An Adaptive Federated Domain Generalization Framework for Consumer Electronics Manufacturing Equipment Cross-Factory Fault Detection

Haodong Li, Xingwei Wang, Ying Li, Bo Yi, *Member, IEEE*, Peng Cao, Min Huang, *Member, IEEE*, and Keqin Li, *Fellow, IEEE*

***Abstract*—Reliability and operational efficiency of equipment are crucial in the manufacturing of consumer electronics. Existing fault detection methods often face limitations such as dataset dependence, poor scenario generalization, and data privacy issues when addressing the complex and diverse operating conditions in product manufacturing. To address these issues, this paper proposes a cross-factory fault detection framework for consumer electronics production equipment based on adaptive federated domain generalization. This framework reconsiders the limitations of Sharpness-Aware Minimization(SAM) and, by jointly considering local personalization and global generalization objectives, designs an adaptive weighting scheme to balance the trade-off between loss minimization and sharpness during optimization, thereby improving the model's robustness and accuracy under various working conditions. Then, A parameter momentum aggregation scheme is proposed on the server side to incorporate historical gradient information, reducing client drift impact and improving model convergence and stability. Finally, extensive scenario experiments were conducted on two public datasets. The results indicate that the proposed framework achieves an average improvement of 22.5% in fault detection accuracy over the baseline model across varying operating conditions and data distribution scenarios, demonstrating its effectiveness in addressing the challenges of complex condition variations and data privacy in consumer electronics manufacturing.**

***Index Terms*—Federated domain generalization, sharpness-aware minimization, fault detection.**

Received 16 September 2024; revised 10 November 2024; accepted 6 January 2025. Date of publication 20 January 2025; date of current version 14 August 2025. This work was supported in part by the National Key Research and Development Program of China Grant 2021YFB3300900, and in part by the National Natural Science Foundation of China under Grant 92267206 and Grant 62032013. *(Corresponding author: Xingwei Wang.)*

Haodong Li, Xingwei Wang, Ying Li, and Bo Yi are with the College of Computer Science and Engineering, Northeastern University, Shenyang 110819, China (e-mail: lihaodong0811@163.com; wangxw@mail.neu.edu.cn; liying1771@163.com; yibo@cse.neu.edu.cn).

Peng Cao is with the Key Laboratory of Intelligent Computing in Medical Image, Northeastern University, Shenyang 110819, China (e-mail: caopeng@mail.neu.edu.cn).

Min Huang is with the College of Information Science and Engineering, Northeastern University, Shenyang 110819, China (e-mail: mhuang@mail.neu.edu.cn).

Keqin Li is with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410012, China (e-mail: likq@hnu.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TCE.2025.3529613, provided by the authors.

Digital Object Identifier 10.1109/TCE.2025.3529613

I. Introduction

WITH the rapid development of Industry 5.0, the consumer electronics manufacturing industry is undergoing unprecedented transformations. Industry 5.0 emphasizes human-machine collaboration, intelligent manufacturing, and personalized production, further enhancing product quality and production efficiency, which in turn improves customer satisfaction [1]. However, with the increasing complexity of the consumer electronics manufacturing process and the diversification of production equipment, the condition monitoring and fault detection of industrial equipment have become crucial steps to ensure production quality and stable equipment operation [2].

In recent years, the rapid development of the industrial Internet has led to the emergence of data-driven fault detection methods for consumer electronics production equipment, significantly improving the performance and accuracy of traditional models [3]. Luo et al. proposed a deep learning algorithm that leverages dynamic attribute recognition to detect early mechanical faults under time-varying operating conditions [4]. Liu et al. developed a novel wind turbine condition monitoring and fault isolation system based on supervisory control and data acquisition to enhance operational reliability and maintenance efficiency [5]. Oh et al. introduced a deep transferable adaptive fault detection method for industrial robotic equipment, effectively ensuring stability in consumer electronics production lines [6]. However, in real industrial settings, individual users have limited fault data and computational resources, which poses challenges for existing fault detection methods when applied to consumer electronics production equipment.

Firstly, most existing methods rely heavily on the completeness and annotation of datasets; however, obtaining labeled data in a cross-factory environment is often costly, time-consuming, and labor-intensive [7], [8]. Secondly, the distribution of equipment data varies significantly between factories. Differences in operating conditions, equipment types, and environmental factors lead to distinct fault patterns in each factory, making it challenging for centralized models to generalize to unknown environments. Although federated training of edge device data can alleviate this issue, the

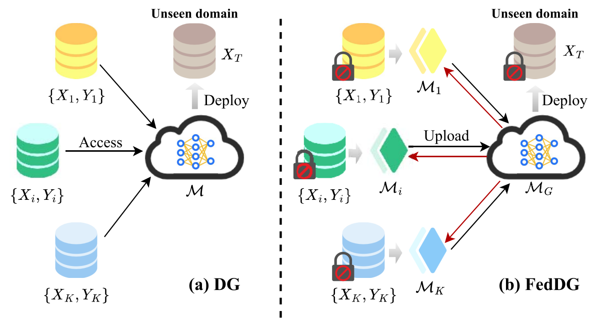


Fig. 1. Comparison between DG and FedDG.

bottleneck of computational resources continues to limit model performance, and data heterogeneity presents an additional challenge to model stability [9]. Additionally, data privacy issues are a major challenge for centralized methods, especially when sensitive production data is involved, leading to severe industrial data silo problems [10].

Consequently, Google proposed a novel distributed federated learning(FL) framework, which enables collaborative learning while ensuring data privacy, has gained widespread attention [11]. However, FL still faces limitations in generalization for cross-factory equipment fault detection, particularly in addressing data distribution heterogeneity and unknown fault patterns. This limitation stems from the assumption in current FL methods that training and testing data share the same distribution. Nonetheless, this assumption is often invalid in the consumer electronics production industry, where failure modes vary significantly across different task requirements and production environments [12]. This variability leads to a decline in federated model performance under unfamiliar conditions and across different factory environments.

Against this background, introducing domain generalization(DG) algorithms into the FL framework is a promising yet challenging task [12]. DG aims to enhance the model's generalization ability on unseen target domains by training on multiple source domains, as illustrated in Figure 1(a). However, existing technologies face challenges in direct application to FL settings due to data privacy concerns and data silo limitations [13]. Therefore, this paper identifies the problem of cross-factory fault detection in consumer electronics production equipment based on federated domain generalization(FedDG), aiming to address the generalization issue of federated models under unknown working conditions and factories, as shown in Figure 1(b). Inspired by the argument that generalization performance is closely related to the flatness of the model's minima [14], this paper proposes a federated domain generalization framework based on adaptive sharpness-aware minimization (FedASAM). On the one hand, FedASAM leverages the sharpness-aware minimization (SAM) algorithm to seek a flat global loss landscape to achieve global model generalization. On the other hand, FedASAM introduces adaptive weights to flexibly balance the trade-off between loss minima and sharpness during the optimization process. This approach enables the global model to better adapt and generalize across different clients, thereby mitigating the impact of varying degrees of flatness on clients with different performance in heterogeneous scenarios. Additionally, in the server-side aggregation phase, FedASAM employs an adaptive momentum aggregation method, which updates the model by incorporating historical gradient information to reduce the impact of client drift on the global model.

The main contributions of this paper are as follows:

1) We propose a federated domain generalization framework for cross-factory fault detection in the consumer electronics manufacturing industry. By employing an adaptive flatness search scheme, the framework enhances the generalization performance of federated models under unknown working conditions while considering heterogeneous data distribution across clients.
2) We design a server-side momentum aggregation scheme for federated domain generalization, which effectively enhances the model's convergence speed, stability, and generalization capability.
3) We provide a rigorous proof of model convergence in the federated domain generalization scenario..
4) Extensive experimental scenarios are designed, and experimental results show that the proposed method improves accuracy by an average of 22.5% over the baseline across all domain generalization scenarios.

