Federated Meta-Learning Framework for Few-shot Fault Diagnosis in Industrial IoT

Jiao Chen*, Jianhua Tang*† and Jie Chen*
* Shien-Ming Wu School of Intelligent Engineering, South China University of Technology, China
† Pazhou Lab, Guangzhou, China
202110190459@mail.scut.edu.cn, jtang4@e.ntu.edu.sg, jiechean@163.com

***Abstract*—Learning-based mechanical fault diagnosis (FD) methods have been widely investigated in recent years. To overcome the shortages of centralized learning techniques from the perspective of data privacy and high communication overhead, federated learning (FL) is emerging as a promising method for FD. However, a large number of labeled fault data is required for the FL technique, which is not accessible in real-world industrial Internet-of-Things (IIoT) scenarios. To address the data scarcity challenge (i.e., few-shot), we propose a collaborative learning method that incorporates meta-learning into the federated learning framework. Specifically, our approach learns an effectively global meta-learner, which can quickly adapt to a new machine or a newly encountered fault category with just a few labeled examples and training iterations. Further, we theoretically analyze the convergence of the proposed algorithm in a non-convex setting. We conduct an extensive empirical evaluation of two real-world fault diagnosis datasets and they demonstrate that our proposed method achieves significantly faster convergence and higher accuracy, compared with the existing approaches.**

***Index Terms*—Federated Learning, Meta-learning, Few-shot Learning, Industrial IoT (IIoT), Intelligent Fault Diagnosis**

I. Introduction

Nowadays, the Internet-of-Things (IoT) is changing the shape of communication and has an increasing number of application scenarios. Industrial Internet of Things (IIoT), an emerging sub-paradigm of IoT, is promoting the development of Industry 4.0 [1], [2]. With many distinctive features, such as multiple services, high connectivity, low latency, and scalability, IIoT is highly successful in various industrial scenarios, e.g., anomaly detection, digital twin, and robotic system navigation. Other than these, IIoT also has great value in facilitating intelligent fault diagnosis (IFD).

Fault diagnosis (FD) is an enduring topic in industry. In recent years, with the prosperity of machine learning techniques, there are many works focused on learning-based FD. Most of them are centralized learning-based (CL) methods. Typically, the CL method collects large quantities of data from multiple institutions into a unified data node, then utilizes machine learning methods to learn a model. Despite this surge in research works, the CL method does not apply to every IIoT scenario, since (1) the required training data are mostly generated from edge devices (e.g., numerous IoT devices and gateways), which may have limited communication capability to upload the high amount of data [3]; (2) due to the growing concerns of data privacy, some data owners are reluctant from sharing their data. Federated learning (FL) [4] is a representative member in distributed machine learning paradigm. It accomplishes machine learning modeling while protecting the privacy of the clients (organizations or users) involved in the training. Several recent studies [5], [6] that demonstrate FL related methods, including self-supervised and supervised learning, perform well on mechanical FD.

However, both self-supervised learning and supervised learning methods are currently difficult to implement in real industrial FD. In particular, self-supervised learning typically requires more computational resources and network bandwidth, and thus cannot achieve real-time industrial edge intelligence, while traditional supervised learning requires a large amount of labeled training data, which is difficult to obtain in real IIoT scenarios. Finding an effective approach to overcome the limited data challenges (i.e., few-shot fault diagnosis) is still an unsolved problem in the existing research. In this paper, we aim to solve the data scarcity challenge in privacy-protected FD and propose an approach that overcomes the limitation of self-supervised learning and supervised learning.

Recall that the above learning-based FD methods all require a large number of samples to train the model, while humans can learn new concepts via a small number of examples. For example, a person who knows how to ride a bicycle can quickly master riding a motorbike without even needing a demonstration. Why can humans learn quickly and accurately with very little direct supervision? Probably because humans are adept at using experience to accelerate learning. Initialization-based meta-learning methods (e.g. MAML [7]), and its recent developments, aim to learn a good model (a model with high performance initialization weights), such that the model can solve newly encountered learning tasks after only a small number of training samples or several gradient descents. To this end, we incorporate this fast learning mechanism of meta-learning into our approach.

Concretely, we incorporate meta-learning into the federated learning framework, resulting in a **F**ew-shot **F**ault **D**iagnosis method with **Fed**erated **Meta**-learning framework (FedMeta-

This work was supported in part by the National Nature Science Foundation of China under Grant 62001168 and in part by the Foundation and Application Research Grant of Guangzhou under Grant 202102020515. The code of this work is available at https://github.com/SCUT-WUSIE-ICLab/FedMeta-FFD.

