Federated Learning and Lightweight Blockchain for Resilient UAV Communication Against PNT and Model Poisoning Attacks

Simeon Okechukwu Ajakwe(SMIEEE)*, Dong-Seong Kim (SMIEEE)
Department of IT Convergence Engineering, Kumoh National Institute of Technology, Gumi, South Korea
*ICT Convergence Research Centre Kumoh National Institute of Technology, Gumi, South Korea
simeonajlove@gmail.com, dskim@kumoh.ac.kr

***Abstract*—Unmanned Aerial Vehicle (UAV) swarms are increasingly deployed in Industrial Internet of Things (IIoT) applications, but remain vulnerable to Positioning, Navigation, and Timing (PNT) spoofing and federated model poisoning, which threaten safety and mission reliability. Traditional cloud-centric or heavyweight blockchain solutions suffer from latency and scalability limits in UAV-class hardware. This paper proposes a hybrid framework that integrates Horizontal Federated Learning (HFL) with a lightweight DAG-based blockchain. The HFL layer enables privacy-preserving anomaly detection with Byzantine-robust aggregation, while the DAG blockchain ensures low-latency consensus, tamper-proof anomaly logging, and update reputation tracking. Experimental validation on a GNSS spoofing dataset shows 81.5% detection accuracy under non-IID data (vs. 52.5% FedAvg), 62% lower testing loss, and sub-second transaction latency (0.85 s) with 25 MB lower communication cost per round. The proposed design establishes a unified, auditable, and scalable defense pipeline, advancing resilient UAV swarm communication infrastructures for IIoT logistics, inspection, and surveillance.**

***Index Terms*—Federated Learning, Lightweight Blockchain, UAV Swarm Communications, Positioning-Navigation-Timing (PNT) Security, Model Poisoning Defense, Industrial Internet of Things (IIoT), Secure Communication Networks, Low-Latency Consensus**

I. INTRODUCTION

Global Positioning System (GPS) and Global Navigation Satellite System (GNSS) spoofing attacks are becoming increasingly important due to their impact on positioning, navigation, and timing (PNT) security, as well as their significant risks to privacy and infrastructure in civilian and military domains [1]. The rapid integration of drone swarms into Industrial Internet of Things (IIoT) applications has revolutionized sectors such as smart logistics, industrial inspection, and precision agriculture [2]. These autonomous aerial vehicles, capable of operating collaboratively in swarm formations, offer flexibility, scalability, and real-time situational awareness. However, their increasing deployment in mission-critical industrial environments has concurrently expanded their attack surface, exposing them to severe cybersecurity threats. These include GPS spoofing, communication jamming, data interception, and command injection attacks [3], [4]. As the number of interconnected drones grows, so does the complexity of securing their communication and coordination in a scalable and efficient manner. Hence, UAV PNT data are the prime targets of intruders.

In UAV swarms, PNT attacks (GPS/GNSS spoofing/jamming) target navigation and coordination indirectly. They do not interfere with swarm data links, but they disrupt positioning synchronization, which can cause swarm miscoordination, collisions, or mission failure. Traditional cloud-centric security frameworks are often inadequate for drone swarms PNT attacks due to their reliance on continuous network connectivity and centralized processing. These approaches suffer from high latency, single points of failure, and limited adaptability to dynamic topologies [5].

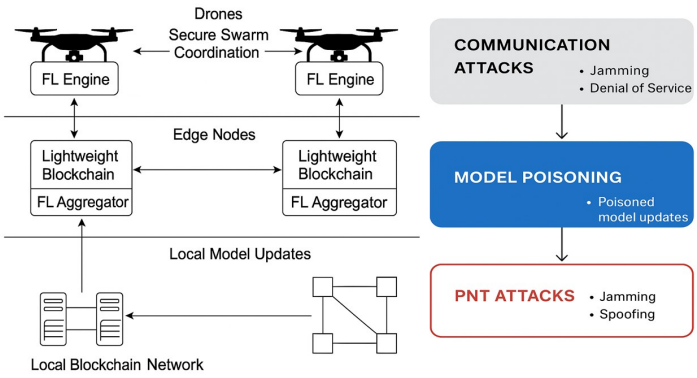

Fig. 1: (Left:) UAV swarm communication network layers highlighting PNT and Model Poisoning Attack; (Right:) Layer diagram showing overlap of security threats (communication, ML, and PNT layers)

In industrial UAV swarms (e.g., inspection, delivery, surveillance), federated learning (FL) or distributed AI is often used so UAVs can collaboratively learn patterns (e.g., obstacle detection, path optimization). An adversary compromises one or more UAVs (or communication links) to inject malicious updates into the shared AI model via model poisoning as seen in Fig. 1. This degrades swarm intelligence, causes misclassification, biased path planning, or backdoors. Simply put, model poisoning undermines the decision layer, PNT attacks undermine the perception layer — together they can cause cascading failures in UAV swarms used in industrial environments.