II. Related Works

A. Consumer Electronics Manufacturing Equipment Fault Detection

In recent years, with the upgrading and development of industrial manufacturing, fault detection methods for production equipment in the consumer electronics manufacturing industry have continuously evolved. From early methods based on rules and traditional statistical models to recent intelligent detection systems that relying on big data and machine learning technologies, the accuracy and real-time performance of fault detection have significantly improved, greatly enhancing the stability of consumer electronics. Pichette and Thibeault proposed a hybrid approach combining knowledge modeling and case-based reasoning to address data insufficiency in high-mix, low-volume production environments, aiming to automate the diagnostic process for assembling printed circuit boards [15]. Ding et al. designed a highly fault-tolerant production network control system by integrating communication protocols, control mechanisms, and fault diagnosis algorithms [16]. Chen et al. proposed a high-order dynamic mode decomposition method for fault detection and isolation in dynamic industrial production processes, aiming to improve fault detection accuracy and the identification of faulty variables [17] . Zhong and Ali proposed a method combining a self-attention mechanism and residual network for the automated monitoring and fault diagnosis of intelligent sensors, effectively capturing fault characteristics and enhancing the accuracy and reliability of sensor status monitoring [18].

However, these methods may encounter a series of practical production problems, such as data scarcity or untimely knowledge updates, missed detection of sudden faults, and high computational complexity, when faced with complex and ever-changing production environments. Therefore, there is an urgent need for a new fault detection framework that can enhance the model's generalization capability in different production environments and unknown operating conditions while ensuring data privacy.

B. Sharpness-Aware Minimization

SAM is an optimization technique proposed in recent years [19]. It aims to improve the generalization ability of models by controlling the flatness of the minima of the loss function during the optimization process [20]. Traditional optimization methods, such as Stochastic Gradient Descent (SGD), may converge to sharp minima of the loss function. While these sharp minima perform well on the training data, they often generalize poorly to unseen test data [21]. SAM effectively improves the model's generalization performance on unseen data by seeking flat regions of the loss function during the optimization process. Researchers have already introduced the SAM algorithm into the federated learning environment. Caldarola et al. incorporated SAM and its adaptive version in client optimization, significantly enhancing the generalization capability of federated models [22]. Dai et al. addressed client drift issues using global SAM techniques [23]. Sun et al. combined dynamic regularization with SAM to effectively improve the convergence speed and generalization accuracy of global models [24]. However, these methods fail to adequately balance the flat regions with the minimum loss, making it challenging to meet the generalization requirements in complex and heterogeneous industrial production environments.

C. Federated Domain Generalization

Federated domain generalization, as an emerging research area in FL, aims to train models on multiple heterogeneous federated nodes that perform well in unseen domains without the need for collaborators to share data [25]. Although research in this direction is still limited, there have been some important explorations. Wu and Gong proposed a strategy called collaborative optimization and aggregation to effectively learn domain-invariant features [26]. Peng et al. suggested solving the domain transfer issue in federated learning by generating domain-invariant features through adversarial learning [27]. Zhang et al. and Yuan et al. have independently developed domain-invariant models that excel in unseen target domains by aggregating local source models using distinct and hierarchical weights [28], [29]. While these explorations may face challenges such as data privacy security and networking communication burdens in complex industrial production environments. Some researchers have already made effective explorations in data privacy and security. Wang et al. designed an FL framework based on differential privacy mechanisms to address privacy issues in the FL process [30]. Lin et al. combined differential privacy with model compression techniques to mitigate privacy leakage caused by inference attacks while maintaining model accuracy [31]. Nonetheless, these methods also increase computational overhead while enhancing data

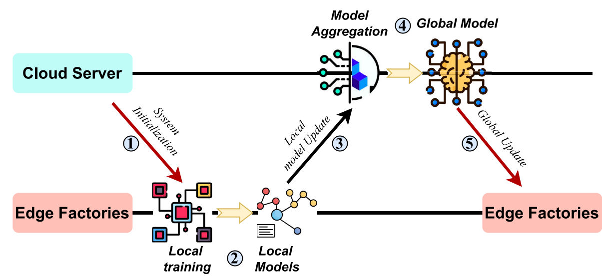


Fig. 2. The flow diagram of FedAvg Framework.

privacy protection. Therefore, improving the efficient generalization capability of FL models under the premise of data security remains a highly challenging area of research.

III. Proposed Method

A. Preliminaries

Before presenting the method proposed in this paper, we first provide the definition of the cross-factory fault detection problem in the federated setting:

Definition 1: Assuming that there are K factories willing to participate in the federated fault detection system, with each factory $i(1 \le i \le K)$ having a private dataset $D_i = \{(x_i^1, y_i^1), (x_i^2, y_i^2), \ldots, (x_i^N, y_i^N)\}$ that follows distribution $\mathcal{B}_i$, N represents the size of the datasets at each client. It is worth noting that the distributions $\mathcal{B}_i$ among different factories are different, corresponding to the data heterogeneity between factories. At the same time, Each factory can only process its local data and cannot access data from other factories. The objective of the federated fault detection model is to learn a global model $\mathcal{M}_G$ with parameters θ, which can adapt to all factories participating in federated fault detection. The generalized FL problem can be formulated as follows:

$$\arg\min_{\theta} \; F(\theta) := \frac{1}{K}\sum_{i=1}^{K} f_i(\theta),$$
$$\text{with } f_i(\theta) = \mathbb{E}_{\mathcal{B}_i(x,y)}\left[\mathcal{L}\left(\theta; (x_i^{(n)}, y_i^{(n)})\right)\right] \quad (1)$$

Here, $F(\theta)$ represents the global objective function, $f_i(\theta)$ denotes the empirical risk associated with the local distribution, $\mathcal{L}(\cdot;\cdot)$ signifies the loss function,$(x_i^{(n)}, y_i^{(n)}) \in D_i$ represents a single data sample within a factory. To solve Eq. (1), Google proposed a federated aggregation algorithm known as FedAvg [11]. In each communication round t, the server sends the global model parameters θ^t to the participating factories. Upon receiving them, each factory performs E rounds of SGD in parallel using local data, then uploads the updated parameters $\theta_{i,E}^t$. The server aggregates all updates to generate new global model parameters θ^{t+1}, initiating the next communication round. The overall process is illustrated in Figure 2.

B. Problem Formulation

The aforementioned FL framework assumes that training and testing data are extracted from the same independent and

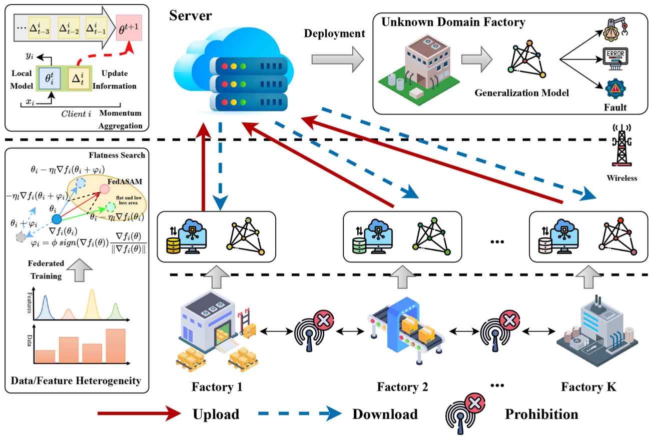


Fig. 3. Overview of the proposed FedASAM.

identically distributed (i.i.d.). However, in industrial production, data distributions exhibit heterogeneity within a single factory and across different factories, making it difficult for simple weighted aggregation models to generalize to unknown domains. Here, we present the definition of the FedDG problem in the scenario of unknown factory fault detection:

Definition 2: In FedDG, assuming there are I source domains $S_{source} = \{S_j | j = 1, 2, \ldots, I\}$, with different joint distributions between each pair of domains: $\mathcal{B}_j \neq \mathcal{B}_{\hat{j}}, 1 \leq j \neq \hat{j} \leq I$. This paper defines different source domains corresponding to different operational conditions of the equipment. The difference between the number of domains and the number of clients forms different types of FedDG problems. Bai et al. provided a detailed definition of the relationship between the number of domains and the number of clients in their study [32]. In this paper, we mainly explore scenarios where the number of clients is greater than the number of domains, that is, each factory contains multiple domain distribution data. Our goal is to develop a robust and generalized FL model using K factories containing I source domains that can minimize fault prediction errors in unknown domain factories S_{target}(where $\mathcal{B}_{target} \neq \mathcal{B}_j$):

$$\arg\min_{\theta} \; \mathbb{E}_{\mathcal{B}_{target}(x,y)}\left[\mathcal{L}(\theta; (x, y))\right] \tag{2}$$

C. FedASAM Algorithm

The FedASAM is divided into two main parts, as shown in Figure 3. In the local client, we adopt the Adaptive Sharpness Aware Minimization method, aiming to improve generalization performance by balancing the loss minimum and sharpness of model. On the server side, we use a momentum aggregation method that integrates historical gradient information into the current updates, thereby enhancing the model's generalization capability and stability. The following sections will provide a detailed description, explaining their implementation details and theoretical foundations.

