Research on Trusted Perception and Flexible Interaction Interface Design Optimization of Industrial Internet of Things Based on Federated Learning

Wan Chen

Nanhai Academy of Art, Haikou University of Economics, Haikou 570100, China
E-mail: 13322023067@phc.ac.cn

Received 06 August 2026; Accepted 23 August 2026

Abstract

This study aims to evaluate whether a risk-adaptive federated learning framework can improve the robustness and system-level efficiency of Industrial Internet of Things (IIoT) anomaly detection under simulated malicious-client behavior and feedback noise. The proposed framework coordinates risk-adaptive security policy generation, sketch-assisted robust aggregation under privacy-constrained uploading, reliability-weighted trustworthiness updating, and a flexible human–machine feedback prototype to support anomaly detection without transferring local data. Federated simulation experiments on six public IIoT/ICS datasets, including Edge-IIoTset, X-IIoTID, TON_IoT, WUSTL-IIoT, SWaT, and HAI, show relatively stable detection performance, with clearer advantages in Robust-F1, ASR control, Drop-F1, communication cost, convergence rounds, and privacy–latency–communication–security trade-offs than in universal superiority of conventional detection accuracy. These results indicate that the main value of the framework lies in the coordinated handling of privacy-aware uploading, aggregation robustness, communication efficiency, and feedback reliability, rather than revolutionary accuracy gains. Since client partitioning, resource perturbation, malicious

Journal of ICT Standardization, Vol. 14_4, 645–690.
doi: 10.13052/jicts2245-800X.1446
© 2026 River Publishers

behaviors, interface feedback, and feedback reliability are experimentally constructed, the results should be interpreted as controlled simulation evidence rather than real cross-factory deployment validation or a complete cryptographic security guarantee, and further validation with genuine industrial clients and operator feedback is required.

Keywords: Sketch-assisted robust aggregation, risk-adaptive privacy protection, human–machine feedback regularization, industrial anomaly detection, secure federated learning, flexible interaction optimization.

1 Introduction

With the development of the Industrial Internet of Things (IIoT) toward edge intelligence and collaborative operation and maintenance, industrial anomaly detection and intrusion identification models are required to support collaborative training under decentralized data, privacy constraints, and dynamically changing edge resources [1, 2]. In typical industrial scenarios, sensor states, control commands, network traffic, device logs, and review feedback are usually distributed across different devices, production lines, or edge nodes, making direct centralized collection difficult. Federated learning provides a privacy-preserving collaborative modeling paradigm for industrial detection tasks by enabling model updates without transferring local data [3].

However, existing studies on federated industrial detection still have several limitations. First, parameters related to privacy protection, secure uploading, and communication compression are often configured statically, with limited consideration of joint adaptation to client risk status, resource load, communication latency, and model uncertainty. This makes them less suitable for dynamic industrial edge environments [4, 5]. Second, many robust aggregation methods rely on plaintext coordinates, norms, distances, or directional similarities of client updates to identify anomalous updates. Under secure aggregation or privacy-constrained uploading, however, the server cannot directly access complete plaintext updates, which limits the direct applicability of such methods. Third, human confirmation, correction, and uncertain feedback in industrial alarm systems are usually used for alarm handling or result interpretation, rather than being modeled as weak supervisory signals that can participate in training. Moreover, feedback may suffer from delay, mislabeling, and subjective inconsistency; treating it directly as ground-truth labels may introduce additional noise [6, 7]. Fourth, existing industrial anomaly detection systems often lack flexible interaction interfaces

that jointly present alarm confidence, prediction uncertainty, feature attribution, risk level, and feedback entry. As a result, operator feedback is difficult to collect, evaluate, and incorporate into subsequent federated optimization in a reliable manner. Therefore, a practical gap remains between privacy-constrained secure federated training and trustworthy industrial review support when plaintext updates are restricted and feedback may be delayed, noisy, or inconsistent.

To address these issues, this paper proposes a risk-adaptive secure federated learning framework for trusted IIoT perception and flexible interaction optimization. The framework consists of four core components. First, it dynamically adjusts the differential privacy budget, secure uploading level, sparsity ratio, and quantization bit-width according to client risk status, resource load, communication latency, prediction uncertainty, and historical trustworthiness, thereby improving the adaptability of security and communication strategies to heterogeneous client states. Second, it employs noisy random projection sketches and bounded influence functions to approximately estimate the deviation of client updates without accessing complete plaintext updates, reducing the impact of abnormal uploads on the global model. Third, human review or simulated feedback is transformed into reliability-weighted weak supervision signals, which are used to update client trustworthiness and further affect subsequent risk scoring and aggregation weights. Fourth, a flexible human–machine interaction prototype is designed for industrial anomaly review. The prototype integrates alarm confidence, prediction uncertainty, feature attribution, risk level, and feedback entry, and its effectiveness is evaluated through feedback delay, feedback reliability, review workload, and before–after interface comparison. It should be noted that this study focuses on feedback-assisted trusted perception and flexible interaction modeling in federated simulation settings based on public datasets, and does not claim to provide a complete cryptographic security proof or large-scale real industrial deployment validation.

1.1 Novelty and Positioning

The contribution of this study is not the invention of each individual module, but the round-level coordination of risk-adaptive uploading, sketch-based deviation estimation, and reliability-weighted feedback under privacy-constrained federated IIoT anomaly detection. Noisy sketches support approximate deviation estimation when full plaintext updates are unavailable, and delayed or uncertain feedback affects subsequent trustworthiness and risk

scoring rather than serving as direct ground-truth supervision. The framework is therefore evaluated as a controlled simulation study of robustness, ASR control, communication efficiency, privacy-related cost, and feedback reliability.

The main contributions of this paper are as follows. First, a risk-adaptive security policy generation mechanism is proposed to jointly adjust privacy budgets, secure uploading levels, and communication compression parameters according to client risk, resource load, communication latency, prediction uncertainty, and historical trustworthiness. Second, a sketch-assisted robust aggregation mechanism is developed for privacy-constrained uploading, enabling approximate estimation of client-update deviation without accessing complete plaintext updates. Third, a closed-loop feedback trustworthiness updating mechanism is introduced, in which human review or simulated feedback is converted into reliability-weighted weak supervision and used to update client trustworthiness, risk scores, and aggregation weights. Fourth, a flexible interaction interface prototype is designed for industrial anomaly review. The interface integrates alarm confidence, uncertainty, feature attribution, risk level, and feedback entry, and its effectiveness is evaluated through feedback delay, feedback reliability, review workload, and before–after interface comparison.

2 Related Work

This section reviews industrial anomaly detection, federated industrial detection, privacy-preserving and communication-efficient federated learning, robust aggregation, and feedback-assisted interaction. Existing studies have improved IIoT perception from detection modeling, privacy protection, heterogeneous collaboration, poisoning defense, and explainable interaction perspectives. However, a unified mechanism linking client risk, secure uploading, communication efficiency, robust aggregation, feedback reliability, and interaction evaluation remains insufficiently explored.

2.1 Industrial Anomaly Detection and Intrusion Identification

Early industrial anomaly detection relied on statistical analysis, rule-based reasoning, and shallow machine learning methods, such as support vector machines, random forests, isolation forests, and Bayesian inference [8]. These methods are interpretable but depend heavily on manual feature engineering and are less adaptable to high-dimensional, nonlinear, and multi-source IIoT data.

Deep learning methods, including Convolutional Neural Network, Recurrent Neural Network, Long Short-Term Memory, autoencoders, variational autoencoders, and attention-based models, have been widely used for industrial traffic, sensor time series, logs, and control states [9–12]. Although they can learn complex abnormal patterns, they still face label scarcity, class imbalance, distribution shift, privacy constraints, and cross-node collaboration difficulties. Therefore, industrial anomaly detection needs to be further combined with federated learning, privacy protection, and feedback-assisted mechanisms.

2.2 Federated Industrial Anomaly Detection and Heterogeneous Collaboration

Federated learning enables collaborative modeling without sharing raw industrial data, providing a feasible solution for privacy-constrained IIoT detection [13–15]. Recent studies have introduced federated deep models for distributed intrusion detection and anomaly identification [16–18]. Methods such as FedAvg, FedProx, SCAFFOLD, personalized FL, clustered FL, and hierarchical FL have been used to handle non-IID data, client drift, and edge collaboration.

However, most studies focus on data heterogeneity and collaborative training, while dynamic industrial factors such as risk level, device load, communication latency, and participation quality are less fully considered. Fixed privacy budgets, secure uploading levels, sparsity ratios, and quantization bit-widths may lead to insufficient protection or redundant communication overhead. Therefore, this study introduces risk-adaptive policy generation for heterogeneous industrial clients. This motivates a client-state-aware design in which privacy budgets, compression levels, and aggregation weights are coordinated rather than configured independently. Bakirci's distributed swarm architecture provides relevant background for resource-aware collaboration, decentralized communication, task distribution, and optimized routing in communication-constrained edge systems [19].

2.3 Privacy-Preserving and Communication-Efficient Federated Learning

Differential privacy, secure aggregation, homomorphic encryption, gradient compression, pruning, sparse uploading, and quantization have been widely studied to reduce privacy leakage and communication costs in federated learning [20, 21]. Recent adaptive FL methods further adjust participation

or uploading strategies according to client resources, training quality, or communication status.

Nevertheless, these mechanisms are often used as independent modules or fixed hyperparameters. A fixed privacy budget may ignore client risk differences, while fixed compression may overlook resource heterogeneity. Few studies jointly consider privacy sensitivity, resource pressure, prediction uncertainty, and historical trustworthiness. This paper incorporates these factors into a unified policy generation process to analyze privacy–communication–latency trade-offs. Recent IIoT-oriented privacy-preserving and secure ML frameworks have also considered irregular clients, secure learning, and cybersecurity protection in next-generation IoT or IIoT environments [22]. These studies further indicate that industrial anomaly detection should be evaluated not only by detection accuracy, but also by privacy protection, robustness, communication cost, and deployment constraints under heterogeneous edge conditions.

2.4 Robust Aggregation and Poisoning Defense

Federated learning is vulnerable to malicious clients, label flipping, model poisoning, and backdoor attacks. Robust aggregation methods such as Krum, Trimmed-Mean, Median, Bulyan, geometric median, FLTrust, and FoolsGold suppress abnormal updates through distance, trimming, directional similarity, trusted root data, or client similarity analysis [23–25].

However, many robust methods require plaintext update coordinates or statistics, which are difficult to access under secure aggregation or privacy-constrained uploading. Recent studies have explored encrypted similarity computation, verifiable aggregation, functional encryption, and secure sketches to balance privacy and robustness [26–28]. This paper does not provide a complete cryptographic proof; instead, it uses noisy random projection sketches and bounded influence functions to estimate update deviations under controlled semi-honest simulation settings. This design is motivated by the practical constraint that plaintext update distances or coordinate-wise statistics may be unavailable under privacy-constrained uploading.

2.5 Human-in-the-Loop Feedback and Flexible Interaction Interfaces

Industrial perception systems should provide not only anomaly detection results but also interpretable and auditable support for operators and engineers. Existing studies use explainable AI, visual analytics, digital twins,

and interaction interfaces to present equipment states, anomaly causes, confidence, feature importance, and propagation paths [29, 30]. Human-in-the-loop anomaly detection, active learning, and weak supervision further use expert review, label correction, and uncertain feedback to improve models [31, 32].

However, feedback is often treated only as additional labels, while feedback delay, reliability, review workload, and interface comparison are rarely modeled. In federated industrial settings, feedback quality may vary across operators, devices, and client nodes. Therefore, this study designs a flexible interaction interface and incorporates interface-level metrics into the federated trustworthiness updating loop. The key difference is that feedback delay and reliability are modeled as factors affecting trustworthiness and subsequent aggregation, rather than treating feedback simply as additional labels.

2.6 Positioning of this Study

Existing studies have advanced industrial anomaly detection, federated learning, privacy protection, robust aggregation, and feedback-assisted learning. However, these aspects are often studied separately. Recent IIoT and IoT privacy-preserving learning studies also highlight irregular-client handling, secure learning, and resource-aware collaboration, but their integration with feedback reliability and trust-aware aggregation remains limited. When clients operate under heterogeneous resources, privacy-constrained uploading, malicious-update risks, and delayed or uncertain feedback, a coordinated mechanism is still insufficiently explored.

To address these issues, this paper proposes a risk-adaptive secure federated learning framework with risk-adaptive policy generation, sketch-assisted robust aggregation, closed-loop feedback trustworthiness updating, and flexible interaction interface optimization. The key difference from existing separated designs is the round-level coordination among protected uploading, sketch-based deviation estimation, aggregation weighting, and feedback reliability modeling.

The objective is to evaluate robustness, communication efficiency, calibration, feedback reliability, review workload, and overall stability under controlled federated simulation settings. Therefore, the framework should be understood as a simulation-based coordination mechanism rather than a complete cryptographic security solution or a validated real industrial deployment system.

3 Methodology

To improve readability and reproducibility, the key symbols used in the methodology are summarized in Table 1. These variables are used to describe client states, adaptive uploading policies, sketch-assisted robust aggregation, feedback modeling, and trustworthiness updating.