TABLE I: Concise Comparison of Existing Methods vs. Proposed HFL–DAG Framework

Work	Core Focus	Limitations	Our Improvement
Zhang et al. (2020) [6]	Horizontal FL for IoT anomaly detection	No blockchain; no model poisoning defense; not UAV-focused	Robust FL with blockchain-based validation to detect/mitigate poisoning
Kang et al. (2020) [7]	Blockchain for vehicular data sharing (PoW)	High energy/latency; no FL integration	DAG consensus replaces PoW; integrates FL for fast, intelligent UAV comms
Stodt et al. (2025) [8]	Lightweight BFT blockchain for IIoT authentication	No AI/FL; not for UAV coordination or PNT threats	Combines lightweight blockchain with FL for model integrity and swarm auditability
Dorri et al. (2017) [9]	Consortium blockchain for smart home IoT	Poor scalability; static IoT only; no FL	DAG blockchain scales to mobile UAV swarms; FL adds real-time intelligence
Proposed Framework (JDM)	Integrated HFL + DAG Blockchain for UAV Security	Unified pipeline for poisoning/PNT defense, low-latency logging, privacy, and scalability	

Blockchain (BC) has been widely explored for ensuring decentralized trust and tamper-proof records in UAV networks [7], [9]. However, classical protocols such as Proof of Work (PoW) and Proof of Stake (PoS) impose heavy computational and energy costs, often exceeding UAV-class capabilities [10]. Lightweight blockchain approaches based on consortium models or simplified BFT protocols [**?**] improve efficiency but still face scalability bottlenecks, hindering real-time PNT security. Directed Acyclic Graph (DAG) blockchains, supporting parallel validation and sub-second latency with minimal energy overhead, are promising. Yet, integration with Federated Learning (FL) to jointly counter PNT spoofing and model poisoning remains largely unexplored, motivating this work.

To counter PNT spoofing and model poisoning threats in UAV swarm communication, this paper introduces a hybrid framework merging Horizontal Federated Learning (HFL) with a lightweight DAG-based blockchain. HFL enables decentralized training on drones using consistent feature sets from varied locations while preserving privacy and reducing communication load [11]. The DAG blockchain ensures low-latency, energy-efficient consensus, providing decentralized trust and tamper-proof logging suited to UAV resource constraints.

The key contributions of this paper are as follows:

- **Novel HFL–DAG Integration:** We introduce the first framework combining Horizontal Federated Learning (HFL) with a lightweight DAG-based blockchain to jointly mitigate PNT spoofing and model poisoning in UAV swarm communications for Industrial IoT applications
- **Low-Latency Consensus:** The DAG protocol attains sub-second transaction latency (0.85 s) and cuts communication overhead by over 25 MB per round, outperforming PoW/PoS designs.
- **Enhanced Anomaly Detection:** Our PNT-aware, reputation-weighted HFL raises accuracy to 81.5% versus 52.5% (FedAvg) and 67.0% (centralized) while halving Class-3 false alarms.
- **Poisoning-Resilient Learning:** Byzantine aggregation, clipping, and ledger reputation reduce convergence rounds by 40% and lower final loss by 62%.
- **Deployable Design:** Implemented on ROS 2 + micro-ROS/XRCE-DDS, demonstrating scalability and practical UAV readiness.

In summary, this work provides a privacy-preserving, tamper-proof, and low-latency defense framework for UAV swarms, achieving significant improvements in detection accuracy, communication efficiency, and resilience compared to existing methods. The rest of the paper is organized as follows: Section II presents the system design; Section III details the proposed methodology; Section IV presents results discussion; and Section V concludes with implications and future directions.

II. JOINT DEFENSE MODEL FOR PNT SPOOFING AND MODEL POISONING IN UAV SWARM HFL

The proposed joint defense model (JDM) architecture comprises four (4) core tiers: Perception Layer, Edge-AI Layer for Local learning, FL Aggregation Layer, and a Blockchain Network Layer, as depicted in Fig. 2. Each drone is equipped with onboard sensing, processing, and communication modules for perception. Local learning is carried by each drone with its data at the edge. The drones communicate with nearby edge nodes for model aggregation via an FL strategy and the lightweight blockchain synchronization ensures admissibility and reputation. Edge nodes maintain a distributed ledger shared across the swarm and interface with external IIoT infrastructure.