1) Local Model Training on the Client Side: Extensive literature indicates that the generalization of a model is directly related to the local training loss. Existing FL methods are overly focused on the personalization of local models, leading to the local loss easily falling into the steep valleys of the loss surface, thus making it difficult for the model to take into account poorly performing clients, and even more challenging to generalize to unknown domains. Therefore, in the case of data distribution shift, priority should be given to the generality of the global model, avoiding involved training clients from getting stuck in localities, and smoothing the loss function surface to make it closer to the global model. To address this issue, inspired by the SAM algorithm, we design a federated adaptive SAM framework, aiming to create a more general global model that can exhibit good generalization performance in scenarios of client data distribution shift and unknown domain data distribution. Specifically, we first add a perturbation vector φ_i to the local models of each clients, i.e., $\tilde{\theta} = \theta + \varphi_i$. This vector is used to simulate the flatness search of the global model during local training. According to SAM, the sharpness of the perturbed federated loss function at θ is defined as:

$$\tilde{F}(\theta) \; := \underbrace{\max_{\|\varphi_i\|_p \leq \phi} F\left(\tilde{\theta}\right)}_{F^{SAM}\left(\tilde{\theta}\right)} - F(\theta) \tag{3}$$

However, $\tilde{F}(\theta)$ can only be used to find flatter areas rather than the minimum, which may lead to convergence at points where the loss is still relatively high. Therefore, SAM adopts $\tilde{F}(\theta) + F(\theta)$ as the loss function, which is a compromise between $\tilde{F}(\theta)$ and $F(\theta)$, but approach of SAM that assigns equal weight to $\tilde{F}(\theta)$ and $F(\theta)$ is not applicable in heterogeneous federated scenarios. Searching for the same degree of flatness may impact clients with different performance. Therefore, this paper designs a more universal adaptive federated SAM framework, we redefines the sharpness of the federated loss function at θ as follows:

$$\begin{aligned} F^{ASAM}(\theta) &:= F(\theta) + \frac{\beta}{1-\beta}\tilde{F}(\theta) \\ &= \frac{1-2\beta}{1-\beta}F(\theta) + \frac{\beta}{1-\beta}F^{SAM}\left(\tilde{\theta}\right) \end{aligned} \tag{4}$$

and FedASAM can be formalized as follows:

$$\begin{aligned} \min_{\theta} \; & \left\{ \left(\frac{1-2\beta}{1-\beta}\right)\left(\frac{1}{K}\sum_{i=1}^{K} f_i(\theta)\right) \right. \\ & \left. + \left(\frac{\beta}{1-\beta}\right)\left(\frac{1}{K}\sum_{i=1}^{K} \max_{\|\varphi_i\|_p \leq \phi} f_i(\theta + \varphi_i)\right) \right\} \end{aligned} \tag{5}$$

where ϕ is a constant controlling the range of the added perturbation vector, and p is the Euclidean norm, typically taken as $p = 2$. Eq. (5) is a typical $\min - \max$ optimization problem, where the min part corresponds to the update of the outer model parameters, and the max part involves the search for the inner perturbation. The objective of the interior maximization problem (i.e., max problem) is to find the perturbation φ_i such that $f_i(\theta + \varphi_i)$ is maximized, given a specific θ. This is equivalent to finding the direction in the neighborhood of θ where the loss function is least sensitive to parameter perturbations, as these directions correspond to the flat regions of the loss function. However, directly solving the aforementioned inner maximization problem is too complex. Therefore, an approximation can be obtained by performing a first-order Taylor expansion around w as follows:

$$\begin{aligned}\varphi_i &= \arg\max_{\|\varphi_i\|\le\phi} f_i(\theta+\varphi_i)\\ &\approx \arg\max_{\|\varphi_i\|\le\phi} f_i(\theta)+{\varphi_i}^{\top}\nabla f_i(\theta)\\ &\approx \phi\ sign(\nabla f_i(\theta))\frac{\nabla f_i(\theta)}{\|\nabla f_i(\theta)\|}\end{aligned}\tag{6}$$

Eq. (6) is equivalent to scaling the magnitude of the gradient using ϕ under the condition $\|\varphi_i\| \le \phi$, such that the direction of φ_i is the same as $\nabla f_i(\theta)$. Subsequently, the perturbed model at each client can be represented as $\tilde{\theta} = \theta + \phi\ sign(\nabla f_i(\theta))\frac{\nabla f_i(\theta)}{\|\nabla f_i(\theta)\|}$. We define $g_{i,e}^t$ and $\tilde{g}_{i,e}^t$ as the original gradient and the perturbed model gradient, respectively, after the e-th iteration during the t-th round of communication at the client, where $g_{i,e}^t = \nabla f_i(\theta_{i,e}^t;\xi_i^t)$, $\tilde{g}_{i,e}^t = \nabla f_i(\tilde{\theta}_{i,e}^t;\xi_i^t)$. The parameter iteration process of local FedASAM for solving the max-min problem can be expressed as follows:

$$\begin{cases}\tilde{\theta}_{i,e}^t = \theta_{i,e}^t + \phi\frac{g_{i,e}^t}{\left\|g_{i,e}^t\right\|}\\ \theta_{i,e+1}^t = \theta_{i,e}^t - \eta_l\left(\frac{1-2\beta}{1-\beta}g_{i,e}^t + \frac{\beta}{1-\beta}\tilde{g}_{i,e}^t\right)\end{cases}\tag{7}$$

where η_l denotes the local learning rate. Eq. (7) indicates that each local client first uses a gradient ascent algorithm to estimate the point where the local loss is maximized at the perturbed parameter $\theta_{i,e}^t + \varphi_i$, and computes the gradient $\tilde{g}_{i,e}^t$ of the perturbed parameter at this point. Finally, it performs a gradient descent update at $\theta_{i,e}^t$ by adaptively combining the original gradient with the perturbed gradient. This gradient combination method better balances the model's performance across different clients, thereby improving overall generalization.

2) Federated Aggregation Model on the Server Side: In existing server parameter aggregation schemes, unrealistic assumptions are often made regarding system heterogeneity and client synchronization, ignoring the impact of client drift, leading to poor generalization performance. Therefore, inspired by momentum learning, this paper proposes a momentum aggregation scheme on the server side, integrating historical information into current updates, making the global model more resistant to local drift. Specifically, we define a momentum buffer m^t, representing the variable that stores and updates historical gradient information during the t-th training round. Then the server-side optimization can be represented as:

$$\begin{cases} m^{t+1} = \gamma m^t + (1-\gamma)\Delta^t\\ \theta^{t+1} = \theta^t - \eta_g m^{t+1}\end{cases}\tag{8}$$

Here, γ represents the momentum coefficient, Δ^t denotes the aggregated gradient on the server side during the t-th training round, and η_g represents the global learning rate. Through the aforementioned momentum aggregation scheme, the server can effectively utilize historical gradient information to mitigate the impact of client drift on the global model, thereby improving the model's generalization ability. The complete FedASAM algorithm is shown in Algorithm 1.