978-1-6654-3540-6/22/$31.00

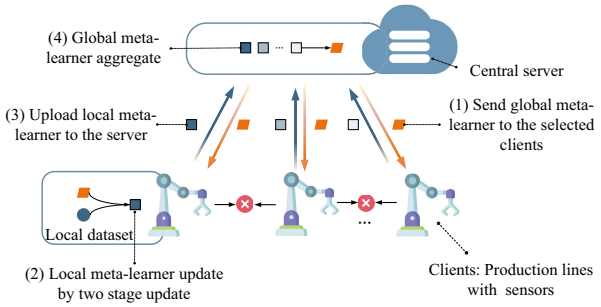

Fig. 1. The pipelines of our FedMeta-FFD framework.

FFD). We summarize the main contributions of this paper below:

(1) We propose a novel few-shot Fault Diagnosis method with a Federated Meta-learning framework (FedMeta-FFD), which relies on initialization-based meta-learning and federated learning to solve few-shot FD tasks.

(2) Theoretically, we perform a convergence analysis of the proposed FedMeta-FFD algorithm on the non-convex setting.

(3) Empirically, we conduct an extensive empirical evaluation on two real-world FD datasets and demonstrate that our method achieves significantly higher convergence and accuracy performance, compared with the other approaches.

II. Related Works

Federated learning not only makes full use of the computational power of all clients but also guarantees data privacy. Owing to that, FL has garnered much attention in IFD. Chen *et al.* [6] propose an FL method with dynamic weighted averaging for bearing fault diagnosis. Huang *et al.* [7] propose a compound fault identification and decoupling method using an IIoT-based monitoring system and achieve promising diagnostic performance.

Few-shot learning, based on the N-way K-shot [8] training setting, aims to learn the ability to adapt quickly to new tasks. Meta-learning is naturally adapted to few-shot learning and can improve model performance [9]. Li *et al.* [10] propose a meta-learning fault diagnosis method for 10-way cross-domain IFD from drive-end bearing to fan-end bearing. Feng *et al.* [11] propose a meta-learning-based method to adapt to a newly encountered fault category using a few sample.

Notations: We use calligraphy letters to represent the sets. Vectors and matrices are in the form of lowercase and uppercase bold letters, respectively. $\mathbf{I}$ means the identity matrix. $(\cdot)^T$ denotes matrix transpose. $\mathbb{E}(\cdot)$ and $\|\cdot\|$ stand for the expectation and the Euclidean norm, respectively. $\nabla(\cdot)$ represents the gradient. The d-dimensional real spaces is denoted by $\mathbb{R}^d$.

III. Method

In this section, we present our approach and the pipeline is shown in Fig. 1.

A. Federated Learning Formulation

As illustrated in Fig. 1, we consider a set of clients (organizations or users) $\mathcal{K} = \{1, \cdots, K\}$ connect to a central server through wireless communication. Clients only interact with the server but cannot exchange data with each other. Each client $k \in \mathcal{K}$ has it own dataset, denoted as $\mathcal{D}_k = \{\boldsymbol{x}_{k,j}, y_j\}_{j=1}^{|\mathcal{D}_k|}$, where $(\boldsymbol{x}_{k,j}, y_j)$ is a datapoint with $\boldsymbol{x}_{k,j}$ being the input and y_j being the class label. We further assume that the datasets in different clients follow a different distribution P_k, i.e., Non - independent and identically distributed (Non-IID).

Firstly, we aim to obtain a model that is trained over all the clients without exchanging their local data with other clients or the central server. To achieve that, we leverage federated learning as the main framework. Defining $f_k(\cdot) : \mathbb{R}^d \to \mathbb{R}$ as the loss (or network) corresponding to client k, federated learning is to obtain a model with parameter $\boldsymbol{\theta}$ by solving an optimization problem, i.e.,

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^d} f(\boldsymbol{\theta}) := \frac{1}{K} \sum_{k=1}^{K} f_k(\boldsymbol{\theta}), \tag{1}$$

where $f_k(\boldsymbol{\theta})$ represents the expected loss of client k, i.e.,

$$f_k(\boldsymbol{\theta}) := \mathbb{E}_{(\boldsymbol{x}_{k,j}, y_j) \sim P_k} \left[l_k(\boldsymbol{x}_{k,j}, y_j; \boldsymbol{\theta}) \right].$$

In machine learning, $l_k(\boldsymbol{x}_{k,j}, y_j; \boldsymbol{\theta})$ is the loss of the prediction at datapoint $(\boldsymbol{x}_{k,j}, y_j)$ measured with parameter $\boldsymbol{\theta}$. To solve problem (1), we can use the FedAvg [4], a federated learning algorithm in which the server learns a shared model by averaging the gradient updates of local clients.

B. Federated Meta-learning Framework

Note that scheme (1) develops a common output for all clients, which relies on massive labeled data to train the diagnosis model. However, due to data scarcity (i.e., few-shot) and heterogeneity, the model obtained by (1) yields poor results in the field of mechanical FD.