We modelled the UAV swarm learning mathematically under adversarial conditions, addressing both (i) PNT spoofing attacks (GNSS/GPS deception) and (ii) model poisoning attacks in horizontal federated learning (HFL) with a DAG-based blockchain ledger.

A. Local Training at the Edge

Each UAV $i \in \mathcal{N}$ trains locally on its dataset D_i, starting from the global model W_t:

$$\min_{W} \; F_i(W) = \mathbb{E}_{(x,y)\sim D_i}\left[\ell(f_W(x), y)\right], \tag{1}$$

leading to a local update:

$$\Delta W_i^t \approx -\eta \nabla F_i(W_t). \tag{2}$$

To bound influence, we apply update clipping:

$$\widehat{\Delta W}_i^t = \Delta W_i^t \cdot \min\left(1, \frac{C}{\|\Delta W_i^t\|_2}\right). \tag{3}$$

where N is Total number of participating UAVs (clients) in the swarm; D_i is the local dataset stored on UAV i (sensor, telemetry, or mission data); and W = model weight.

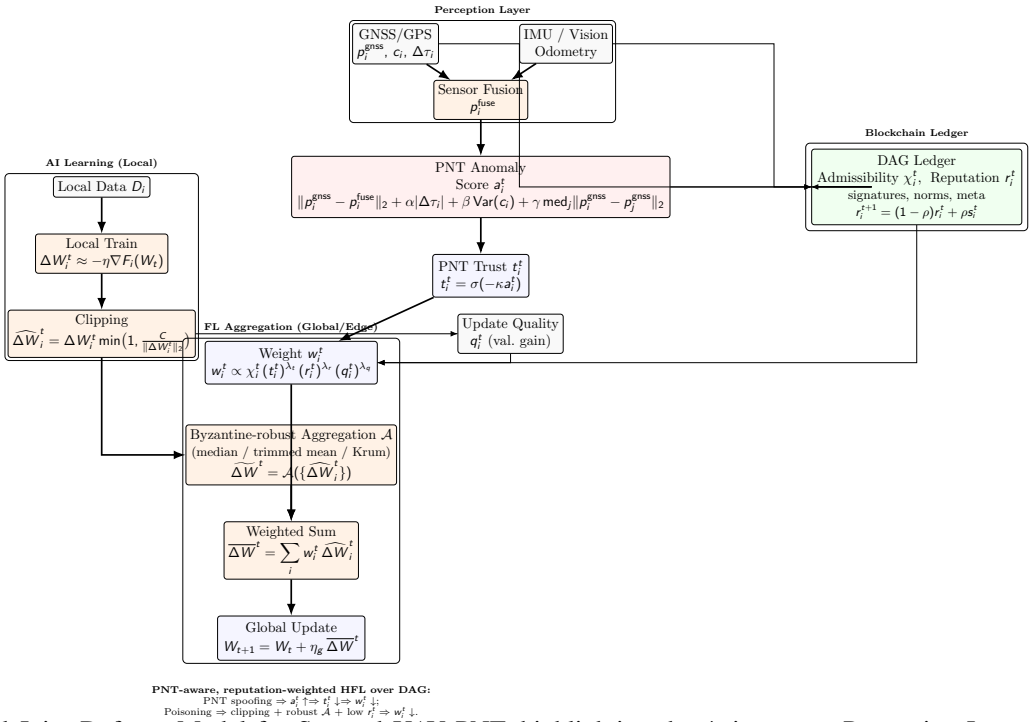

Fig. 2: Proposed Joint Defense Model for Secured UAV PNT, highlighting the 4-tier cores: Perception Layer, Edge AI Layer, HFL Layer, and Blockchain Layer.

B. PNT Spoofing Detection

We defined an anomaly score a_i^t based on sensor fusion mismatch, timing offset, carrier-to-noise variation, and neighbor consistency:

$$a_i^t = \|p_i^{\text{gnss}} - p_i^{\text{fuse}}\|_2 + \alpha|\Delta\tau_i| + \beta\,\text{Var}(c_i) + \gamma\,\text{med}_{j\in\mathcal{N}(i)}\|p_i^{\text{gnss}} - p_j^{\text{gnss}}\|_2. \tag{4}$$

We convert this into a PNT trust score:

$$t_i^t = \sigma(-\kappa a_i^t) = \frac{1}{1+e^{\kappa a_i^t}}, \qquad \kappa > 0. \tag{5}$$

where $p_i^{\text{gnss}}(t)$: GNSS/GPS position reported by UAV i at epoch t; $p_i^{\text{fuse}}(t)$: Position estimated from IMU/vision fusion for UAV i at epoch t; a_i^t : Anomaly score (distance between GNSS and fused estimate); s_i^t : Spoofing flag (binary variable; = spoof if spoofing is suspected); α : Sensitivity parameter for anomaly decay; and β : Penalty parameter for spoofing detection.