Algorithm 1 FedASAM

Input: local datasets D_i, number of communication rounds T, number of factories K, number of local epochs E, local learning rate η_l, global learning rate η_g
Output: The global model parameters θ^T.
Server executes:

```
Initialize server model θ^t.
for t = 0, 1, 2, ..., T − 1 do
    for i = 1, 2, ..., K in parallel do
        send global model θ^t to F_i
        Δ_i^t ← ClientUpdate(t, i, θ^t)
    end for
    Server aggregates Δ_t = (1/|K_i|) Σ_{i∈K_i} Δ_i^t
    Server updates θ^{t+1} = θ^t − η_g(γ m^t + (1 − γ)Δ^t)
end for
return θ^{t+1}
```

ClientUpdate(t, i, θ^t):

```
for epoch = 1, 2, ..., E do
    for each batch ξ_i = (x_i, y_i) in D_i do
        Compute original gradient g_{i,e}^t = ∇f_i(θ_{i,e}^t; ξ_i^t)
        Compute perturbed gradient g̃_{i,e}^t = ∇f_i(θ̃_{i,e}^t; ξ_i^t)
        Compute local model θ_{i,e+1}^t by Eq. (7)
    end for
    Δ_i^t = θ^t − θ_{i,E}^t
end for
return Δ_i^t to server
```

D. Theoretical Analysis

This section presents the convergence analysis of the FedASAM framework. For the sake of subsequent analysis, we first provide some definitions. In this paper, the clients participating in FL perform e-step SGD updates locally with a learning rate of η_l, while momentum aggregation is performed on the server with a learning rate of η_g. Unlike the traditional FedAvg update rule, the update sequence in this paper defined as $\theta^{t+1} - \theta^t = \eta_g(\gamma m^t + (1-\gamma)\Delta^t)$. Therefore, to simplify the proof process, an auxiliary sequence $\{d^t\}_{t=0}^T$ is constructed. $\{d^t\}_{t=0}^T$ is defined as follows:

$$d^t = \theta^t - \frac{\eta_g\gamma}{1-\gamma}m^t\tag{9}$$

The above equation satisfies $d^{t+1} - d^t = -\eta_g\Delta_t$, and the detailed proof can be found in Appendix A im the supplementary material.

The accumulated parameter difference is denoted as $\Delta_t^i = \theta_{i,E}^t - \theta_{i,0}^t - = -\eta_l\sum_{e=0}^{E-1} g_{i,e}^t$. In the following narrative, we list the assumptions used in the convergence analysis proof in this paper.

Assumption 1 (L-Lipschitz Gradient Continuity): The overall loss function is continuously differentiable, and there exists a constant $L > 0$ satisfying the following conditions:

$\|\nabla F(\theta_1) - \nabla F(\theta_2)\| \le L\|\theta_1 - \theta_2\|, \forall \theta_1, \theta_2 \in R_d$, and $i \in [K]$ [33].

Assumption 2 (Unbiased Gradient Estimation): Let ξ_i^t represent a random sample data within the i-th factory updated to the t-th step, satisfying the following conditions: $\mathbb{E}[\nabla l_i(\theta^t, \xi_i^t)] = \nabla l_i(\theta^t), \forall i \in [K]$ [34].

Assumption 3 (Bounded Gradient Variance): The variance of the stochastic gradient for each internal factory model satisfies the following conditions: $\mathbb{E}[\|\nabla l_i(\theta^t, \xi_i^t)] - \nabla l_i(\theta^t)\|^2] \le \Gamma$, $\mathbb{E}[\|\nabla l_i(\theta^t, \xi_i^t)\|^2] \le \Omega, \forall i \in [K]$ [35].

Theorem 1: After $T \ge 1$ rounds of communication, if the local learning rate η_l and global learning rate η_g satisfies $\eta_l \le \frac{(1-\gamma)^2}{2E(C\eta_g^2 L^2 \gamma^2 + (1-\gamma)^2 + \eta_g L(1-\gamma)^2)}$, the convergence upper bound of Algorithm 1 based on Assumptions 1-3 can be formalized as follows:

$$\min_{t \in [T]} \mathbb{E}_t\left[\|\nabla F(\theta^t)\|^2\right] \le \frac{2\left(F(\theta^0) - F(\theta^*)\right)}{\eta_g \eta_l E T} + \Pi \qquad (10)$$

where, $\Pi = \frac{\eta_l \Gamma}{K} + \eta_l{}^2 L^2 E \Omega + \frac{L \eta_g \eta_l \Gamma}{K} + \frac{\eta_g^2 \eta_l L^2 \gamma^2 \Gamma}{K(1-\gamma^2)^2}$

Theorem 1 shows that as the number of communication rounds T increases, the upper bound of the expected norm of the gradient can be determined by the difference between the initial value and the optimal value of the function, as well as terms related to the learning rate. This implies that by choosing an appropriate learning rate, the gradient's descent can be ensured, leading to the global convergence of the algorithm. The detailed proof can be found in Appendix B in the supplementary materials.

IV. Experiment Study

A. Dataset Selection and Task Scenario Setup

1) Dataset: To validate the generalization performance of the proposed method in complex industrial scenarios, we selected two standard datasets from industrial domains to conduct extensive generalization experiments.

- Paderborn University (PU) dataset: The PU dataset provides fault data generated by two methods: artificial damage and accelerated life testing. Various fault types occurring during bearing operation were simulated using methods such as drilling and electric engraving machines [36]. We randomly selected bearings with different processing methods and fault depths to form a dataset comprising 8 different fault types.
- Case Western Reserve University (CWRU) dataset: The CWRU dataset provides four working conditions, with vibration signals collected under each condition using a 16-channel data recorder. Single-point damage was artificially created at different bearing locations using electrical discharge machining, resulting in fault types with diameters of 0.007, 0.014, and 0.021 inches [37]. Fault data from both the drive end and fan end are provided. In our experimental setup, we selected data from four types: healthy bearings, inner race faults, outer race faults, and rolling element faults, each with different working conditions and fault severity, to create a comprehensive test scenario. Detailed information about the two datasets is shown in Table I.

TABLE I
Detailed Information on Fault Modes in PU and CWRU Datasets

PU	Fault modes	(0) Normal; (1) Outer ring damage of EDM; (2) Outer ring damage of electric engraver; (3) Outer ring damage of drilling(level 1); (4) Outer ring damage of drilling(level 2); (5) Inner ring damage of EDM; (6) Inner ring damage of electric engraver(level 1); (7) Inner ring damage of electric engraver(level 2)
	Working condition (Rotational speeds, load torques, and radial forces)	A=1500 rpm, 0.7 Nm, 1000 N; B=900 rpm, 0.7 Nm, 1000 N; C=1500 rpm, 0.1 Nm, 1000 N; D=1500 rpm, 0.7 Nm, 400 N
CWRU	Fault modes	(0) Normal; (1) Rolling element damage with level 0.007inch; (2) Inner ring with level 0.014inch; (3) Outer ring with level 0.021inch
	Working condition (Rotational speeds)	E = 1797 r/min; F = 1772 r/min; G = 1750 r/min; H = 1730 r/min
	Fault location	Drive end(DE); Fan end(FE)

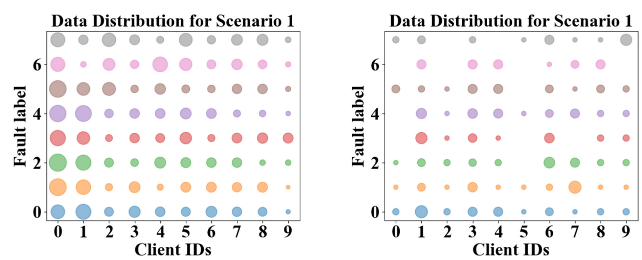


Fig. 4. Data distribution for different scenarios.

2) Task Scenario Setup: This study simulates scenarios of federated domain generalization based on the aforementioned datasets. In the experimental setup, each client is considered an independent domain to mimic different working conditions and environments that may be encountered in real applications. Specifically, each training client is configured to include mixed data from three different bearing fault conditions, simulating complex real-world operating environments. The test clients contain only one unseen condition to evaluate the model's generalization capability in new conditions. Additionally, on the CWRU dataset, we set up federated generalization experiments with cross-condition and cross-location mixtures to further test the robustness and adaptability of the proposed method in the face of broader domain variations. Table II shows detailed information about the detection tasks designed on the PU and CWRU datasets.