Table 1 Key symbols and operational definitions

Symbol	Meaning	Range / Type	Source / Estimation	Used in
i	Client index	Integer	Federated client set	Client-level modeling
t	Federated round index	Integer	Training protocol	Round-level update
D_i	Local dataset of client i	Dataset	Public IIoT/ICS data partition	Local training
p_i^t	Privacy sensitivity of client i in round t	[0, 1]	Simulated protocol	Privacy-risk scoring
r_i^t	Resource load of client i in round t	[0, 1]	Simulated protocol	Resource-risk scoring
l_i^t	Communication latency of client i in round t	Non-negative value	Simulated protocol	Resource-pressure estimation
u_i^t	Prediction uncertainty of client i in round t	[0, 1]	Model output	Review triggering and risk scoring
τ_i^t	Trustworthiness of client i in round t	[0, 1]	Updated variable	Aggregation weighting and next-round risk
d_i^t	Update deviation of client i in round t	Non-negative value	Sketch-based estimation	Robust aggregation
ϵ_i^t	Differential privacy budget of client i in round t	Positive value	Policy output	Privacy protection
q_i^t	Secure uploading level of client i in round t	Discrete level	Policy output	Secure uploading cost simulation
s_i^t	Sparsity ratio of client i in round t	[0, 1]	Policy output	Communication compression
b_i^t	Quantization bit-width of client i in round t	Positive integer	Policy output	Communication compression
Δw_i^t	Local model update before protection	Vector	Local training	Client update generation
$\widetilde{\Delta w}_i^t$	Protected local update after clipping, noise, sparsification, and quantization	Vector	Uploading policy and local update	Secure aggregation
R	Random projection matrix	Matrix	Shared projection seed	Sketch generation

Table 1 Continued

Symbol	Meaning	Range / Type	Source / Estimation	Used in
z_i^t	Noisy random projection sketch of client i	Low-dimensional vector	Sketch generation	Deviation estimation
ξ_i^t	Sketch noise	Random noise	Sketch perturbation	Privacy-constrained sketching
$\bar{z}^t$	Robust center in sketch space	Vector	Server-side robust estimation	Raw deviation computation
$\tilde{d}_i^t$	Policy-corrected update deviation	Non-negative value	Sketch deviation with policy correction	Aggregation weighting
α_i^t	Aggregation weight of client i in round t	[0, 1], normalized	Server-side weighting	Global model update
B_i^t	Feedback cache of client i in round t	Set / buffer	Delayed feedback storage	Feedback regularization
$\hat{y}^{fb}$	Weak supervision label from feedback	Label / soft label	Operator or simulated expert feedback	Feedback loss
ω^{fb}	Feedback reliability weight	[0, 1]	Delay, reliability, and consistency estimation	Feedback regularization
C_{comm}	Communication and computation cost	Non-negative value	Policy-dependent cost estimation	Decomposed optimization
C_{int}	Interface-related cost	Non-negative value	Feedback delay, review burden, and reliability	Interface evaluation

3.1 Problem Formulation

Assume that an IIoT system contains K edge clients. The k-th client holds a local dataset:

$$D_k = \{(x_i^k, y_i^k)\}_{i=1}^{N_k}, x_i^k \in \mathbb{R}^{T\times d}, y_i^k \in \mathcal{Y}, \tag{1}$$

where $\mathcal{Y}$ denotes the set of normal and attack classes. For binary industrial process anomaly detection, $\mathcal{Y} = \{0, 1\}$. The data of client k follow a local distribution $P_k(X, Y)$, and non-IID differences usually exist among clients. The standard federated learning objective is:

$$\min_{\theta} \sum_{k=1}^{K} \pi_k \mathbb{E}_{(X,Y)\sim P_k}[\ell(f_\theta(X), Y)], \pi_k = \frac{N_k}{\sum_j N_j}. \tag{2}$$

Beyond detection loss, IIoT federated detection must also consider privacy protection, communication latency, resource constraints, malicious

updates, and feedback uncertainty. Therefore, this study introduces client states, uploading policies, robust aggregation weights, and trustworthiness variables. In round t, the state and uploading policy of client k are defined as:

$$z_k^t = [p_k^t, r_k^t, \tau_k^t, u_k^t, q_k^{t-1}, \hat{d}_k^{t-1}], a_k^t = [\varepsilon_k^t, h_k^t, \rho_k^t, b_k^t]. \tag{3}$$

Here, p_k^t, r_k^t, τ_k^t, u_k^t, q_k^{t-1}, and $\hat{d}_k^{t-1}$ denote privacy sensitivity, resource load, communication latency, prediction uncertainty, previous-round trustworthiness, and previous-round update deviation, respectively. The variables ε_k^t, h_k^t, ρ_k^t, and b_k^t denote the per-round differential privacy budget, secure uploading level, sparsity ratio, and quantization bit-width.

Trustworthiness in this study refers to a bounded empirical reliability indicator, not a social-trust concept or a cryptographic trust guarantee. It summarizes feedback utility, update-deviation behavior, and trust-related risk, and is used only for aggregation weighting and next-round risk assessment.

After local training, client k obtains $\Delta\theta_k^t$, which is clipped, perturbed, sparsified, and quantized to form the protected update:

$$\widehat{\Delta\theta}_k^t = Q_{b_k^t}(S_{\rho_k^t}(\mathrm{Clip}(\Delta\theta_k^t, C_\Delta) + \xi_k^t)), \tag{4}$$

where ξ_k^t is differential privacy noise. Since ε_k^t varies across clients and rounds, each client's privacy trajectory is recorded, and an RDP accountant is used to estimate the worst-case cumulative, $(\varepsilon_{\mathrm{total}}^{\max}, \delta_{\mathrm{DP}})$-DP guarantee.

To estimate update deviation under privacy-constrained conditions, each client generates a noisy random projection sketch:

$$g_k^t = R^t \widehat{\Delta\theta}_k^t + \nu_k^t, \tag{5}$$

where R^t is a low-dimensional random projection matrix and ν_k^t is sketch noise. The server observes g_k^t, but does not access the complete plaintext update. The sketch is not equivalent to cryptographic encryption and does not guarantee zero leakage; it only reduces complete-update exposure under a semi-honest server and controlled simulation setting. Therefore, the sketch branch is used only as a privacy-constrained deviation-estimation tool, rather than as a formal privacy-preserving or cryptographic security mechanism. Stronger threats such as sketch inversion, sketch forgery, client collusion, or malicious-server behavior are beyond the default threat model of this study.

The server computes aggregation weights using sketch-based deviation, trustworthiness, and risk status, and updates the global model as:

$$\theta^{t+1} = \theta^t + \eta_g \sum_{k \in S_t} \alpha_k^t \widehat{\Delta\theta}_k^t. \tag{6}$$

Here, α_k^t is a server-side aggregation weight or a protocol-constrained weighting coefficient, rather than a local operation relying on malicious clients' voluntary execution. This study assumes that malicious clients may upload abnormal updates, but cannot forge sketch consistency, bypass server-side weights, or compromise the secure aggregation protocol. Stronger attacks require verifiable sketches or verifiable secure aggregation and are beyond the default threat model.

3.2 Overall Framework

As shown in Figure 1, the proposed framework consists of four closely connected components: risk-adaptive security policy generation, sketch-assisted robust aggregation, flexible interaction and feedback collection, and closed-loop feedback trustworthiness updating. Rather than being a simple combination of separate modules, these components are coupled through client states, protected updates, sketch-based deviations, aggregation weights, feedback reliability, and next-round trustworthiness. They are selected to match the main constraints of this setting: heterogeneous clients require adaptive uploading, restricted update visibility requires sketch-based deviation estimation, abnormal uploads require robust weighting, and delayed or uncertain feedback requires reliability-aware trust updating. These components do

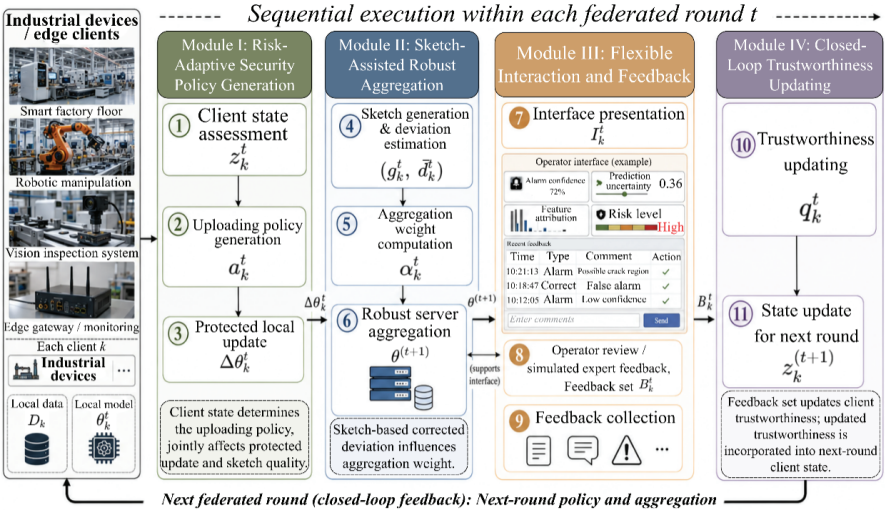


Figure 1 Overall framework of the proposed risk-adaptive federated learning system with flexible interaction.

not constitute a single end-to-end jointly optimized model. Instead, they are executed sequentially in each round of federated training, including client-state assessment, policy generation, protected update uploading, sketch-based deviation estimation, robust aggregation, interface-assisted feedback collection, and trustworthiness updating.

The overall variable transmission process is formulated as:

$$z_k^t \to a_k^t \to \widehat{\Delta\theta}_k^t \to (g_k^t, \hat{d}_k^t) \to \alpha_k^t \to \theta^{t+1}, \quad F_k^t \to B_k^t \to q_k^t \to z_k^{t+1}, \tag{7}$$

where the client state z_k^t determines the uploading policy a_k^t. The uploading policy affects the protected update $\widehat{\Delta\theta}_k^t$ and the quality of the sketch g_k^t. The sketch-based corrected deviation $\hat{d}_k^t$ further influences the aggregation weight α_k^t, which is used to update the global model θ^{t+1}. Meanwhile, the flexible interaction interface selects review samples F_k^t and stores the collected feedback in the feedback cache B_k^t. The cached feedback then affects the client trustworthiness q_k^t, and the updated trustworthiness is incorporated into the next-round client state z_k^{t+1}.

This closed loop describes the dependency relationships among security policy generation, protected uploading, robust aggregation, interface-assisted feedback, and trustworthiness updating. It should be interpreted as a round-level coordination mechanism rather than strict causal proof. The empirical contribution of each component is evaluated in subsequent ablation studies, attack-sensitivity experiments, feedback-noise experiments, and interface evaluation.

Sequential modules integrate client state, policy generation, sketch-based aggregation, flexible feedback, and trustworthiness updating.

3.3 Core Modules

3.3.1 Risk-adaptive security policy generation

To avoid semantic overloading caused by a single risk score, this study decomposes the client state into privacy-sensitive risk, resource-pressure risk, and trust-related risk:

$$\begin{aligned} r_{\mathrm{priv},k}^t &= \mathrm{clip}(w_p p_k^t + w_u u_k^t + w_q(1 - q_k^{t-1}), 0, 1), \\ r_{\mathrm{res},k}^t &= \mathrm{clip}(w_r r_k^t + w_\tau \tau_k^t, 0, 1), \\ r_{\mathrm{trust},k}^t &= \mathrm{clip}(w_d \hat{d}_k^{t-1} + w_{q'}(1 - q_k^{t-1}) + w_{u'} u_k^t, 0, 1). \end{aligned} \tag{8}$$

Here, $r^t_{\mathrm{priv},k}$ controls the differential privacy budget and secure uploading level, $r^t_{\mathrm{res},k}$ controls the sparsity ratio and quantization bit-width, and $r^t_{\mathrm{trust},k}$ is used in robust aggregation weighting and trustworthiness updating. The above risk scores are obtained through rule-based mapping rather than end-to-end learning. Their purpose is to enable clients with different states to adopt differentiated uploading policies. The rule-based design is used for transparency, auditability, and safety controllability, while learning-based policies such as reinforcement learning or meta-learning require additional exploration, reward design, and stability guarantees.

The uploading policy is defined as:

$$\begin{aligned}\varepsilon^t_k &= \varepsilon_{\max} - (\varepsilon_{\max} - \varepsilon_{\min}) r^t_{\mathrm{priv},k}, h^t_k = \mathbb{I}(r^t_{\mathrm{priv},k} > \delta_1) + \mathbb{I}(r^t_{\mathrm{priv},k} > \delta_2), \\ \rho^t_k &= \rho_{\min} + (\rho_{\max} - \rho_{\min}) r^t_{\mathrm{res},k}, b^t_k = b_{\max} - |(b_{\max} - b_{\min}) r^t_{\mathrm{res},k}|.\end{aligned} \tag{9}$$

When the privacy-sensitive risk is high, the client adopts a smaller ε^t_k and a higher secure uploading level. When the resource pressure is high, the client adopts a higher sparsity ratio and a lower quantization bit-width. As shown in Figure 2, to avoid protocol inconsistency caused by mixing heterogeneous ciphertext spaces, this study adopts a unified secure aggregation interface and records communication and computational overhead according to the secure

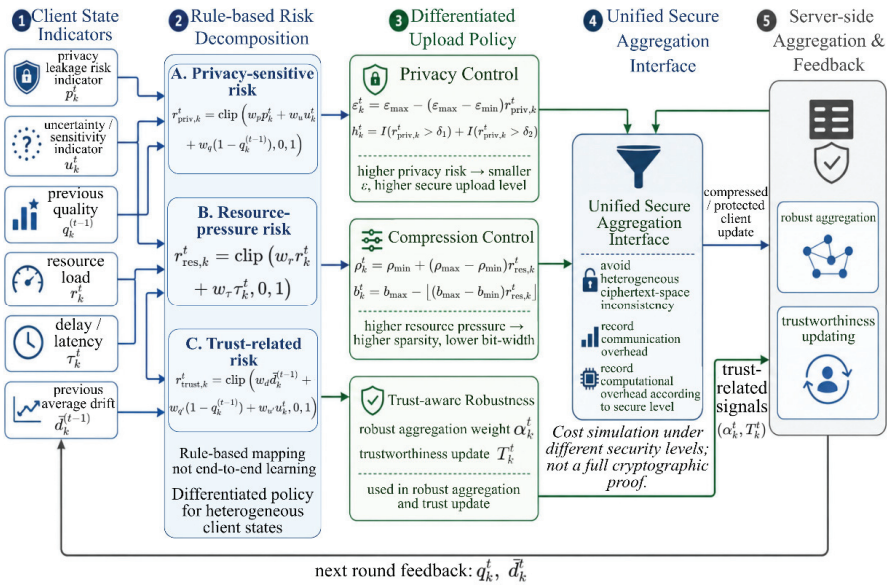


Figure 2 Risk-decomposed adaptive upload policy for federated IIoT clients.

uploading level. This design is used to simulate system costs under different security levels and is not equivalent to a complete cryptographic protocol proof.

