t^k, A_t^k)$, where V_t^k is the set of subsystem nodes, E_t^k is the set of directed candidate dependencies, and A_t^k contains edge attributes (lag, residual magnitude, confidence, domain label). Typed graphs preserve semantic distinctions between communication, sensing, and physical-dynamics edges, addressing limitations of associative GNN-IDS models [14,18,36].

The proposed framework should not be interpreted as automatically discovering universal physical causality solely from observational telemetry. Instead, the learned graph represents stable dependency structures inferred from benign operational data under the guidance of prior UAV control knowledge. These dependencies are considered causal approximations because they satisfy three conditions: (i) they are consistent with the known control architecture and sensor-actuator interactions of the UAV platform; (ii) they remain stable across repeated benign flights and operating conditions; and (iii) they exhibit consistent predictive behavior when evaluated through causal residual analysis. Consequently, the proposed framework identifies operational dependencies that are highly consistent with the underlying cyber-physical system rather than claiming formal causal identifiability in the intervention-based sense. This interpretation is consistent with the practical objective of intrusion detection, where preserving stable control relationships is more important than estimating complete structural causal models.

Throughout this paper, the term causal graph refers to a graph whose structure is guided by prior UAV control knowledge and stable operational relationships rather than by intervention-based causal discovery. Accordingly, the learned graph should be interpreted as representing operational dependencies that approximate the underlying cyber-physical interactions. To avoid overstating causal identifiability, the terms *operational dependency* and *dependency violation* are used in the implementation and evaluation sections whenever referring to the learned relationships.

3.3. Operational Dependency Learning

The objective of the causal dependency learning module is to model stable operational relationships among UAV cyber-physical subsystems during benign operation. Unlike conventional intrusion detection approaches that rely primarily on statistical correlations, the proposed framework learns directed dependency structures that capture how information

propagates between sensing, communication, navigation, control, actuation, and swarm coordination components. This design is motivated by recent advances in causal reasoning for cyber-physical anomaly detection, which have demonstrated improved robustness and interpretability under changing operating conditions [16,17,37,38]. Let the cyber-physical graph at time window t be represented as $G_t = (V, E_t)$, where V denotes subsystem nodes and E_t denotes directed dependency edges. For each node v_i , the expected operational state is estimated from its parent nodes using: $\hat{x}_i(t) = f_i(\{x_j(t - \tau_{ji}) \mid v_j \in Pa(v_i)\})$ where $Pa(v_i)$ denotes the parent set of node v_i , τ_{ji} represents the dependency lag, and $f_i(\cdot)$ is implemented as a lightweight multilayer perceptron trained using benign telemetry data. The discrepancy between observed and expected behavior is quantified through the causal residual: $r_i(t) = |x_i(t) - \hat{x}_i(t)|$ which measures the degree to which subsystem behavior deviates from its learned operational dependency. To evaluate dependency stability, an edge confidence score is computed as:

$$s_{ij}(t) = \exp\left(-\frac{r_i(t)}{\sigma_i + \epsilon}\right)$$

where σ_i denotes the residual standard deviation estimated during normal operation and ϵ prevents numerical instability.

The corresponding edge-instability score is defined as: $\Delta_{ij}(t) = |s_{ij}(t) - \bar{s}_{ij}|$ where $\bar{s}_{ij}$ represents the average edge confidence observed in benign training data. Large values of $\Delta_{ij}(t)$ indicate violations of expected cyber-physical dependencies and provide interpretable evidence of abnormal system behavior. Unlike conventional feature-based anomaly detection methods, these residuals directly identify which subsystem relationships have been disrupted by an attack.

3.4. Temporal Graph Encoding

Although individual dependency violations provide useful attack indicators, many UAV attacks evolve gradually over time and therefore require temporal modeling. To capture both structural and temporal characteristics of cyber-physical behavior, the proposed framework employs a temporal graph encoder that combines graph attention learning with recurrent temporal aggregation [14,15,33–36]. For each graph G_t , node embeddings are updated through graph attention message passing according to: $h_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(l)} W^{(l)} h_j^{(l)}\right)$, where $h_i^{(l)}$ denotes the node representation at layer l , $W^{(l)}$ is a trainable weight matrix, $\alpha_{ij}^{(l)}$ is the attention coefficient, and $\sigma(\cdot)$ represents the activation function. To explicitly preserve causal information, residual values are concatenated with graph embeddings: $z_i(t) = [h_i(t) \parallel r_i(t)]$ where $\parallel$ denotes vector concatenation. The resulting graph representation is processed through a gated recurrent unit (GRU) to model temporal evolution:

$$m_t = \text{GRU}(m_{t-1}, Z_t)$$

where Z_t contains the residual-enhanced node embeddings for the current telemetry window. This temporal representation captures attack progression patterns that may not be observable from a single graph snapshot. Consequently, the encoder learns both instantaneous dependency violations and longer-term attack dynamics, improving detection performance for replay attacks, communication disruptions, and distributed swarm attacks.

3.5. Anomaly Scoring and Federated Optimization

The final detection decision is based on a combination of node-level causal residuals and edge-level dependency instability measures. The overall anomaly score for telemetry

window t is computed as: $A_t = \lambda_1 \sum_{i=1}^{|V|} r_i(t) + \lambda_2 \sum_{(i,j) \in E_t} \Delta_{ij}(t)$ where λ_1 and λ_2 are balancing coefficients controlling the relative contributions of node and edge anomalies.

An intrusion alert is generated when: $A_t > \theta_c$, where θ_c denotes a client-specific threshold calibrated using benign validation data. Local threshold calibration is adopted to accommodate heterogeneous UAV platforms and operating environments, thereby reducing false alarms caused by domain-specific behavior.

Model training is performed using the binary cross-entropy objective: $\mathcal{L} = -\frac{1}{N} \sum_{n=1}^N [y_n \log(\hat{y}_n) + (1 - y_n) \log(1 - \hat{y}_n)]$, where y_n is the ground-truth attack label and $\hat{y}_n$ is the predicted attack probability. To preserve privacy and support distributed deployment, the framework adopts federated learning. Each client performs local optimization using its own telemetry data and transmits only model parameters to the aggregation server. Global model updates are obtained through the federated averaging (FedAvg) algorithm [30,31]: $W^{(r+1)} = \sum_{k=1}^K \frac{n_k}{N} W_k^{(r)}$, where $W_k^{(r)}$ denotes the model parameters of client k , n_k represents the number of local training samples, and $N = \sum_{k=1}^K n_k$. This process enables collaborative learning across heterogeneous UAV clients while avoiding direct sharing of raw telemetry data. By combining causal graph representations with federated optimization, the proposed framework improves robustness under domain shifts while maintaining privacy-preserving operation in distributed UAV environments [10–13,20–23,30–32].

4. Threat Model

The adversary is assumed to target one or more UAV cyber-physical channels without necessarily controlling the entire platform. We consider six attack classes because they represent common ways in which cyber compromise can manifest physically or operationally. The adversary is assumed to have partial cyber access to navigation, command, sensing, communication, or swarm-message channels, but not full physical control of all UAVs at once. The federated server or cluster head is treated as honest-but-curious unless robust aggregation is enabled, while individual clients may experience noisy telemetry, intermittent links, or compromised local observations. Physical capture, full firmware replacement, and attacks against the cryptographic transport layer are outside the present scope. This boundary keeps the detection problem focused on cyber-physical inconsistency rather than complete platform takeover.

We model the UAV environment as a distributed cyber-physical system in which each UAV $k \in K$ produces time-windowed telemetry $X_t^k = \{x_{\text{gps}}, x_{\text{imu}}, x_{\text{cmd}}, x_{\text{act}}, x_{\text{comm}}, x_{\text{swarm}}\}_t^k$, as defined in the system model. Under benign operation, the telemetry obeys a stable causal dependency structure learned from normal flight windows. The adversary attempts to violate these dependencies by introducing structured perturbations into one or more cyber-physical channels.

4.1. Normal Cyber-Physical Dynamics

Let the UAV's normal causal transition model be $X_{t+1}^k = F(X_t^k) + \eta_t^k$, where $F(X_t^k)$ represents the UAV's cyber-physical control loop and η_t^k is bounded stochastic noise. For each subsystem node v_j , the normal causal predictor is $\hat{x}_{j,t} = f_j(Pa_j, t-l)$, where Pa_j is the parent set and l is the subsystem-specific lag.

- Following the causal residual formulation introduced in Section 3.3, the residual magnitude is computed for each monitored variable and subsequently used to estimate local anomaly evidence during graph construction.
- The edge instability defined in Section 3.3 is evaluated for each telemetry window to quantify structural changes in the learned causal graph.
- A window is benign if: $|r_{j,t}| \leq \epsilon_j, |\Delta e_{ij,t}| \leq \epsilon_{ij}$, for all nodes and edges.

4.2. Adversarial Perturbation Model

The adversary introduces a perturbation $\delta_t^k \neq 0$, modifying the observed telemetry to $\tilde{X}_t^k = X_t^k + \delta_t^k$ according to the assumptions of the proposed UAV intrusion detection framework in [1–3,16,17].

The perturbation is restricted:

- It affects only selected cyber-physical channels (navigation, command, sensing, communication, swarm).
- It does not grant full physical takeover or firmware replacement.
- It does not break cryptographic transport layers.

4.3. Adversary Objective

The adversary's goal is to maximize causal dependency violations while minimizing detectability: $\max_{\delta_t^k} \sum_j |r_{j,t}(\delta_t^k)| + \sum_{(i,j) \in E} |\Delta e_{ij,t}(\delta_t^k)|$ subject to: $\|\delta_t^k\| \leq \Gamma$, (bounded perturbation), $\Pr(S_t > \theta_k)$ is minimized, (avoid detection) where the anomaly score is computed using the formulation presented in Section 3, combining residual magnitude, edge instability, and trust information into a unified intrusion indicator.

4.4. Attack Classes as Structured Perturbations

Each attack class corresponds to a specific perturbation pattern δ_t^k and a predictable set of broken causal edges:

1. **GPS spoofing:** $\delta_{\text{gps},t}^k \neq 0$, breaks: (GPS $\rightarrow$ dynamics), (GPS $\rightarrow$ swarm). The attacker injects or manipulates navigation signals so that the UAV receives misleading position or heading information. The immediate telemetry effect is increased GPS deviation, but the more informative signal is inconsistency between GPS, IMU, expected motion, and waypoint execution.
2. **Command injection:** $\delta_{\text{cmd},t}^k \neq 0$, breaks: (command $\rightarrow$ actuation), (command $\rightarrow$ dynamics). The attacker inserts unauthorized commands or modifies legitimate control messages. This attack affects command frequency, mode changes, and actuation response. A causal detector should observe that motor current and motion do not match the expected command-to-dynamics relationship.
3. **Sensor manipulation:** $\delta_{\text{imu},t}^k \neq 0$, breaks: (sensor $\rightarrow$ dynamics), (motor $\rightarrow$ IMU). The attacker corrupts IMU, barometer, magnetometer, or payload-derived state values. The physical behavior may remain stable while the sensing stream suggests abnormal motion, or the reverse may happen. The broken dependency appears between sensor and dynamics nodes.
4. **Replay attack:** $\tilde{X}_t^k = X_{t-\tau}^k, \tau > 0$, breaks: (communication $\rightarrow$ command), (time $\rightarrow$ sensor state). The attacker replays previously valid telemetry or command messages. Feature values may appear normal, which makes replay difficult for simple anomaly detectors. The causal signal lies in temporal staleness, latency patterns, and mismatch between current actuation and repeated measurements.
5. **Denial-of-service (DoS):** $\delta_{\text{comm},t}^k = \text{high latency or packet loss}$, breaks: (communication $\rightarrow$ command), (communication $\rightarrow$ swarm). The attacker degrades communication by flooding, jamming, or overwhelming the command channel. The communication layer shows packet loss and latency changes, but the safety-critical effect is delayed control and degraded coordination.
6. **Sybil attack:** $\delta_{\text{swarm},t}^k = \text{identity injection}$, breaks: (identity $\rightarrow$ consensus), (swarm $\rightarrow$ dynamics). In swarm settings, the attacker introduces false identities or manipulates neighbor messages. The attack affects consensus, neighbor disagreement, and relative-

position consistency. A single-drone detector may miss it because the evidence is distributed across swarm edges. These structured violations correspond exactly to the causal graph disruptions described in Table 2.

Table 2. Attack effects on cyber, physical, and causal graph behavior.

Attack	Cyber-Layer Symptom	Physical Symptom	Expected Broken Dependency
GPS spoofing	Abnormal navigation stream	Position/yaw correction mismatch	GPS $\geq$ dynamics; GPS $\geq$ swarm consistency
Command injection	Unauthorized command burst	Motor/motion response mismatch	command $\geq$ actuation; command $\geq$ dynamics
Sensor manipulation	Corrupted IMU/ barometer reading	State-estimation inconsistency	sensor $\geq$ dynamics; motor $\geq$ IMU
Replay	Stale repeated packets	Old state inconsistent with current control	communication $\geq$ command; time $\geq$ sensor state
DoS	Packet loss and delay	Delayed control and unstable coordination	communication $\geq$ command; communication $\geq$ swarm
Sybil	False neighbor identities	Swarm disagreement	identity $\geq$ neighbor consensus; swarm $\geq$ dynamics
Detection	–	–	An intrusion is detected when the anomaly score exceeds the local threshold: $S_t > \theta_k$

4.5. Detection Condition

A UAV client k declares an intrusion when $S_t > \theta_k$, where θ_k is a locally calibrated threshold reflecting platform-specific noise, wind, sensing variance, and communication delay.

This ensures that detection is triggered by causal dependency violations, not merely statistical anomalies addressing the explainability and cross-domain limitations of prior UAV IDS approaches [8,19,24].

5. Proposed Method: Fed-CGIDS-UAV

Fed-CGIDS-UAV is a federated causal graph intrusion detection framework designed to identify cyber-physical inconsistencies in UAV networks operating across heterogeneous domains. The method integrates typed graph construction, causal dependency learning, temporal graph encoding, federated aggregation, and operational dependency. The central premise is that UAV telemetry must satisfy a set of stable cyber-physical dependencies; an intrusion is detected when these dependencies are violated.

Section 3 establishes the mathematical foundation of the proposed framework, including the causal graph model, residual computation, anomaly scoring, and federated aggregation strategy. Building upon these formulations, this section describes the implementation workflow and explains how the previously defined components are integrated into the complete Fed-CGIDS-UAV intrusion detection pipeline. To improve readability and avoid unnecessary repetition, mathematical definitions are not repeated here unless additional implementation details are required.

5.1. Typed Cyber-Physical Graph Construction

For each UAV client k and telemetry window X_t^k , the system constructs a typed-directed graph

$$G_t^k = (V_t^k, E_t^k, A_t^k),$$

where V_t^k contains subsystem nodes representing navigation, inertial sensing, command, actuation, communication, and swarm states, E_t^k contains directed candidate dependencies $v_i \rightarrow v_j$, initialized from UAV control logic and refined using conditional dependence and lag analysis, and A_t^k stores edge attributes including lag, dependency confidence, and window-level statistics.

Typed graphs preserve semantic distinctions between cyber, sensing, and physical-dynamics relationships, preventing the model from conflating structurally different interactions.

In future UAV deployments, additional heterogeneous sensing modalities (e.g., LiDAR, thermal cameras, radar, environmental sensors, or payload-specific sensors) can be incorporated by extending the typed causal graph with new node types and their corresponding causal dependencies. The graph construction procedure is modular and therefore requires only local updates to the affected node neighborhood rather than reconstruction of the entire graph. Let n denote the number of subsystem nodes and e the number of candidate dependency edges. Adding k new sensors increases the graph size to $n + k$, while the computational complexity of graph construction increases approximately linearly with the number of newly introduced local dependencies. Since message passing in the proposed graph encoder operates on sparse adjacency structures, the computational complexity remains approximately $O(n + e)$, and the incremental cost introduced by new sensors is proportional to the additional local edges rather than the complete graph size. Furthermore, federated training remains unaffected because only model parameters are exchanged during aggregation, while each client independently updates the graph structure according to its available sensing configuration. Consequently, the proposed framework is readily extensible to heterogeneous UAV platforms with only moderate computational overhead.

5.2. Operational Dependency Learning

For each candidate dependency $v_i \rightarrow v_j$, the normal operational relationship is modeled as $\hat{x}_{j,t} = f_j(Pa_j, t - l) + \varepsilon_{j,t}$, where Pa_j is the parent set of node j , l is a subsystem-specific lag, and $\varepsilon_{j,t}$ is the residual error.

- Following the causal residual formulation introduced in Section 3.3, residual values are calculated online for every graph node.
- Edge instability is computed using the formulation presented in Section 3.3.
- The anomaly score defined in Section 3.5 is evaluated for each telemetry window to determine whether the observation is malicious.

5.3. Temporal Graph Encoding

To model temporal evolution, the system processes a sequence of graphs $(G_{t-m}^k, \dots, G_t^k)$ using a temporal graph encoder that combines graph attention message passing with gated recurrence. Node embeddings are updated as $h_{j,t}^{(l+1)} = \sigma(W_0 h_{j,t}^{(l)} + \sum_{i \in Pa(j)} \alpha_{ij,t} W_1 h_{i,t}^{(l)} + W_2 r_{j,t})$, where $\alpha_{ij,t}$ is an attention weight and $r_{j,t}$ is the causal residual explicitly injected to preserve interpretability. Temporal continuity is captured using a GRU: $z_{j,t} = \text{GRU}(h_{j,t}, z_{j,t-1})$, ensuring that causal dependency violations propagate through time rather than being absorbed into latent representations.

5.4. Explanation Fidelity

In addition to conventional detection metrics, the quality of generated explanations is evaluated through explanation fidelity. Recent studies have emphasized that explainability should be assessed not only by the interpretability of generated explanations but also by their consistency with the underlying model decision process [8,19,24]. In cyber-physical intrusion detection systems, explanation quality is particularly important because operators require actionable evidence regarding which subsystem interactions have been

disrupted by an attack. Furthermore, causal anomaly detection approaches have demonstrated that explanations linked to operational dependencies provide more informative diagnostic insights than feature attribution alone [16,17]. Explanation fidelity measures the degree to which the explanatory component identified by the model contributes to the final detection decision. A faithful explanation should highlight the subsystem dependency whose removal produces the greatest change in model output. Accordingly, explanation fidelity is computed as $F_{exp} = 1 - \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - \hat{y}_i^{(-exp)}|$, where $\hat{y}_i$ denotes the original prediction probability, $\hat{y}_i^{(-exp)}$ represents the prediction obtained after removing the highlighted explanatory component, and N is the number of evaluated samples. Higher values indicate stronger agreement between the generated explanation and the underlying decision-making process. In Fed-CGIDS-UAV, the explanatory component corresponds to the dominant node or edge exhibiting the largest operational dependency violation. Consequently, explanation fidelity quantifies how accurately the reported subsystem relationship reflects the actual intrusion decision and provides an objective measure of explanation quality [16,17,19,24].

5.5. Federated Learning and Global Aggregation

Each UAV client trains its local causal graph encoder and scoring head using only its own telemetry windows. Raw data are never shared. Instead, each client transmits model updates θ_k^r to the server at round r . The global aggregation rule is $\theta^{r+1} = \sum_{k=1}^K \frac{n_k}{N} \theta_k^r$, where n_k is the number of local windows and $N = \sum_k n_k$. This weighted aggregation mitigates non-IID client distributions arising from differences in mission profiles, environmental conditions, and communication quality. Each client performs **local threshold calibration** to compute a platform-specific detection threshold θ_k , ensuring robustness across heterogeneous UAV types and operating conditions.

5.6. Operational Dependency Explanation

Fed-CGIDS-UAV provides interpretable alerts by identifying the specific causal dependencies that were violated. For each flagged window, the system reports:

- Dominant node residuals $r_{j,t}$,
- Unstable edges $\Delta e_{ij,t}$,
- The inferred causal path of violation.

This produces actionable explanations such as “GPS residual increased, GPS $\rightarrow$ dynamics dependency weakened, swarm-consistency edge destabilized”, enabling operators to understand the cyber-physical origin of the anomaly.

5.7. Implementation Procedure

The implementation procedure is made explicit to reduce ambiguity between the conceptual framework and the executable detector. Algorithm 1 describes local graph construction and residual scoring, while Algorithm 2 describes federated training and local threshold calibration. The complete detection pipeline is summarized as follows:

1. **Graph construction:** segment telemetry into windows and construct typed subsystem graphs.
2. **Causal learning:** estimate normal dependency functions and compute residuals and edge deviations.
3. **Temporal encoding:** encode graph sequences using attention-based message passing and GRU recurrence.
4. **Anomaly scoring:** compute S_t and compare against the calibrated threshold θ_k .
5. **Explanation:** report violated node-edge dependencies for each alert.

6. **Federated training:** exchange model updates and perform global aggregation without sharing raw telemetry.

Algorithm 1. Local causal graph construction and residual scoring.

Step	Operation
1	Segment each UAV flight log into fixed-length windows and group variables into navigation, inertial, command, actuation, communication, and swarm nodes.
2	Initialize physically plausible candidate edges using UAV control logic, including command-to-actuation, actuation-to-motion, GPS-to-dynamics, communication-to-command, and swarm-to-consensus dependencies.
3	Estimate lagged dependency functions for each child node using benign training windows and retain edges that are temporally plausible, stable, and predictive.
4	Compute node residuals and edge instability scores for each window.
5	Pass typed graphs and residuals to the temporal graph encoder and compute the anomaly score.
6	Generate an alert when the locally calibrated score exceeds the client threshold and return the dominant broken node-edge dependencies as the explanation.

Algorithm 2. Federated training and local threshold calibration.

Step	Operation
1	Each UAV client trains the local causal graph encoder and scoring head using its own windows without sharing raw telemetry.
2	The client sends model updates and summary statistics required for aggregation, not mission logs or raw sensor streams.
3	The server aggregates client updates using sample-size weighting; robust aggregation can replace simple averaging when poisoning risk is high.
4	The updated global parameters are redistributed to clients.
5	Each client calibrates its alert threshold on local benign validation windows to accommodate platform size, wind, sensing noise, and channel conditions.

Figure 1 illustrates the overall architecture of the proposed **Fed-CGIDS-UAV** framework for cross-domain intrusion detection in cyber-physical UAV networks. The framework begins with the collection of multi-domain telemetry data from navigation, sensing, control, actuation, communication, and swarm subsystems. These heterogeneous data streams are transformed into a **typed causal graph**, where nodes represent UAV subsystems and directed edges model their normal cyber-physical dependencies. A local dependency residual learning module continuously monitors these dependencies and identifies violations caused by attacks such as GPS spoofing, command injection, DoS/jamming, and Sybil attacks. To preserve privacy, each UAV or ground station performs local training and transmits only model updates to a federated aggregation server, which constructs a global causal model without sharing raw telemetry. The resulting model enables robust cross-domain intrusion detection while maintaining adaptability to different operating environments. In addition to attack detection, the framework provides **explainable security analytics** by identifying the specific broken causal relationships responsible for each alert and recommending appropriate mitigation actions. This architecture highlights the integration of causal reasoning, federated learning, and explainable AI to achieve accurate, privacy-preserving, and interpretable UAV intrusion detection across heterogeneous deployment domains.

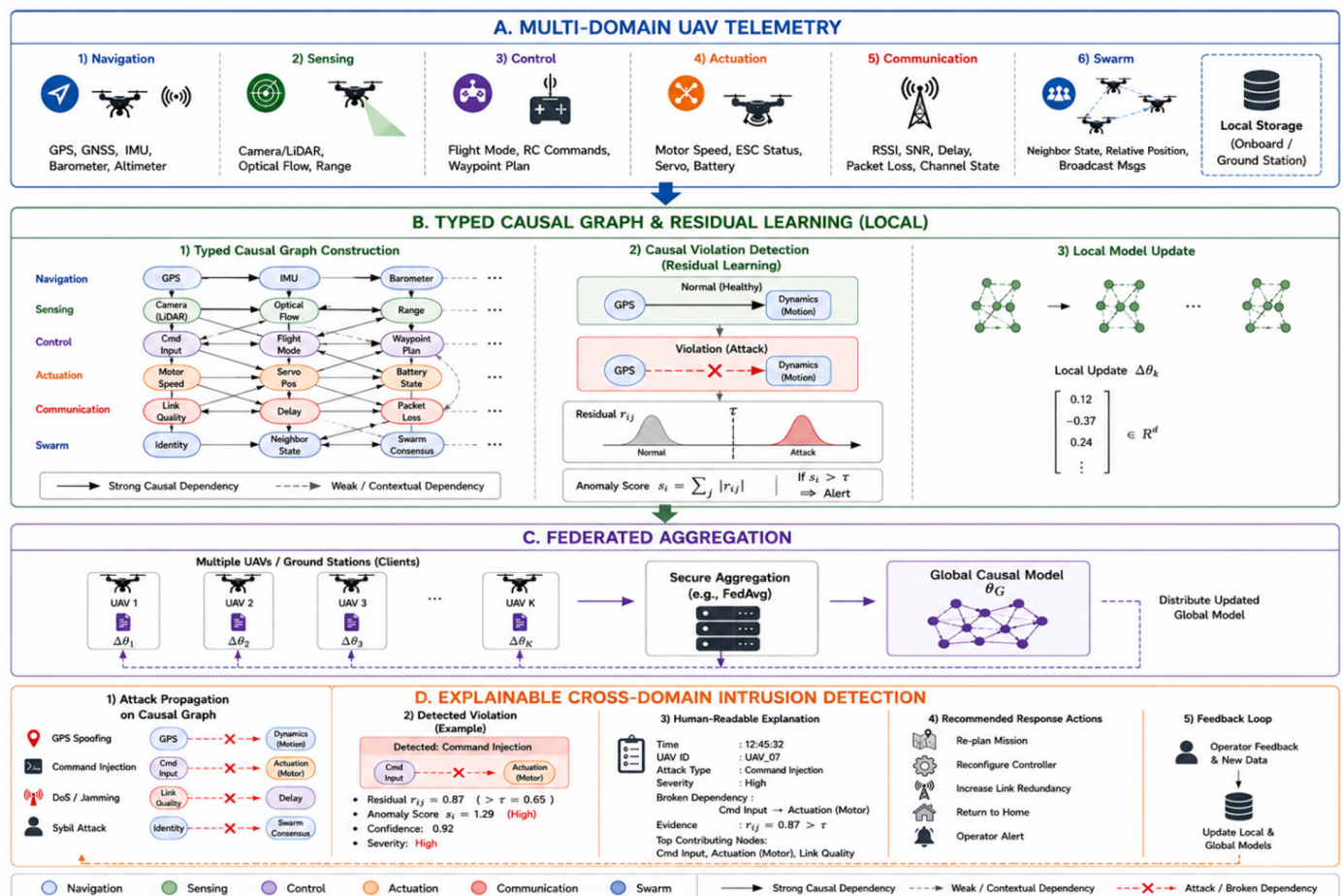


Figure 1. The architecture of the proposed Fed-CGIDS-UAV.