To comprehensively simulate realistic industrial production scenarios, we have designed two data distribution schemes, taking the PU dataset as an example, as illustrated in Figure 4, the color of each circle represents a different fault type, and the size indicates the data volume for that fault type. In scenario 1, each client contains all fault types, but the data volume for each fault type is inconsistent. This distribution

TABLE II
DESCRIPTION OF CROSS-FACTORY DOMAIN GENERALIZATION TASK SCENARIOS FOR PU AND CWRU DATASETS

PU task name	Multiple source clients	Unknown target clients	CWRU task name	Multiple source clients (DE)	Unknown target clients (FE)
P1	A B C	D	C1	E F G	H
P2	A B D	C	C2	E F H	G
P3	A C D	B	C3	E G H	F
P4	B C D	A	C4	F G H	E

method is more reflective of the data distribution in real industrial environments. In scenario 2, the fault types and data volumes for each client are randomly assigned, making the generalization task in scenario 2 more challenging.

B. Approaches for Comparison and Implementation Details

1) Comparison Methods: To fully validate the effectiveness and superiority of the FedASAM algorithm when dealing with unknown domain clients, we compared the performance of FedASAM with a total of 7 FL frameworks under four strategies, including FedAvg [11], FedProx [38], FedALA [39], FedADG [13], FedSoup [40], FedSpeed [41], and FedGAMMA [23]. FedAvg mainly updates the global model by averaging the model updates from each client, which is also set as the baseline in this paper. FedProx introduces a regularization term based on FedAvg, which helps better control the deviation of each client from the global model. FedALA is a personalized FL framework that considers both the global model and the specific needs of each client. FedADG, FedSoup, FedSpeed, and FedGAMMA, as current state-of-the-art (SOTA) federated domain generalization frameworks, aim to improve the model's performance on unseen client domains.

2) Implementation Details: For all scenarios, this paper uses Resnet-18 as the main framework. However, appropriate modifications were made to data input and feature extraction to address the characteristics of industrial time series data. In each scenario, 10 clients are set up for federated training, and 5 clients with unknown data distributions are used for testing. The initial global training cycle for all methods is set to 100; the local training cycle is 5. The control variables $\beta = 0.6$, $\phi = 0.3$, momentum coefficient $\gamma = 0.1$, and local batch size is 32. The hyperparameters for all comparative methods are selected based on the literature to achieve satisfactory performance while meeting experimental requirements.

3) Metrics: In this study, the performance of the proposed model is evaluated using the following metrics: the average accuracy(Acc) of fault classification across all clients and the average Area Under the Curve (AUC).

- Acc: The average accuracy of all generalization test factories involved in joint detection, i.e., $Acc = 1/K \sum_{i=1}^{K} a_i$, where a_i represents the accuracy of fault classification for an individual factory.
- AUC: The average AUC for all generalization test factories participating in joint detection. i.e., $AUC = 1/K \sum_{i=1}^{K} AUC_i$. The AUC value is not sensitive to the imbalance of sample categories, making it suitable for reflecting the overall classification performance of the model in situations with imbalanced samples.

C. Performance Evaluation and Result Analysis

In Tables III and IV, we report the comparison of the unknown domain generalization accuracy between the method proposed in this paper and the current SOTA methods across two industrial datasets, two data distribution scenarios, and four cross-domain task scenarios. From the experimental results, we can observe that in scenario 1, which includes complete fault types, FedASAM significantly outperforms other methods on average, demonstrating its excellent performance in cross-domain fault detection tasks. Specifically, on the PU dataset, compared to the widely compared FedAvg, our method improved accuracy and AUC by 36.20% and 4.44%, respectively. Additionally, in cross-location fault detection tasks on the CWRU dataset, accuracy and AUC improved by 53.21% and 5.89%, respectively, indicating that the proposed method can effectively adapt to generalization tasks in scenarios with imbalanced sample distributions and cross-location fault detection.

When facing scenario 2, which involves missing data types, most methods showed a significant performance decline, primarily due to the severe insufficiency of fault type samples, which affected the training of the aggregated model. In contrast, the FedASAM framework maintained a high level of generalization accuracy and robustness. Specifically, in the cross-location generalization tasks on the CWRU dataset, FedASAM showed an average improvement of 54.15% in accuracy and 14.03% in AUC compared to the baseline, indicating that FedASAM can more effectively distinguish between fault and non-fault states in cross-domain generalization tasks, reducing false positives and false negatives. Overall, the experimental results demonstrate that by introducing adaptive SAM and momentum aggregation, FedASAM significantly enhances the performance of FL models in cross-domain scenarios, offering stronger generalization ability and robustness.

To further evaluate the performance of each client participating in the federated domain generalization task, we selected the PU dataset scenario 1, task P1 (P1(S1)), scenario 2, task P3 (P3(S2)) and the CWRU dataset scenario 1, task C2 (C2(S1)), scenario 2, task C4 (C4(S2)) as the experimental subjects. We visualized the accuracy of the five clients in the test, and the experimental results are shown in Figure 5. From the comparative radar chart, it can be seen that FedASAM performed excellently in scenario 1 of both datasets, achieving high fault detection accuracy on almost all clients. This indicates that FedASAM exhibits strong generalization capability in situations with uneven data distribution but comprehensive fault types. In the more complex scenario 2, despite the significant performance differences between clients, FedASAM still maintained a superior performance. This further demonstrates its robustness in handling complex

TABLE III
COMPARISON OF GENERALIZATION PERFORMANCE(%) WITH STATE-OF-THE-ART METHODS IN SCENARIO 1

Method	P1		P2		P3		P4		C1		C2		C3		C4	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
FedAvg	63.82	92.66	75.37	94.73	66.09	92.22	74.48	93.85	79.75	97.64	79.27	98.26	50.74	95.94	55.01	96.18
FedProx	65.20	92.25	73.17	93.30	70.01	92.40	70.08	93.17	81.38	97.82	64.98	95.65	64.25	96.65	76.26	97.33
FedALA	58.42	91.03	80.76	94.45	71.07	91.14	66.41	92.02	67.47	96.34	78.25	97.88	81.21	96.56	70.48	97.04
FedADG	61.25	92.36	72.90	93.45	46.58	86.94	62.07	91.89	84.40	98.25	71.90	97.64	73.32	97.67	78.22	97.84
FedSoup	86.51	95.33	90.68	96.28	81.82	96.22	90.75	96.47	87.72	94.98	74.43	94.72	74.69	97.30	78.06	98.01
FedSpeed	79.41	92.09	89.68	97.17	87.59	96.72	89.63	97.03	91.46	98.34	89.07	98.93	87.93	99.16	89.14	98.59
FedGAMMA	83.03	93.82	94.07	97.24	86.00	96.80	91.73	97.00	94.94	**98.59**	80.45	98.47	93.51	**99.73**	92.26	99.37
FedASAM	**90.33**	**96.92**	**94.61**	**97.94**	**88.52**	**97.28**	**93.44**	**97.87**	**95.69**	98.46	**91.94**	**99.29**	**95.14**	99.32	**93.91**	**99.80**

TABLE IV
COMPARISON OF GENERALIZATION PERFORMANCE(%) WITH STATE-OF-THE-ART METHODS IN SCENARIO 2

Method	P1		P2		P3		P4		C1		C2		C3		C4	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
FedAvg	78.65	92.65	76.99	93.85	67.57	91.52	73.76	95.02	56.34	93.97	67.47	96.58	67.22	95.81	63.05	96.50
FedProx	71.01	90.30	74.54	93.30	72.58	93.29	78.87	95.51	61.88	94.01	64.19	94.89	54.25	94.82	74.96	97.69
FedALA	70.14	89.53	76.92	94.23	75.15	92.63	72.17	92.14	57.39	92.71	85.41	97.26	65.77	97.21	73.66	97.88
FedADG	76.22	93.00	83.98	95.99	64.12	90.94	80.46	96.42	61.63	94.47	59.02	94.72	60.71	96.09	74.78	97.20
FedSoup	84.46	89.60	90.84	95.61	78.81	87.84	89.50	96.26	88.25	**98.41**	85.33	97.47	83.97	98.33	77.25	98.84
FedSpeed	87.43	94.91	91.04	96.92	81.18	95.30	**90.81**	95.79	72.84	96.78	90.25	98.43	**88.69**	98.61	87.27	98.97
FedGAMMA	89.86	97.03	92.19	97.33	84.97	**97.23**	88.33	95.39	92.49	97.80	81.55	98.46	81.76	98.84	91.34	99.24
FedASAM	**92.77**	**97.39**	**93.62**	**97.77**	**85.92**	96.88	90.40	**96.88**	**94.29**	98.35	**90.63**	**98.54**	87.76	**98.91**	**92.34**	**99.35**