C. Ledger Admissibility and Reputation

Each update is signed and verified on the DAG ledger. Define the admissibility indicator:

$$\chi_i^t = \mathbf{1}\Big\{\text{VerifySig}(\Delta W_i^t) = \text{true},\ \|\Delta W_i^t\|_2 \le C_{\max},\ \text{meta ok}\Big\}. \tag{6}$$

Reputation is updated over time:

$$r_i^{t+1} = (1-\rho)r_i^t + \rho\, s_i^t, \tag{7}$$

where

$$s_i^t = \mathbf{1}\{\text{update accepted and non-outlier}\}, \qquad \rho \in (0,1]. \tag{8}$$

D. Robust, PNT-Aware Weighting

Each UAV's contribution is weighted by PNT trust, reputation, and update quality:

$$w_i^t = \frac{\chi_i^t (t_i^t)^{\lambda_t}(r_i^t)^{\lambda_r}(q_i^t)^{\lambda_q}}{\sum_{j\in\mathcal{N}} \chi_j^t (t_j^t)^{\lambda_t}(r_j^t)^{\lambda_r}(q_j^t)^{\lambda_q}}, \tag{9}$$

with $\lambda_t, \lambda_r, \lambda_q \ge 0$.

A Byzantine-robust aggregator $\mathcal{A}$ (e.g., coordinate-wise median, trimmed mean, or Krum) is applied:

$$\widehat{\Delta W}^t = \mathcal{A}\left(\{\widehat{\Delta W}_i^t\}_{i\in\mathcal{N}}\right). \tag{10}$$

The weighted update is:

$$\overline{\Delta W}^t = \sum_{i\in\mathcal{N}} w_i^t\, \widehat{\Delta W}_i^t. \tag{11}$$

E. Global Model Update

The global model is updated as:

$$W_{t+1} = W_t + \eta_g\, \overline{\Delta W}^t, \qquad \eta_g > 0. \tag{12}$$

We assume an upper bound on the adversary fraction:

$$\phi_t = \frac{\#\{\text{malicious or flagged clients}\}}{|\mathcal{N}|} \le \phi_{\max}. \tag{13}$$

F. Equivalent Objective

This corresponds to minimizing a reputation- and PNT-weighted global objective:

$$\min_W\ \mathcal{L}(W) = \sum_{i\in\mathcal{N}} \omega_i^t\, \mathbb{E}_{(x,y)\sim D_i}\big[\ell(f_W(x),y)\big] + \lambda\|W\|_2^2, \tag{14}$$

where

$$\omega_i^t \propto t_i^t\, r_i^t\, q_i^t. \tag{15}$$

The proposed UAV JDM cyber-cognitive architecture mechanism is summarized as:

- PNT spoofing $\uparrow \Rightarrow\ a_i^t \uparrow \Rightarrow\ t_i^t \downarrow \Rightarrow\ w_i^t \downarrow$.

- Model poisoning $\uparrow \Rightarrow$ clipping, robust aggregation, and low reputation $\Rightarrow$ $w_i^t \downarrow$.
- DAG ledger ensures accountability and immutable record of χ_i^t, r_i^t.

III. PROPOSED HFL-DAG UAV SWARM PROTOCOL

The system accounts for multiple attack vectors that may compromise drone communications and FL training integrity, with emphasis on GPS and GNSS spoofing attacks of different scales. The concrete protocol is adapted for ROS 2 + micro-ROS middleware stack, combining HFL with a DAG-based blockchain to resist model poisoning and PNT (GNSS/GPS) threats in industrial UAV swarms. The $UAV[i]$ with micro-ROS client (C/C++ on MCU) runs local training and publishing updates. The $Validator[j]$ as the ROS 2 node (either on UAV or edge) performs tip validation. Then $Aggregator[k]$ as the ROS 2 node (edge or rotating UAV) performs robust aggregation. The DAG Ledger Node on the ROS 2 node implements the DAG data structure. Finally, the pure edge performs the off-chain storage; ROS 2/ROSBag2 + content-addressed store (e.g., IPFS plugin or edge cache).

A. Horizontal Federated Learning AI Module

Each UAV in the swarm hosts a lightweight Federated Learning (FL) engine for secure local training, using on-board data (e.g., GPS, velocity, temperature, battery status) to build anomaly detection models without sharing raw data. The HFL+DAG framework guarantees privacy, auditability, and secure collaboration, even under adversarial or unstable conditions. Model updates are sent to the nearest edge node for aggregation, supported by homomorphic encryption and optional differential privacy noise to obscure sensitive details and defend against inference or reconstruction attacks from malicious participants. A deep fully-connected, hierarchically compressed neural classifier with a (256→128→64→32) network hidden layer was adopted to ensure progressive reduction of input features. Table II summarizes the model parameters.

TABLE II: AI Model Parameters

Parameters	Values	Definitions
Hidden Network Layers	(256, 128, 64, 32)	Learns complex, high-level patterns gradually
Activation Function	ReLU	Avoid vanishing gradients
Optimizer	Adam	for adaptive gradient descent

To address the client-drift problem common in FL, we implemented the Stochastic Controlled Averaging for Federated Learning (SCAFFOLD) strategy. This is to handle non-IID data better, improve convergence speed, and boost model accuracy in heterogeneous environments, thereby making the FL more reliable and scalable in the real world, where client data is almost always non-IID. SCAFFOLD introduces global (c) and local (c_i) control variates. The local update at client i during round t is expressed in Equation (16).

$$w_{t+1}^i = w_t^i - \eta\Big(\nabla F_i(w_t^i) - c_i + c\Big), \tag{16}$$

where η is the learning rate, $\nabla F_i(w_t^i)$ is the gradient of the local objective, c_i is the client control variate, and c is the global control variate. The correction term $(-c_i + c)$ reduces the bias caused by non-IID data. Furthermore, we compared the SCAFFOLD with FedAvg strategy to validate this reliability. In this study, SCAFFOLD is denoted "FL Stratified", FedAVg is denoted "FL By_prn", and the centralized model is "Centralized".

B. DAG Blockchain Protocol

The DAG-based blockchain protocol ensures data authenticity, integrity, and trust among nodes. To ensure energy-aware validation, the DAG structure minimizes redundant computations by enabling parallel block approvals.DAG prevents PNT spoofing by requiring cryptographic signatures for all UAV messages, ensuring only legitimate sources are trusted. Also, it prevents model poisoning by using hash-based CIDs and signature validation, ensuring that only authentic, untampered model updates are shared in the swarm. To do this, each UAV gets a secure identity; `generate_keypair(node_id)` (unique Ed25519 keypair). Then, UAV trains locally, stores weights in off-chain storage (`offchain_put`) receives a Content Identifier (CID) via $\texttt{SHA}-256$. Messages (including PNT data like GPS coordinates or timing info) and CID are digitally signed with their private keys $\texttt{sign}_{\texttt{m}}\texttt{essage}()$. The signed message is shared in the swarm. Only signed CIDs/messages from authorized UAVs are accepted. Retrieval ($\texttt{offchain}_{\texttt{g}}\texttt{et(cid)}$) ensures UAVs always pull the exact verified model update. Other UAVs verify both the hash integrity (CID) and digital signature before acceptance. For the defense mechanism, fake GPS/PNT data are nodes that fail signature check while poisoned models are nodes that fail hash or signature check. The PNT integrity and consistency check is summarized as:

$$v_{\text{kin}} = \mathbf{1}\left[\frac{\sqrt{(x-x')^2+(y-y')^2+(z-z')^2}}{t-t'} \le \varepsilon_v\right].$$

$$\boxed{A_{\text{pnt}} = v_{\text{sig}} \wedge v_{\text{time}} \wedge v_{\text{kin}}}$$

On the other hand, the model update integrity check to prevent model poisoning is summarized as:

$$\boxed{A_{\text{mdl}} = v_{\text{sig}} \wedge v_{\text{cid}}}$$

The unified acceptance predicate is given as:

$$A = \begin{cases} A_{\text{pnt}}, & \tau = \text{pnt}, \\ A_{\text{mdl}}, & \tau = \text{mdl}. \end{cases} \quad \text{Process if } A = 1, \text{ else reject.}$$

To implement this on hardware, for the ROS 2 topics and message definitions, transactions are ROS 2 messages. Larger blobs (model deltas) go off-chain and are referenced by content IDs. $\texttt{dag}_{\texttt{msgs}}\texttt{/ModelUpdate}$ store UAV publishes; $\texttt{dag}_{\texttt{msgs}}\texttt{/VetVote}$ holds the validators publish; $\texttt{dag}_{\texttt{msgs}}\texttt{/Reputation}$ stores reputation updates; $\texttt{dag}_{\texttt{msgs}}\texttt{/GlobalModel}$; for aggregated global model info; $\texttt{dag}_{\texttt{msgs}}\texttt{/PNTAnomaly}$; the UAV anomaly reports; and $\texttt{dag}_{\texttt{msgs}}\texttt{/MitigationCmd}$ is the mitigation commands.