As such, we overcome this issue by incorporating meta-learning into (1). The goal now is not to find a model which performs well on all tasks. Instead, we explore an initialization that performs well once it is performed on a new task, possibly by one or a few steps of gradient descent, to solve the problem of limited fault data. In particular, suppose each client takes the same initialization and updates its loss function using one-step gradient descent, we rewrite the form of (1) as

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^d} F(\boldsymbol{\theta}) := \frac{1}{K} \sum_{k=1}^{K} f_k(\boldsymbol{\theta} - \alpha \nabla f_k(\boldsymbol{\theta})), \tag{2}$$

where F is the global meta-learner to be learned. The learning rate $\alpha \geqslant 0$ can be fixed as a hyperparameter or learned [12]. (2) is a joint optimization problem, which incorporates meta-learning into the federated learning framework, and is called our *Federated Meta-learning* framework.

To solve (2), we can follow the similar principles of FedAvg, i.e., the server updates the global meta-learner F by calculating

the average of the local meta-learners $F_1, \cdots, F_K$, where the local meta-learner F_k associated with client k is defined as

$$F_k(\boldsymbol{\theta}) := f_k(\boldsymbol{\theta} - \alpha \nabla f_k(\boldsymbol{\theta})). \tag{3}$$

It is worth noting that FedAvg's server performs averaging of local client-side gradient updates, whereas our Federated Meta-learning framework's server performs averaging of local meta-learner. Due to this difference, we perform an algorithm convergence analysis in Section IV.

Next, we describe the learning process of (2). At each communication round $t \in [1, T]$, the server chooses a fraction of clients with size rK ($r \in (0, 1]$) and send current global meta-learner $\boldsymbol{\theta}_t$ to these clients. Each selected client k sets initializes $\boldsymbol{\theta}^k_{t+1,0} = \boldsymbol{\theta}_t$, and performs local computations based on its local dataset. In particular, these local computations generate a local sequence $\{\boldsymbol{\theta}^k_{t+1,i}\}_{i=0}^{\tau}$ by

$$\boldsymbol{\theta}^k_{t+1,i} = \boldsymbol{\theta}^k_{t+1,i-1} - \beta \nabla F_k(\boldsymbol{\theta}^k_{t+1,i-1}), \tag{4}$$

where τ is a hyperparameter representing the number of local iterations ($1 \leqslant i \leqslant \tau$), and β is the learning rate. Then, $\boldsymbol{\theta}^k_{t+1,i}$ is used as the starting point for the next iteration at client k. Similar to most initialization based meta learning algorithms [7], in this paper, $\nabla F_k(\boldsymbol{\theta}^k_{t+1,i-1})$ is computed by two-stage updates, which are called as **inner update** and **outer update** respectively. In meta-learning, a model is first trained on a large number of tasks $\mathcal{T}$, which are generated by sampling client data. The task $\mathcal{T}_k$ of client k in the meta-training consists of a *support* set $D_S^{\mathcal{T}_k}$ and a *query* set $D_Q^{\mathcal{T}_k}$. With these definitions, we introduce **Inner update** and **Outer update** as follows:

- **Inner Update:** After pulling global meta-learner $\boldsymbol{\theta}_t$ from server, client k firstly adapts $\boldsymbol{\theta}^k_{t+1,i-1}$ on the support set $D_S^{\mathcal{T}_k}$, the parameters $\boldsymbol{\theta}^k_{t+1,i-1}$ become $\overline{\boldsymbol{\theta}}^k_{t+1,i}$ by

$$\overline{\boldsymbol{\theta}}^k_{t+1,i} = \boldsymbol{\theta}^k_{t+1,i-1} - \alpha \nabla f_k(\boldsymbol{\theta}^k_{t+1,i-1}; D_S^{\mathcal{T}_k}), \tag{5}$$

  where $\boldsymbol{\theta}^k_{t+1,0} = \boldsymbol{\theta}_t$.
- **Outer Update:** Then, the model with parameter $\overline{\boldsymbol{\theta}}^k_{t+1,i}$ is evaluated on the query set $D_Q^{\mathcal{T}_k}$, and some test loss $f_k(\overline{\boldsymbol{\theta}}^k_{t+1,i}; D_Q^{\mathcal{T}_k})$ is computed to reflect the training ability of local meta-learner $F_k(\boldsymbol{\theta}^k_t)$. Thus $\nabla F_k(\boldsymbol{\theta}^k_{t+1,i-1})$ in (4) is calculated by

$$\begin{aligned}
\nabla F_k(\boldsymbol{\theta}^k_{t+1,i-1}) &= \frac{\partial f_k(\overline{\boldsymbol{\theta}}^k_{t+1,i-1}; D_Q^{\mathcal{T}_k})}{\partial \boldsymbol{\theta}^k_{t+1,i-1}} \\
&= \frac{\partial f_k(\overline{\boldsymbol{\theta}}^k_{t+1,i-1}; D_Q^{\mathcal{T}_k})}{\partial \overline{\boldsymbol{\theta}}^k_{t+1,i-1}} \frac{\partial \overline{\boldsymbol{\theta}}^k_{t+1,i-1}}{\partial \boldsymbol{\theta}^k_{t+1,i-1}} \\
&= \nabla f_k(\boldsymbol{\theta}^k_{t+1,i-1} - \alpha \nabla f_k(\boldsymbol{\theta}^k_{t+1,i-1}); D_Q^{\mathcal{T}_k}) \\
&\quad (\mathbf{I} - \alpha \nabla^2 f_k(\boldsymbol{\theta}^k_{t+1,i-1}; D_S^{\mathcal{T}_k})).
\end{aligned}$$