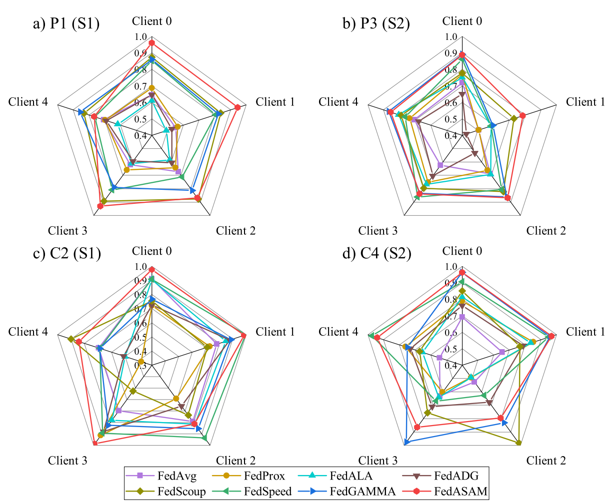


Fig. 5. Generalization performance of FedASAM on each unseen client across different scenarios.

and heterogeneous data distributions, validating its potential and advantages in federated domain generalization tasks.

D. In-Depth Study

1) Sharpness Quantification: Recent studies on network generalization have shown that the top Hessian eigenvalue (α_{max}) and the Hessian trace (α_T) can serve as important indicators for assessing the generalization capability of models [42], [43]. Models with lower α_{max} and α_T tend to exhibit stronger robustness, leading to smoother loss surfaces and flatter minima during the training process. Therefore, this paper selected several representative FL methods' global models and calculated their α_{max} and α_T. The results are shown in Table V and Figure 6. From the experimental results, both traditional federated learning and personalized federated learning struggle to adapt to domain generalization tasks under unknown conditions, showing relatively high α_{max} and α_T. In contrast, the federated domain generalization framework with the introduction of the SAM method significantly outperforms traditional domain generalization methods, further proving the importance of minimizing the loss surface in federated domain generalization tasks. In all task scenarios, FedASAM consistently achieves the lowest α_{max} and α_T, demonstrating its lower sensitivity to changes in the environment, stronger generalization ability, and more stable optimization process.

2) Visualization of the Loss Landscape: To provide a more intuitive demonstration of the effectiveness of the FedASAM method, we visualized the loss functions optimized by FedAvg, FedSpeed, FedGamma, and FedASAM in a random task scenario (P2(S1)), as shown in Figure 7. As observed from the figure, the loss landscape of FedAvg is very sharp, indicating its poor generalization ability. The two more advanced federated generalization models, FedSpeed

TABLE V
COMPARISON OF TOP EIGENVALUE(α_{max}) ACROSS DIFFERENT FL METHODS

Method	P1(S1)	P3(S2)	C2(S1)	C4(S2)
FedAvg	1481.65	891.98	1165.83	3019.05
FedALA	2285.32	1468.45	602.92	730.51
FedADG	1382.21	1425.52	1344.25	678.64
FedSpeed	693.31	**734.17**	<u>531.37</u>	627.42
FedGAMMA	<u>506.66</u>	777.83	676.42	<u>480.83</u>
FedASAM	**479.71**	<u>758.92</u>	**522.47**	**350.44**

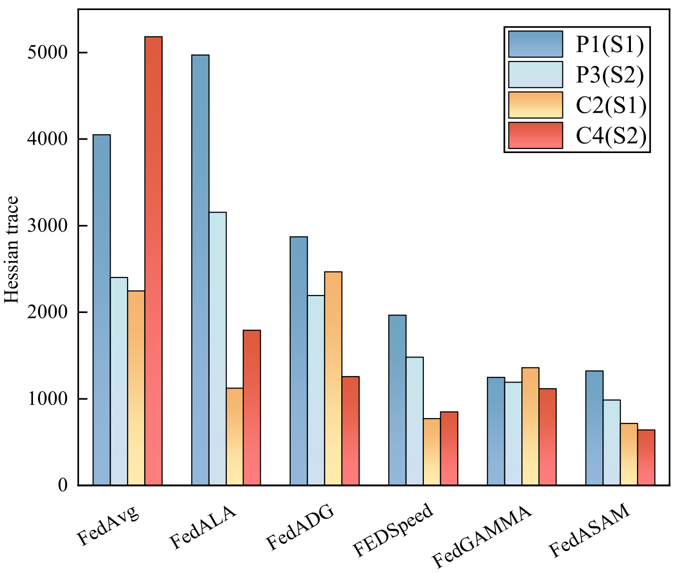


Fig. 6. Comparison of Hessian trace (α_T) across different FL methods.

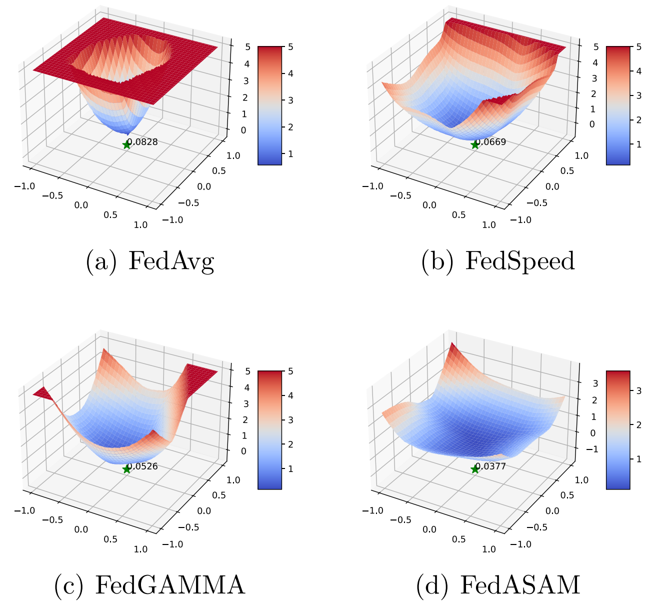


Fig. 7. 3-D landscape visualizations of the empirical loss of the global model across different FL frameworks.

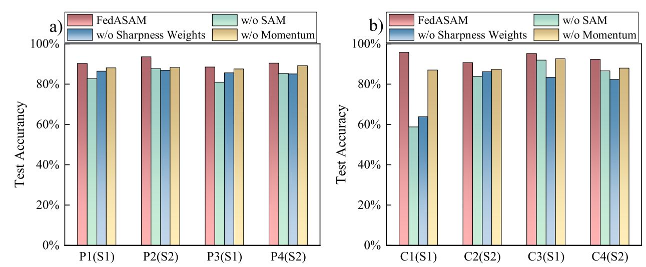


Fig. 8. Ablation study of the FedASAM on generalization tasks across different scenarios.

and FedGamma, show a significant improvement in their loss surfaces compared to FedAvg. This indicates that they are better at optimizing model loss under heterogeneous client data distributions, leading to improvements in both generalization ability and robustness. However, these methods only average the local flat regions of each client, which might remain in sharp regions due to client drift issues. FedASAM achieves the best performance in both loss values and loss landscape. This is due to the introduction of an adaptive sharpness control technique, which allows the FedASAM method to better balance model sharpness and the minimum loss, thereby improving generalization across unknown factories and conditions. Compared to other methods, FedASAM demonstrates higher robustness and accuracy when dealing with complex condition variations.