Rule-based risks guide privacy protection, compression, and trust-aware aggregation for heterogeneous federated clients.

3.3.2 Sketch-assisted robust aggregation

The server estimates client-update deviation based on noisy sketches. Standard secure aggregation protects update contents but also limits the server's ability to identify abnormal uploads. The sketch branch is therefore introduced not as a stronger cryptographic primitive, but as an auxiliary low-dimensional signal for deviation estimation under restricted visibility. First, the robust center and the raw deviation are computed in the sketch space:

$$\mu_g^t = \mathrm{Median}(\{g_k^t\}_{k\in S_t}), d_k^t = \|g_k^t - \mu_g^t\|_2. \tag{10}$$

Since differential privacy noise, sparsification, and quantization errors may also cause deviations in normal clients, this study introduces a policy-perturbation correction term:

$$\bar{d}_k^t = \max(0, d_k^t - a_\varepsilon \sigma^2(\varepsilon_k^t) - a_\rho \rho_k^t - a_b 2^{-b_k^t}). \tag{11}$$

This correction term is used only to approximately compensate for the expected deviation caused by DP perturbation, compression, and quantization. It does not claim to strictly distinguish non-IID drift, normal compression errors, and truly malicious updates. Its effectiveness is mainly validated empirically through ablation studies and attack-sensitivity experiments.

The server then computes aggregation weights using a bounded influence function and trustworthiness information:

$$\alpha_k^t = (1-\zeta)\frac{\exp[-\beta\psi(\bar{d}_k^t) + \gamma q_k^{t-1} - \lambda_r r_{\mathrm{trust},k}^t]}{\sum_{j\in S_t}\exp[-\beta\psi(\bar{d}_j^t) + \gamma q_j^{t-1} - \lambda_r r_{\mathrm{trust},j}^t]} + \frac{\zeta}{|S_t|}, \tag{12}$$

where $\psi(\bar{d}) = \min(\bar{d}, \delta_\psi)$. A larger deviation and higher trust-related risk reduce the aggregation weight, whereas higher historical trustworthiness increases the aggregation weight. Finally, the server obtains the global update through a unified weighted secure aggregation interface:

$$\Delta\theta gt = SecAgg(\{\alpha kt\Delta\theta^{kt}\}k \in St), \theta t + 1 = \theta t + \eta g\Delta\theta gt. \tag{13}$$

As shown in Figure 3, this study assumes that the secure aggregation interface can correctly execute weighted aggregation. If clients can forge sketches,

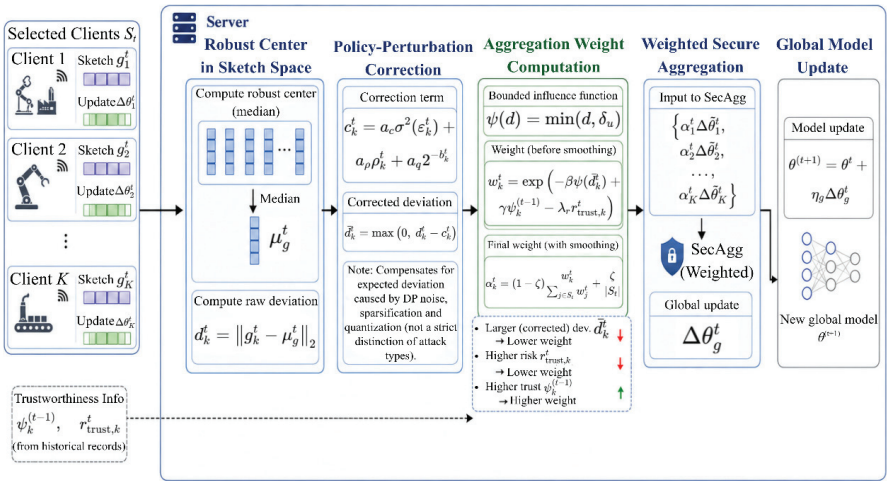


Figure 3 Sketch-assisted robust aggregation workflow (server-side).

upload updates inconsistent with their sketches, or bypass weighting constraints, additional mechanisms such as verifiable sketches, consistency proofs, or verifiable secure aggregation protocols are required. Such attacks are beyond the default experimental setting of this paper. Thus, the practical role of sketch-assisted aggregation is to provide approximate deviation signals for robust weighting under restricted update visibility, not to replace verifiable secure aggregation or provide a complete defense against fully adaptive adversaries. Accordingly, all security-related results should be interpreted within the semi-honest, non-collusive, and controlled simulation threat model used in this study.

3.3.3 Flexible Feedback Interaction and Trustworthiness Updating

For samples with high anomaly scores, high prediction uncertainty, low client trustworthiness, or abnormal update deviation, the system triggers an interface-assisted review entry:

$$F_k^t = \{x_i^k | s_i^{(k,t)} \geq \tau_s \quad \text{or} \quad u_i^{(k,t)} \geq \tau_u \quad \text{or} \quad q_k^{t-1} \leq \tau_q \quad \text{or} \quad \bar{d}_k^t \geq \tau_d\}. \tag{14}$$

For each reviewed sample, the flexible interaction interface presents the anomaly score, prediction confidence, uncertainty level, risk score, top feature attribution, historical client trustworthiness, and recommended

action. The operator or simulated expert can confirm the anomaly, reject the alarm, mark it as uncertain, correct the attack type, or indicate unreliable sensor/traffic sources.

For $x_i^k \in F_k^t$, human review or simulated feedback is mapped to a weak supervision label $\tilde{y}_i^{(k,t)}$. The feedback weight $\lambda_i^{(k,t)}$ is generated according to feedback delay, operator reliability, consistency with model confidence, and consistency with historical feedback. The feedback regularization loss is defined as:

$$L_{\mathrm{fb},k}^t = -\frac{1}{|F_k^t|} \sum_{x_i^k \in F_k^t} \lambda_i^{(k,t)} \sum_{c=1}^{|\mathcal{Y}|} \tilde{y}_{i,c}^{(k,t)} \log p_{i,c}^{(k,t)}. \tag{15}$$

To avoid temporal-order conflicts, local training in round t uses only feedback cached from the previous round:

$$L_k^t = L_{\mathrm{sup},k}^t + \lambda_f L_{\mathrm{fb},k}^{t-1}. \tag{16}$$

Operationally, this score is treated only as a bounded reliability state that records whether a client has shown consistent feedback behavior and non-abnormal update deviation in previous rounds. Client trustworthiness is smoothly updated according to feedback utility, update deviation, and trust-related risk:

$$q_k^t = \mu q_k^{t-1} + (1-\mu)\mathrm{clip}[\lambda_V V_k^t + \lambda_D(1-\bar{D}_k^t) + \lambda_R(1-r_{\mathrm{trust},k}^t), 0, 1]. \tag{17}$$

When no feedback samples are available, V_k^t is removed. As shown in Figure 4, feedback affects the framework through two pathways: it participates in later local training through feedback regularization and influences next-round risk scoring and aggregation weights through trustworthiness updating. Since feedback utility, trustworthiness, and aggregation weights are endogenously related, this study interprets them as controlled mechanism effects rather than strict causal evidence. Its effectiveness is evaluated through ablation studies, feedback-noise sensitivity, ECE, Trust-AUC, feedback delay, feedback reliability, and review workload.

Feedback drives both local training and next-round trust adaptation.

3.4 Flexible Interaction Interface Design Optimization

To improve feedback collection and support trustworthiness updating, this study designs a flexible interaction interface for industrial anomaly review.

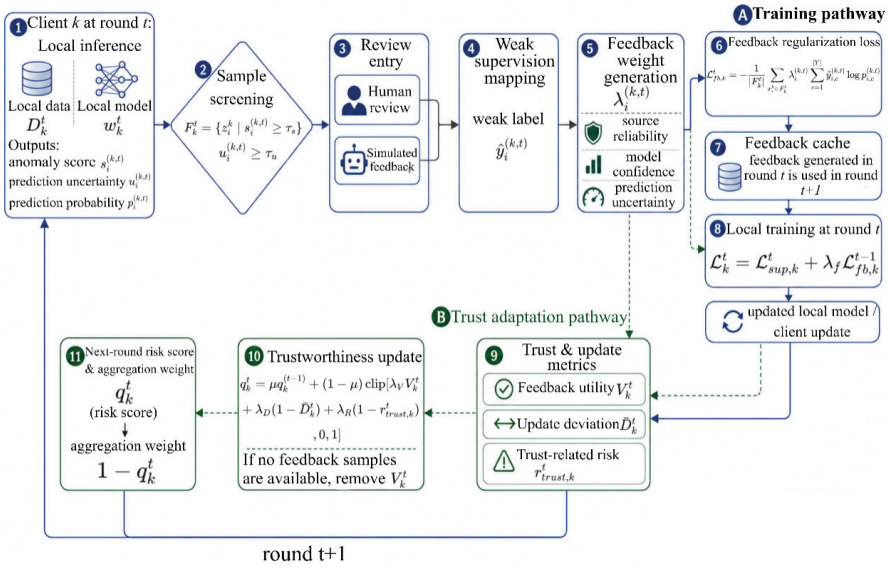


Figure 4 Closed-loop feedback trustworthiness updating mechanism.

The interface aims to reduce manual review burden, improve feedback reliability, and shorten feedback delay. High-anomaly, high-uncertainty, low-trustworthiness, or abnormal-deviation samples are prioritized for review.

The interface contains five functional parts: an alarm overview panel for anomaly time, device ID, predicted attack type, and risk level; an uncertainty and confidence panel for prediction confidence, uncertainty score, and threshold position; a feature attribution panel for Flow, Protocol, Temporal, Sensor/Actuator, and Log/Event contributions; a feedback entry panel for confirming, rejecting, marking uncertainty, or correcting attack types; and a trustworthiness panel for client trustworthiness and historical feedback consistency.

The interface optimization objective is defined as:

$$\min \mathcal{J}_{ui} = \lambda_1 D_{fb} + \lambda_2 B_{review} - \lambda_3 R_{fb} + \lambda_4 U_{uncertain}, \tag{18}$$

where D_{fb} denotes feedback delay, B_{review} denotes manual review burden, R_{fb} denotes feedback reliability, and $U_{uncertain}$ denotes high-uncertainty sample coverage. The collected feedback is further used for feedback weighting, client trustworthiness updating, and next-round risk scoring and aggregation.

3.5 Decomposed Optimization

This study does not treat model parameters, uploading policies, aggregation weights, interface feedback, and client trustworthiness as jointly optimized variables within a single differentiable objective. Instead, it adopts a decomposed optimization procedure. Model parameters are updated by local optimizers; uploading policies are generated through state-mapping rules; aggregation weights are computed by the server based on sketch-based deviations; interface feedback is collected through the flexible interaction layer; and client trustworthiness is updated according to feedback utility and update deviation.

To describe the round-level coordination among different modules, we define:

$$J^t = \sum_{k \in S_t} \alpha_k^t \left(L_{\text{sup},k}^t + \lambda_f L_{\text{fb},k}^{t-1} \right) + \lambda_{\text{rob}} \sum_{k \in S_t} \alpha_k^t \psi(\bar{d}_k^t) + \lambda_{\text{cost}} \sum_{k \in S_t} C(a_k^t) + \lambda_{\text{ui}} \mathcal{J}_{ui}^t, \tag{19}$$

where $C(a_k^t)$ denotes the communication and computational cost determined by privacy budget, secure uploading level, sparsity ratio, and quantization bit-width. $\mathcal{J}_{ui}^t$ denotes the interface-related cost involving feedback delay, review burden, feedback reliability, and uncertainty coverage. Equation (19) is not an end-to-end differentiable objective. It only describes the trade-off among detection loss, feedback loss, robust deviation penalty, system overhead, and interface-assisted feedback quality. In implementation, these components are executed sequentially through local training, state mapping, sketch-based deviation estimation, robust weighting, interface-assisted feedback collection, and trustworthiness updating.

3.6 Stability Discussion

This study discusses stability only from the perspective of boundedness, rather than providing a rigorous convergence proof. Under the assumptions that local gradients, stochastic variance, clipped updates, differential privacy noise, compression error, quantization error, sketch perturbation, and feedback weights are bounded, the global update error can be described as:

$$\Delta_{\text{err}}^t = \Delta_{\text{sgd}}^t + \Delta_{\text{DP}}^t + \Delta_{\text{comp}}^t + \Delta_{\text{sketch}}^t + \Delta_{\text{nonIID}}^t + \Delta_{\text{weight}}^t + \Delta_{\text{fb}}^t. \tag{20}$$

These terms correspond to stochastic optimization error, privacy perturbation, compression and quantization error, sketch estimation error, non-IID

bias, aggregation-weight error, and feedback-induced error. Since aggregation weights are normalized, the bounded influence function satisfies $0 \leq \psi(\bar{d}_k^t) \leq \delta_\psi$, trustworthiness is clipped into $[0, 1]$, and feedback weights are bounded, we have:

$$\sum_{k \in S_t} \alpha_k^t = 1, \alpha_k^t \geq \frac{\zeta}{|S_t|}, \quad q_k^t \in [0, 1], 0 \leq \lambda_i^{(k,t)} \leq 1. \tag{21}$$

Therefore, under the default protocol assumptions, no single client or feedback source can exert unbounded influence on the global update. This discussion does not cover malicious servers, client collusion, adaptive backdoors, sketch inversion, feedback manipulation, or protocol disruption. Under catastrophic client dropout, sustained malicious participation, systematic feedback manipulation, or sketch-update inconsistency, the current bounded-influence and trust-smoothing mechanisms cannot guarantee recovery or correctness and require additional verifiable and fault-tolerant mechanisms. Empirical stability is evaluated through controlled federated simulations, poisoning attacks, feedback noise, and interface-assisted review experiments.

3.7 Algorithm Procedure

The secure aggregation interface in Algorithm 1 provides a unified description of the weighted aggregation process for protected updates. This study mainly evaluates abnormal updates, poisoning attacks, feedback noise, and interface-assisted review under controlled simulation settings. Stronger threats, such as sketch forgery, update-sketch inconsistency, client collusion, feedback manipulation, or protocol disruption, require additional verifiable mechanisms and are beyond the default experimental setting.

Algorithm 1 Training and Inference Procedure of the Proposed Framework

Phase	Step	Operation
Input	–	Client datasets $\{D_k\}$, initial global model θ^0, initial trustworthiness q_k^0, feedback cache B_k^0, initial corrected deviation $\bar{d}_k^0$, total rounds T.
Output	–	Final global model θ^T, client trustworthiness q_k^T.
Training	1	Initialize the RDP privacy accountant.
	2	For each global round $t = 0, 1, \ldots, T-1$, the server selects participating clients S_t.
	3	The server broadcasts the current global model θ^t and the random projection seed.
	4	For each client $k \in S_t$, compute the client state $z_k^t = [p_k^t, r_k^t, \tau_k^t, u_k^t, q_k^{t-1}, \bar{d}_k^{t-1}]$.
	5	Compute privacy-sensitive risk, resource-pressure risk, and trust-related risk.
	6	Generate the uploading policy $a_k^t = (\varepsilon_k^t, h_k^t, \rho_k^t, b_k^t)$.

Train the local model using supervised loss and cached feedback loss from the previous round.
Obtain the local update $\Delta\theta_k^t$.
Clip, perturb, sparsify, and quantize $\Delta\theta_k^t$ to obtain the protected update $\widehat{\Delta\theta}_k^t$.
Generate the noisy random projection sketch $g_k^t = R^t \widehat{\Delta\theta}_k^t + \nu_k^t$.
Upload the protected update and sketch statistics through the unified secure uploading interface.
The server computes the robust sketch center, raw deviation d_k^t, and policy-corrected deviation $\bar{d}_k^t$.
The server computes aggregation weights α_k^t according to $\bar{d}_k^t$, q_k^{t-1}, and $r_{\mathrm{trust},k}^t$.
The server obtains the global update through weighted secure aggregation: $\Delta\theta_g^t = \mathrm{SecAgg}(\{\alpha_k^t \widehat{\Delta\theta}_k^t\}_{k\in S_t})$.
Update the global model: $\theta^{t+1} = \theta^t + \eta_g \Delta\theta_g^t$.
Trigger the flexible interaction interface for samples with high anomaly scores, high uncertainty, low trustworthiness, or abnormal deviation.
The interface presents alarm confidence, uncertainty, feature attribution, risk level, and feedback entry.
Operator review or simulated expert feedback is mapped into weak labels and stored in the feedback cache.
Compute feedback utility and update client trustworthiness q_k^t through exponential smoothing.
Pass q_k^t, $\bar{d}_k^t$, feedback cache, and privacy-accounting state to the next round.

Inference

Given a test sample, the global model outputs the predicted class, anomaly score, and prediction uncertainty.
If the anomaly score or uncertainty exceeds the threshold, the flexible review entry is triggered.
Feedback recorded during testing is used only to illustrate the interaction mechanism and is not used to retroactively adjust test results.

4 Experiments and Analysis

4.1 Dataset Settings

4.1.1 Dataset

This study uses Edge-IIoTset and X-IIoTID as primary datasets, and TON_IoT, WUSTL-IIoT, SWaT, and HAI as supplementary datasets to evaluate IIoT intrusion detection, ICS anomaly detection, non-IID federated training, and malicious-client attacks. These datasets are public IIoT/ICS datasets rather than naturally cross-factory federated datasets. Thus, experiments are conducted in a simulated federated environment: samples and labels are taken from public datasets, while client partitioning, non-IID distributions, resource load, communication latency, privacy sensitivity, malicious behaviors, and feedback signals are generated by a unified protocol. Therefore, client partitioning, resource load, communication latency,

Table 2 Dataset statistics and federated partition settings

Dataset	Samples Total / Train / Val / Test	Features	Classes	Clients	Partition
Edge-IIoTset	1,258,360 / 880,852 / 125,836 / 251,672	61	2 / 15	20	Dirichlet non-IID, $\alpha = 0.5$
X-IIoTID	632,412 / 442,688 / 63,241 / 126,483	67	2 / 18	20	Dirichlet non-IID, $\alpha = 0.5$
TON_IoT	461,043 / 322,730 / 46,104 / 92,209	52	2 / 10	20	By modality or class ratio
WUSTL-IIoT	780,866 / 546,606 / 78,087 / 156,173	28	2	20	By class ratio
SWaT	496,320 / 342,180 / 38,020 / 116,120	51	2	20	By time, condition, or subsystem
HAI	368,640 / 254,016 / 28,224 / 86,400	78	2	20	By time, condition, or subsystem

privacy sensitivity, malicious-client behavior, and feedback signals should be regarded as protocol-generated variables, and they are used only for controlled evaluation rather than as evidence of real industrial deployment performance. The results should be interpreted as controlled simulation evidence rather than real deployment validation.

To prevent data leakage, each dataset is first split into training, validation, and test sets, and federated clients are constructed only from the training set. Preprocessing parameters are fitted only on the training set and fixed for validation and testing. Potential leakage fields such as timestamps, IP addresses, ports, and session identifiers are removed. For SWaT and HAI, chronological or official splits and sliding windows are used. Simulated variables are used only for risk policy generation, communication-cost simulation, and trustworthiness updating, not as model inputs. The dataset statistics and federated partition settings are summarized in Table 2.

All methods use the same data versions, splits, client partitions, random seeds, and preprocessing settings. Cross-dataset testing is interpreted only after common feature mapping or similar feature-group alignment.

4.1.2 Baseline methods

To ensure fair comparison, the compared methods are grouped into five categories: basic federated baselines, privacy/communication baselines, plaintext robust references, strong industrial FL baselines, and ablation variants. All methods use the same data splits, client settings, training protocol, preprocessing pipeline, and random seeds. For FedAvg, FedProx, SCAFFOLD, FedAvg+DP, FedAvg+Compression, FedAvg+SecAgg, and all ablations, the

Table 3 Compared methods and protocol roles

Category	Methods / Variants	Role
Non-federated reference	Centralized-MLP	Centralized performance reference only; not a direct competitor
Basic FL baselines	Local-Only, FedAvg, FedProx, SCAFFOLD	Evaluate local training, standard FL, and non-IID correction
Privacy / communication baselines	FedAvg+DP, FedAvg+Compression, FedAvg+SecAgg	Compare fixed DP, fixed compression, and secure aggregation without robustness
Plaintext robust references	Krum, Trimmed-Mean	Robust references under plaintext-update visibility only
Strong industrial FL baselines	Privacy-Robust FL, SecuFL-IoT, Lightweight FL-IDS	Main baselines for privacy robustness, industrial FL detection, and edge efficiency
Full model	Ours	Complete proposed framework
Ablation variants	w/o RiskPolicy, ResourceOnly, PrivacyOnly, CompressionOnly, w/o SketchRobust, w/o StrategyCorrection, Plain SecAgg, w/o FeedbackTrust, DirectFeedback, WeightedFeedbackOnly, NoTrustUpdate, w/o Risk+Sketch, w/o Sketch+Feedback	Analyze module contributions and complementarity

same backbone is used to control model-capacity differences. For SecuFL-IoT and Lightweight FL-IDS, results are reported with their original structures and, when applicable, with the unified backbone. Efficiency metrics are reported separately to distinguish architectural effects from aggregation-mechanism effects. The compared methods and their protocol roles are summarized in Table 3.

Centralized-MLP is used only as a centralized reference, while Krum and Trimmed-Mean are not fully fair competitors under privacy-constrained uploading because they require plaintext update statistics. The practical competitiveness of the proposed method is mainly assessed against Privacy-Robust FL, SecuFL-IoT, and Lightweight FL-IDS. Ablation studies are used to analyze module effects on detection, robustness, communication efficiency, calibration, and feedback utilization, rather than to prove that every module is significant in all scenarios. Privacy-Robust FL, SecuFL-IoT, and Lightweight FL-IDS are selected as recent baselines because they represent privacy robustness, industrial FL detection, and edge efficiency, respectively. Other recent methods are discussed in Section 2 Related Work but are not used as executable baselines because of different modalities, unavailable implementations, or incompatible deployment assumptions.

4.1.3 Implementation details

All experiments are conducted on a unified platform with an NVIDIA RTX 4090 GPU, 64 GB memory, Ubuntu 22.04, Python 3.10, PyTorch 2.x, and CUDA 12.x. Communication volume includes model updates, sketch statistics, and secure aggregation overhead. Simulated variables, such as resource status, latency, privacy sensitivity, malicious behavior, and feedback signals, are not used as detection-model inputs.

The default setting uses $K = 20$ clients, a participation rate of $\eta_c = 0.5$, 100 global rounds, and 5 local epochs. Network-traffic datasets use a three-layer MLP with hidden dimensions of 256, 128, and 64, while SWaT and HAI use a two-layer LSTM with a hidden size of 128. AdamW is adopted with a learning rate of 1×10^{-3}, batch size 256, and weight decay 1×10^{-4}. Each experiment is repeated five times.

For privacy and compression, $\epsilon_k^t \in [0.5, 8.0]$, $\delta_{\mathrm{DP}} = 10^{-5}$, $\rho_k^t \in [0.1, 0.5]$, quantization bit-widths are $\{4, 8, 16\}$, and the sketch dimension is $p = 0.1d_\theta$. Robustness experiments use 10%, 20%, and 30% malicious clients with Gaussian noise, sign-flipping, and model-scaling attacks. Feedback simulation is used only during training and validation. The main experimental settings are summarized in Table 4.

Table 4 Experimental settings

Module	Default Setting
Environment	RTX 4090; Ubuntu 22.04; Python 3.10; PyTorch 2.x; CUDA 12.x
Federated training	$K = 20$, $\eta_c = 0.5$, $R = 100$, $E = 5$
Model	MLP: 256–128–64; LSTM hidden size: 128
Optimization	AdamW; LR 1×10^{-3}; batch size 256; weight decay 1×10^{-4}
DP / compression	$\varepsilon_k^t \in [0.5, 8.0]$, $\delta_{\mathrm{DP}} = 10^{-5}$, $\rho_k^t \in [0.1, 0.5]$, $b_k^t \in \{4, 8, 16\}$
Sketch / attack / feedback	$p = 0.1d_\theta$; 20% malicious clients by default; 5% feedback ratio

Unless otherwise specified, all methods use the same splits, client partitions, training protocol, random seeds, and preprocessing pipeline.

4.1.4 Evaluation metrics

To evaluate the system-level trade-off, this study uses the privacy–latency–communication–security score (PLCS). For each method, PLCS is defined as a weighted sum of normalized privacy, latency, communication, and security scores. Unless otherwise specified, equal weights are assigned to the four components. Min–max normalization is applied within the same experimental setting, with higher-is-better and lower-is-better metrics normalized separately. The privacy score is related to the cumulative DP budget,

the latency score includes training, inference, and aggregation latency, the communication score is based on per-round communication volume, and the security score is calculated from Robust-F1 and ASR. A higher PLCS indicates a better overall balance among privacy protection, efficiency, communication cost, and robustness. PLCS is used only as a supplementary trade-off indicator rather than a replacement for individual metrics.

4.1.5 Simulated expert review and interface evaluation protocol

Since no real operators are involved, interface evaluation is conducted using a simulated expert review protocol. In each federated round, samples with high anomaly scores or high prediction uncertainty are selected into the review queue. Four simulated interface settings are compared: Baseline, Confidence-only, Explanation-assisted, and Proposed Flexible. These settings differ in displayed information, feedback options, and reliability modeling, as further summarized in Section 4.9.

Feedback reliability is set to 0.7, 0.8, and 0.9, while feedback delay is set to 1, 3, and 5 federated rounds. Delayed feedback is stored in the feedback cache and used only in later rounds. The evaluation compares FD, FR, MRB, HRC, FEG, Trust-AUC, and IES, all of which are generated by the simulated review protocol. These results should be interpreted as a simulation-based assessment of feedback prioritization and reliability-aware feedback flow, rather than as a real user study or direct evidence of operator usability, workload reduction, or decision-support effectiveness.

4.2 Main Experimental Results

Experiments are conducted on six datasets: Edge-IIoTset, X-IIoTID, TON_IoT, WUSTL-IIoT, SWaT, and HAI. Macro-F1 is used for Edge-IIoTset, X-IIoTID, TON_IoT, and WUSTL-IIoT, while anomaly detection F1 is used for SWaT and HAI. All results are reported as mean $\pm$ standard deviation over five independent runs. Centralized-MLP is used only as a non-federated reference, and Krum and Trimmed-Mean are used only as plaintext-visible robust references.