Finally, for $i = \tau$, the selected clients transmit their local meta-learner $\boldsymbol{\theta}^k_{t+1,\tau}$ to the server. The server updates the global meta-learner by computing the average of the local meta-learner from these selected clients using

$$\boldsymbol{\theta}_{t+1} = \frac{1}{rK} \sum_{k=1}^{K} \boldsymbol{\theta}^k_{t+1,\tau}. \tag{6}$$

As such, once the t-th communication round ends, then the $(t+1)$-th round follows.

Algorithm 1: FedMeta-FFD Framework with MAML

Input: Learning rate α and β, and fraction of active clients r.
Output: global meta-learner F with parameter $\boldsymbol{\theta}_T$.

1 Server initializes global meta-learner $F(\boldsymbol{\theta}_0)$;
2 **for** *each round* $t = 1, 2, \cdots$ *to* T **do**
3 Sample a set $\mathcal{K}_t$ with size rK;
4 **for** *each client* $k \in \mathcal{K}_t$ *in parallel* **do**
5 Pull global meta-learner $\boldsymbol{\theta}_t$ from server, set $\boldsymbol{\theta}^k_{t+1,0} \leftarrow \boldsymbol{\theta}_t$;
6 **for** $i : 1$ *to* τ **do**
7 Sample support set $D_S^{\mathcal{T}_k}$ and query set $D_Q^{\mathcal{T}_k}$;
  `// Inner update;`
8 Compute $\overline{\boldsymbol{\theta}}^k_{t+1,i}$ by (5);
  `// Outer update;`
9 Compute $\boldsymbol{\theta}^k_{t+1,i}$ by (4);
10 **end**
11 Return $\boldsymbol{\theta}^k_{t+1,\tau}$ to server;
12 **end**
  `// Global aggregation`
13 Server update meta-learner parameters by (6) ;
14 **end**

C. Federated Meta-learning Framework for IFD Task

By solving problem (2), we obtain an initial model (global meta-learner). Once applied to the few-shot fault datasets, Eqn.(2) is termed as **F**ew-shot **F**ault **D**iagnosis method with **Fed**erated **Meta**-learning framework (FedMeta-FFD). Algorithm 1 provides the the procedures of FedMeta-FFD with MAML, i.e., FedMeta-FFD(MAML).

- Step 1: The selected clients pull the global meta-learner from the server and set local meta-learner $\boldsymbol{\theta}^k_{t+1,0} \leftarrow \boldsymbol{\theta}_t$ (line 5);
- Step 2: Each selected client updates local meta-learner through two-stage update, inner update (line 8) and outer updates (line 9), respectively.
- Step 3: Each client returns updated local meta-learner $\boldsymbol{\theta}^k_{t+1,\tau}$ to the server (line 11);
- Step 4: The server aggregates all updated local meta-learner (line 13) and then the next round follows until convergence.

In addition, we implement another version of FedMeta-FFD, i.e., FedMeta-FFD(SGD), in which clients perform training using Meta-SGD [12]. Meta-SGD supports a plug-and-play

improvement to enhance the performance of meta-learning, where the learning rate of inner update is a learned parameter.

IV. Convergence Analysis

In this section, we present the convergence analysis of algorithm 1 in a non-convex setting. In contrast to existing research works, we explicitly show that *task similarity* and the number of *local updates* have a significant impact on convergence.

Assumption 1: Consider a set of clients $\mathcal{K}$, for every $k \in \mathcal{K}$, the loss function f_k is $L_k - smooth$, i.e.,

$$\|\nabla f_k(\boldsymbol{\theta}_1) - \nabla f_k(\boldsymbol{\theta}_2)\| \leq L_k \|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|, \forall \boldsymbol{\theta}_1,\ \boldsymbol{\theta}_2 \in \mathbb{R}^d, \quad (7)$$

and also, the gradient ∇f_k is bounded by a nonnegative constant B_k, i.e.,

$$\|\nabla f_k(\boldsymbol{\theta})\| \leq B_k. \quad (8)$$

Assumption 2: Each loss function f_k is $H-strongly$ convex, i.e. for all $\theta_1, \theta_2 \in \mathcal{R}^d$,

$$\|\nabla f_k(\boldsymbol{\theta}_1) - \nabla f_k(\boldsymbol{\theta}_2)\| \geq H \|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|. \quad (9)$$

Our Assumption 1 and 2 simply follows many convergence analyses [13], [14]. $L - smooth$ and $H - strongly$ convex characterize the maximum/minimum rate of change of the corresponding gradient of the loss function. The outer update needs second order derivative in f_k Section III. Therefore, we apply a regularity condition to the Hessian of f_k.

Assumption 3: The Hessian of each f_k is $\rho_k - Lipschitz$, i.e, for all $\theta_1, \theta_2 \in \mathcal{R}^d$,

$$\|\nabla^2 f_k(\boldsymbol{\theta}_1) - \nabla^2 f_k(\boldsymbol{\theta}_2)\| \leq \rho_k \|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|, \forall \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \mathbb{R}^d. \quad (10)$$

For simplicity, we let $B := \max_k B_k, L := \max_k L_k$, and $\rho := \max_k \rho_k$.

Assumption 4: There exists nonnegtive constant $\gamma_G \geq 0$ and $\gamma_H \geq 0$ such that the variances of gradient and Hessian satisfy

$$\frac{1}{K}\sum_{k=1}^{K} \|\nabla f_k(\boldsymbol{\theta}) - \nabla f(\boldsymbol{\theta})\|^2 \leq \gamma_G^2, \quad (11)$$

$$\frac{1}{K}\sum_{k=1}^{K} \|\nabla^2 f_k(\boldsymbol{\theta}) - \nabla^2 f(\boldsymbol{\theta})\|^2 \leq \gamma_H^2. \quad (12)$$

Assumption 4 characterizes the similarity between clients. Intuitively, a small constant implies that the tasks are more similar in different clients. With Assumption 1-3, the local meta-learner $F_k(\boldsymbol{\theta})$ and their average function $F(\boldsymbol{\theta}) = (1/K)\sum_{k=1}^{K} F_k(\boldsymbol{\theta})$ are smooth and strongly convex. To complete the convergence analysis we also need the following intermediate results.

Lemma 1: Let $\psi : \mathbb{R}^d \to \mathbb{R}$ be an arbitrary function which is $L - sommth$ and $H - strongly$ convex. The Jacobian of $U(\cdot)$ is given by $U(\boldsymbol{\theta}) = \mathbf{I} - \alpha\nabla^2\psi(\boldsymbol{\theta})$. Since $H\mathbf{I} \leq \nabla^2\psi(\boldsymbol{\theta}) \leq L\mathbf{I}$ for $\forall \boldsymbol{\theta} \in \mathbb{R}^d$, we can bound be Jacobian as

$$(1 - \alpha L)\mathbf{I} \leq \nabla U(\boldsymbol{\theta}) \leq (1 - \alpha H)\mathbf{I}. \quad (13)$$

Lemma 2: If Assumptions 1-3 hold, local meta-learner F_k is $L_F - smooth$ convex and $H_F - strongly$ convex with parameter $L_F = (1 - \alpha H)^2 L + \alpha\rho B$ and $H_F = (1 - \alpha L)^2 H - \alpha\rho B$. As a consequence, the average function $F(\boldsymbol{\theta}) = (1/K)\sum_{k=1}^{K} F_k(\boldsymbol{\theta})$ is also smooth and strongly convex with parameter L_F and H_F.

Two additional metrics are required for convergence proof, i.e., the similarity between local meta-learner and global meta-learner, and the upper bound of the gradient estimation variance of the loss function. Therefore, we provide Lemma 3 and Lemma 4 next.

Lemma 3: There exists constants γ_F such that for any $\boldsymbol{\theta} \in \mathbb{R}^d$ and $k \in \mathcal{K}$

$$\frac{1}{K}\sum_{k=1}^{K} \|\nabla F_k(\boldsymbol{\theta}) - \nabla F(\boldsymbol{\theta})\|^2 \leq \gamma_F^2,$$

where

$\gamma_F^2 := 3B^2\alpha^2\gamma_H^2 + 3(1 + \alpha^2 L^2)\left[\alpha(1 - \alpha H)^2 + 8\alpha^2 L^2\right]\gamma_G^2$.

Due to the page limitation, all the detailed proof of this paper will be given in our extended version. Lemma 3 describes the similarities between local meta-learners, while establishing the connections between local meta-learners and global target, which makes it possible for analyzing the global loss.

Lemma 4: We further define a virtual index, i.e., the average of local update at round t on τ step $\boldsymbol{\theta}_{t,i} = (1/K)\sum_{k=1}^{K} \boldsymbol{\theta}_{t,i}^k$. Suppose that the conditions in Assumption 1 and 4 are satisfied, for any $0 \leq i \leq \tau$, we have

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K} \|\boldsymbol{\theta}_{t,i}^k - \boldsymbol{\theta}_{t,i}\|^2\right] \leq \sigma_\theta^2 := 35\beta^2(\tau - 1)\tau\gamma_F^2.$$

Based on Lemma 1-4, we present our main conclusion next.

Theorem 1: Suppose that Assumptions 1-4 hold, and the local update and global aggregation satisfy (4) and (6) respectively. Considering performing Algorithm 1 for T rounds with τ local updates in each round, the following fact holds true:

$$\begin{aligned}\mathbb{E}(T) &= \frac{1}{\tau T}\sum_{t=0}^{T-1}\sum_{i=0}^{\tau-1} \mathbb{E}\left[\|\nabla F(\boldsymbol{\theta}_{t+1,i})\|^2\right] \\ &\leq \frac{F(\boldsymbol{\theta}_0) - F(\boldsymbol{\theta}^*) + \beta\tau T(1 + 2L_F\beta)L_F^2\sigma_\theta^2}{\beta\tau T(1 - 2L_F\beta)}.\end{aligned}$$

We obtain the upper bound of $\mathbb{E}(T)$, which indicates the convergence rate, i.e., as $t = 1, 2, \cdots, T$, if $E(T)/T \to 0$, and hence consider the algorithm to be convergent. Note that, $\sigma_\theta^2 := 35\beta^2(\tau - 1)\tau\gamma_F^2$, and $\boldsymbol{\theta}_{t+1,i}$ is the average of local updates at time i of round t, i.e., $\boldsymbol{\theta}_{t+1,i} = (1/K)\sum_{k=1}^{K} \boldsymbol{\theta}_{t+1,i}^k$, and in particular, $\boldsymbol{\theta}_{t+1,0}^k = \boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_{t+1} = (1/K)\boldsymbol{\theta}_{t+1,\tau}^k$.

Note that σ_θ^2 positive correlation with γ_F^2, which increases with γ_H^2 and γ_G^2. Thus γ_F^2 indicates how the *task similarity* impacts the convergence performance, i.e., γ_H^2 and γ_G^2 decrease accelerates the global meta-learner gradient decrease, resulting in faster convergence. In addition, note that β is the learning rate, we can make it arbitrary and small, i.e., if given a fixed train rounds T the convergence increases with the number of *local updates* τ increasing.

V. Experiments

In this section, we study the effectiveness of FedMeta-FFD with limited fault diagnosis data (i.e., few-shot). In particular, we consider the multi-class classification problem over two real-world fault diagnosis datasets, CWRU [15] and PU [16].

The data among clients are divided as follows: (1) We use 50 clients for training (containing 75% of the data) and 50 clients for testing (containing 25% of the data). The data in each client is divided into a support set (containing 20% of the data) and a query set (containing 80% of the data). (2) To simulate Non-IID, we follow [17], where each client is configured to contain only 2 labels and the amount of data in different clients is uneven (see Table I). (3) During testing, each *local client* has the same data distribution as the training client, and each *new client* has a data distribution that is completely different from all the clients participating in the training.

A. Experiment Setup

We conducted experiments on a computer server with NVIDIA RTX 3090. All approaches are implemented based on FedML, which is a popular federated learning library [18].

1) Baselines. We compare it with the following methods:

- FedAvg: A celebrated FL algorithm [4].
- FedAvgMeta: This algorithm allows the global model obtained by the FedAvg algorithm to perform a fine-tune (one or several times) on the client's support set during the inference phase.

2) Evaluation Metrics. The study evaluates the model through the following metrics: accuracy in correlation with all data points (acc_{micro}), accuracy in correlation with all clients (acc_{macro}), and F1-score ($F1_{macro}$).

3) Model Architecture. The model receives flattened input of size (1×784). Two linear layers were used for feature extraction with a hidden layer size of 100 and an output layer size of 10 (note that for the PU dataset the output layer size is 8). The activation functions used are ReLU and Softmax.

4) Hyperparameters. In this paper, hyperparameters include the number of rounds of communication between the server and the client ($T = 300$), the number of clients involved in each round of training ($|\mathcal{K}_t| = 5$), the number of client local updates ($steps = 1$), the batchsize ($batchsize = 64$), and the learning rate of the client for different datasets and methods (we set the learning rate of FedMeta-FFD (MAML) to 0.001 for both inner update and outer update).

B. Results and Discussion

The results are in Table II. First, comparing different methods, we notice that FedAvg and FedAvgMeta perform significantly worse than FedMeta-FFD(MAML) and FedMeta-FFD(SGD). This is mainly due to 1) the scarcity of client training samples, as in Table I, there are clients with only 35 fault samples in the CWRU dataset; 2) the data for each client is Non-IID, FedAvg and FedAvgMeta obtained from the global model is difficult to adapt to new clients. Second, FedMeta-FFD(SGD) achieves the highest accuracies in different cases, increasing the final acc_{micro} by 35.52%-47.05%.

C. Ablation Experiment

Our analysis in Section IV shows that *task similarity* and *local updates* significantly affect the convergence performance. Here, we designed ablation experiments to verify one of the metrics, i.e., *local updates*. We also vary the fraction of data used as a support set for each client to examine how efficiently FedMeta-FFD adapts to new clients with few-shot data.

1) Local Updates: We set the number of local updates to be 1 and 3 for all methods on the CWRU dataset, and the results are in Fig. 2 (local client) and Fig. 3 (new client). We found that the performance of all methods increased when the number of local updates was slightly increased. Secondly, comparing the convergence of FedAvg and FedMeta-FFD, the latter has much faster and more stabler convergence with the merit of meta-learning. This validates the positive impact of meta-learning on the federated learning system.

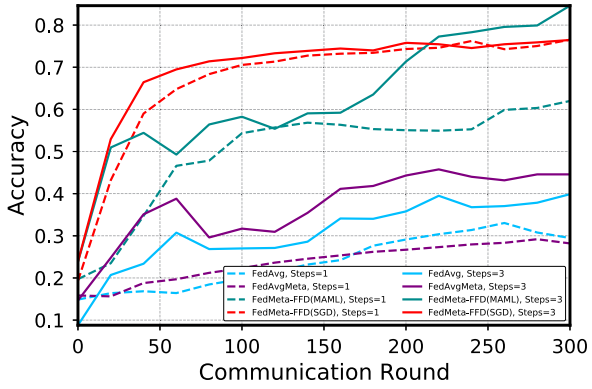

Fig. 2. acc_{micro} on local client, CWRU dataset.

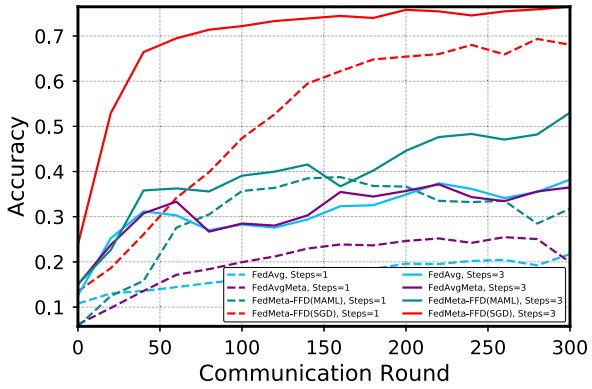

Fig. 3. acc_{micro} on new client, CWRU dataset.

2) The Fraction of Support Set: We vary the fraction $p \in \{0.2, 0.5, 0.8\}$ of data used as support set for each client to study how efficiently that FedMeta-FFD(SGD) adapts to new clients with few-shot data. The results are presented in Fig. 4 (new client). We note that when $p = 0.5$, FedMeta-FFD(SGD) achieves the highest accuracies. Compared to $p = 0.2$ (or 0.8), $p = 0.5$ indicates a more balanced amount of data in

TABLE I
STATISTICS ON CWRU AND PU DATASET

Dataset	#clients	#samples	#classes	#samples per client				#classes/client
				min	mean	std	max	
CWRU	50	28,000	10	35	560	566	2,568	2
PU	50	52,497	8	420	5,291	4,381	18,970	2

TABLE II
CLASSIFICATION RESULTS (%) IN CWRU AND PU DATASETS. BEST RESULTS PER METRICS ARE BLODFACED

		CWRU			PU		
		acc_{micro}	acc_{macro}	$F1_{macro}$	acc_{micro}	acc_{macro}	$F1_{macro}$
local client	FedAvg	29.48	21.90±21.85	18.03±17.37	36.71	22.08±16.18	18.06±12.23
	FedAvgMeta	28.19	26.37±29.00	20.25±23.10	43.28	25.83±27.56	19.81±21.49
	FedMeta-FFD(MAML)	62.00	43.21±28.78	38.64±27.63	71.13	67.43±18.22	46.70±15.32
	FedMeta-FFD(SGD)	**76.53**	**72.28±15.20**	**50.70±18.86**	**72.23**	**71.38±11.75**	**47.69±13.00**
unseen client	FedAvg	21.72	18.04±25.72	14.46±21.64	34.89	18.08±16.47	16.06±13.11
	FedAvgMeta	20.00	17.48±22.10	13.00±16.00	25.44	20.81±13.44	18.83±11.21
	FedMeta-FFD(MAML)	31.78	21.75±22.58	18.19±18.92	63.04	47.19±31.79	34.00±23.90
	FedMeta-FFD(SGD)	**68.04**	**64.59±21.02**	**46.08±19.55**	**73.28**	**71.96±14.40**	**47.00±15.75**

the support and query sets. Thus we can learn that the local meta-learner achieves better performance when the ratio of data in the support and query sets is more balanced.