3) Ablation Experiment: To validate the effectiveness of the components in the proposed framework, this section conducts ablation experiments with three variants based on the FedASAM framework i.e., 'w/o SAM', 'w/o Sharpness Weight' and 'w/o Momentum'. Specifically, 'w/o SAM' removes sharpness-aware minimization, 'w/o Sharpness Weight' removes adaptive sharpness weighting, and 'w/o Momentum' removes server momentum. The experimental results are shown in Figure 8. It can be observed that when SAM is removed, the generalization performance of the model across different clients significantly decreases, indicating that SAM plays a crucial role in maintaining model stability and robustness. After removing the adaptive sharpness weight, the test accuracy also decreases. The role of adaptive sharpness weight is to balance updates between minimizing curvature and minimizing loss. This is particularly important in complex scenarios, such as scenario 2, where the lack of adaptive sharpness adjustment makes the model more prone to getting trapped in local optima, thus affecting overall generalization performance. Figure 8 also shows that without server momentum aggregation, the model's performance fluctuates significantly across different clients, further verifying the importance of momentum aggregation in FL.

E. Hyperparameter Analyses

This section mainly discusses the impact of perturbation radius ϕ, sharpness term β, and server momentum aggregation coefficient γ on model performance. The P1(S1), P3(S2), C2(S1), and C4(S2) task scenarios are used as experimental subjects, and the experimental results are shown in Figure 9.

1) Perturbation Radius ϕ: The parameter ϕ represents the step size in the gradient ascent process during flatness search.

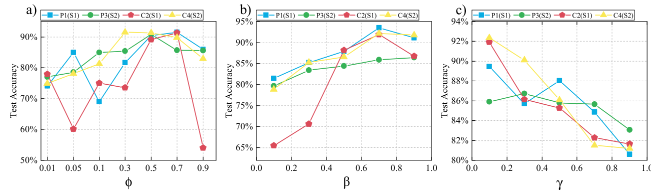


Fig. 9. The sensitivity of FedASAM's performance to the choice of ϕ, β, γ.

It is used to measure how much area the model needs to explore to find the flattest minimum. As shown in Figure 9(a), when ϕ is small, the model's generalization ability is weak. This is because a smaller search area may cause the model to get trapped in the flat minimum within steep valleys. As ϕ increases, the model's generalization ability improves significantly. However, an excessively large ϕ may cause the model to favor larger flat minima, thereby reducing the original data processing performance. This is especially evident in the cross-location fault detection tasks on the CWRU dataset, where a larger ϕ causes the model's generalization performance to drop sharply. Therefore, it is necessary to reasonably choose the search range for global flatness search based on the specific task context.

2) Sharpness Term β: The parameter β is the weight that controls sharpness. As observed in Figure 9(b), when $\beta < \frac{1}{2}$, the federated model focuses more on minimizing the loss function itself during the optimization process, rather than considering the impact of sharpness. In this case, the model may tend to find a minimum with lower loss but higher curvature, meaning it could converge at a narrow and steep valley. The model's generalization ability might be weaker than that in flatter regions with lower curvature. When $\beta > \frac{1}{2}$, the model tends to minimize curvature, leading it to converge in flatter regions with lower curvature. This reduced curvature in flatter regions means the model is less sensitive to small input perturbations (i.e., data noise), thereby effectively improving its generalization ability.

3) Server Momentum Value γ: In the momentum aggregation scheme, the hyperparameter γ plays a crucial role in balancing historical gradient information with newly aggregated gradients. This balance is essential for mitigating the effects of client drift and enhancing the global model's generalization ability. As observed in Figure 9(c), the generalization performance of the model shows a significant decline with the increase of γ. This indicates that over-reliance on historical gradients may reduce the model's ability to adapt to new data patterns, while noise in the historical information can also significantly impact model updates. This effect is particularly pronounced in cross-location generalization tasks, where a larger momentum coefficient severely impairs the model's generalization ability. Experimental results indicate that while the momentum aggregation method enhances the model's resistance to client drift, careful tuning is required.

V. Conclusion

This paper proposes the FedASAM framework for cross-factory fault detection of industrial equipment in consumer electronics production. By incorporating an adaptive SAM algorithm, the framework significantly improves the generalization capability of the global model under heterogeneous client data distributions, while ensuring data privacy. Additionally, an adaptive momentum aggregation strategy is implemented on the server side, which greatly enhances the convergence speed and stability of the global model, reducing the impact of client drift on the global model's performance. Experimental results show that the framework demonstrates excellent detection accuracy and generalization performance across different working conditions and data distribution scenarios, effectively addressing the limitations of existing fault detection methods in complex conditions. The proposed method provides a novel solution for cross-factory fault detection in unknown settings, addressing diverse working conditions and fault types in the consumer electronics production process while ensuring data privacy, thus offering significant practical value.

References

[1] C. Zhang et al., "Towards new-generation human-centric smart manufacturing in industry 5.0: A systematic review," *Adv. Eng. Inf.*, vol. 57, Aug. 2023, Art. no. 102121.

[2] S.-K. S. Fan, C.-Y. Hsu, D.-M. Tsai, F. He, and C.-C. Cheng, "Data-driven approach for fault detection and diagnostic in semiconductor manufacturing," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 4, pp. 1925–1936, Oct. 2020.

[3] H. Chen and B. Jiang, "A review of fault detection and diagnosis for the traction system in high-speed trains," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 450–465, Feb. 2020.

[4] B. Luo, H. Wang, H. Liu, B. Li, and F. Peng, "Early fault detection of machine tools based on deep learning and dynamic identification," *IEEE Trans. Ind. Electron.*, vol. 66, no. 1, pp. 509–518, Jan. 2019.

[5] X. Liu, J. Du, and Z.-S. Ye, "A condition monitoring and fault isolation system for wind turbine based on SCADA data," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 986–995, Feb. 2022.

[6] Y. Oh, Y. Kim, K. Na, and B. D. Youn, "A deep transferable motion-adaptive fault detection method for industrial robots using a residual–convolutional neural network," *ISA Trans.*, vol. 128, pp. 521–534, Sep. 2022.

[7] Y. Wang, X. Yuan, Y. Lin, J. Gu, and M. Zhang, "A semi-supervised multi-scale deep adversarial model for fan anomaly detection," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 3539–3547, Feb. 2024.

[8] T. Li, Z. Zhao, C. Sun, R. Yan, and X. Chen, "Domain adversarial graph convolutional network for fault diagnosis under variable working conditions," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–10, 2021.

[9] W. Zhang and X. Li, "Federated transfer learning for intelligent fault diagnostics using deep adversarial networks with data privacy," *IEEE/ASME Trans. Mechatronics*, vol. 27, no. 1, pp. 430–439, Feb. 2022.

[10] J. Chen, J. Li, R. Huang, K. Yue, Z. Chen, and W. Li, "Federated transfer learning for bearing fault diagnosis with discrepancy-based weighted federated averaging," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–11, 2022.

[11] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Stat.*, 2017, pp. 1273–1282.

[12] C. Zhao and W. Shen, "Federated domain generalization: A secure and robust framework for intelligent fault diagnosis," *IEEE Trans. Ind. Informat.*, vol. 20, no. 2, pp. 2662–2670, Feb. 2024.

[13] L. Zhang, X. Lei, Y. Shi, H. Huang, and C. Chen, "Federated learning with domain generalization," 2021, *arXiv:2111.10487*.

[14] N. S. Keskar, J. Nocedal, P. T. P. Tang, D. Mudigere, and M. Smelyanskiy, "On large-batch training for deep learning: Generalization gap and sharp minima," in *Proc. 5th Int. Conf. Learn. Rep. (ICLR)*, 2017, pp. 1–9.

[15] S. Pichette and C. Thibeault, "Knowledge-intensive diagnostics using case-based reasoning and synthetic case generation," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 42, no. 7, pp. 2404–2417, Jul. 2023.

[16] S. X. Ding, P. Zhang, S. Yin, and E. L. Ding, "An integrated design framework of fault-tolerant wireless networked control systems for industrial automatic control applications," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 462–471, Feb. 2013.