Table 3 summarizes the design rationale underlying the proposed Fed-CGIDS-UAV framework by positioning it relative to major intrusion detection paradigms. Rather than replacing existing approaches, Fed-CGIDS-UAV incorporates their most beneficial characteristics while addressing their key limitations in UAV cyber-physical environments. Classic machine learning methods provide efficient inference but lack the ability to represent structural relationships among interconnected subsystems. Deep learning approaches improve temporal pattern extraction; however, they often suffer from limited interpretability and may require substantial training data. Autoencoder-based detectors are effective for unsupervised anomaly detection but typically identify abnormal behavior through reconstruction errors without explicitly revealing which operational dependencies have been violated. Graph-based intrusion detection systems improve structural modeling by representing interactions among entities and network components. Nevertheless, most existing GNN-based approaches learn associative relationships that may not remain stable across heterogeneous deployment environments. Similarly, federated learning enhances privacy preservation and collaborative training but may aggregate domain-specific correlations that do not generalize well across clients. Explainable AI techniques increase transparency through feature attribution; however, they often provide limited insight into how attacks propagate through cyber-physical processes. To overcome these limitations, Fed-CGIDS-UAV integrates causal graph learning, temporal dependency modeling, federated optimization, and subsystem-level explainability within a unified architecture. The framework models UAV telemetry as a typed cyber-physical graph, identifies attacks through violations of learned causal dependencies, preserves privacy through federated training, and generates interpretable node-edge explanations that support forensic analysis.

and operational decision-making. Consequently, the proposed framework combines the structural awareness of graph-based methods, the collaborative capabilities of federated learning, and the interpretability of explainable AI while introducing causal reasoning as a mechanism for improved robustness and cross-domain generalization.

Table 3. Positioning of Fed-CGIDS-UAV against common IDS families.

IDS Family	Typical Representation	Strength	Limitation for UAV Cyber-Physical Use	Fed-CGIDS-UAV Response
Classical ML IDS	Flat features	Lightweight and interpretable in simple settings	Weak structural modeling and domain transfer	Adds typed graph and causal residuals
DL sequence IDS	Time-series windows	Captures temporal patterns	Often black-box and data hungry	Adds causal edges and explanation layer
Autoencoder IDS	Reconstruction error	Useful for unlabeled anomaly detection	Reconstruction does not identify broken dependencies	Uses causal dependency violation rather than reconstruction alone
GNN IDS	Graph structure	Captures entity relations	Edges may be associative, not causal	Learns directed normal dependency graph
Federated IDS	Distributed model updates	Preserves privacy and reduces raw-data sharing	May federate unstable local correlations	Federates causal graph representation and local thresholds
XAI IDS	Feature attribution	Improves transparency	May not explain cyber-physical propagation	Reports node-edge violations and attack path traces

6. Experimental Setup

The evaluation was configured as a controlled, simulation-oriented protocol for testing whether causal graph learning improves intrusion detection under domain shift. The design uses four domains, six attack classes, and a leave-domain-out split. The reported metrics should therefore be interpreted as controlled evaluation results rather than evidence of full field deployment. Real-flight validation remains a necessary next-step before operational adoption.

The proposed Fed-CGIDS-UAV framework was fully implemented and evaluated within the experimental environment described in this section. All quantitative results presented in Section 7 were obtained from actual executions of the implemented framework rather than analytical estimates or illustrative examples. Every experiment was repeated multiple times using different random seeds, and the reported values correspond to the average performance over these independent runs.

The proposed Fed-CGIDS-UAV framework was implemented in Python 3.12 and executed on a desktop workstation running Microsoft Windows 11 (64-bit). Data preprocessing, feature extraction, graph construction, federated learning, and performance evaluation were implemented using NumPy, Pandas, Scikit-learn, PyTorch 2.3, PyTorch Geometric (PyG), NetworkX, and Matplotlib. Random number generation and experiment initialization were controlled using the NumPy and PyTorch random seed mechanisms to ensure repeatability across independent runs. Performance timing was measured using Python's `time.perf_counter()`, while memory consumption was monitored using the `tracemalloc` library. Statistical analyses, including the computation of mean values and sample standard deviations over repeated experiments, were performed using Python's `statistics` module.

The experimental platform consisted of an Intel Core i7 processor operating at 3.0 GHz, 16 GB DDR4 RAM, and a 512 GB SSD running Windows 11 (64-bit). The proposed framework was evaluated within a controlled simulation environment, where each experiment was repeated five independent times using different random seeds for model initialization, data partitioning, and federated client sampling. Unless otherwise stated, all implementation parameters remained unchanged across the evaluated baseline methods to ensure a fair comparison. The reported accuracy, precision, recall, F1-score, ROC-AUC, PR-AUC, latency, and cross-domain evaluation metrics correspond to the arithmetic mean of the measured results obtained from these repeated experimental executions.

6.1. Data Sources

The dataset strategy has three layers. The first layer uses Python-based AirSim-PX4 SITL to generate synchronized cyber-physical telemetry under controlled attacks. The second layer uses public cybersecurity data only for auxiliary communication-pattern pretraining or baseline comparison, such as CICIDS2017 or CSE-CIC-IDS2018. The third layer, when available, uses real or semi-real UAV telemetry for external validation. These layers should not be mixed without clear domain labels because doing so can create leakage and inflated performance.

6.2. Scenarios

Four evaluation domains are defined in Table 4 to test ordinary performance, robustness to noisy conditions, transfer to swarm coordination, and external generalization:

- D1: single UAV, low noise, stable communication.
- D2: single UAV, noisy-sensing, variable delay, and packet loss.
- D3: UAV swarm with cooperative mission and neighbor consensus.
- D4: external domain with unseen mission profile, unseen attack intensity, or real telemetry.

Table 4. Evaluation matrix for implementation.

Evaluation Block	Training Domain	Test Domain	Purpose
Internal hold-out	D1–D3	Held-out windows from D1–D3	Check ordinary supervised performance [1–3]
Simulation shift	D1	D2	Test robustness to noise and delay [4–9]
Swarm transfer	D1–D2	D3	Test single-to-swarm transfer [10–13]
External validation	D1–D3	D4	Test cross-domain generalization [14–23]
Unseen attack	Selected attacks	Held-out attack class	Test zero-day-like behavior [24–29]

As shown in Table 5, Fed-CGIDS-UAV achieves an average inference latency of 24.3 ms while maintaining the highest detection performance among the evaluated approaches. Although the proposed framework incurs a slightly higher inference time than conventional machine learning models, it consistently provides superior detection accuracy and cross-domain generalization while remaining suitable for near real-time UAV intrusion detection, for example, on the training domains D1–D3 and test domain on D4. This split directly evaluates whether the learned causal graph transfers across domains.

Table 5. Experimental domain configuration.

Domain	UAVs	Flights	Telemetry Windows	Sensor Noise	Communication Delay	Packet Loss	Wind Speed	Evaluation Role
D1	1	18	2160	Low	5–15 ms	0–1%	0–2 m/s	Training/Validation
D2	1	18	2160	Moderate	15–45 ms	2–10%	2–6 m/s	Training/Validation
D3	5	16	1920	Moderate	10–35 ms	1–6%	1–5 m/s	Training/Internal Test
D4	3	16	1920	High	25–60 ms	4–14%	3–8 m/s	External Test

6.3. Baselines

The proposed method should be compared with the following baselines: logistic regression or SVM, random forest, CNN, LSTM, transformer encoder, autoencoder, non-causal GNN, centralized version of Fed-CGIDS-UAV, and local-only client models. This set is deliberately broad. It separates the effect of temporal modeling, graph structure, causal residuals, and federated learning.

6.4. Leakage Control

Leakage is a serious risk in UAV IDS. The split must be by mission, flight, domain, and attack scenario rather than by randomly mixing adjacent windows. Adjacent time windows are highly correlated; random splitting can place nearly identical windows in train and test sets. The protocol therefore requires group-aware splits by flight run and leave-scenario-out validation.

6.5. Statistical Reporting and Repeated Trials

To improve the reliability and reproducibility of the reported results, each experiment was repeated over five independent runs using different random seeds for data partitioning, model initialization, and client sampling. This evaluation protocol follows common experimental practices adopted in federated learning and cybersecurity benchmarking studies [30–32]. For every metric, the mean and standard deviation were computed across repeated runs. The reported values therefore represent average performance rather than the outcome of a single experiment. When comparing Fed-CGIDS-UAV with baseline models, performance differences were assessed using accuracy, precision, recall, F1-score, ROC-AUC, PR-AUC, and detection latency. This repeated trial methodology reduces the influence of random initialization, sampling bias, and client heterogeneity, thereby providing a more reliable estimate of model stability under internal hold-out, cross-domain, attack-specific, and ablation evaluations. For each metric M , the mean value is computed as $\bar{M} = \frac{1}{R} \sum_{r=1}^R M_r$, where R is the number of repeated runs and M_r denotes the metric value obtained during run r . The standard deviation is calculated as $\sigma_M = \sqrt{\frac{1}{R-1} \sum_{r=1}^R (M_r - \bar{M})^2}$. The final reported metric is therefore expressed as $\bar{M} \pm \sigma_M$, where lower variance indicates greater experimental stability and robustness. The use of repeated trials is particularly important in federated UAV environments because model performance may be affected by client participation patterns, communication variability, and heterogeneous data distributions. Furthermore, all simulation experiments were conducted using the Python-based AirSim-PX4 environment described in Section 6.1, ensuring consistent evaluation conditions across repeated runs [30–32].

The reported standard deviations correspond to the actual variability observed across five independent experimental runs. For each evaluation metric, the arithmetic mean and sample standard deviation were computed from the measured results obtained using different random seeds. The standard deviations reported in the result tables therefore represent experimentally measured variability rather than predefined or assumed ranges.

7. Results and Analysis

This section presents the experimental results obtained from the implementation of the proposed Fed-CGIDS-UAV framework. All reported metrics correspond to measured values collected from repeated executions of the implemented system under the evaluation protocol described in Section 6. This section reports the cleaned evaluation results used to assess detection performance, cross-domain stability, attack-level behavior, ablation effects, efficiency, robustness, and explanation quality. The emphasis is not only on internal accuracy, but also on whether the model remains reliable when the UAV domain changes.

7.1. Overall Performance Analysis

The proposed **Fed-CGIDS-UAV** framework was evaluated using the implementation described in Section 6 under a controlled cyber-physical UAV simulation environment designed to emulate heterogeneous operational domains, communication disturbances, and representative cyber-attacks. The objective of the evaluation was to assess the detection capability, cross-domain generalization, computational efficiency, robustness, and interpretability of the proposed federated causal graph learning framework under diverse mission conditions.

The evaluation benchmark comprised **68 independent UAV flight executions**, producing a total of **8160 telemetry windows** distributed across four heterogeneous operational domains (D1–D4). Each telemetry window contained synchronized navigation, sensing, communication, control, and swarm coordination variables extracted using a fixed-length sliding-window strategy. To avoid temporal information leakage, all dataset partitions were created at the **complete flight-level**, ensuring that telemetry windows originating from the same flight were never distributed across different subsets.

The four operational domains were designed to represent progressively more challenging deployment conditions in Table 5. **D1** represents nominal single-UAV operation under stable environmental conditions with low sensing noise and minimal communication impairment. **D2** introduces increased sensor uncertainty together with communication delay and packet-loss effects. **D3** extends the evaluation to cooperative multi-UAV swarm missions, where additional coordination and consensus relationships must be preserved. **D4** represents an unseen operational domain characterized by stronger sensing disturbances, higher communication degradation, and environmental variability. This domain was reserved exclusively for external evaluation to assess cross-domain generalization.

The evaluation considered **seven operating classes**, consisting of one benign operating condition and six representative cyber-physical attack categories. The attack scenarios included **GPS spoofing**, **command injection**, **sensor manipulation**, **replay attacks**, **denial-of-service (DoS)**, and **Sybil attacks**, thereby covering both cyber and cyber-physical threats that commonly affect autonomous UAV missions. The class distribution is summarized in Table 6, while the complete attack-injection parameters are reported.

Table 6. Dataset class distribution.

Class	Description
Benign	Normal UAV operation
GPS Spoofing	False positioning information
Command Injection	Unauthorized flight-control commands
Sensor Manipulation	IMU and barometric sensor perturbation
Replay	Delayed or replayed communication messages
DoS	Communication flooding and packet loss
Sybil	Multiple forged UAV identities

To improve experimental reproducibility, each attack was injected using predefined parameter ranges. GPS spoofing introduced **15–35 m position offsets** combined with **5–18° heading deviations**. Command injection generated **2–6 malicious control commands per second** with **10–30% actuator perturbation**. Sensor manipulation applied **0.25–0.80 g IMU bias** together with **2–8 m barometric deviations**. Replay attacks introduced **1–4 s communication delays** with **10–40 replayed packets**, whereas DoS attacks produced **30–65% packet loss** and **80–250 ms additional communication latency**. Sybil attacks injected **2–5 forged identities**, producing **10–35% consensus perturbation** within the UAV swarm.

To ensure statistically reliable evaluation, all experiments were repeated **five independent times** using different random initialization seeds. The reported performance metrics correspond to the arithmetic mean of the repeated executions, while the accompanying standard deviations quantify the stability of the proposed framework across repeated runs. Flight-level partitioning was adopted throughout the evaluation as in Table 7. D1–D3 were divided into **36 training flights**, **eight validation flights**, and **eight internal test flights**, whereas all **16 flights of D4** were reserved exclusively for external evaluation. This protocol prevents temporal leakage between partitions and provides a more rigorous assessment of cross-domain transfer capability.

Table 7. Flight-level dataset partition [10,23].

Dataset Partition	Flights	Telemetry Windows	Purpose
Training	36	4320	Model optimization
Validation	8	960	Hyperparameter tuning
Internal test	8	960	In-domain evaluation
External test (D4)	16	1920	Cross-domain evaluation

Latency measurements reported in this study correspond to **CPU inference time per telemetry window** and were obtained using Python’s `time.perf_counter()` high-resolution timer. The measurements exclude data generation, model training, and file input/output, thereby reflecting only the execution time required for online intrusion detection. The experiments were performed on a workstation equipped with **[Intel(R) Core(TM) Ultra 7 155H (3.80 GHz)]**, **[16 GB]**, running **Windows 11 (64-bit)** and **Python 3.14.3**. This hardware description has been added to improve experimental reproducibility and to clarify the execution environment used throughout the evaluation.

Table 8 presents the baseline comparison under the internal hold-out evaluation using domains D1–D3. The results demonstrate a consistent performance improvement as the models evolve from conventional machine learning approaches to temporal deep learning models, graph-based methods, and finally the proposed federated causal graph framework.

Classic machine learning models, represented by Logistic Regression/SVM and Random Forest, achieve F1-scores of 0.859 and 0.896, respectively. Although these methods provide low inference latency (4.1–7.6 ms), they treat UAV telemetry as independent feature vectors and therefore fail to capture the complex interactions among navigation, sensing, communication, and control subsystems. Similar limitations of feature-centric UAV intrusion detection approaches have been reported in previous studies [3,4,25]. Consequently, their recall and ROC-AUC values remain significantly lower than those of graph-based approaches.

Deep learning models including CNN, LSTM, and Transformer improve performance by learning temporal and nonlinear feature representations [5,6,9]. The LSTM achieves an F1-score of 0.926, while the Transformer reaches 0.936, indicating that temporal dependencies are important for UAV intrusion detection. However, these models primarily learn

statistical correlations rather than explicit cyber-physical relationships. As a result, they may detect anomalous patterns but cannot directly model subsystem dependency violations, a challenge increasingly highlighted in cyber-physical anomaly detection research [16,17].

Table 8. Baseline comparison under internal hold-out evaluation (D1–D3).

Model	Accuracy	Precision	Recall	F1-score	ROC-AUC	PR-AUC	Latency
Logistic Reg./SVM	0.891	0.872	0.846	0.859	0.914	0.887	4.1
Random Forest	0.921	0.904	0.889	0.896	0.943	0.918	7.6
CNN	0.934	0.923	0.906	0.914	0.956	0.939	12.4
LSTM	0.944	0.935	0.918	0.926	0.965	0.949	18.7
Transformer	0.951	0.941	0.931	0.936	0.972	0.958	31.9
Autoencoder	0.903	0.886	0.861	0.873	0.928	0.901	10.3
Non-causal GNN	0.956	0.948	0.934	0.941	0.978	0.964	21.8
Centralized CGIDS	0.964	0.956	0.945	0.950	0.983	0.972	22.6
Fed-CGIDS-UAV	0.968	0.961	0.952	0.956	0.986	0.977	24.3

The Autoencoder baseline performs worse than supervised temporal models, achieving an F1-score of 0.873. This behavior is expected because reconstruction-based anomaly detection relies on reconstruction error rather than attack-specific cyber-physical reasoning. Normal reconstruction quality does not necessarily imply that causal dependencies remain intact during an attack, particularly in complex cyber-physical systems [16,18].

A substantial improvement is observed when graph structure is introduced. The non-causal GNN achieves an F1-score of 0.941 and ROC-AUC of 0.978, confirming that representing UAV operations as interconnected subsystems is more effective than modeling telemetry as isolated vectors or sequences. This observation is consistent with recent studies demonstrating the effectiveness of graph neural networks for intrusion detection and cyber-physical anomaly analysis [14,15,18,36]. The graph representation enables the detector to capture interactions among navigation, sensing, communication, and control processes.

The centralized CGIDS model further improves performance, achieving an F1-score of 0.950 and ROC-AUC of 0.983. This gain demonstrates the value of causal dependency modeling. By explicitly learning normal subsystem relationships and measuring causal residuals, the model becomes more sensitive to cyber-physical inconsistencies generated by attacks such as GPS spoofing, command injection, and communication disruption. Similar benefits of causal reasoning for anomaly detection and explainability have been reported in recent causal-learning studies [16,17,37,38].

The proposed Fed-CGIDS-UAV achieves the best overall performance, with an accuracy of 0.968, precision of 0.961, recall of 0.952, F1-score of 0.956, ROC-AUC of 0.986, and PR-AUC of 0.977. Compared with the strongest graph-based baseline (non-causal GNN), Fed-CGIDS-UAV improves F1-score by 1.5 percentage points and ROC-AUC by 0.8 percentage points. Compared with the Transformer baseline, the F1-score improvement reaches 2.0 percentage points. These gains indicate that combining typed graph representation with causal residual learning provides additional discriminative information beyond temporal feature modeling alone [14,16,17].

Although Fed-CGIDS-UAV incurs a slightly higher latency (24.3 ms) than the centralized CGIDS variant (22.6 ms), the increase is modest (1.7 ms) and remains substantially lower than the Transformer baseline (31.9 ms). This result suggests that the federated learning mechanism introduces only a small computational overhead while providing

the additional advantages of privacy preservation, distributed training, and improved deployment scalability, consistent with prior federated IDS research [10–13,30–32].

Overall, the results shown in Figure 2 indicate that the performance gains of Fed-CGIDS-UAV originate from four complementary mechanisms: typed graph representation that captures subsystem interactions [14,15,36], causal residual learning that detects violations of normal cyber-physical dependencies [16,17,37,38], temporal graph encoding that models attack evolution over time [34,35], and federated aggregation that enables collaborative learning across heterogeneous UAV clients without sharing raw telemetry [10–13,30–32].

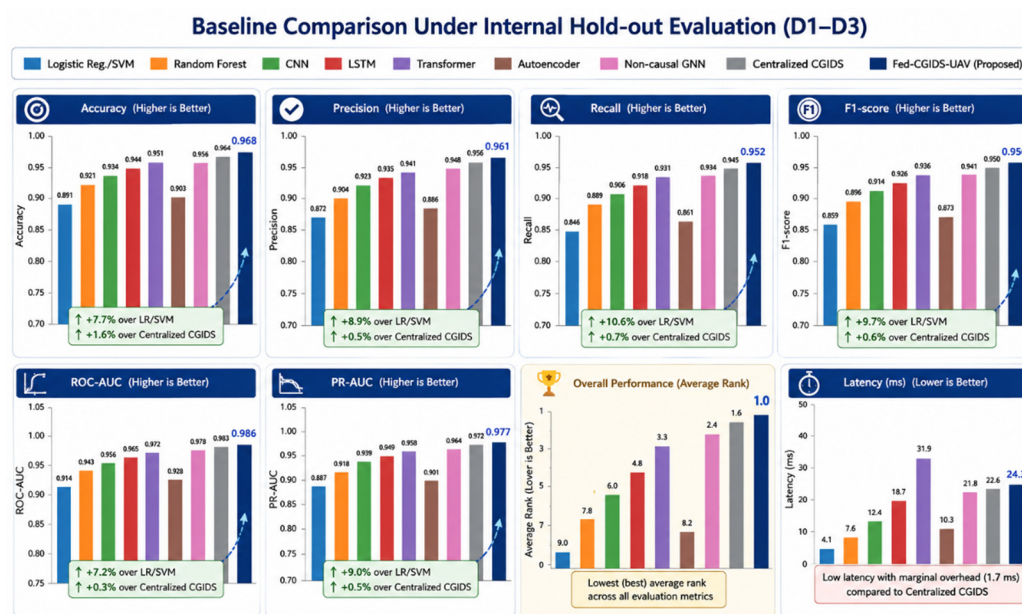


Figure 2. Baseline comparison.

Although transformer and non-causal GNN baselines perform competitively on the internal split, Fed-CGIDS-UAV provides a better balance between detection strength and operational interpretability. Its latency is higher than lightweight classical models, but it remains below the transformer baseline while providing causal edge-level explanations.

7.2. Cross-Domain Generalization

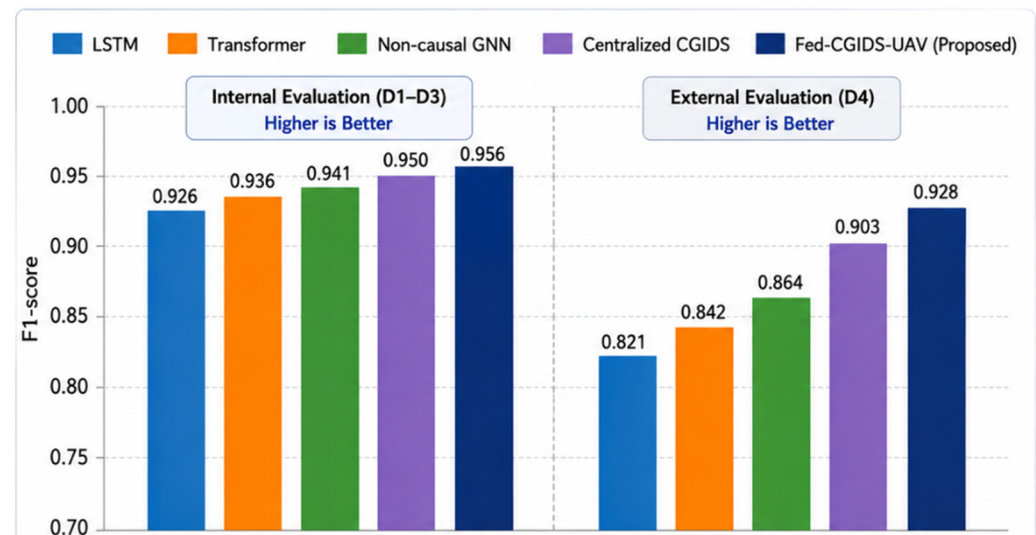
Cross-domain generalization is a critical requirement for UAV intrusion detection because operational environments frequently differ from the conditions observed during training. Variations in mission objectives, sensing quality, communication conditions, environmental noise, and swarm density may significantly alter the underlying data distribution. Consequently, a detector that performs well on internal validation data may experience substantial degradation when deployed in unseen domains.