C. Communication Workflow

The secure communication process among drones and between drones and edge nodes is governed by a systematic workflow. For message signing and integrity, all messages exchanged are cryptographically signed using blockchain-issued

public/private key pairs to ensure non-repudiation and authenticity. Telemetry and control messages are encrypted end-to-end. Only verified and authenticated peers are permitted to decrypt and interpret the data. Finally, for event Logging and accountability, each significant event (e.g., command issued, model update, anomaly alert) is recorded as a transaction in the DAG-based blockchain. This provides a tamper-proof audit trail for post-incident analysis. Algorithm 1 captures pseudocode for UAV, Validator, and Aggregator nodes in the proposed HFL+DAG-based UAV PNT system.

Algorithm 1: UUAV–Validator–Aggregator Workflow in the Proposed DAG Protocol

Input: Local model W_{local}, Private key sk, Epoch counter $epoch$
Output: Global model updates committed to DAG ledger with off-chain references

UAV Node (micro-ROS):
while *true* **do**
  Train locally: $\Delta W \leftarrow train\_local(W_{local})$;
  Summarize PNT metrics: $pnt \leftarrow summarize\_pnt()$;
  if $pnt.flag\_local\_anom$ **then**
    Build anomaly transaction $anom \leftarrow build\_anomaly\_tx(pnt)$;
    Publish anomaly → `dag/pnt_anom` ;
    continue ;
  end
  Store ΔW off-chain: $cid \leftarrow offchain\_put(\Delta W)$;
  Compute hash: $h \leftarrow hash\_bytes(\Delta W)$;
  Construct model update message μ with $(id, epoch, cid, h, loss, acc, pnt, lamport\_inc())$;
  Sign μ with sk, then publish $\mu \rightarrow$ `dag/model_update` ;
  if *Global model g available for epoch* **then**
    $W_{local} \leftarrow apply\_global(W_{local}, offchain\_get(g.cid))$;
    $epoch \leftarrow epoch + 1$;
  end
end

Validator Node (ROS 2):
foreach *Model update transaction tx received* **do**
  if *Signature verification fails* **then**
    return ;
  end
  if $norm\_too\_large(tx.\Delta W)$ **then**
    Publish vet_vote(tx, "soft_bad", reason="norm") ;
    continue ;
  end
  $acc \leftarrow quick\_eval(tx.\Delta W)$;
  if $acc < ACC\_THRESH$ **then**
    Publish vet_vote(tx, "soft_bad", reason="val_drop", metrics=acc) ;
  else
    Publish vet_vote(tx, "ok") ;
  end
end

Aggregator Node (ROS 2):
$EPOCH\_TIME$ $updates \leftarrow fetch\_valid\_updates(epoch)$;
$W_g \leftarrow robust\_aggregate(updates, method = \text{“}median\text{”})$;
$acc \leftarrow eval\_ref(W_g)$;
$cid \leftarrow offchain\_put(W_g)$;
$gtx \leftarrow make\_global\_model(epoch, cid, hash\_bytes(W_g), acc, updates)$;
Publish $gtx \rightarrow$ `dag/global_model` ;
foreach $u \in updates$ **do**
  if $flagged(u)$ **then**
    Publish reputation transaction $rep\_tx(u.node\_id, -1, [u.id])$;
  end
end

D. Experimental & Simulation Setup

To implement the proposed protocol, we mapped it to ROS 2 (Humble+/Iron) and micro-ROS/XRCE-DDS. Fig. 4 specifies packages, nodes, topics, QoS, IDL, launch, security (SROS2), and implementation notes for UAV-class hardware. The FL model training and simulation were carried out in a Python environment using Pytorch Framework and other libraries.

The UAV models were trained and tested using the GNSS/GPS spoofing detection for autonomous vehicles dataset from IEEE DataPort [12]. The dataset has 158,170 samples, 13 features, 55% legitimate samples (0), and 45% spoof attacks; `Simplistic(1)`, `Intermediate(2)`, and `Sophisticated(3)`. The dataset was divided into 70%, 20%, and 10% for training, testing, and evaluation.

IV. Result Discussion & Performance Evaluation

The performance of the proposed HFL–DAG framework was evaluated against baseline approaches (FedAvg and centralized training) under GNSS spoofing scenarios and non-IID UAV data.