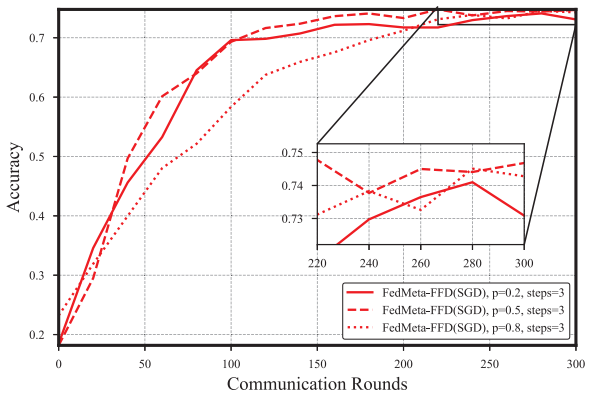

Fig. 4. acc_{micro} of FedMeta-FFD (SGD) for different p values, new client.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed the FedMeta-FFD for intelligent fault diagnosis in Industrial IoT. Our method can quickly adapt to a new machine or a newly encountered fault category with just a few labeled examples and training iterations. We did the necessary convergence analysis and build the corresponding ablation experiments. Our method outperforms existing approaches in two real-world fault diagnosis datasets, sometimes surpassing large margins. In the future, we plan to explore a robust personalized federated learning framework for intelligent fault diagnosis tasks.

REFERENCES

[1] J. Li, J. Tang, and Z. Liu, "On the data freshness for industrial Internet of Things with mobile-edge computing," *IEEE Internet Things J.*, vol. 9, no. 15, pp. 13 542–13 554, Aug. 2022.

[2] B. Yin, J. Tang, and M. Wen, "Maximizing the connectivity of wireless network slicing enabled industrial Internet-of-Things," in *Proc. IEEE GLOBECOM*, Madrid, Spain, Dec. 2021, pp. 1–6.

[3] T. Zhang, C. He, T. Ma, L. Gao, M. Ma, and S. Avestimehr, "Federated learning for Internet of Things: a federated learning framework for on-device anomaly data detection," in *Proc. ACM Conference on Embedded Networked Sensor Systems*, Coimbra, Portugal, Nov. 2021, pp. 413–419.

[4] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. AISTATS*, vol. 54, Ft. Lauderdale, FL, USA, Apr. 2017, pp. 1273–1282.

[5] W. Zhang, X. Li, H. Ma, Z. Luo, and X. Li, "Federated learning for machinery fault diagnosis with dynamic validation and self-supervision," *Knowledge-Based Systems*, vol. 213, p. 106679, Feb. 2021.

[6] J. Chen, J. Li, R. Huang, K. Yue, Z. Chen, and W. Li, "Federated learning for bearing fault diagnosis with dynamic weighted averaging," in *Proc. IEEE ICSMD*, Nanjing, China, Oct. 2021, pp. 1–6.

[7] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. ICML*, Sydney, Australia, Aug. 2017, pp. 1126–1135.

[8] O. Vinyals, C. Blundell, T. Lillicrap, k. kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," vol. 29, Dec. 2016.

[9] M. Ren, E. Triantafillou, S. Ravi, J. Snell, K. Swersky, J. B. Tenenbaum, H. Larochelle, and R. S. Zemel, "Meta-learning for semi-supervised few-shot classification," 2018. [Online]. Available: https://arxiv.org/abs/1803.00676

[10] C. Li, S. Li, A. Zhang, Q. He, Z. Liao, and J. Hu, "Meta-learning for few-shot bearing fault diagnosis under complex working conditions," *Neurocomputing*, vol. 439, pp. 197–211, Jun. 2021.

[11] Y. Feng, J. Chen, J. Xie, T. Zhang, H. Lv, and T. Pan, "Meta-learning as a promising approach for few-shot cross-domain fault diagnosis: Algorithms, applications, and prospects," *Knowledge-Based Systems*, vol. 235, p. 107646, Jan. 2022.

[12] Z. Li, F. Zhou, F. Chen, and H. Li, "Meta-SGD: Learning to learn quickly for few-shot learning," 2017. [Online]. Available: https://arxiv.org/abs/1707.09835

[13] A. Fallah, A. Mokhtari, and A. Ozdaglar, "Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach," in *Proc. NeurIPS*, Virtual, Dec. 2020, pp. 3557–3568.

[14] H. Zhao, F. Ji, Q. Li, Q. Guan, S. Wang, and M. Wen, "Federated meta-learning enhanced acoustic radio cooperative framework for ocean of things," *IEEE J. Sel. Top. Signal Process.*, vol. 16, no. 3, pp. 474–486, Apr. 2022.

[15] W. A. Smith and R. B. Randall, "Rolling element bearing diagnostics using the case western reserve university data: A benchmark study," *Mechanical systems and signal processing*, vol. 64-65, pp. 100–131, Jun. 2015.

[16] 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 Society European Conference*, vol. 3, no. 1, Jul. 2016.

[17] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with Non-IID data," 2018. [Online]. Available: https://arxiv.org/abs/1806.00582

[18] C. He, S. Li, J. So, M. Zhang, H. Wang, X. Wang, P. Vepakomma, A. Singh, H. Qiu, L. Shen, P. Zhao, Y. Kang, Y. Liu, R. Raskar, Q. Yang, M. Annavaram, and S. Avestimehr, "FedML: A research library and benchmark for federated machine learning," 2020. [Online]. Available: https://arxiv.org/abs/2007.13518