Table 5 indicates that the proposed method achieves the best average performance among the federated methods, but its advantage over strong industrial baselines is limited. Lightweight FL-IDS still performs competitively on some traffic datasets, suggesting that conventional detection accuracy alone is insufficient to fully demonstrate the benefit of the proposed framework. Therefore, as shown in Figure 5, the proposed method should

Table 5 Primary metric comparison across multiple datasets

Method	Edge-IIoTset ↑	X-IIoTID ↑	TON_IoT ↑	WUSTL-IIoT ↑	SWaT ↑	HAI ↑	Average ↑
Centralized-MLP	94.2 ± 0.6	92.5 ± 0.8	89.7 ± 1.0	92.0 ± 0.7	87.4 ± 1.4	84.1 ± 1.9	90.0
Local-Only	83.4 ± 2.0	82.7 ± 2.2	79.1 ± 1.8	84.6 ± 1.7	76.8 ± 2.5	75.2 ± 2.1	80.3
FedAvg	88.5 ± 1.3	87.6 ± 1.5	83.9 ± 1.7	88.1 ± 1.2	80.2 ± 2.1	78.5 ± 1.8	84.5
FedProx	89.4 ± 1.2	88.0 ± 1.3	84.7 ± 1.6	88.5 ± 1.4	81.6 ± 1.9	79.4 ± 2.0	85.3
SCAFFOLD	90.1 ± 1.0	89.3 ± 1.2	86.0 ± 1.4	89.0 ± 1.2	82.5 ± 1.8	80.6 ± 1.7	86.3
FedAvg + DP	87.9 ± 1.6	86.6 ± 1.7	83.2 ± 1.9	87.3 ± 1.5	79.3 ± 2.2	77.8 ± 2.1	83.7
FedAvg + Compression	88.8 ± 1.4	87.3 ± 1.6	84.0 ± 1.8	88.5 ± 1.2	80.1 ± 2.3	77.9 ± 2.2	84.4
FedAvg + SecAgg	89.6 ± 1.1	88.7 ± 1.3	85.1 ± 1.5	89.2 ± 1.3	81.3 ± 2.0	79.5 ± 1.9	85.6
Krum	88.6 ± 1.7	87.9 ± 1.9	84.6 ± 1.8	87.4 ± 1.6	83.2 ± 1.9	80.1 ± 1.8	85.3
Trimmed-Mean	90.2 ± 1.4	88.1 ± 1.6	85.3 ± 1.6	89.4 ± 1.3	83.1 ± 1.7	81.5 ± 1.7	86.3
Privacy-Robust FL	91.2 ± 1.2	89.4 ± 1.4	87.0 ± 1.3	90.2 ± 1.2	84.3 ± 1.7	81.8 ± 1.6	87.3
SecuFL-IoT	91.5 ± 0.9	91.0 ± 1.2	88.2 ± 1.4	90.7 ± 1.1	84.6 ± 1.6	83.0 ± 1.7	88.2
Lightweight FL-IDS	90.9 ± 1.2	90.4 ± 1.3	88.7 ± 1.2	91.2 ± 1.0	85.5 ± 1.5	82.4 ± 1.8	88.2
Ours	92.3 ± 1.0	91.7 ± 1.1	88.3 ± 1.3	90.9 ± 1.2	86.0 ± 1.4	83.4 ± 1.7	88.8

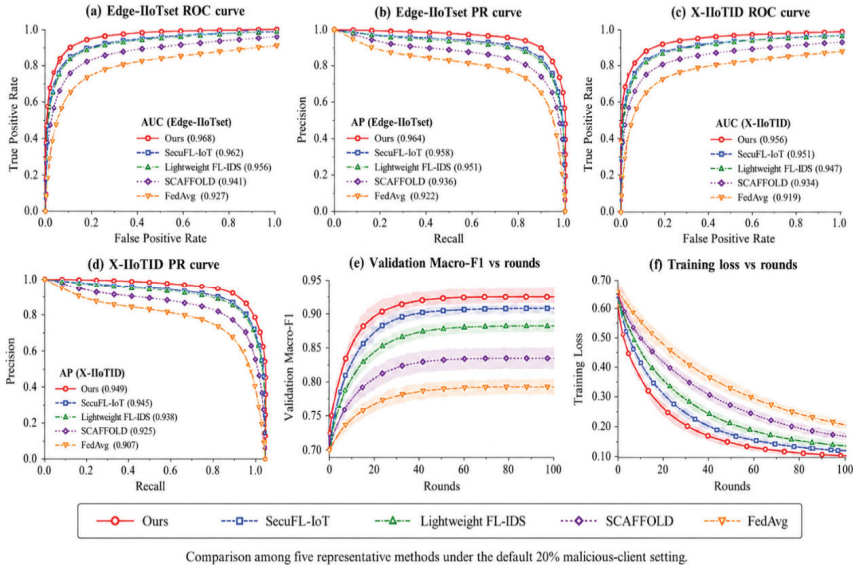


Figure 5 ROC/PR and convergence comparison under the default setting: (a) Edge-IIoTset ROC curve; (b) Edge-IIoTset PR curve; (c) X-IIoTID ROC curve; (d) X-IIoTID PR curve; (e) validation Macro-F1 versus rounds; (f) training loss versus rounds.

not be interpreted as consistently outperforming strong industrial baselines in conventional detection metrics; its clearer benefits are reflected in robustness, ASR control, convergence efficiency, and system-level trade-offs.

Table 6 reports the security performance on Edge-IIoTset. As shown in Table 7, the proposed method achieves the lowest FNR and ASR and the highest Robust-F1, showing better missed-detection control and attack suppression. However, SecuFL-IoT remains slightly better in Precision and FPR, indicating that the proposed method does not outperform all baselines on every metric. Krum and Trimmed-Mean are used only as plaintext-update robust references.

The cross-dataset results indicate that all methods suffer performance degradation under distribution shifts across industrial network environments. The proposed method remains relatively stable in most transfer settings, but it cannot eliminate cross-domain degradation. Since feature dimensions and semantics differ across datasets, these results are interpreted only after common feature mapping or similar feature-group alignment, and should not

Table 6 Core security metrics under 20% malicious clients

Method	Precision ↑	Recall ↑	FPR ↓	FNR ↓	Robust-F1 ↑	Drop-F1 ↓	ASR ↓
FedAvg	89.5 ± 1.4	88.8 ± 1.6	4.8 ± 0.8	11.2 ± 1.6	75.6 ± 2.2	8.9 ± 1.6	39.6 ± 3.8
FedProx	90.6 ± 1.2	89.0 ± 1.3	4.2 ± 0.7	11.0 ± 1.3	77.5 ± 2.0	8.1 ± 1.4	36.5 ± 3.5
SCAFFOLD	91.2 ± 1.1	89.8 ± 1.2	3.8 ± 0.6	10.2 ± 1.2	78.6 ± 1.9	7.8 ± 1.3	34.3 ± 3.2
FedAvg + SecAgg	91.4 ± 1.2	89.5 ± 1.4	3.5 ± 0.6	10.5 ± 1.4	78.2 ± 2.1	7.5 ± 1.5	33.5 ± 3.1
Krum	90.3 ± 1.6	88.2 ± 1.7	4.4 ± 0.9	11.8 ± 1.7	82.4 ± 1.8	3.2 ± 1.2	22.8 ± 2.9
Trimmed-Mean	90.6 ± 1.5	89.1 ± 1.6	4.1 ± 0.8	10.9 ± 1.6	81.4 ± 1.7	4.7 ± 1.1	24.4 ± 2.7
Privacy-Robust FL	91.9 ± 1.1	90.4 ± 1.3	3.4 ± 0.6	9.6 ± 1.3	83.7 ± 1.6	3.9 ± 1.0	19.5 ± 2.5
SecuFL-IoT	92.9 ± 1.0	90.7 ± 1.1	2.9 ± 0.5	9.3 ± 1.1	84.0 ± 1.5	4.1 ± 1.0	20.6 ± 2.4
Lightweight FL-IDS	92.1 ± 1.1	91.1 ± 1.2	3.5 ± 0.7	8.9 ± 1.2	83.1 ± 1.7	5.0 ± 1.1	24.2 ± 2.6
Ours	92.5 ± 0.9	92.1 ± 1.1	3.1 ± 0.6	7.9 ± 1.1	85.4 ± 1.4	3.4 ± 0.9	16.9 ± 2.3

Note: Results are obtained on Edge-IIoTset under the default federated setting with 20% simulated malicious clients.

Table 7 Cross-dataset generalization results

Training Set → Test Set	FedAvg	SCAFFOLD	SecuFL-IoT	Lightweight FL-IDS	Ours	Ours Drop ↓
X-IIoTID → Edge-IIoTset	77.8 ± 2.3	80.1 ± 2.0	82.6 ± 1.8	81.4 ± 1.9	83.4 ± 1.7	8.9
Edge-IIoTset → X-IIoTID	76.4 ± 2.5	79.2 ± 2.1	81.0 ± 1.9	80.9 ± 2.0	82.1 ± 1.8	9.6
X-IIoTID → WUSTL-IIoT	79.4 ± 2.1	82.2 ± 1.8	83.5 ± 1.7	84.4 ± 1.6	84.2 ± 1.5	6.7
TON_IoT → Edge-IIoTset	75.7 ± 2.6	78.4 ± 2.2	81.5 ± 2.0	80.2 ± 2.1	81.0 ± 2.0	11.3

Table 8 Efficiency and resource overhead comparison

Method	Params M ↓	FLOPs M ↓	Training Time s ↓	Rounds to Target F1 ↓	Inference ms ↓	Communication MB/Round ↓	Aggregation ms ↓	PLCS ↑
FedAvg	0.18	0.42	18.6	94	0.41	2.92	31	0.598
SCAFFOLD	0.18	0.42	22.5	73	0.44	5.78	52	0.561
FedAvg + SecAgg	0.18	0.42	24.8	80	0.42	3.24	84	0.594
Krum	0.18	0.42	31.0	87	0.44	2.92	136	0.579
Trimmed-Mean	0.18	0.42	28.9	82	0.42	2.92	116	0.591
Privacy-Robust FL	0.24	0.56	31.6	68	0.53	2.45	98	0.649
SecuFL-IoT	0.35	0.78	28.2	64	0.65	2.31	108	0.667
Lightweight FL-IDS	0.16	0.34	21.5	70	0.39	1.52	71	0.705
Ours	0.21	0.49	24.9	58	0.48	1.86	93	0.716

be regarded as strict proof of cross-domain generalization. The remaining performance drop indicates that risk-adaptive aggregation does not solve domain adaptation; differences in protocol semantics, device configurations, attack patterns, and feature spaces still cause substantial cross-domain degradation.

Table 8 shows that the proposed method is not the lowest-cost solution. Lightweight FL-IDS has lower parameter size, FLOPs, inference latency, and per-round communication volume. As further illustrated in Figure 6, the proposed method does not dominate every single efficiency metric, but it achieves a more favorable overall trade-off among efficiency, robustness, and privacy. Specifically, it requires fewer rounds to reach the target F1, obtains the highest PLCS, and maintains stronger Robust-F1 and lower ASR under moderate resource overhead. Its aggregation latency is higher than FedAvg, SCAFFOLD, and Lightweight FL-IDS, mainly because of sketch-based deviation estimation, robust weight computation, and the secure aggregation interface.

The proposed method achieves a better system-level balance among Robust-F1, ASR control, PLCS, communication cost, training rounds, and latency, although it is not the lowest-cost method in all individual efficiency metrics.

Table 9 summarizes the additional overhead of the main components, where N is the number of clients, d is the update dimension, m is the sketch dimension, K is the number of state variables, and B is the number of feedback items.

The main extra overhead comes from sketch generation and robust-center computation, while risk-policy generation, strategy correction, and trust updating remain lightweight.

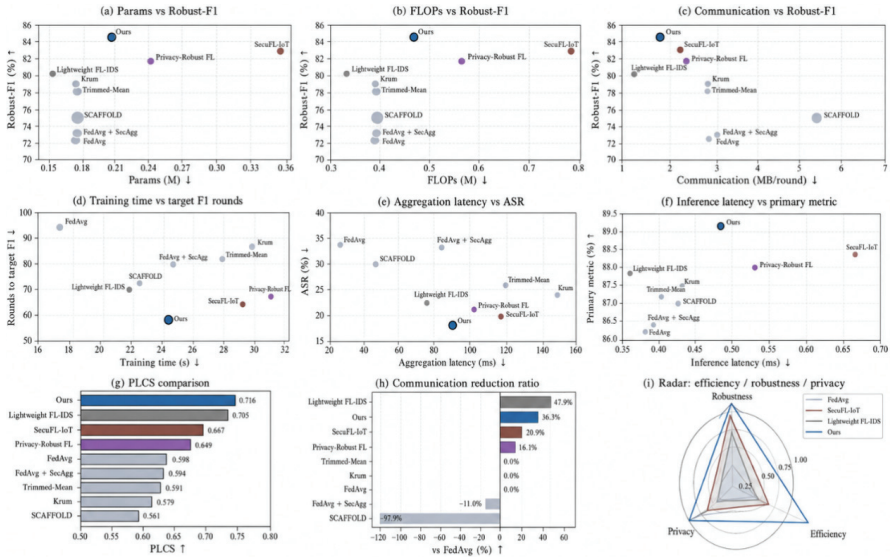


Figure 6 Efficiency-robustness-privacy trade-off analysis.

Table 9 Component-wise complexity and overhead

Component	Computation	Communication	Storage
RiskPolicy	$O(K)$	None	Client state
Sketch generation	$O(md)$	$O(m)$ sketch	Sketch vector
Robust center	$O(Nm)$–$O(Nm \log N)$	None	Client sketches
StrategyCorrection	$O(N)$	None	Policy states
Aggregation weighting	$O(N)$	None	Client weights
Trust update	$O(B)$	Feedback records	Feedback cache
Interface review	Queue-dependent	Feedback records	Review history

Overall, the main message of the experiments is not that the proposed method consistently dominates strong industrial baselines in conventional detection accuracy. Rather, under the specified federated and attack simulation settings, it provides a more balanced system-level behavior in ASR control, malicious-client robustness, convergence efficiency, and privacy–communication–latency trade-offs, while specialized lightweight methods remain preferable for some individual efficiency metrics. Because the federated clients, resource states, malicious behaviors, and feedback signals are protocol-generated from public datasets, these results should be interpreted as controlled simulation evidence rather than direct evidence of real industrial deployment effectiveness.

4.3 Statistical Significance Analysis

To assess whether the differences between the proposed method and the main baselines are statistically significant, this study conducts a two-sided Welch's t-test based on five independent runs. The Holm–Bonferroni method is used for multiple-comparison correction within each dataset, with the significance level set to $p_{\mathrm{Holm}} < 0.05$. Since only five runs are used, the statistical results serve as auxiliary evidence for performance stability rather than independent proof of superiority. Future work should include more repeated runs and effect-size measures, such as Cohen's d or Cliff's delta, to complement corrected p-values.

As summarized in Table 10, the proposed method shows stable advantages over traditional FL methods and plaintext robust references on Edge-IIoTset and X-IIoTID. However, most differences against strong baselines are not significant after Holm–Bonferroni correction. Therefore, the results do not support universal superiority across all datasets. The strongest statistical evidence is observed in ASR control, where the proposed method significantly outperforms Lightweight FL-IDS with $p_{\mathrm{Holm}} = 0.002$.

Table 10 Summary of statistical significance results

Dataset / Setting	Main Result	Conclusion
Edge-IIoTset	Significantly outperforms SCAFFOLD, Krum, and Trimmed-Mean; not significant against strong baselines	Stable advantage over traditional FL and plaintext robust references, but limited advantage over strong baselines
X-IIoTID	Significantly outperforms SCAFFOLD, Krum, and Trimmed-Mean; close to significance against Privacy-Robust FL	Shows a relatively stable advantage on the primary dataset
TON_IoT	Significant only against Krum; lower than Lightweight FL-IDS without significance	No clear advantage in conventional detection
WUSTL-IIoT	Significant only against Krum; lower than Lightweight FL-IDS without significance	Cannot claim superiority over the lightweight baseline
SWaT	Significant only against SCAFFOLD	Shows an advantage trend in time-series anomaly detection
HAI	No major comparison reaches significance	Evidence is insufficient for a strong conclusion
Robust-F1	Higher mean than strong baselines, but not significant after correction	Robustness improvement trend exists, but evidence is limited
ASR	Significantly lower than Lightweight FL-IDS after Holm–Bonferroni correction, $p_{\mathrm{Holm}} = 0.002 < 0.05$	ASR control is one of the more stable advantages.

4.4 Representation and Feature Space Analysis

To examine representation behavior, this study visualizes feature-space distribution, anomaly-score distribution, and state-variable relationships. These results are used only as auxiliary interpretation, not as causal evidence or independent proof of module effectiveness.

Figure 7(a) compares the t-SNE representation spaces of FedAvg, SecuFL-IoT, Ours-w/o SketchRobust, and Ours on the X-IIoTID multi-class test set with balanced class sampling and a fixed seed. The proposed method shows clearer local separation between Normal samples and typical attacks such as DoS/DDoS and Injection, while overlaps remain among Malware, MITM, and Reconnaissance samples. Figure 7(b) presents the test-set anomaly-score distribution using Histogram/KDE, with the threshold selected only on the validation set. Normal samples are mainly concentrated in low-score regions, whereas typical attacks tend to appear in high-score regions. However, overlapping samples still exist near the threshold, which is consistent with the limited performance advantage on some datasets. Figure 7(c) shows a correlation heatmap based on client-round statistics from training and validation stages. It includes risk scores, resource status, communication latency, uncertainty, feedback utility, and trustworthiness. Since some variables are generated by the simulation protocol or endogenously linked by the proposed mechanism, these correlations should be interpreted only as controlled empirical observations. Overall, Figure 7 suggests

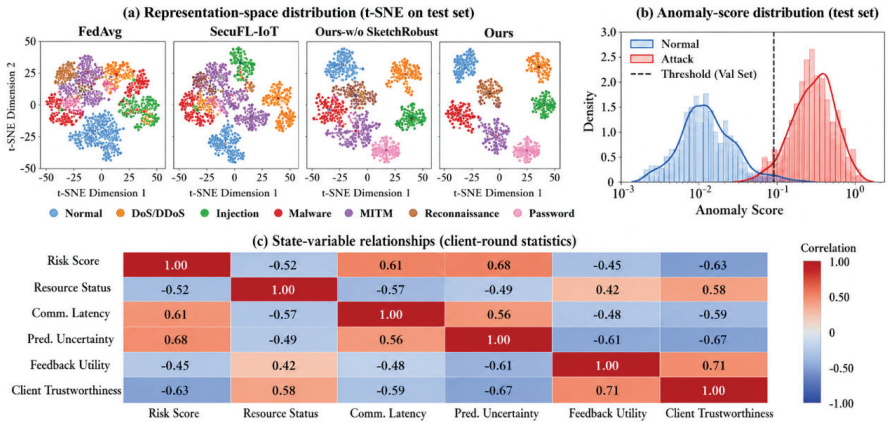


Figure 7 Representation and feature-space visualization:(a) t-SNE representation distribution on X-IIoTID; (b) anomaly-score distribution with validation-selected threshold; (c) client-round state-variable correlation heatmap.

improved local representation separation and typical-attack discrimination, but boundary anomalies, stealthy attacks, and cross-domain samples still remain challenging.

4.5 Feature Attribution and Stability Analysis

To analyze feature dependence, this study uses Integrated Gradients and feature permutation perturbation instead of attention weights, because the backbone models are mainly MLP and LSTM without explicit attention layers. To avoid label leakage and simulated-variable interference, attribution is performed only on real detection-model inputs, while variables such as resource status, communication latency, feedback reliability, and trustworthiness are excluded. Feature attribution is used only as auxiliary diagnostic evidence to inspect feature dependence, and it does not provide a complete explanation guarantee or prove causal interpretability.

Input features are grouped into Flow, Protocol, Temporal, Sensor/Actuator, and Log/Event. Integrated Gradients scores are computed on fixed validation samples and normalized to $[0, 1]$. Feature permutation perturbation further measures the change in F1-Score or anomaly score after disturbing each feature group.

Figure 8 shows that Flow and Protocol dominate traffic-based datasets, Temporal and Sensor/Actuator dominate SWaT and HAI, and permutation perturbation confirms similar sensitivity. Feature-importance stability across five runs is highest for the proposed method, suggesting a more consistent feature-dependence pattern. Results are auxiliary interpretability evidence only.

4.6 Error and Failure Case Analysis

Although the proposed method shows relatively stable performance in controlled federated simulation settings, several failure cases remain. Figure 9 summarizes typical errors, including gradual anomalies, boundary-sample confusion, missing key variables, cross-domain misclassification, and feedback-noise interference. These cases are analyzed only to clarify applicability boundaries, not as independent evidence of model effectiveness.

Figure 9(a) shows a SWaT gradual-anomaly case, where the anomaly score increases slowly and crosses the validation threshold only after several windows, causing delayed detection. Figure 9(b) indicates that weak attacks and normal fluctuations may overlap near the threshold, leading to

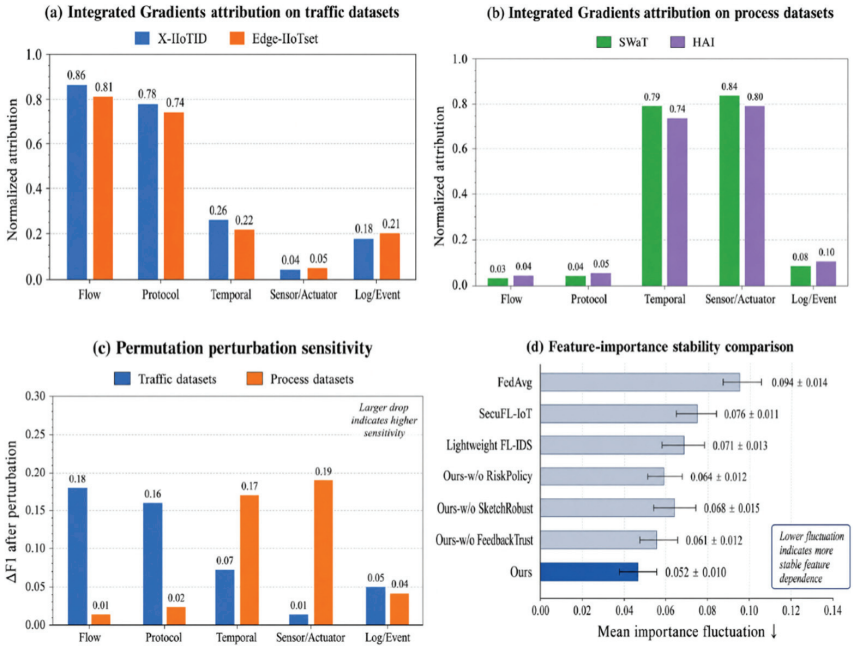


Figure 8 Feature attribution and stability analysis:(a) Integrated Gradients attribution on traffic datasets; (b) Integrated Gradients attribution on process datasets; (c) permutation perturbation sensitivity of different feature groups; (d) feature-importance stability comparison across different methods.

low-confidence predictions and possible false positives or false negatives. Figure 9(c) shows that target-domain performance still decreases under protocol, device, or attack-pattern shifts, even after feature alignment. Figure 9(d) shows that noisy or biased feedback can disturb trustworthiness updating, while audited feedback produces more stable trust estimates.

To further connect these failure patterns with practical industrial review scenarios, Figure 10 presents a case-based comparison between a strong baseline and the proposed method. The comparison does not serve as additional quantitative proof, but illustrates how anomaly scores, uncertainty estimation, risk indication, trust-aware review triggering, and interface-assisted feedback can support decision interpretation in representative IIoT/ICS cases.

Figures 9 and 10 show that the proposed method improves anomaly interpretation and review support across network intrusion, process anomalies, and high-uncertainty cases, but gradual anomalies, boundary samples, cross-domain shifts, and feedback noise remain challenging.

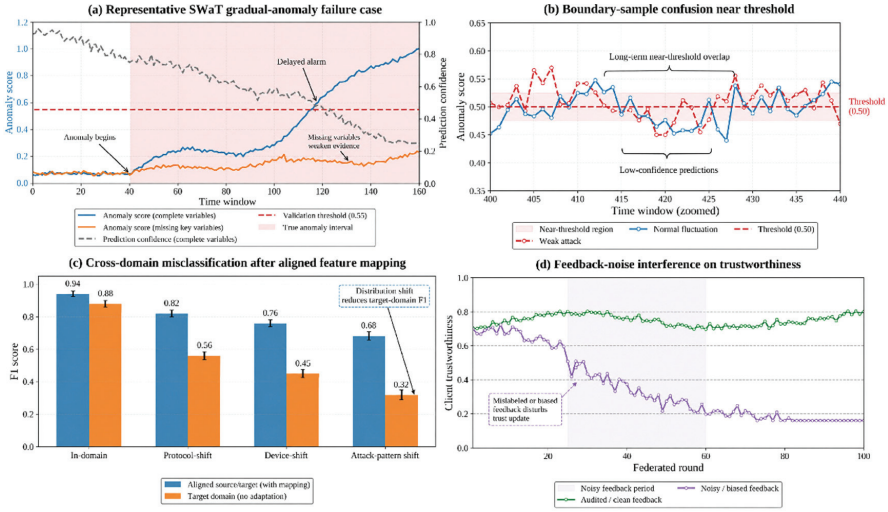


Figure 9 Error and failure case analysis:(a) gradual anomaly and delayed alarm; (b) boundary-sample confusion near the threshold; (c) cross-domain misclassification after feature alignment; (d) feedback-noise interference on trustworthiness updating.

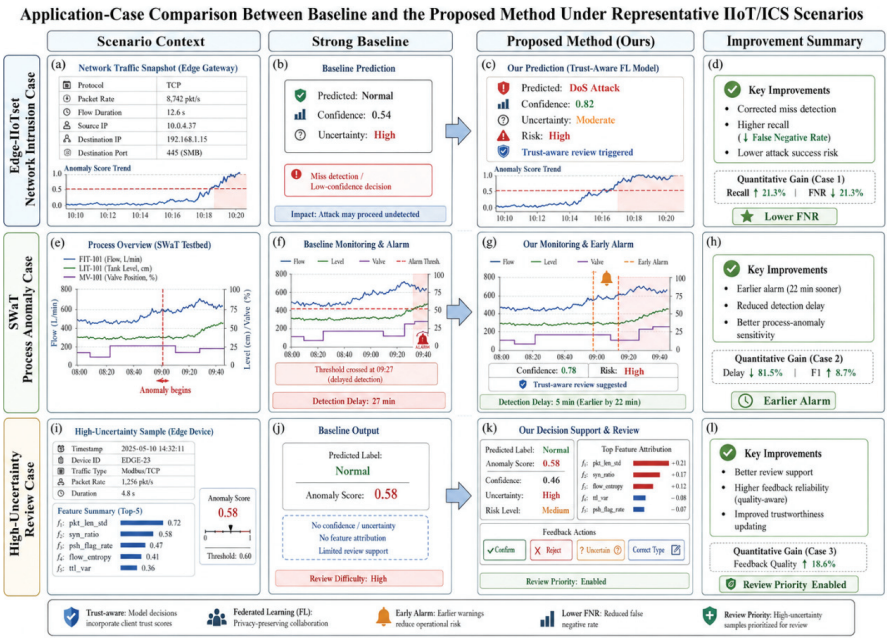


Figure 10 Application-case comparison between baseline and the proposed method under representative IIoT/ICS scenarios.

4.7 Ablation Study

To evaluate the contribution of different modules, this study conducts ablation experiments on detection performance, attack robustness, communication efficiency, and calibration reliability. The core results are summarized in Table 11. The full model is used as the reference, and Holm–Bonferroni correction is applied for significance testing. Here, n.s. indicates no statistical significance after correction, while * indicates $p_{\mathrm{Holm}} < 0.05$.

Table 11 shows that different modules affect different objectives. RiskPolicy mainly improves communication efficiency and privacy-latency-communication trade-offs. SketchRobust contributes more to attack robustness, while Plain SecAgg shows that secure aggregation alone is insufficient to suppress abnormal updates. Compared with Plain SecAgg, the full model mainly improves Robust-F1 and ASR, indicating that the practical benefit of sketches lies in robustness under privacy-constrained visibility rather than stronger encryption security. StrategyCorrection reduces deviation misjudgment caused by privacy noise, sparsification, and quantization. FeedbackTrust mainly affects calibration and trust updating rather than conventional detection accuracy. Although most single-module removals are not statistically significant after correction, the feedback-related variants suggest that

Table 11 Core ablation results on edge-IIoTset under the default 20% malicious-client setting

Model Variant	Macro-F1 ↑	Robust-F1 ↑	ASR ↓	PLCS ↑	ECE ↓	Comm. MB/ Round ↓	Avg. ↑	Drop	Sig.
Full model	92.4 ± 0.9	85.6 ± 1.3	16.7 ± 2.1	0.718 ± 0.020	0.042 ± 0.005	1.83	89.7	–	–
w/o Risk Policy	91.3 ± 0.8	83.8 ± 1.5	20.4 ± 2.7	0.661 ± 0.026	0.048 ± 0.006	2.42	88.9	−0.8	n.s.
Resource Only	91.5 ± 0.7	84.1 ± 1.4	19.9 ± 2.6	0.674 ± 0.024	0.047 ± 0.006	1.95	89.1	−0.6	n.s.
Privacy Only	91.7 ± 0.7	84.4 ± 1.4	19.2 ± 2.5	0.682 ± 0.023	0.046 ± 0.006	1.90	89.2	−0.5	n.s.
Compression Only	91.4 ± 0.8	83.9 ± 1.6	20.8 ± 2.8	0.656 ± 0.027	0.049 ± 0.007	1.88	89.0	−0.7	n.s.
w/o Sketch Robust	91.0 ± 0.8	80.9 ± 1.8	31.4 ± 3.4	0.679 ± 0.027	0.050 ± 0.007	1.80	88.5	−1.2	n.s.
w/o Strategy Correction	91.6 ± 0.6	83.2 ± 1.7	23.7 ± 2.8	0.690 ± 0.025	0.046 ± 0.006	1.82	89.1	−0.6	n.s.
Plain SecAgg	90.7 ± 0.7	79.8 ± 2.0	33.1 ± 3.7	0.603 ± 0.032	0.055 ± 0.008	2.15	88.1	−1.6	n.s.
w/o Feedback Trust	91.6 ± 0.6	84.2 ± 1.6	21.9 ± 2.7	0.703 ± 0.023	0.055 ± 0.008	1.84	89.2	−0.5	n.s.
Direct Feedback	91.8 ± 0.6	83.8 ± 1.8	22.5 ± 2.9	0.706 ± 0.022	0.050 ± 0.007	1.84	89.2	−0.5	n.s.
Weighted Feedback Only	91.9 ± 0.6	84.4 ± 1.6	20.8 ± 2.5	0.710 ± 0.021	0.046 ± 0.006	1.84	89.3	−0.4	n.s.
NoTrust Update	91.5 ± 0.6	84.0 ± 1.7	22.1 ± 2.8	0.704 ± 0.024	0.052 ± 0.007	1.84	89.1	−0.6	n.s.
w/o Risk + Sketch	89.5 ± 0.8	78.6 ± 2.1	39.2 ± 3.9	0.579 ± 0.037	0.064 ± 0.010	2.46	86.6	−3.1	*
w/o Sketch + Feedback	89.3 ± 0.8	78.1 ± 2.2	40.5 ± 4.1	0.586 ± 0.036	0.070 ± 0.011	1.86	86.4	−3.3	*

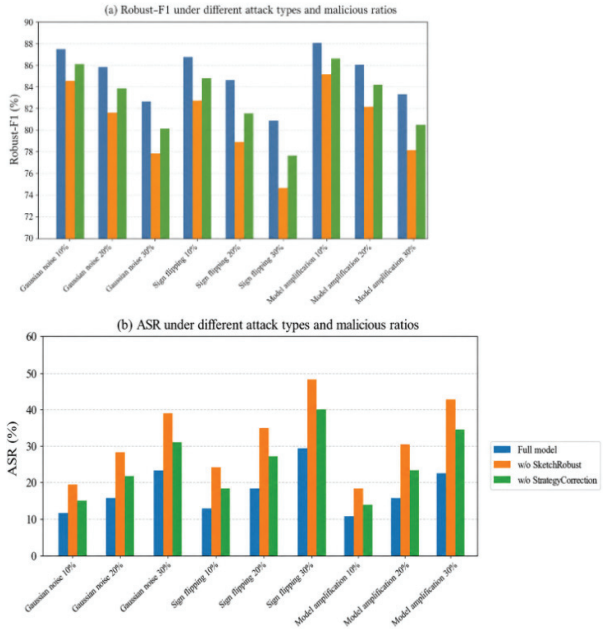


Figure 11 Robustness ablation under different attack types and malicious-client ratios. (a) Robust-F1 under different attack types and malicious ratios. (b) ASR under different attack types and malicious ratios.

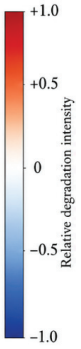
reliability weighting improves calibration and feedback-noise control, while trust updating affects next-round risk scoring and aggregation weights.

Figure 11 further analyzes robustness under different attack types and malicious-client ratios. As the malicious-client ratio increases, Robust-F1 generally decreases, while ASR increases. The full model maintains relatively stable performance across attack settings. Removing SketchRobust causes clearer robustness degradation, especially under sign-flipping and model-amplification attacks. The w/o StrategyCorrection variant usually performs between the full model and w/o SketchRobust, suggesting that correction-based deviation estimation provides auxiliary robustness but cannot replace robust aggregation.

Figure 12 summarizes module-removal effects relative to the full model. Single-module removals usually lead to moderate and metric-specific degradation, whereas combined removals cause broader performance drops across multiple metrics. Overall, the results suggest that the full framework benefits from complementary mechanisms. Therefore, the ablation results should be

Values show change relative to the Full model. Negative values indicate lower metric values; for ASR, ECE, and Comm., positive changes indicate degradation.

	ΔMacro-F1	ΔRobust-F1	ΔASR	ΔPLCS	ΔECE	ΔComm.	ΔAvg.
w/o RiskPolicy	−1.1	−1.8	+3.7	−0.057	+0.006	+0.59	−0.8
ResourceOnly	−0.9	−1.5	+3.2	−0.044	+0.005	+0.12	−0.6
PrivacyOnly	−0.7	−1.2	+2.5	−0.036	+0.004	+0.07	−0.5
CompressionOnly	−1.0	−1.7	+4.1	−0.062	+0.007	+0.05	−0.7
w/o SketchRobust	−1.4	−4.7	+14.7	−0.039	+0.008	−0.03	−1.2
w/o StrategyCorrection	−0.8	−2.4	+7.0	−0.028	+0.004	−0.01	−0.6
Plain SecAgg	−1.7	−5.8	+16.4	−0.115	+0.013	+0.32	−1.6
w/o FeedbackTrust	−0.8	−1.4	+5.2	−0.015	+0.013	+0.01	−0.5
DirectFeedback	−0.6	−1.8	+5.8	−0.012	+0.008	+0.01	−0.5
WeightedFeedbackOnly	−0.5	−1.2	+4.1	−0.008	+0.004	+0.01	−0.4
NoTrustUpdate	−0.9	−1.6	+5.4	−0.014	+0.010	+0.01	−0.6
w/o Risk + Sketch	−2.9	−7.0	+22.5	−0.139	+0.022	+0.63	−3.1
w/o Sketch + Feedback	−3.1	−7.5	+23.8	−0.132	+0.028	+0.03	−3.3



For Macro-F1, Robust-F1, PLCS, and Avg., lower values indicate degradation.
For ASR, ECE, and Comm., higher values indicate degradation.

Figure 12 Module-removal effect heatmap. The heatmap shows metric changes relative to the full model. Darker cells indicate stronger effects, and combined removals cause broader degradation than single-module removals.

interpreted as evidence that combined modules improve overall stability, rather than as proof that every individual component independently brings statistically significant improvement.

4.8 Sensitivity Analysis of Risk-Policy Hyperparameters

To examine whether the proposed framework is overly dependent on manually selected risk-policy parameters, we provide a trend-level one-factor-at-a-time sensitivity summary around the default setting. The analysis focuses on risk weights, sketch dimension, trust smoothing, feedback ratio, review threshold, sparsity ratio, and quantization bit-width. Since the purpose is to identify parameter sensitivity rather than to introduce new baselines, the results are reported as trend-level effects on robustness, communication cost, feedback quality, and interface workload.

Table 12 summarizes the observed sensitivity trends. The risk weights mainly affect PLCS and communication cost, while the sketch dimension has a clearer influence on ASR and aggregation latency. The trust smoothing coefficient mainly affects Trust-AUC and ECE. Feedback ratio and uncertainty threshold influence feedback reliability, high-risk coverage, and manual review burden. Compression-related parameters directly affect communication volume, with moderate influence on F1 when compression becomes aggressive. Overall, the framework is most sensitive to sketch

Table 12 Trend-level sensitivity summary of risk-policy hyperparameters

Parameter	Meaning	Main Affected Metrics	Observed Trend
$\lambda_p, \lambda_r, \lambda_t$	Weights of privacy, resource, and trust risks	PLCS, communication, Robust-F1	Moderate changes mainly affect system trade-offs; detection performance remains relatively stable.
M	Sketch dimension	ASR, Robust-F1, aggregation latency	Larger m improves deviation estimation and ASR control but increases aggregation latency.
β	Trust smoothing coefficient	Trust-AUC, ECE	Too small β causes unstable trust updates; too large β slows response to abnormal clients.
Feedback ratio	Proportion of reviewed samples	Feedback reliability, MRB, Trust-AUC	Higher feedback ratio improves reliability and trust estimation but increases review burden.
Uncertainty threshold	Review trigger threshold	HRC, MRB, FD	Lower threshold improves high-risk coverage but increases manual review burden.
Sparsity ratio	Sparse uploading level	Communication, F1, PLCS	Higher sparsity reduces communication cost but may reduce F1 under aggressive compression.
Bit-width	Quantization precision	Communication, F1, latency	Lower bit-width reduces communication but may increase quantization error.

dimension, trust smoothing, and review-trigger settings, while moderate changes in risk-weight combinations produce relatively stable detection performance.

Table 12 indicates that the framework is not equally sensitive to all hyperparameters. Parameters directly affecting deviation estimation and feedback dynamics, such as sketch dimension, trust smoothing, and review-trigger threshold, have clearer effects, whereas moderate changes in risk-weight settings mainly change the privacy–communication–latency balance.

4.9 Simulation-Based Flexible Interaction Prototype Evaluation

Under the simulated expert-review protocol, four interface settings are compared. Their differences are summarized in Table 13.

FD, FR, MRB, HRC, FEG, Trust-AUC, and IES are generated by the simulated review protocol and describe feedback delay, feedback reliability, manual review burden, high-risk coverage, feedback evidence gain,

Table 13 Interface settings in the simulated review protocol

Interface	Displayed Information	Feedback Options	Reliability Modeling
Baseline	Label + anomaly score	Confirm / reject	No
Confidence-only	Label + score + confidence	Confirm / reject	Weak
Explanation-assisted	Confidence + attribution	Confirm / reject / correct	Partial
Proposed Flexible	Confidence + uncertainty + risk + attribution + trustworthiness	Multi-type feedback	Yes

Table 14 Interface evaluation metrics

Interface	FD ↓	FR ↑	MRB ↓	HRC ↑	FEG ↑	Trust-AUC ↑	IES ↑
Baseline	4.8 ± 0.6	72.4 ± 2.8	18.6 ± 1.9	61.3 ± 3.1	0.32 ± 0.08	0.681 ± 0.026	0.542 ± 0.031
Confidence-only	3.9 ± 0.5	77.8 ± 2.5	16.2 ± 1.7	67.9 ± 2.8	0.51 ± 0.10	0.714 ± 0.024	0.608 ± 0.028
Explanation-assisted	3.1 ± 0.4	82.6 ± 2.2	13.7 ± 1.5	74.5 ± 2.6	0.73 ± 0.12	0.756 ± 0.022	0.687 ± 0.026
Proposed Flexible	2.4 ± 0.3	87.9 ± 1.9	10.8 ± 1.3	82.1 ± 2.4	0.96 ± 0.15	0.813 ± 0.020	0.774 ± 0.024

Note: FD is measured in federated rounds. FR, MRB, HRC, and FEG are reported in %. IES is normalized to $[0, 1]$.

trust-discrimination ability, and normalized interface efficiency, respectively. These metrics are not real usability or workload measurements.

Table 14 shows that the proposed interface reduces feedback delay and review burden while improving feedback reliability, high-risk coverage, Trust-AUC, and interface efficiency.

As shown in Figure 13, the interface includes device lists, anomaly timelines, alarm cards, confidence, uncertainty, feature attribution, feedback actions, trustworthiness trends, and operation logs.

As shown in Figure 14, compared with the baseline interface, the proposed interface adds uncertainty, risk level, review priority, multi-type feedback entry, and feedback reliability recording.

These results reflect only the simulated feedback protocol and should not be interpreted as evidence of real operator workload reduction, usability improvement, or human–machine decision-support effectiveness.

5 Discussion

The proposed framework achieves relatively stable results in most federated simulation settings. This may be because it integrates risk-adaptive policy generation, sketch-assisted robust aggregation, flexible interaction feedback, and closed-loop trustworthiness updating into the same training process.



Figure 13 Flexible interaction interface prototype.

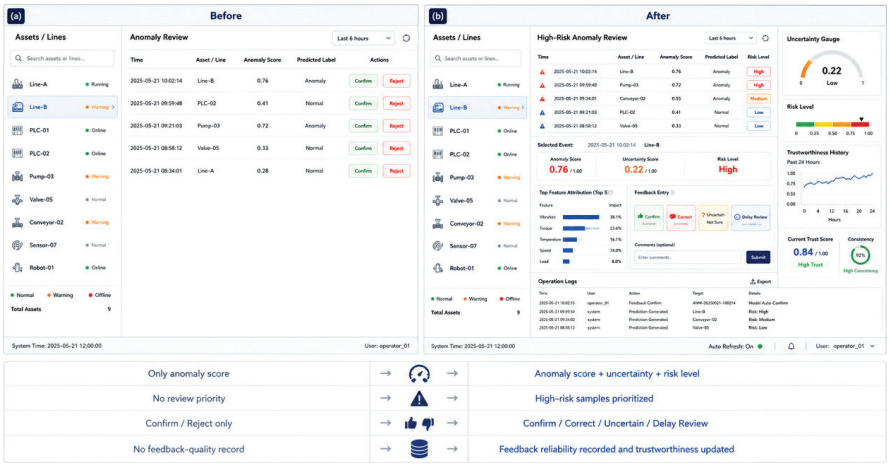


Figure 14 Before–after interface comparison.

The risk-adaptive policy allows clients with different privacy, resource, uncertainty, and trust states to adopt differentiated uploading configurations, which is consistent with adaptive federated coordination in heterogeneous edge environments [33,34]. Sketch-assisted aggregation estimates update deviations through low-dimensional noisy sketches and adjusts aggregation weights without directly accessing complete plaintext updates. The flexible interaction interface presents anomaly confidence, uncertainty, feature

attribution, risk level, and feedback entry, allowing simulated review signals to be converted into reliability-weighted weak supervision.

The results show that the advantage of the proposed method is not mainly reflected in universal superiority in conventional detection performance. Compared with strong industrial federated and lightweight baselines, its improvement in primary detection metrics is limited on some datasets, which is consistent with studies showing that personalization, heterogeneity handling, and task-specific model design can strongly affect federated intrusion detection performance [35]. Its clearer contribution lies in malicious-client robustness, ASR control, aggregation stability, convergence efficiency, and privacy–communication–latency trade-offs. The interface evaluation further suggests that richer review information can reduce feedback delay and review burden while improving feedback reliability and Trust-AUC under simulated expert-review settings.

The ablation results indicate that different modules serve different roles. RiskPolicy mainly improves communication coordination and system trade-offs; SketchRobust contributes more to robustness under attacks; StrategyCorrection reduces deviation-estimation interference caused by privacy perturbation and compression errors; and FeedbackTrust with the flexible interface mainly affects calibration, feedback reliability, and trustworthiness updating. Since several single-module ablations are not statistically significant, the results should be interpreted as empirical stability from combined mechanisms rather than decisive superiority of each component.

Beyond model robustness, the feedback-assisted trust mechanism can also be interpreted as a lightweight governance layer for AI-assisted industrial decision-making. By exposing uncertainty, feature attribution, risk level, and feedback reliability, the interface supports traceability, human oversight, and responsibility-aware review. This interpretation is consistent with recent discussions on AI-assisted decision-making governance, which emphasize reasoning transparency, human control, legitimacy, and accountability. However, this governance role remains conceptual in this study because no real operator study or organizational deployment validation is conducted [36].

This study has limitations. The experiments are based on public IIoT/ICS datasets with simulated client partitioning, resource perturbation, malicious behavior, expert feedback, and interface interaction. The practical effectiveness of the adaptive policy and trust-updating mechanism remains to be validated in real cross-factory IIoT environments, where device load, network latency, operator routines, maintenance policies, and organizational constraints may differ from the simulation protocol. The trustworthiness

score is a protocol-level empirical reliability indicator, not a validated form of social, organizational, operator, or cryptographic trust. Therefore, the results should be interpreted as evidence for a simulation-validated coordination mechanism rather than a deployment-ready industrial system. The proposed framework is not plug-and-play; resource-constrained deployment requires consideration of hardware limits, latency, operator training, feedback logging, maintenance cost, and additional sketch, secure-uploading, trust-updating, and feedback-review overhead [37]. Small- and medium-sized facilities may require simplified deployment with fixed security levels, lightweight sketches, cloud-managed aggregation, and optional feedback review. The interface evaluation is simulation-based and still requires validation with real industrial operators. Real feedback reliability may be affected by operator fatigue, expertise differences, alarm pressure, shift schedules, and organizational procedures, which are not captured by the simulated expert protocol. In addition, the secure uploading interface and sketches do not constitute a complete cryptographic security proof, and FL-based IoT intrusion detection remains vulnerable to poisoning without explicit defense or verification mechanisms [38, 39]. The sketch-assisted mechanism should be interpreted within a semi-honest and non-collusive simulation setting and does not cover malicious servers, client collusion, sketch-update inconsistency, or systematic protocol disruption. Future work will focus on real industrial federated deployment, genuine operator-feedback collection, verifiable secure aggregation, sketch-consistency verification, stronger adaptive attack models, domain adaptation with feature alignment and cross-factory calibration, and long-term validation under operating-condition drift.

6 Conclusion

This paper proposes a risk-adaptive secure federated learning framework for feedback-assisted trusted perception and flexible interaction optimization in IIoT anomaly detection. The framework integrates risk-adaptive policy generation, sketch-assisted robust aggregation, interface-assisted feedback collection, and closed-loop trustworthiness updating to coordinate privacy protection, communication efficiency, attack robustness, and feedback utilization.

Experiments on multiple public IIoT/ICS datasets show that the proposed method achieves stable average performance. However, its improvements over strong industrial baselines in conventional detection metrics are modest and not always statistically significant after correction. Its main advantages

are reflected in malicious-client robustness, ASR control, convergence efficiency, and privacy–communication–latency trade-offs, rather than universal superiority in conventional detection accuracy. Ablation and interface evaluation further indicate that different modules contribute to efficiency, robustness, calibration, feedback reliability, and trustworthiness updating.

The main contribution of this work is a coordinated framework for studying privacy-aware, risk-adaptive, and feedback-assisted federated IIoT anomaly detection under controlled simulation settings. Future work will further validate the framework in real cross-factory federated environments with genuine operator feedback, verifiable security mechanisms, stronger attack models, domain adaptation with feature alignment and cross-factory calibration, and long-term operating-condition drift. Scalability under larger client populations, longer operating periods, and higher feedback-review loads will also be examined.

References

[1] Rodríguez M, Tobón D P, and Múnera D, (2023) 'Anomaly classification in industrial internet of things: a review,' Intelligent Systems with Applications 18, 200232.

[2] He N, Zhang Z, Wang X, and Gao T, (2023) 'Efficient privacy-preserving federated deep learning for network intrusion of industrial IoT,' *International Journal of Intelligent Systems* 2023(1), 2956990.

[3] Yang R, He H, Wang Y, Qu Y, and Zhang W, (2023) 'Dependable federated learning for IoT intrusion detection against poisoning attacks,' *Computers & Security* 132, 103381.

[4] Hidayat M A, Nakamura Y, and Arakawa Y, (2023) 'Privacy-preserving federated learning with resource-adaptive compression for edge devices,' *IEEE Internet of Things Journal* 11(8), 13180–13198.

[5] Hu G, Li H, Fan W, and Zhang Y, (2023) 'Efficient byzantine-robust and privacy-preserving federated learning on compressive domain,' *IEEE Internet of Things Journal* 11(4), 7116–7127.

[6] Sánchez P M S, Celdrán A H, Xie N, Bovet G, Pérez G M, and Stiller B, (2024) 'Federatedtrust: A solution for trustworthy federated learning,' *Future Generation Computer Systems* 152, 83–98.

[7] Bajic B, Rikalovic A, Suzic N, and Piuri V, (2024) 'Toward a human-cyber-physical system for real-time anomaly detection,' *IEEE Systems Journal* 18(2), 1308–1319.

[8] Ni C and Li S C, (2024) 'Machine learning enabled industrial IoT security: Challenges, trends and solutions,' *Journal of Industrial Information Integration* 38, 100549.

[9] Lu Y, Chai S, Suo Y, Yao F, and Zhang C, (2024) 'Intrusion detection for Industrial Internet of Things based on deep learning,' *Neurocomputing* 564, 126886.

[10] Nandanwar H and Katarya R, (2024) 'Deep learning enabled intrusion detection system for Industrial IoT environment,' *Expert Systems with Applications* 249, 123808.

[11] Seo J K, Lee J, Kim B, Shim W, and Seo J T, (2025) 'AI-based anomaly detection in industrial control and cyber–physical systems: A data-type-oriented systematic review,' *Electronics* 15(1), 20.

[12] Li Z, Yan Y, Wang X, Ge Y, and Meng L, (2025) 'A survey of deep learning for industrial visual anomaly detection,' *Artificial Intelligence Review* 58(9), 279.

[13] Wang X, Wang Y, Javaheri Z, Almutairi L, Moghadamnejad N, and Younes O S, (2023) 'Federated deep learning for anomaly detection in the internet of things,' *Computers and Electrical Engineering* 108, 108651.

[14] Chaurasia N, Ram M, Verma P, Mehta N, and Bharot N, (2024) 'A federated learning approach to network intrusion detection using residual networks in industrial IoT networks,' *The Journal of Supercomputing* 80(13), 18325–18346.

[15] Kaur A, (2024) 'Intrusion detection approach for industrial internet of things traffic using deep recurrent reinforcement learning assisted federated learning,' *IEEE Transactions on Artificial Intelligence* 6(1), 37–50.

[16] Lee S J and Lee I G, (2025) 'Lightweight federated learning-based intrusion detection system for industrial internet of things,' ICT Express 11(4), 690–695.

[17] Li H, Ge L, and Tian L, (2024) 'Survey: Federated learning data security and privacy-preserving in edge-Internet of Things,' *Artificial Intelligence Review* 57(5), 130.

[18] Yazdinejad A, Dehghantanha A, Karimipour H, Srivastava G, and Parizi R M, (2024) 'A robust privacy-preserving federated learning model against model poisoning attacks,' *IEEE Transactions on Information Forensics and Security* 19, 6693–6708.

[19] Bakirci M, (2023) 'A novel swarm unmanned aerial vehicle system: Incorporating autonomous flight, real-time object detection, and coordinated intelligence for enhanced performance,' *Traitement du Signal* 40(5), 2063–2078.

[20] Ni L, Gong X, Li J, Tang Y, Luan Z, and Zhang J, (2024) 'rfedfw: Secure and trustable aggregation scheme for byzantine-robust federated learning in internet of things,' *Information Sciences* 653, 119784.

[21] Mbonu W E, Maple C, Epiphaniou G, and Panchev C, (2025) 'A verifiable, privacy-preserving, and poisoning attack-resilient federated learning framework,' *Big Data and Cognitive Computing* 9(4), 85.

[22] Yazdinejad A, Dehghantanha A, Srivastava G, Karimipour H, and Parizi R M, (2024) 'Hybrid privacy preserving federated learning against irregular users in next-generation Internet of Things,' *Journal of Systems Architecture* 148, 103088.

[23] Siriwardhana Y, Porambage P, Liyanage M, Marchal S, and Ylianttila M, (2024) 'Robust aggregation technique against poisoning attacks in multi-stage federated learning applications,' in *2024 IEEE 21st Consumer Communications & Networking Conference (CCNC)*, pp. 956–962, Las Vegas, NV, USA, Jan. 2024.

[24] Liu J, Li X, Liu X, Zhang H, Miao Y, and Deng R H, (2024) 'Defendfl: A privacy-preserving federated learning scheme against poisoning attacks,' *IEEE Transactions on Neural Networks and Learning Systems* 36(5), 9098–9111.

[25] Rashid M M, Khan S U, Eusufzai F, Redwan M A, Sabuj S R, and Elsharief M, (2023) 'A federated learning-based approach for improving intrusion detection in industrial internet of things networks,' *Network* 3(1), 158–179.

[26] Khraisat A, Alazab A, Alazab M, Obeidat A, Singh S, and Jan T, (2025) 'Federated learning for intrusion detection in IoT environments: A privacy-preserving strategy,' *Discover Internet of Things* 5(1), 72.

[27] Sun S, Zhou L, Wang Z, and Han L, (2025) 'Robust intrusion detection based on personalized federated learning for IoT environment,' *Computers & Security* 154, 104442.

[28] Anwer R W, Abrar M, Ullah M, Salam A, and Ullah F, (2025) 'Advanced intrusion detection in the industrial internet of things using federated learning and LSTM models,' *Ad Hoc Networks*, 103991.

[29] Jakubowski J, Bobek S, and Nalepa G, (2023) 'Human-in-the-loop anomaly detection in industrial data streams,' *arXiv preprint arXiv:2312.12345*.

[30] Mathuros K, Venugopalan S, and Adepu S, (2024) 'Waxai: Explainable anomaly detection in industrial control systems and water systems,' in *Proceedings of the 10th ACM Cyber-Physical System Security Workshop*, pp. 3–15.

[31] Kim D, Sim Y S, Yoon S, Yoon S, Yoa S, Lee S, and Lim W, (2026) 'AEGIS: Toward expert-in-the-loop industrial anomaly detection,' in *Proceedings of the AAAI Conference on Artificial Intelligence* 40(48), 41619–41621.

[32] Deng Z, Xuan X, Ma K L, and Kong Z, (2026) 'A reliable framework for human-in-the-loop anomaly detection in time series,' *ACM Transactions on Interactive Intelligent Systems* 16(2), 1–23.

[33] Mughal F R, He J, Das B, Dharejo F A, Zhu N, Khan S B, and Alzahrani S, (2024) 'Adaptive federated learning for resource-constrained IoT devices through edge intelligence and multi-edge clustering,' *Scientific Reports* 14(1), 28746.

[34] Li X, Gao Y, Deng Y, and Jiang X, (2025) 'Federated learning with adaptive aggregation weights for non-IID data in edge networks,' *IEEE Transactions on Cognitive Communications and Networking* 11(5), 3425–3439.

[35] Albanbay N, Tursynbek Y, Graffi K, Uskenbayeva R, Kalpeyeva Z, Abilkaiyr Z, and Ayapov Y, (2025) 'Federated learning-based intrusion detection in IoT networks: Performance evaluation and data scaling study,' *Journal of Sensor and Actuator Networks* 14(4), 78.

[36] Susilo E, (2026) 'Judging with practical wisdom in digital courts: Paul Ricœur's little ethics and the ethical governance of AI in the judiciary,' *Ceniiac* 2, e0011.

[37] Bakirci M, (2025) 'Performance evaluation of low-power and lightweight object detectors for real-time monitoring in resource-constrained drone systems,' *Engineering Applications of Artificial Intelligence* 159, 111775.

[38] Bai Y, Jiang W, Mu J, Liu S, Gu W, and Wang S, (2025) 'Enhancing IoT security via federated learning: A comprehensive approach to intrusion detection,' *IET Information Security* 2025(1), 8432654.

[39] Islam M M, Abdullah W M, and Saha B N, (2025) 'Privacy-preserving hierarchical fog federated learning (PP-HFFL) for IoT intrusion detection,' *Sensors* 25(23), 7296.

Biography



Wan Chen received the bachelor's, master's, and Ph.D. from Hainan University, China. She is currently a lecturer at Haikou University of Economics, Haikou. Her research interests include artificial intelligence, economic forecasting, machine learning, and AI-driven applications in the information and communication technology (ICT) sector.