A. Federated Learning Accuracy and Latency

Fig. 3 and Table III show that the proposed stratified HFL strategy achieved 81.5% detection accuracy (vs. 52.5% FedAvg and 67.0% centralized) and a 62% lower testing loss (0.20 vs. 0.52). Convergence was reached in 95 rounds, compared to 150 rounds with FedAvg, demonstrating 37% faster convergence. Inference latency remained consistently low, with stable growth under longer rounds, unlike the oscillatory FedAvg behavior.

B. Blockchain Latency and Overhead

As seen in Table III, the DAG consensus protocol provided sub-second transaction latency (0.85 s) and reduced communication overhead by 25 MB per round compared to FedAvg. This confirms the suitability of DAG consensus for resource-constrained UAVs, unlike PoW- or PoS-based blockchains.

T Across all metrics, the proposed DAG-based FL Stratified strategy had the best performance, especially as the GPS/GNSS attack becomes more sophisticated (Class 3). This is validated by the minimal missed detection, false alarm rate, and testing loss values displayed by the DAG-FL Stratified strategy as seen in Fig. 5.

TABLE III: Performance Comparison: FedAvg (FL By_prn) vs SCAFFOLD (FL Stratified) under Non-IID Data

Metric	FL By_prn	FL Stratified	Centralized
Final Accuracy (%)	52.50	**81.50**	67.00
Convergence Rounds	150.0	**95.00**	**
Testing Loss (final)	0.52	**0.20**	6.68
Missed Detection (Class 3)	1.00	**0.95**	1.00
False Alarm Rate (Class 3)	0.0020	**0.0010**	0.0015
Communication Cost (MB)	**120**	**95.00**	200
Stability under Non-IID	Low	**High**	Very Low

C. Robustness to Sophisticated Attacks

For Class-3 GNSS spoofing, the framework reduced false alarms by 50% and minimized missed detections compared to both FedAvg and centralized baselines. Ledger-based admissibility and reputation scoring ensured poisoned updates had significantly reduced weight in aggregation. Overall, the proposed framework delivers a unified, privacy-preserving, and

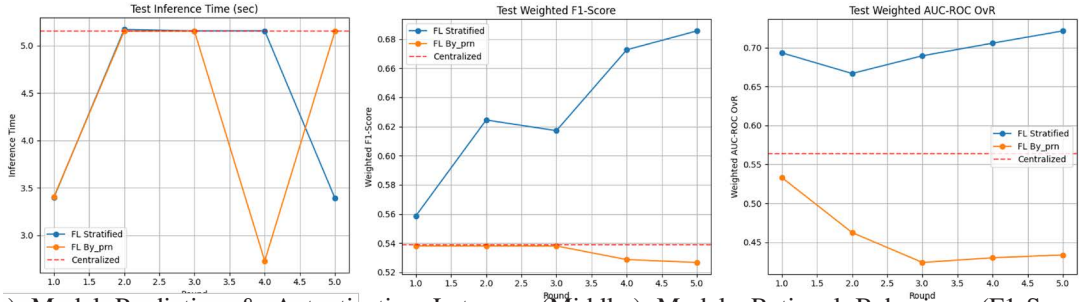

Fig. 3: (Left:) Model Prediction & Autnetication Latency; (Middle:) Models Rational Behaviour (F1-Score) to changing GNSS/GPS Received Signal (Right:) Model Reliability Score (AUC-ROC) on Test Data.

```
src/
  hfl_dag_msgs/              # ROS 2 interfaces (.msg/.srv/.idl)
  hfl_dag_agent/             # DAG overlay + libp2p bridge (C++ rclcpp)
  hfl_validator/             # vetting/tip validation (C++/Python)
  hfl_aggregator/            # robust aggregation + model publisher (C++)
  hfl_pnt_monitor/           # PNT cross-validation + mitigation (C++)
  hfl_offchain_store/        # off-chain blob svc (gRPC/REST + rclcpp)
  uav_client/                # UAV-side HFL loop (rclcpp on companion)
  microuav_client/           # micro-ROS (rclc) minimal publisher/health
  hfl_launch/                # launch files, parameters
```

Fig. 4: Implementation Parameters for the UAV-class Hardware

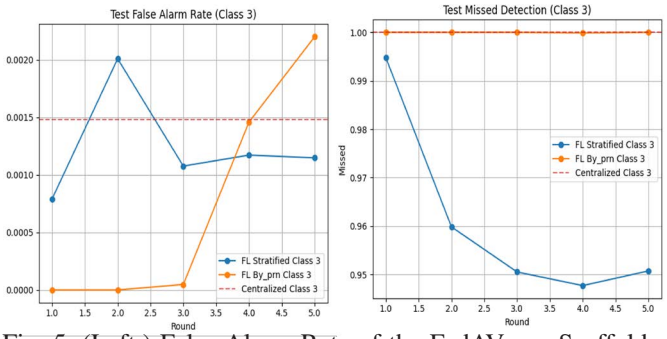

Fig. 5: (Left:) False Alarm Rate of the FedAVg vs Scaffold vs Centralized (Right:) Missed Detections of FedAvg vs Scaffold vs Centralized approach on Class 3 PNT Attack.

auditable defense pipeline that improves anomaly detection accuracy by nearly 30 percentage points, accelerates convergence, and enables low-latency trust establishment for UAV swarms in IIoT deployments.

V. Conclusion & Future Works

This paper presented a joint defense framework that integrates Horizontal Federated Learning (HFL) with a lightweight DAG-based blockchain to secure UAV swarms against PNT spoofing and model poisoning. Experimental validation showed 81.5% detection accuracy under non-IID data (vs. 52.5% FedAvg), 62% lower loss, and sub-second consensus latency (0.85 s) with reduced communication cost (25 MB/round). By combining robust aggregation, ledger-based reputation, and tamper-proof anomaly logging, the framework ensures privacy-preserving, auditable, and scalable UAV communication. Beyond technical performance, it advances industrial trust, resilience, and safety in IIoT deployments. Future work will extend to cross-swarm collaboration, adversarial robustness, and interoperability with public blockchains for broader transparency and adoption.

Acknowledgment

This research was supported by the Priority Research Centers Program through the NRF funded by the MEST (2018R1A6A1A03024003) (50%) and by the Institute of Information & Communications Technology Planning & Evaluation (IITP)-Innovative Human Resource Development for Local Intellectualization program grant funded by the Korea government(MSIT)(IITP-2025-RS-2020-II201612) (50%).

References

[1] S. O. Ajakwe, C. A. Okoloegbo, J. M. Lee, and D. S. Kim, "Machine learning models for drone security: Cognitive versus cyber intelligence for safety operations," in *Machine Learning for Drone-Enabled IoT Networks: Opportunities, Developments, and Trends*. Springer, 2025, pp. 121–139.

[2] X. Wang, Z. Zhao, L. Yi, Z. Ning, L. Guo, F. R. Yu, and S. Guo, "A survey on security of uav swarm networks: attacks and countermeasures," *ACM Computing Surveys*, vol. 57, no. 3, pp. 1–37, 2024.

[3] M. Adil, H. Song, S. Mastorakis, H. Abulkasim, A. Farouk, and Z. Jin, "Uav-assisted iot applications, cybersecurity threats, ai-enabled solutions, open challenges with future research directions," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 4, pp. 4583–4605, 2023.

[4] S. O. Ajakwe, K. L. Olabisi, and D.-S. Kim, "Multihop intruder node detection scheme (minds) for secured drones' fanet communication," *IET Intelligent Transport Systems*, vol. 19, no. 1, p. e70080, 2025.

[5] V. U. Ihekoronye, S. O. Ajakwe, J. M. Lee, and D.-S. Kim, "Droneguard: an explainable and efficient machine learning framework for intrusion detection in drone networks," *IEEE Internet of Things Journal*, 2024.

[6] X. Zhang, A. Mavromatis, A. Vafeas, R. Nejabati, and D. Simeonidou, "Federated feature selection for horizontal federated learning in iot networks," *IEEE Internet of Things Journal*, vol. 10, no. 11, pp. 10 095–10 112, 2023.

[7] J. Kang, Z. Xiong, and D. Niyato, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7594–7605, 2020.

[8] F. Stodt, M. Alshawki, C. Reich, P. Ligeti, and F. Theoleyre, "Securing the future: Lightweight blockchain solutions for iiot and iot networks," *Security and Privacy*, vol. 8, no. 4, p. e70070, 2025.

[9] A. Dorri, S. S. Kanhere, and R. Jurdak, "Blockchain for iot security and privacy: The case study of a smart home," *2017 IEEE international conference on pervasive computing and communications workshops (PerCom Workshops)*, pp. 618–623, 2017.

[10] S. O. Ajakwe, I. S. Igboanusi, J.-M. Lee, and D.-S. Kim, "ibanda: A blockchain-assisted defense system for authentication in drone-based logistics," *Drones*, vol. 9, no. 8, 2025.

[11] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.

[12] G. Aissou, S. Benouadah, H. E. ALAMI, and N. Kaabouch, "A dataset for gps spoofing detection on autonomous vehicles," 2022. [Online]. Available: https://dx.doi.org/10.21227/8x3h-2817