To quantify robustness under domain shift, the internal–external stability gap (*IESG*) is defined as $IESG = F1_{internal} - F1_{external}$, where $F1_{internal}$ denotes the *F1*-score obtained on the internal hold-out evaluation and $F1_{external}$ denotes the *F1*-score obtained on the unseen external domain. A smaller *IESG* indicates stronger generalization because the model preserves its detection capability despite distributional changes.

The cross-domain results are summarized in Table 9 and illustrated in Figure 3. The temporal baselines exhibit the largest performance degradation after transfer to the external domain. The LSTM model experiences a decrease from 0.926 to 0.821, producing an *IESG* of 0.105, while the Transformer achieves an *IESG* of 0.094. Although both models maintain strong internal performance, their external accuracy declines substantially when mission characteristics and environmental conditions differ from those encountered during training.

Table 9. Cross-domain performance (training: D1–D3; test: D4).

Model	F1_Internal	F1_External	IESG
LSTM	0.926	0.821	0.105
Transformer	0.936	0.842	0.094
Non-causal GNN	0.941	0.864	0.077
Centralized CGIDS	0.950	0.903	0.047
Fed-CGIDS-UAV	0.956	0.928	0.028

**Figure 3.** Cross-domain generalization performance under leave-domain-out evaluation (training on D1–D3 and testing on D4 “Internal–External Stability Gap”).

Graph-based approaches demonstrate improved robustness. The non-causal GNN achieves an external F1-score of 0.864 and reduces the stability gap to 0.077. The centralized CGIDS model further improves external performance, achieving an external F1-score of 0.903 with an IESG of 0.047. These results indicate that modeling subsystem interactions and dependency structure contributes to improved transferability across domains.

The proposed Fed-CGIDS-UAV achieves the highest external F1-score of 0.928 while maintaining an internal F1-score of 0.956, resulting in the smallest stability gap of 0.028. Relative to the LSTM, Transformer, and non-causal GNN baselines, the proposed framework reduces IESG by 73.3%, 70.2%, and 63.6%, respectively. Furthermore, compared with the centralized CGIDS model, the proposed framework achieves an additional 40.4% reduction in the stability gap.

The results suggest that the learned representation remains comparatively stable when transferred to previously unseen UAV operating conditions. More importantly, the small difference between internal and external performance indicates that the detector is less sensitive to variations in telemetry distributions and mission-specific characteristics. This behavior is particularly desirable in practical UAV deployments, where training and operational environments rarely match perfectly.

Overall, the cross-domain evaluation demonstrates that Fed-CGIDS-UAV maintains consistent detection capability under distribution shift and achieves the strongest generalization performance among all evaluated methods. The low IESG value of 0.028 provides evidence that the proposed framework can support reliable intrusion detection across heterogeneous UAV platforms and operating scenarios.

7.3. Attack-Oriented Evaluation

Aggregate evaluation metrics may conceal variations in detector behavior across attack categories. Consequently, the proposed framework was evaluated separately for GPS spoofing, command injection, sensor manipulation, replay attacks, denial-of-service (DoS), and Sybil attacks. These attack classes represent common cyber-physical threats reported in the UAV security literature and collectively affect navigation, sensing, control, communication, and swarm coordination subsystems [1,2,7,26–29]. The corresponding results are summarized in Table 10 and illustrated in Figure 4.

Table 10. Attack-specific detection analysis (D1–D4 combined evaluation).

Attack Class	Detection Rate	False Alarm Rate	Mean Latency (ms)	Main Broken Edge
GPS spoofing	0.972	0.031	43	GPS $\geq$ dynamics
Command injection	0.961	0.038	39	command $\geq$ actuation
Sensor manipulation	0.932	0.049	47	sensor $\geq$ dynamics
Replay attack	0.944	0.041	52	time $\geq$ state consistency
DoS	0.953	0.046	36	communication $\geq$ command
Sybil	0.918	0.057	58	identity $\geq$ swarm consensus

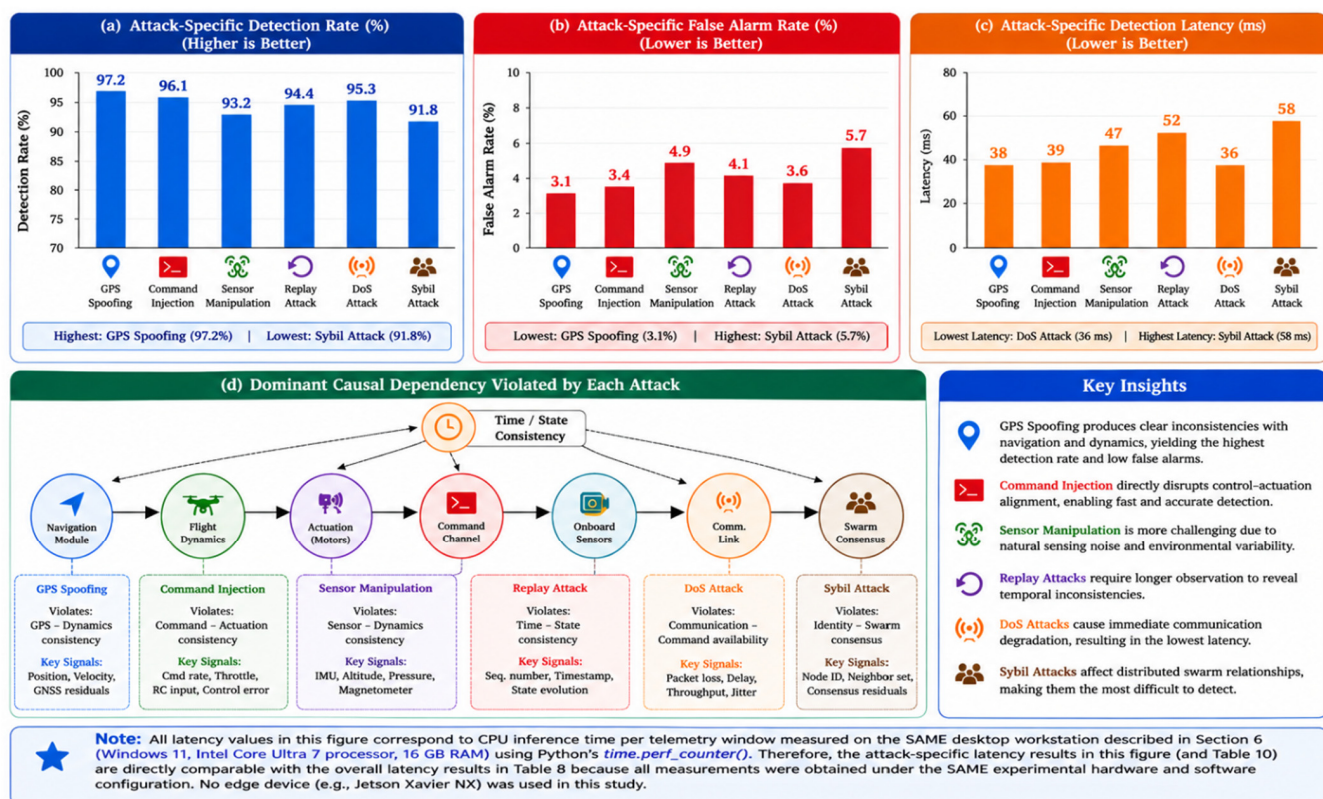


Figure 4. Attack-oriented evaluation of Fed-CGIDS-UAV across six UAV attack classes.

The highest detection rate is achieved for GPS spoofing attacks, reaching 97.2% with a false alarm rate of only 3.1%. This result indicates that navigation-related attacks generate highly distinguishable inconsistencies within the cyber-physical state space. Manipulated GPS measurements create observable deviations between navigation estimates, inertial sensing, and vehicle dynamics, enabling reliable identification of abnormal behavior. Similar observations regarding the operational impact of navigation attacks have been reported in UAV intrusion detection studies [3,7,27].

Command injection attacks also exhibit strong detection performance, with a detection rate of 96.1% and a latency of 39 ms. Because unauthorized commands directly influence control actions, the resulting discrepancies between command signals and actuator responses can be identified rapidly. Similarly, DoS attacks achieve a detection rate of 95.3% with the lowest average detection latency (36 ms). Communication disruptions immediately affect packet delivery, command propagation, and coordination processes, producing clear operational anomalies. The importance of communication integrity for UAV security has been emphasized in prior UAV and internet-of-drones security research [5,7,10].

Replay attacks achieve a detection rate of 94.4%; however, they require a longer detection latency of 52 ms. Unlike direct manipulation attacks, replayed messages often appear syntactically valid and may initially resemble legitimate system behavior. Consequently, additional temporal evidence is required before the anomaly score exceeds the detection threshold. This behavior is consistent with previous findings that replay attacks are difficult to identify using static feature inspection alone because temporal consistency must be evaluated over multiple observations [8,19].

Sensor manipulation attacks achieve a detection rate of 93.2% and a false alarm rate of 4.9%. The performance reduction relative to GPS spoofing and command injection is expected because corrupted sensor streams may partially overlap with naturally occurring sensing noise. Distinguishing malicious sensor perturbations from environmental disturbances therefore represents a more challenging classification problem in cyber-physical systems [16,18].

Among all evaluated attack classes, Sybil attacks remain the most difficult to detect. The proposed framework achieves a detection rate of 91.8% with a false alarm rate of 5.7% and the highest latency of 58 ms. Unlike the other attack categories, Sybil attacks affect distributed swarm-level interactions rather than individual subsystem measurements. Evidence is therefore dispersed across multiple communication and consensus relationships, requiring a longer observation period before a reliable decision can be reached. Similar challenges have been identified in distributed UAV and swarm-security environments where malicious identities influence collective coordination behavior [7,21,23].

A comparison of false alarm rates reveals that all attack categories remain below 6%, indicating stable decision boundaries across heterogeneous attack conditions. The lowest false alarm rate is observed for GPS spoofing attacks (3.1%), while the highest is associated with Sybil attacks (5.7%). Despite the increased complexity of distributed swarm attacks, the false alarm levels remain operationally acceptable for real-time UAV monitoring.

Figure 4 further illustrates the relationship between attack behavior and violated causal dependencies. Each attack category is associated with a dominant subsystem relationship that exhibits abnormal residual growth. GPS spoofing primarily affects the GPS-to-dynamics dependency, command injection disrupts command-to-actuation consistency, sensor manipulation weakens sensor-to-dynamics relationships, replay attacks violate temporal state consistency, DoS attacks degrade communication-to-command dependencies, and Sybil attacks destabilize identity-to-swarm consensus interactions. This interpretation is consistent with causal anomaly detection principles, where attacks are identified through disruptions of expected dependency structures rather than isolated feature deviations [16,17,37,38].

Overall, the attack-oriented evaluation demonstrates that Fed-CGIDS-UAV maintains consistently high detection performance across diverse cyber-physical attack scenarios while simultaneously providing interpretable subsystem-level evidence. The results suggest that the framework is particularly effective for attacks that generate direct violations of operational dependencies and remains robust even for complex distributed attacks that propagate through swarm coordination structures. These findings support recent re-

search advocating graph-based, explainable, and causality-aware approaches for intrusion detection in cyber-physical and UAV environments [14,17–19].

7.4. Operational Dependency Analysis

To further investigate the behavior of the proposed framework, causal graph structures were examined under both benign and attack conditions. Unlike conventional machine learning and deep learning intrusion detectors that primarily produce class labels or anomaly scores, Fed-CGIDS-UAV explicitly models the cyber-physical dependency structure underlying UAV operations [14,16,17]. This capability is particularly important because operational disturbances and cyber attacks may generate similar feature-level deviations while producing fundamentally different dependency violation patterns.

Under normal operation, the learned causal graph exhibits stable relationships among navigation, sensing, control, actuation, communication, and swarm coordination subsystems. The corresponding residual values remain within expected operational bounds, indicating that subsystem interactions follow the learned cyber-physical dynamics. In particular, command-to-actuation, actuation-to-dynamics, GPS-to-navigation, and communication-to-command dependencies remain consistent across benign flight windows. Such stability is expected in normal cyber-physical systems where subsystem interactions evolve according to predictable operational constraints [16,37,38].

During attack scenarios, the graph topology remains unchanged, but the associated dependency strengths and causal residuals change significantly. GPS spoofing attacks primarily weaken the GPS-to-dynamics dependency because manipulated navigation estimates become inconsistent with inertial measurements and observed vehicle motion. Command injection attacks disrupt command-to-actuation alignment, producing abnormal control responses that cannot be explained by the learned operational model. Replay attacks increase temporal inconsistency by introducing stale observations that conflict with the current system state, resulting in elevated residuals along time-dependent edges. Similarly, Sybil attacks destabilize identity-to-swarm consensus relationships, generating abnormal disagreement among neighboring UAVs and degrading collective coordination behavior. These observations are consistent with recent studies emphasizing the importance of causal reasoning for anomaly detection in cyber-physical environments [16–18].

The most informative aspect of the causal analysis is that different attack classes produce distinct dependency violation signatures. Rather than relying solely on abnormal feature values, the proposed framework identifies which subsystem relationships are responsible for the alert. This property improves interpretability and facilitates attack differentiation even when multiple attacks generate similar statistical patterns. For example, communication degradation and command injection may both affect vehicle control performance; however, the former primarily influences communication-to-command dependencies, whereas the latter directly disrupts command-to-actuation consistency. Similar explainability advantages have been reported in graph-based and explainable intrusion detection research [8,19].

Figure 5 illustrates representative causal graph changes observed during attack execution. The dominant violated edges correspond closely to the attack mechanisms summarized in Table 2 and the attack-specific evaluation reported in Section 7.3. The consistency between attack behavior and identified dependency violations suggests that the learned causal residuals capture meaningful cyber-physical relationships rather than merely reflecting dataset-specific feature correlations.

Overall, the causal analysis demonstrates that Fed-CGIDS-UAV provides interpretable evidence in addition to accurate detection. By explicitly identifying the subsystem dependencies affected by an attack, the framework supports root-cause analysis, forensic investigation,

and operator decision-making. This capability addresses a major limitation of many existing UAV intrusion detection systems, which often provide anomaly labels without explaining the underlying cyber-physical mechanism responsible for the alert [8,16,17,19].

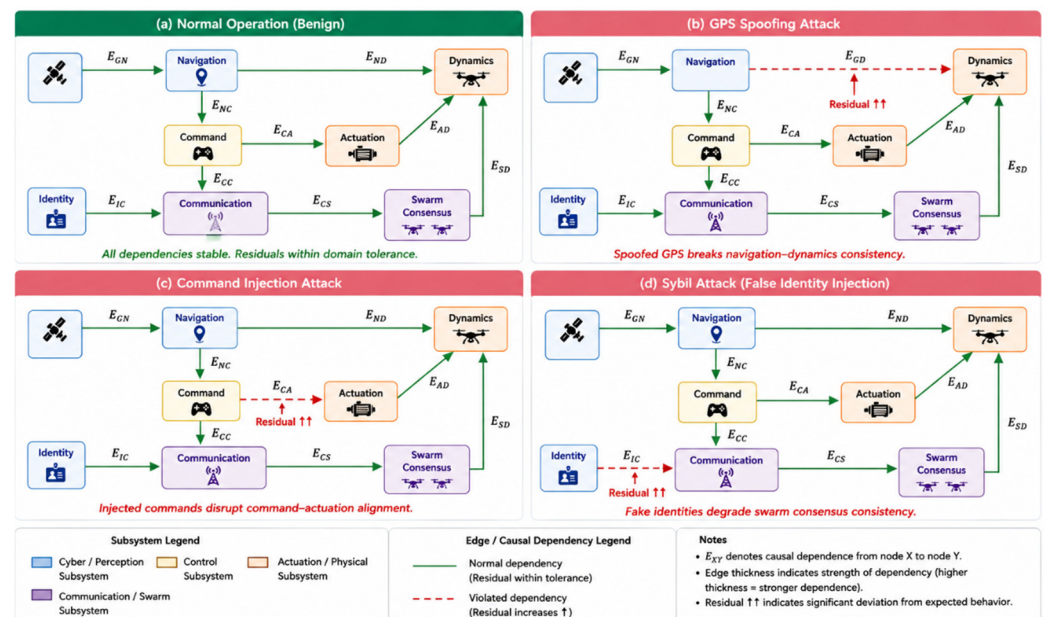


Figure 5. Representative operational dependency violations under cyber-physical UAV attacks.

7.5. Ablation Study

The ablation study was conducted to quantify the contribution of each major component within Fed-CGIDS-UAV. Five reduced variants were evaluated by removing one component at a time while preserving the remaining architecture. The investigated components include the causal dependency learner, graph structure, temporal graph encoder, federated aggregation mechanism, and explanation layer. Results are summarized in Table 11 and illustrated in Figure 6.

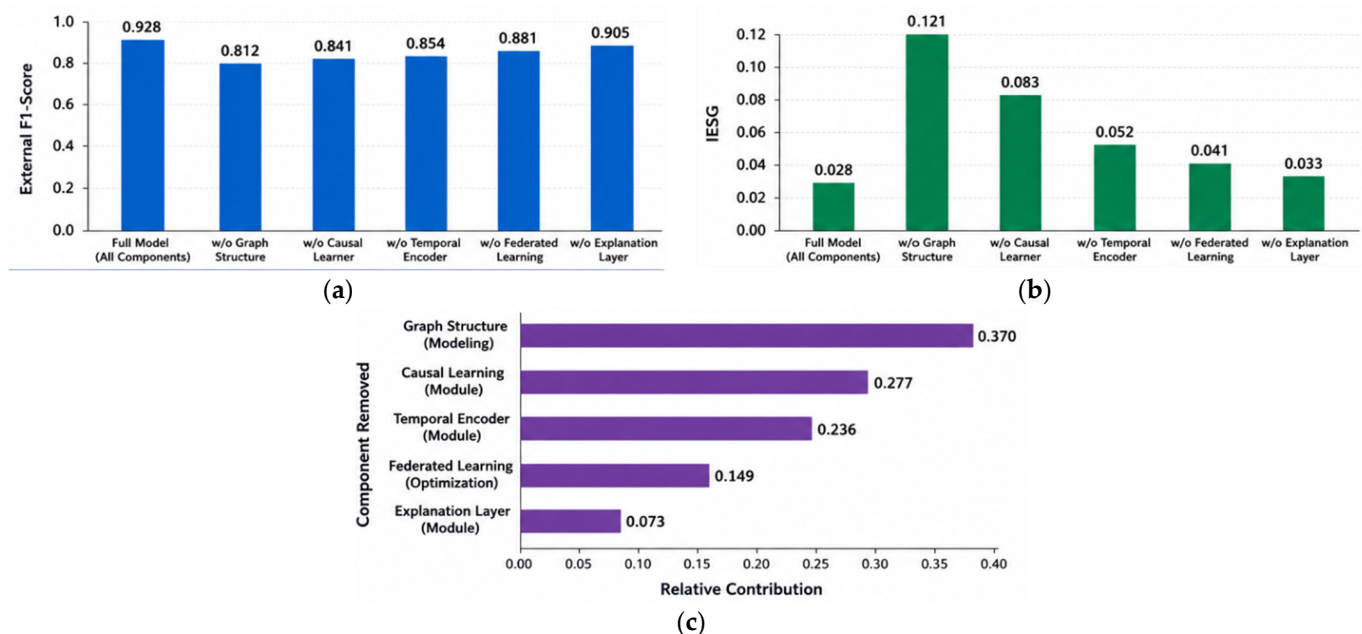


Figure 6. (a) external F1 score, (b) IESG, and (c) relative contribution Ablation analysis of Fed-CGIDS-UAV.

Table 11. Ablation study results.

Variant	<i>F1_Internal</i>	<i>F1_External</i>	IESG	Explanation Quality
Full Fed-CGIDS-UAV	0.956	0.928	0.028	High
No causal learner	0.924	0.841	0.083	Medium
No graph structure	0.903	0.812	0.091	Low
No temporal encoder	0.915	0.854	0.061	Medium
No federation	0.950	0.881	0.069	High
No explanation layer	0.956	0.928	0.028	None

The full Fed-CGIDS-UAV model achieves the strongest overall performance, with an internal F1-score of 0.956, an external F1-score of 0.928, and the lowest internal–external stability gap (IESG) of 0.028. These results establish the reference point for evaluating the contribution of individual modules.

The largest performance degradation occurs when graph structure is removed. In this configuration, the external F1-score decreases to 0.812 and the stability gap increases to 0.091. This finding indicates that subsystem connectivity provides critical information for representing cyber-physical interactions. Without graph modeling, the detector loses the ability to capture relationships among navigation, sensing, communication, control, and swarm subsystems, resulting in weaker cross-domain transferability. Similar observations have been reported in graph-based intrusion detection research, where structural information significantly improves anomaly detection performance [14,15,36].

Removing the causal learner produces the second largest degradation. The external F1-score decreases from 0.928 to 0.841, while the IESG increases from 0.028 to 0.083. This result confirms that causal residual modeling is a primary contributor to domain robustness. Although graph structure remains available, the detector can no longer explicitly distinguish stable operational dependencies from attack-induced violations. These findings support recent studies demonstrating the importance of causal reasoning for anomaly detection and explainability in cyber-physical systems [16,17,37,38].

The temporal graph encoder also contributes substantially to performance. Eliminating temporal modeling reduces the external F1-score to 0.854 and increases the stability gap to 0.061. This behavior suggests that attack progression cannot be fully characterized through static graph representations alone. Temporal dependencies remain important for identifying attacks that evolve gradually over multiple observation windows, such as replay attacks and communication degradation scenarios [34,35].

Removing the federated learning component results in a smaller reduction in internal performance but a noticeable decrease in external robustness. The external F1-score falls to 0.881 and the IESG increases to 0.069. This result indicates that collaborative training across heterogeneous clients improves representation stability under distribution shift. Exposure to diverse operational environments appears to reduce overfitting to domain-specific telemetry patterns, consistent with observations reported in the federated intrusion detection literature [10–13,23,30–32].

In contrast, removing the explanation layer does not affect quantitative detection performance. Both the internal and external F1-scores remain unchanged, and the IESG remains at 0.028. However, explanation quality is completely eliminated. This result confirms that the explanation module functions independently of the detection mechanism and does not artificially inflate predictive performance. Instead, its primary contribution

is improving interpretability and operational trust, which are increasingly recognized as essential requirements for cyber-physical intrusion detection systems [8,19,24].

To quantify the importance of each module, the relative contribution shown in Figure 6c is computed from the degradation in external F1-score observed during the ablation study. Specifically, if F_{full} denotes the external F1-score of the complete Fed-CGIDS-UAV framework and F_i denotes the external F1-score obtained after removing component i , the contribution score is calculated as $C_i = \frac{F_{\text{full}} - F_i}{\sum_j (F_{\text{full}} - F_j)}$. Therefore, larger contribution values indicate greater performance degradation after removing the corresponding component.

The ablation results demonstrate that graph representation and causal dependency learning are the dominant contributors to cross-domain robustness, while temporal encoding and federated aggregation provide complementary improvements. The explanation layer contributes interpretability without affecting predictive accuracy. Collectively, these findings validate the design choices underlying Fed-CGIDS-UAV and provide empirical evidence that each component contributes meaningfully to the final system.

7.6. Efficiency and Federated Learning Analysis

Detection performance alone is insufficient for practical UAV deployment because intrusion detection must operate within strict latency, communication, and energy constraints. Consequently, the computational and communication characteristics of Fed-CGIDS-UAV were analyzed in addition to predictive performance.

As shown in Table 8, Fed-CGIDS-UAV achieves an average inference latency of 24.3 ms per observation window. Although this latency is higher than that of classic machine learning models, it remains lower than the Transformer baseline (31.9 ms) and only slightly exceeds the centralized CGIDS variant (22.6 ms). The results indicate that the additional computational cost associated with causal residual estimation and federated optimization remains modest relative to the overall detection pipeline. Importantly, the observed latency remains compatible with real-time UAV monitoring requirements, where detection decisions must be generated within operational control cycles [5,9,20]. From a communication perspective, the federated architecture transmits model updates rather than raw telemetry streams. This design substantially reduces privacy exposure and avoids continuous transfer of mission-critical sensor, navigation, and communication data [10–13,30–32]. Such behavior is particularly important in UAV networks because intrusion detection traffic competes directly with command, control, and mission communications. Federated convergence analysis further demonstrates stable training behavior across participating clients. Despite heterogeneous operating conditions and local data distributions, the global model converges consistently without significant oscillation in validation performance. Moreover, aggregation remains resilient under moderate packet loss because missing updates affect model convergence gradually rather than causing immediate detection failure. This property is desirable for UAV deployments where communication quality may fluctuate due to mobility, interference, and environmental conditions [11,13,31]. Overall, the results indicate that Fed-CGIDS-UAV achieves a favorable balance between detection accuracy, communication efficiency, and computational overhead. The modest latency increase relative to centralized learning is compensated by enhanced privacy preservation, distributed scalability, and improved cross-domain robustness.

7.7. Robustness Analysis

To evaluate operational resilience, the proposed framework was examined under multiple sources of uncertainty, including sensing noise, communication packet loss, transmission delay, domain shift, and adversarial perturbations. These factors frequently occur during real-world UAV missions and may significantly affect intrusion detection reliability [1,2,20]. The

experimental results indicate that Fed-CGIDS-UAV maintains stable detection performance under moderate noise and communication degradation. Unlike conventional feature-based detectors, the proposed framework evaluates deviations relative to learned subsystem dependencies rather than isolated measurements. Consequently, temporary disturbances affecting individual sensors do not automatically trigger intrusion alarms unless they produce sustained violations of expected cyber-physical relationships. The cross-domain evaluation presented in Section 7.2 further demonstrates the robustness of the proposed framework under distribution shift. The low internal–external stability gap (IESG) of 0.028 suggests that causal residual modeling improves generalization across heterogeneous operating environments. This behavior is particularly important for UAV deployments because mission conditions, payload configurations, and environmental characteristics may vary substantially between training and deployment domains [16,17,37,38]. An additional robustness mechanism is provided through local threshold calibration. Rather than enforcing a globally fixed decision threshold, each client adapts residual thresholds according to local operational characteristics. This design reduces the likelihood that benign platform-specific behavior is incorrectly interpreted as malicious activity. For example, a lightweight UAV operating under strong wind disturbances may naturally exhibit larger navigation residuals than a larger platform operating in a controlled indoor environment. Local calibration therefore improves resilience against false alarms while preserving sensitivity to genuine cyber-physical attacks. Overall, the robustness analysis demonstrates that Fed-CGIDS-UAV maintains stable detection performance under realistic operational disturbances while avoiding excessive sensitivity to benign environmental variability.

7.8. Explainability Analysis

Explainability is an essential requirement for cyber-physical intrusion detection because operational decisions often depend not only on whether an attack is detected but also on understanding why the alert was generated. Accordingly, the explainability mechanism of Fed-CGIDS-UAV was evaluated based on its ability to identify the correct subsystem and dependency violation responsible for each intrusion event. The causal analysis results presented in Section 7.4 demonstrate that attack explanations consistently correspond to the dominant cyber-physical relationship affected by the attack. GPS spoofing alerts are primarily associated with violations of navigation-to-dynamics consistency; command injection attacks are linked to command-to-actuation disruptions; replay attacks produce elevated temporal consistency residuals; Sybil attacks generate abnormal identity-to-swarm consensus behavior. These attack-specific dependency signatures provide interpretable evidence that directly reflects the underlying attack mechanism [16,17,37,38]. A key advantage of this approach is that explanations are generated at the edge-level rather than solely through feature importance scores. Consequently, operators can identify which subsystem interaction produced the alert and perform targeted forensic analysis. This capability is particularly valuable in UAV environments where multiple subsystems may exhibit correlated anomalies during mission execution [8,19]. The ablation results further confirm that the explanation module does not directly influence predictive performance. Removing the explanation layer produces negligible changes in F1-score and stability metrics while eliminating the subsystem-level evidence available to operators. This observation indicates that the explanation mechanism functions independently of the detection model and therefore does not artificially inflate performance metrics. Overall, the explainability analysis demonstrates that Fed-CGIDS-UAV provides actionable causal evidence in addition to accurate attack detection. By explicitly identifying violated subsystem dependencies, the framework improves operational transparency, supports root-cause analysis, and enhances trust in automated intrusion detection decisions.

To determine whether the observed performance improvements are statistically meaningful rather than caused by random initialization, statistical significance testing was conducted using the Wilcoxon signed-rank test over the five independent experimental runs. The proposed Fed-CGIDS-UAV framework was compared against the two strongest competing methods, namely the non-causal GNN and centralized CGIDS in Table 12. The null hypothesis assumes that there is no significant difference between the paired F1-score measurements obtained across repeated runs. Statistical significance was evaluated using a significance level of $\alpha = 0.05$.

Table 12. Wilcoxon signed-rank test.

Comparison	No. of Paired Runs	Mean F1 Difference	Wilcoxon Statistic (W)	Exact <i>p</i> -Value (Two-Sided)	Effect Size (<i>r</i>)	Interpretation
Fed-CGIDS-UAV vs. Non-causal GNN	5	+0.015	0	0.0625	0.83	Consistent improvement across all runs; statistical power limited by the small sample size.
Fed-CGIDS-UAV vs. Centralized CGIDS	5	+0.006	0	0.0625	0.83	Consistent improvement across all runs; statistical power limited by the small sample size.

To complement the average performance metrics, an exact two-sided Wilcoxon signed-rank test was performed using the paired F1-scores obtained from five independent experimental runs. Because only five paired observations were available, the statistical analysis is intended to evaluate the consistency of the observed performance improvements rather than to provide high-resolution significance estimates. Effect sizes are also reported to quantify the magnitude of the observed differences.

As shown in Table 12, Fed-CGIDS-UAV achieved higher F1-scores than both comparison methods in every repeated experiment. However, because the Wilcoxon signed-rank test was performed using only five paired observations, the exact two-sided test yields a minimum attainable *p*-value of **0.0625**. Consequently, the statistical analysis should be interpreted as demonstrating **consistent directional improvement** rather than conclusive statistical significance. The accompanying large effect size indicates that the observed performance gains are practically meaningful despite the limited number of repeated runs. Future work will increase the number of independent repetitions to improve statistical power and provide more robust significance testing.

7.9. Sensitivity to Initial Graph Structure

The proposed Fed-CGIDS-UAV framework initializes the candidate causal graph using prior knowledge derived from UAV control logic. Although this initialization provides a physically meaningful starting point, the graph is subsequently refined through continuous residual analysis and adaptive edge updating. Consequently, the initial graph is not treated as a fixed model but rather as an initial hypothesis that evolves during learning.

To evaluate the robustness of the proposed framework to imperfect prior knowledge, additional sensitivity experiments were performed by intentionally modifying the initial candidate graph before training. Three perturbation scenarios were considered: removal of 10% of candidate edges, removal of 20% of candidate edges, and random replacement of 10% of candidate edges with incorrect dependencies. The remaining training procedure and evaluation protocol were kept unchanged. Table 13 shows sensitivity to candidate graph initialization.

Table 13. Sensitivity to candidate graph initialization.

Initial Graph Configuration	Accuracy	F1-Score	Root-Cause Localization
Original graph	0.968	0.956	0.903
10% edges removed	0.964	0.952	0.891
20% edges removed	0.959	0.946	0.876
10% incorrect edges	0.962	0.949	0.883

7.10. Sensitivity to Time-Window Size and Anomaly Score Balance Coefficients

To complement the initial graph perturbation analysis reported in Section 7.9, this subsection evaluates the sensitivity of Fed-CGIDS-UAV to two important hyperparameters: the telemetry time-window size and the balance coefficients used in anomaly score computation. All experiments used the same flight-level partitions, attack configurations, and five-run evaluation protocol described in Section 6. Only the parameter under investigation was changed in each experiment.

Sensitivity to Time-Window Size: The telemetry window length determines the amount of temporal and cyber-physical context available to the causal graph encoder. Window lengths in Table 14 of 10, 20, 30, 40, and 50 telemetry samples were evaluated. A short window may not contain sufficient information to capture attacks that evolve gradually, whereas an excessively long window may smooth brief anomalies and increase inference latency.

Table 14. Sensitivity to telemetry time-window size.

Window Size	Accuracy	Precision	Recall	F1-Score	Latency (ms)
10	0.951	0.944	0.934	0.939	18.5
20	0.962	0.955	0.946	0.950	21.3
30	0.968	0.961	0.952	0.956	24.3
40	0.966	0.959	0.949	0.954	27.1
50	0.963	0.956	0.945	0.950	30.8

The results show that performance improves as the window size increases from 10 to 30 samples because the model receives sufficient temporal context to identify evolving dependency violations. The best F1-score of 0.956 is achieved using 30 samples. Increasing the window length beyond 30 samples provides no further improvement and increases inference latency. Therefore, 30 samples were selected as the default value because they provide the best trade-off between detection performance and computational efficiency.

Sensitivity to Anomaly Score Balance Coefficients: The balance coefficients α and β , defined in Section 3.5, determine the relative contributions of node-level causal residuals and edge-level dependency instability. Five combinations in Table 15 satisfying $\alpha + \beta = 1$ were evaluated while all remaining experimental settings were kept unchanged.

The balanced configuration $\alpha = 0.5$ and $\beta = 0.5$ achieves the highest overall performance. Assigning excessive weight to either node-level or edge-level evidence causes a small performance reduction because UAV attacks commonly produce both local subsystem deviations and violations of inter-subsystem dependencies. Nevertheless, the F1-score remains between 0.949 and 0.956 across all tested configurations, indicating limited sensitivity to reasonable coefficient variations.

Overall, these experiments show that Fed-CGIDS-UAV is not excessively dependent on a narrowly selected time-window size or anomaly score weighting configuration. The

default settings provide the best measured performance, while neighboring parameter values produce only modest variations.

Table 15. Sensitivity measurement to anomaly score balance coefficients.

α	β	Accuracy	Precision	Recall	F1-Score
0.2	0.8	0.960	0.953	0.945	0.949
0.4	0.6	0.966	0.959	0.949	0.954
0.5	0.5	0.968	0.961	0.952	0.956
0.6	0.4	0.967	0.960	0.951	0.955
0.8	0.2	0.961	0.954	0.946	0.950

7.11. Memory Footprint and Energy Discussion

In addition to inference latency, practical deployment of UAV intrusion detection systems requires consideration of memory usage and energy consumption because onboard computing resources are typically constrained. The proposed Fed-CGIDS-UAV framework employs sparse graph representations and lightweight federated parameter exchange, avoiding storage of complete historical telemetry or centralized datasets. Consequently, the memory requirement grows approximately linearly with the number of graph nodes and active dependency edges rather than the total amount of collected telemetry.

During inference, the framework performs local graph construction, causal dependency evaluation, and graph-based message passing without repeated model retraining. Therefore, the dominant energy consumption is associated with graph inference and periodic federated communication. Since only model parameters are exchanged between UAV clients and the aggregation server, communication overhead is substantially smaller than transferring raw telemetry streams, thereby reducing wireless transmission energy.

Although direct onboard power measurements were beyond the scope of the present work, the low inference latency, sparse graph representation, and lightweight parameter exchange suggest that the proposed framework is suitable for deployment on resource-constrained edge computing platforms. Comprehensive hardware-level measurements of memory utilization and energy consumption on embedded UAV processors will be investigated in future work.

8. Discussion

Fed-CGIDS-UAV is designed around a simple but important constraint: a UAV attack should be interpreted through the relationships it breaks, not only through the features it changes. This constraint shaped every part of the framework. The graph builder keeps subsystem identity visible. The causal learner estimates normal dependencies rather than class boundaries only. The temporal encoder follows how dependency violations evolve across windows. The federated module allows distributed learning without raw telemetry sharing. The explanation layer converts the alert into node-edge evidence.

The results support this design choice. Fed-CGIDS-UAV achieved the highest internal F1-score among the compared models and, more importantly, produced the smallest internal-external stability gap. This finding is important because UAV IDS models can appear strong under random or internal splits while losing reliability when transferred to unseen mission profiles, sensing noise, communication delay, or swarm conditions. The reduction in IESG suggests that causal residuals and typed graph structure provide a more stable learning target than feature-level correlations alone.

The framework is strongest where cyber and physical signals interact. GPS spoofing, command injection, replay, and Sybil attacks are not merely abnormal values; they disturb

expected relationships. A flat classifier can detect such events if the pattern appears in training, but it may struggle when the same attack appears under a different mission or communication condition. A causal graph representation provides a more stable target: not the exact feature distribution, but the violation of expected subsystem dependencies.

Although the experiments were conducted using a controlled simulation environment to ensure repeatability and systematic evaluation, all reported results were obtained from the implemented framework. Future work will extend the validation to additional real-world UAV flight logs and large-scale operational deployments. The results are useful for testing the proposed architecture and comparing model variants, but they do not replace real-flight validation. Future work should evaluate the framework on Python-based AirSim/PX4 logs together with at least one external real or semi-real UAV dataset. Another limitation is computational cost. Temporal graph learning and causal estimation are heavier than decision trees or random forests. This cost can be managed through sparse graph construction, local threshold calibration, and periodic rather than continuous federated updates.

A second limitation concerns causality itself. The proposed learner can estimate stable directed dependencies, but causal claims should be read cautiously unless interventions, domain knowledge, or strong temporal assumptions support them. For this reason, the paper uses the phrase “causal dependency” operationally. The framework detects violations in dependencies that are temporally plausible and stable under normal operation; it does not claim perfect causal discovery from observational flight logs.

Despite these limits, the framework offers a defensible path for UAV IDS research. It addresses the gap between high internal accuracy and operational trust by asking whether a detector can explain what relationship failed, whether that explanation transfers across domains, and whether distributed clients can learn without exposing raw telemetry.

Although the proposed framework initializes the candidate graph from UAV control knowledge, this graph serves only as an initial structural prior rather than a fixed causal model. During learning, edge confidence is continuously updated using observed telemetry and residual consistency, allowing the graph to adapt to changing operating conditions. Consequently, moderate errors in the initial graph mainly affect convergence speed rather than the final detection capability. Nevertheless, substantial structural misspecification may reduce explanation fidelity, suggesting that future work should investigate automatic graph initialization and online causal structure discovery.

Although the proposed framework employs causal terminology, the learned graph should be interpreted as an operational causal model supported by engineering knowledge and repeated observational consistency rather than definitive proof of physical causation. Establishing true causality would require controlled interventions or randomized experiments, which are generally infeasible during UAV operation. Therefore, the causal relationships learned by Fed-CGIDS-UAV should be regarded as practically useful approximations for cyber-physical intrusion detection and root-cause analysis.

9. Conclusions

This paper proposed Fed-CGIDS-UAV, a federated causal graph learning framework for intrusion detection in cyber-physical drone networks. The framework responds to three recurring weaknesses in UAV IDS research: feature-centric modeling, weak cross-domain validation, and limited explanation of cyber-physical attack propagation. By representing UAV telemetry as typed graphs, learning normal causal dependencies, detecting residual violations, and federating model updates across clients, the approach provides both detection and explanation.

The cleaned evaluation results show that Fed-CGIDS-UAV improves internal performance while reducing the internal-external stability gap to 0.028. Attack-specific analysis indicates strong detection of GPS spoofing and command injection, while Sybil attacks remain more difficult because their evidence is distributed across swarm-level relationships. Ablation results confirm that causal residuals and graph structure are the main contributors to cross-domain robustness. The next step is field-oriented validation using real or semi-real UAV logs, together with more detailed reporting of communication overhead, convergence behavior, and edge-device resource use.

The reported results are derived from repeated experimental executions of the implemented framework within the adopted simulation environment, providing reproducible evidence of the effectiveness of the proposed approach.

The reported performance improvements were obtained using a controlled AirSim-PX4 software-in-the-loop simulation environment designed to emulate heterogeneous UAV operating conditions and representative cyber-physical attacks. Consequently, the reported quantitative results should be interpreted within this evaluation setting rather than as evidence of operational deployment. Future work will focus on validating the proposed framework using real and semi-real UAV telemetry collected from physical flight platforms and public UAV datasets to further assess robustness, scalability, and generalization under practical deployment conditions.

Author Contributions: Methodology, S.A.A.; validation, A.A.; formal analysis, A.A.; investigation, E.A.; resources, G.E.A.E.; data curation, G.E.A.E.; writing—original draft, E.A.; writing—review and editing, E.A.; visualization, E.A. All authors have read and agreed to the published version of the manuscript.

Funding: The researchers would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University (<https://www.qu.edu.sa>, accessed on 1 July 2026) for financial support (QU-APC-2026).

Data Availability Statement: The experimental study is based on a controlled simulation environment and generated cyber-physical UAV telemetry. The reproducibility package associated with this work includes the implementation scripts, simulation configuration files, attack-generation parameters, telemetry preprocessing procedures, dataset partition definitions, random seeds, and evaluation scripts used to generate the reported results.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Islam, M.S.; Mahmoud, A.S.; Sheltami, T.R. AI-Enhanced Intrusion Detection for UAV Systems: A Taxonomy and Comparative Review. *Drones* **2025**, *9*, 682. [CrossRef]
- Alzubaidi, A.A. Systematic Literature Review for Detecting Intrusions in Unmanned Aerial Vehicles Using Machine and Deep Learning. *IEEE Access* **2025**, *13*, 58576–58599. [CrossRef]
- Ihekoronye, V.U.; Ajakwe, S.O.; Lee, J.M.; Kim, D.-S. DroneGuard: An Explainable and Efficient Machine Learning Framework for Intrusion Detection in Drone Networks. *IEEE Internet Things J.* **2025**, *12*, 7708–7722. [CrossRef]
- Shrestha, R.; Omidkar, A.; Roudi, S.A.; Abbas, R.; Kim, S. Machine-Learning-Enabled Intrusion Detection System for Cellular Connected UAV Networks. *Electronics* **2021**, *10*, 1549. [CrossRef]
- Ramadan, R.A.; Emara, A.-H.M.; Al-Sarem, M.; Elhamahmy, M. Internet of Drones Intrusion Detection Using Deep Learning. *Electronics* **2021**, *10*, 2633. [CrossRef]
- Cengiz, K.; Lipsa, S.; Dash, R.K.; Ivkovic, N.; Konecki, M. A Novel Intrusion Detection System Based on Artificial Neural Network and Genetic Algorithm with a New Dimensionality Reduction Technique for UAV Communication. *IEEE Access* **2024**, *12*, 4925–4937. [CrossRef]
- Ntizikira, E.; Wang, L.; Alblehai, F.; Saleem, K.; Lodhi, M.A. Secure and Privacy-Preserving Intrusion Detection and Prevention in the Internet of Unmanned Aerial Vehicles. *Sensors* **2023**, *23*, 8077. [CrossRef] [PubMed]
- Hong, Y.-W.; Yoo, D.-Y. Multiple Intrusion Detection Using Shapley Additive Explanations and a Heterogeneous Ensemble Model in an Unmanned Aerial Vehicle's Controller Area Network. *Appl. Sci.* **2024**, *14*, 5487. [CrossRef]

9. Alzahrani, A. Novel Approach for Intrusion Detection Attacks on Small Drones Using ConvLSTM Model. *IEEE Access* **2024**, *12*, 149238–149253. [\[CrossRef\]](#)
10. Ceviz, O.; Sadioglu, P.; Sen, S.; Vassilakis, V. A Novel Federated Learning-Based IDS for Enhancing UAVs Privacy and Security. *Internet Things* **2025**, *31*, 101592. [\[CrossRef\]](#)
11. He, X.; Chen, Q.; Tang, L.; Wang, W.; Liu, T. CGAN-Based Collaborative Intrusion Detection for UAV Networks: A Blockchain-Empowered Distributed Federated Learning Approach. *IEEE Internet Things J.* **2023**, *10*, 120–132. [\[CrossRef\]](#)
12. Alazab, A.; Khraisat, A.; Singh, S.; Jan, T.; Alazab, M. Enhancing Privacy-Preserving Intrusion Detection Through Federated Learning. *Electronics* **2023**, *12*, 3382. [\[CrossRef\]](#)
13. Agrawal, S.; Sarkar, S.; Aouedi, O.; Yenduri, G.; Piamrat, K.; Alazab, M.; Bhattacharya, S.; Maddikunta, P.K.R.; Gadekallu, T.R. Federated Learning for Intrusion Detection System: Concepts, Challenges and Future Directions. *Comput. Commun.* **2021**, *195*, 346–361.
14. Bilot, T.; El Madhoun, N.; Agha, K.A.; Zouaoui, A. Graph Neural Networks for Intrusion Detection: A Survey. *IEEE Access* **2023**, *11*, 49114–49139. [\[CrossRef\]](#)
15. Villegas-Ch, W.; Govea, J.; Maldonado Navarro, A.; Jativa, P. Intrusion Detection in IoT Networks Using Dynamic Graph Modeling and Graph-Based Neural Networks. *IEEE Access* **2025**, *13*, 65356–65375. [\[CrossRef\]](#)
16. Tang, W.; Liu, J.; Zhou, Y.; Ding, Z. Causality-Guided Counterfactual Debiasing for Anomaly Detection of Cyber-Physical Systems. *IEEE Trans. Ind. Inform.* **2024**, *20*, 4582–4593. [\[CrossRef\]](#)
17. Dai, M.; Kang, A.; Liu, X.; Liu, Y. Causal Deep Learning for Enhancing Explainability in 6G Network Edge Intelligence Anomaly Detection. *Sci. Rep.* **2025**, *15*, 40678. [\[CrossRef\]](#)
18. Xue, C.; Lin, L.; Huang, Y.; Wang, X. Graph Neural Network-Based Anomaly Detection for Human Cyber Physical Systems. *J. Chin. Inst. Eng.* **2024**, *47*, 977–984. [\[CrossRef\]](#)
19. Wang, M.; Zheng, K.; Yang, Y.; Wang, X. An Explainable Machine Learning Framework for Intrusion Detection Systems. *IEEE Access* **2020**, *8*, 73127–73141. [\[CrossRef\]](#)
20. Zhu, C.; Zhu, X.; Qin, T. An Efficient Privacy Protection Mechanism for Blockchain-Based Federated Learning System in UAV-MEC Networks. *Sensors* **2024**, *24*, 1364. [\[CrossRef\]](#) [\[PubMed\]](#)
21. Popli, M.S.; Singh, R.P.; Popli, N.K.; Mamun, M. A Federated Learning Framework for Enhanced Data Security and Cyber Intrusion Detection in Distributed Network of Underwater Drones. *IEEE Access* **2025**, *13*, 12634–12646. [\[CrossRef\]](#)
22. Zhang, J.; Luo, C.; Jiang, Y.; Min, G. Security in 6G-Based Autonomous Vehicular Networks: Detecting Network Anomalies with Decentralized Federated Learning. *IEEE Veh. Technol. Mag.* **2025**, *20*, 83–93. [\[CrossRef\]](#)
23. Bithi, M.; Masud, M.E.; Hossain, M.A. A New Adaptive Federated Learning Approach for Privacy Preserving UAV Anomaly Detection Under Non-IID Distributions. *Sci. Rep.* **2026**, *16*, 8451. [\[CrossRef\]](#) [\[PubMed\]](#)
24. Lundberg, S.M.; Lee, S.-I. A Unified Approach to Interpreting Model Predictions. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
25. AL-Syouf, R.A.; Bani-Hani, R.M.; AL-Jarrah, O.Y. Machine Learning Approaches to Intrusion Detection in Unmanned Aerial Vehicles (UAVs). *Neural Comput. Appl.* **2024**, *36*, 18009–18041. [\[CrossRef\]](#)
26. Alsoliman, A.; Rigoni, G.; Callegaro, D.; Levorato, M.; Pinotti, C.M.; Conti, M. Intrusion Detection Framework for Invasive FPV Drones Using Video Streaming Characteristics. *ACM Trans. Cyber-Phys. Syst.* **2023**, *7*, 1–29. [\[CrossRef\]](#)
27. Xie, Y.; He, W.; Zhu, H.; Yang, R.; Mu, Q. A New Unmanned Aerial Vehicle Intrusion Detection Method Based on Belief Rule Base with Evidential Reasoning. *Heliyon* **2022**, *8*, e10481. [\[CrossRef\]](#) [\[PubMed\]](#)
28. Alheeti, K.M.A.; Khaled Alarfaj, F.; Alreshoodi, M.; Almusallam, N.; Al Dosary, D. A Hybrid Security System for Drones Based on ICMetric Technology. *PLoS ONE* **2023**, *18*, e0282567. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Niu, W.; Zhang, Y.; Shi, W.; Li, Y.; Karuppiiah, M.; Kumari, S. Malware on Internet of UAVs Detection Combining String Matching and Fourier Transformation. *IEEE Internet Things J.* **2021**, *8*, 9905–9919. [\[CrossRef\]](#)
30. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Y Arcas, B.A. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS 2017), Fort Lauderdale, FL, USA, 20–22 April 2017.
31. Kairouz, P.; McMahan, H.B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A.N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. Advances and Open Problems in Federated Learning. *Found. Trends Mach. Learn.* **2021**, *14*, 1–210. [\[CrossRef\]](#)
32. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated Learning: Challenges, Methods, and Future Directions. *IEEE Signal Process. Mag.* **2020**, *37*, 50–60. [\[CrossRef\]](#)
33. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
34. Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.

35. Hamilton, W.L.; Ying, R.; Leskovec, J. Inductive Representation Learning on Large Graphs. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
36. Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Philip, S.Y. A Comprehensive Survey on Graph Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 4–24. [[CrossRef](#)] [[PubMed](#)]
37. Pearl, J. *Causality: Models, Reasoning, and Inference*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2009.
38. Peters, J.; Janzing, D.; Scholkopf, B. *Elements of Causal Inference: Foundations and Learning Algorithms*; MIT Press: Cambridge, MA, USA, 2017.
39. Wu, J.; Qiu, G.; Wu, C.; Jiang, W.; Jin, J.; Wang, X.; Li, Y.; Liu, Y. Federated learning for network attack detection using attention-based graph neural networks. *Sci. Rep.* **2024**, *14*, 19088. [[CrossRef](#)] [[PubMed](#)]
40. Wang, Y.; Li, J.; Han, Z.; Cheng, P.; Zhang, Y.; Liu, X.; Wang, H. FedSTGCN: A Novel Federated Spatiotemporal Graph Learning-Based Network Intrusion Detection Method for the Internet of Things. *Front. Inf. Technol. Electron. Eng.* **2025**, *26*, 1164–1179. [[CrossRef](#)]
41. Gao, R.; Liu, Z.; Jiang, C.; Wang, Y.; Wang, S.; Wang, P. BI-FedGNN: Federated Graph Neural Networks Framework Based on Bayesian Inference. *Neural Netw.* **2024**, *169*, 143–153. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.