[17] X. Chen, J. Zheng, C. Zhao, and M. Wu, "Full decoupling high-order dynamic mode decomposition for advanced static and dynamic synergetic fault detection and isolation," *IEEE Trans. Autom. Sci. Eng.*, vol. 21, no. 1, pp. 226–240, Jan. 2024.

[18] S. Zhong and R. Ali, "Joint self-attention mechanism and residual network for automated monitoring of intelligent sensor in consumer electronics," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 1302–1309, Feb. 2024.

[19] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, "Sharpness-aware minimization for efficiently improving generalization," 2020, *arXiv:2010.01412*.

[20] B. Li and G. Giannakis, "Enhancing sharpness-aware optimization through variance suppression," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2024, pp. 1–9.

[21] J. Wang et al., "Generalizing to unseen domains: A survey on domain generalization," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 8, pp. 8052–8072, Aug. 2023.

[22] D. Caldarola, B. Caputo, and M. Ciccone, "Improving generalization in federated learning by seeking flat minima," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 654–672.

[23] R. Dai et al., "FedGAMMA: Federated learning with global sharpness-aware minimization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 12, pp. 17479–17492, Dec. 2024.

[24] Y. Sun, L. Shen, S. Chen, L. Ding, and D. Tao, "Dynamic regularized sharpness aware minimization in federated learning: Approaching global consistency and smooth landscape," in *Proc. Int. Conf. Mach. Learn.*, 2023, pp. 32991–33013.

[25] Y. Wei and Y. Han, "Multi-source collaborative gradient discrepancy minimization for federated domain generalization," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, 2024, pp. 15805–15813.

[26] G. Wu and S. Gong, "Collaborative optimization and aggregation for decentralized domain generalization and adaptation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 6484–6493.

[27] X. Peng, Z. Huang, Y. Zhu, and K. Saenko, "Federated adversarial domain adaptation," 2019, *arXiv:1911.02054*.

[28] R. Zhang, Q. Xu, J. Yao, Y. Zhang, Q. Tian, and Y. Wang, "Federated domain generalization with generalization adjustment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 3954–3963.

[29] J. Yuan, X. Ma, D. Chen, F. Wu, L. Lin, and K. Kuang, "Collaborative semantic aggregation and calibration for federated domain generalization," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 12, pp. 12528–12541, Dec. 2023.

[30] B. Wang, Y. Chen, H. Jiang, and Z. Zhao, "PPEFL: Privacy-preserving edge federated learning with local differential privacy," *IEEE Internet Things J.*, vol. 10, no. 17, pp. 15488–15500, Sep. 2023.

[31] Q. Lin, S. Jiang, Z. Zhen, T. Chen, C. Wei, and H. Lin, "Fed-PEMC: A privacy-enhanced federated deep learning algorithm for consumer electronics in mobile edge computing," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 4073–4086, Feb. 2024.

[32] R. Bai, S. Bagchi, and D. I. Inouye, "Benchmarking algorithms for federated domain generalization," in *Proc. 12th Int. Conf. Learn. Rep.*, 2024, p. 12.

[33] S. Wang et al., "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019.

[34] S. Ghadimi and G. Lan, "Stochastic first-and zeroth-order methods for nonconvex stochastic programming," *SIAM J. Optim.*, vol. 23, no. 4, pp. 2341–2368, 2013.

[35] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on non-IID data," in *Proc. Int. Conf. Learn. Rep.* 2020, p. 12.

[36] C. Lessmeier, J. K. Kimotho, D. Zimmer, and W. Sextro, "Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification," in *Proc. PHM Soc. Eur. Conf.*, vol. 3, 2016, pp. 1–8.

[37] W. A. Smith and R. B. Randall, "Rolling element bearing diagnostics using the case western reserve university data: A benchmark study," *Mech. Syst. Signal Process.*, vols. 64–65, pp. 100–131, Dec. 2015.

[38] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Mach. Learn. Syst.*, vol. 2, 2020, pp. 429–450.

[39] J. Zhang et al., "FEDALA: Adaptive local aggregation for personalized federated learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, 2023, pp. 11237–11244.

[40] M. Chen, M. Jiang, Q. Dou, Z. Wang, and X. Li, "FedSoup: Improving Generalization and Personalization in federated learning via selective model interpolation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2023, pp. 318–328.

[41] Y. Sun, L. Shen, T. Huang, L. Ding, and D. Tao, "FedSpeed: Larger local interval, less communication round, and higher generalization accuracy," in *Proc. 11th Int. Conf. Learn. Rep.*, 2023, pp. 1–9.

[42] Y. Jiang, B. Neyshabur, H. Mobahi, D. Krishnan, and S. Bengio, "Fantastic Generalization measures and where to find them," in *Proc. Int. Conf. Learn. Rep.*, 2020, pp. 1–11.

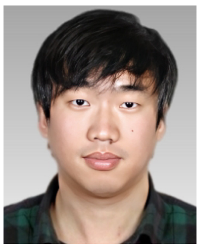
[43] T. Yoon, S. Shin, S. J. Hwang, and E. Yang, "FedMix: Approximation of mixup under mean augmented federated learning," in *Proc. Int. Conf. Learn. Rep.*, 2021, pp. 1–11.

Haodong Li received the B.S. degree in the process equipment and control engineering from the Shenyang University of Chemical Technology, Shenyang, China, in 2017, and the M.S. degree in mechanical engineering from Shenyang Jianzhu University, Shenyang, in 2020. He is currently pursuing the Ph.D. degree in computer science and technology with Northeastern University, Shenyang. His research interests include time series data mining, deep learning, and industrial intelligence.

Xingwei Wang received the B.S., M.S., and Ph.D. degrees in computer science from Northeastern University, Shenyang, China, in 1989, 1992, and 1998, respectively, where he is currently a Professor with the College of Computer Science and Engineering. He has published more than 100 journal articles, books and book chapters, and refereed conference papers. His research interests include cloud computing and future Internet.

Ying Li received the B.S. degree in the Internet of Things from the Anyang Institute of Technology, Anyang, China, in 2017, and the M.S. degree in computer technology from Northeastern University, Shenyang, China, in 2020, where she is currently pursuing the Ph.D. degree in computer science and technology.

Bo Yi (Member, IEEE) received the B.S. and M.S. degrees in computer science from the South-Central University for Nationalities, Wuhan, China, in 2012 and 2015, respectively, and the Ph.D. degree in computer science from Northeastern University, Shenyang, China, in 2019, where he is currently a Lecturer with the College of Computer Science and Engineering. His research interests include routing and service function chain in SDN, NFV, deterministic networking, and cloud computing.



Peng Cao received the Ph.D. degree in computer application from Northeastern University, China, in 2014, where he is an Associate Professor with the Key Laboratory of Intelligent Computing in Medical Image, Ministry of Education. His research papers have been published in or accepted by journals or conference, including *Pattern Recognition*, *Computer Methods and Programs in Biomedicine*, *Medical Image Computing and Computer Assisted Intervention*, and Thirty-Sixth AAAI Conference on Artificial Intelligence in 2022. His research interests include brain network analysis and machine learning for medical data.

Min Huang (Member, IEEE) received the B.S. degree in automatic instrument, the M.S. degree in systems engineering, and the Ph.D. degree in control theory from Northeastern University, Shenyang, China, in 1990, 1993, and 1999, respectively, where she is currently a Professor with the College of Information Science and Engineering. She has published more than 100 journal articles, books, and refereed conference papers. Her research interests include modeling, and optimization for logistics and supply chain system.

Keqin Li (Fellow, IEEE) is currently a SUNY Distinguished Professor of Computer Science with the State University of New York, New Paltz, NY, USA. His current research interests include cloud computing, big data computing, and machine learning. He is among the world's top 5 most influential scientists in parallel and distributed computing based on a composite indicator of the Scopus citation database. He is currently an Associate Editor of the *ACM Computing Surveys* and the *CCF Transactions on High-Performance Computing*. He has served on the editorial board of the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, the IEEE TRANSACTIONS ON COMPUTERS, the IEEE TRANSACTIONS ON CLOUD COMPUTING, the IEEE TRANSACTIONS ON SERVICES COMPUTING, and the IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING.