Distance-Aware Hierarchical Federated Learning in Blockchain-enabled Edge Computing Network

Xiaoge Huang, Yuhang Wu, Chengchao Liang, Qianbin Chen, *Senior Member, IEEE*, and Jie Zhang, *Senior Member, IEEE*

***Abstract*—Federated learning (FL) has been proposed as an emerging paradigm to perform privacy-preserving distributed machine learning in the Internet of Things (IoT). However, the communication overhead caused by partial model aggregations will increase the model training latency. In this paper, a multi-layer blockchain-enabled hierarchical federated learning (HFL) network is proposed for low-latency model training while ensuring data security. Meanwhile, we theoretically analyze the bottleneck of the model accuracy with the total data distance due to the imbalanced data distribution. Moreover, the mathematical expression of the model error with respect to IoT devices (IDs) association and local data distribution is provided, then the upper bound of the model error is represented by the total data distance. To further improve the learning performance, the distance-aware hierarchical federated learning (DAHFL) algorithm is investigated, which optimizes ID association strategy based on dual-distance, and allocates computing and communication resources alternatively. Finally, the working process of the blockchain-enabled HFL system is exhibited by the blockchain simulation platform and the efficiency of the proposed DAHFL algorithm is demonstrated by the simulation results.**

***Index Terms*—Hierarchical federated learning, blockchain, data distance, learning latency.**

I. Introduction

With the rapid development of the Internet of Things (IoT), the service requirements of low latency and fast response in IoTs were daily on the increase. However, these requirements cannot be met by the centralized cloud computing network. Edge computing enables computation to be performed at edge servers of the network and allows data to be downstreamed on behalf of cloud services and upstreamed on behalf of IoT devices (IDs) [1]. In addition, the rapid increase of IDs generates a large amount of data, which supports multiple smart applications, such as smart healthcare, smart home, smart transportation and smart education [2]. Machine learning (ML) is a core technology in enabling smart applications to make predictions and classifications based on data obtained from the real world using methodologies like decision trees, K-means clustering, neural networks, and more [3]. However, the centralized ML for cloud computing networks will cause potential problems, including isolated data islands, ID privacy, and data security [4]. In [5], federated learning (FL) has been proposed as a distributed learning method that utilizes localized ID data and cooperative learning approaches to address these issues. The traditional FL will face the bottleneck of high communication overhead due to the long transmission latency. Therefore, the hierarchical federated learning (HFL) network, which includes cloud layer, edge layer and ID layer, was proposed to reduce the communication overhead [6]. In the HFL network, the edge servers will aggregate edge models based on local models from IDs, and upload them to the cloud server for global aggregation. In this case, the communication overhead for local model transmission could be reduced.

Furthermore, due to the distributed deployment of the edge servers, they become vulnerable to cyber attacks and result in mutual distrust. Blockchain technology is considered as a potential solution to address this issue, which could establish a decentralized trust mode without the requirement for third-party credit endorsement [7]. Blockchain is highly consistent with FL due to its decentralized characteristics. Kim et al. proposed the combining of blockchain with FL in 2018 to ensure data security at edge servers [8]. The models stored on the blockchain are tamper-resistant, undeniable, and publicly verifiable based on the consensus protocol [9].

Moreover, the data distributions of IDs could be significantly different since they belong to different individuals and enterprises, that is, the local data are not independently and identically distributed (Non-IID) [10]. In the HFL network, the difference between the local models trained with the Non-IID data is greater than with the IID data, leading to a larger edge model difference, which will decrease the model accuracy and the global model convergence rate. Additionally, the different computing and communication capabilities among IDs will result in a considerable learning latency. To minimize the learning latency, it is essential to optimize the allocation of computing and communication resources of IDs under some constraints.

A. Related Work

1) Hierarchical Federated Learning: Compared with the traditional FL network, the HFL network could accommodate FL participants under certain constraints to achieve a high-quality global model. In [11], the author proposed an edge computing enabled HFL network that transferred the model aggregation from the cloud server to edge servers to decrease

This work is supported by the National Natural Science Foundation of China (NSFC) (61831002), and Innovation Project of the Common Key Technology of Chongqing Science and Technology Industry (Grant no.cstc2018jcyjAX0383).

Xiaoge Huang, Yuhang Wu, Chengchao Liang and Qianbin Chen are with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China. E-mail: huangxg@ cqupt.edu.cn; wu_yu_hang@yeah.net; liangcc@cqupt.edu.cn; chenqb@cqupt.edu.cn.

Jie Zhang is with the School of Communication and Information Engineering, University of Sheffield, United Kingdom. E-mail: jie.zhang@sheffield.ac.uk.

the communication overhead and improve energy efficiency. In [12], the author optimized the number of edge aggregation times and resource allocation in the HFL network to minimize the training loss and latency. Clustering HFL was proposed in [13] to address the dependence on the central server through a distributed network where the gossip protocol and the feedback mechanism were adopted in the intra-cluster and the inter-cluster communication model, respectively. However, these models required additional network bandwidth. To reduce the bandwidth consumption and storage requirements of the HFL network, a distributed hierarchical tensor depth computing model was proposed in [14], which compressed edge models of the high-dimensional tensor space into a low-dimensional subspace. In [15], the authors assumed that only partial edge servers could upload their models due to limited communication resources, and designed an efficient training algorithm that jointly optimized the IDs selection and communication resource allocation.

2) FL with Non-IID data: In the HFL network, local data of IDs were not shared, resulting in an imbalanced data distribution. In [10], the authors verified that the Non-IID local data would decrease the local model accuracy, as well as the global model, which is the bottleneck of FL with a large number of cooperative IDs. In [16], to explore the influence of the Non-IID local data on the global model accuracy, the authors derived the upper bound of the weight difference between the global model trained with the Non-IID data and the IID data. In [17], the authors emphasized that the low model accuracy was caused by the large weighted distance between local models and the global model. They decreased this distance using the gradient descent method for the edge model aggregation and the average aggregation for the global model, which improved the global model accuracy. In [18], IDs were divided into different groups according to the weighted cosine distances to improve the model accuracy of FL with the Non-IID data. Moreover, Kullback-Leibler Divergence based grouping mechanism was proposed in [19], which ensures the distribution difference between the group model and the global model is within constraints. In [20], the authors discussed a resource limited HFL network with the IID and the Non-IID data. In particular, the model accuracy with the Non-IID data could be improved by optimizing the ID association.

3) Blockchain-enable Federated Learning: The integration of blockchain technology with edge servers could facilitate trust between them and prevent cyber attacks. In [21], the authors proposed a blockchain based FL network to ensure the security of edge servers, then the superior performance, in terms of both data protection and resistance to malicious attacks was verified. However, due to the decentralized structure of the blockchain, block verification could cause significant consensus latency. To accelerate the global model convergence in the blockchain-enabled HFL network, in [22], the authors proposed an asynchronous FL scheme that only requires parts of local models for the global model aggregation. In [23], the authors introduced a framework for blockchain-enabled FL that utilizes a direct acyclic graph structure (DAG-FL). The DAG-FL update algorithm was designed to accelerate the global model convergence while preventing malicious attacks. In addition, the authors in [24] proposed a blockchain-based reputation mechanism to detect malicious IDs and improve the model accuracy. It is worth noting that malicious attacks could originate from both malicious edge servers and IDs. Therefore, to ensure the privacy of IDs, a data submission regulation based edge servers selection algorithm was introduced in [25]. In [26], the classic FL was integrated with the enterprise-level Hyperledger Fabric blockchain to improve industrial robustness, with multiple parallel channels for FL tasks.

B. Contribution

Although FL has been extensively investigated in existing literatures, there are still some open issues that need further discussion. Firstly, it is crucial to enable a high throughput blockchain technology within the HFL network to ensure the security of edge models. Meanwhile, it is difficult to theoretically analyze the impact of parameters on the global model accuracy in the HFL network. Furthermore, while existing works had achieve low latency and energy consumption in the HFL network, they tended to face performance bottlenecks. Finally, it is important to design multiple simulation platforms to exhibit the working process of the blockchain-enabled HFL network. To address these challenges, the major contributions of this paper are summarized as follows

- We propose a multi-layer blockchain-enabled HFL network consisting of cloud layer, intelligent consensus layer, edge layer and ID layer, which support low latency model training and guarantee data security.
- We theoretically analyze the bottleneck of the model accuracy with the total data distance and provide mathematical expression of the model error concerning the ID association and local data distribution. Moreover, the upper bound of the model error is represented by the total data distance. Specifically, we discuss the affect of the ID association strategy on total data distance and model accuracy.
- We develop a distance-aware hierarchical federated learning (DAHFL) algorithm to improve the learning performance, which adjusts the ID association based on the data distance and communication distance between IDs and edge servers to increase the model accuracy. Meanwhile, we optimize the computing and communication resource allocation of IDs to reduce the learning latency.
- We construct a blockchain simulation platform to exhibit the working process of the blockchain-enabled HFL network. In addition, we evaluate the performance of the DAHFL algorithm using various datasets and neural networks to demonstrate the efficiency of the proposed DAHFL algorithm.

The remainder of this paper is organized as follows. Section II describes the network model and the related works. In Section III, the model accuracy and latency of the HFL are verified and analyzed. The distance-aware hierarchical federated learning algorithm is proposed in Section IV. The simulation results are shown in Section V, followed by the conclusions in Section VI.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2023.3279983

II. Network Model

A. Network Model

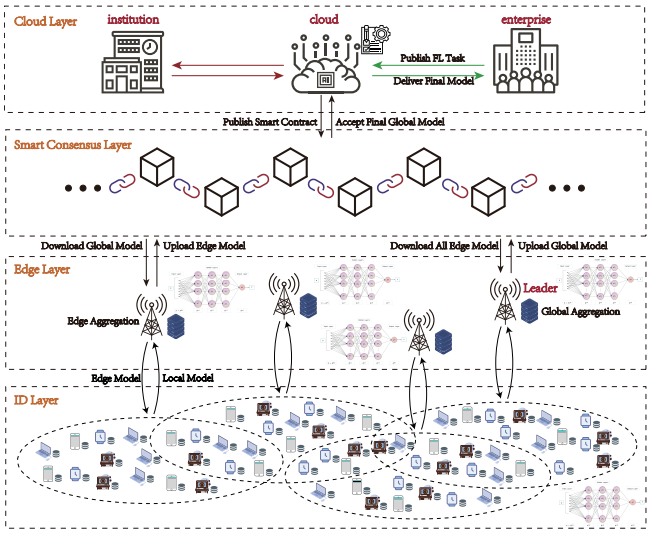


Fig. 1: Network model

Consider a multi-layer blockchain-enabled edge computing network for HFL, which consists of the cloud layer, the smart consensus layer, the edge layer and the ID layer, as illustrated in Fig. 1. The functions of these layers are described as follows

- Cloud layer: The cloud layer consists of a cloud server with powerful computing, communication, and storage capacities, and publishers are institutions or enterprises that have FL tasks such as image classification or recognition [27]. Publishers would send task information, including the publisher's identity, task description and model accuracy requirement to the cloud server. The cloud server will monitor task training epochs and deliver the global model to the publisher when it satisfies the accuracy requirement. The number of classes in the FL task is Z, for which we denote the set $\mathcal{Z} = \{1, \cdots, z, \cdots, Z\}$.
- Smart consensus layer: The smart consensus layer comprises the main-chain layer and the side-chain layer. The main-chain layer and the side-chain layer are based on the Raft consensus and the HotStuff consensus, which are used to record the global model and edge models. In the smart consensus layer, each edge server corresponds to a consensus node or the leader node and the cloud server corresponds to the unique monitoring node. In the scenario, the edge server and the consensus node are called edge nodes (EN). Each consensus node stores its edge model in the side-chain layer and verifies edge models from other consensus nodes. The leader node manages the transfer between the main-chain layer and the side-chain layer, and aggregates the global model from the edge models. The monitoring node generates and broadcasts the block of task information.
- Edge layer: The main elements of this layer are ENs with certain computing and communication capabilities. ENs are deployed close to IDs to reduce the cloud server communication overhead. The set of ENs is denoted as $\mathcal{M} = \{1, \cdots, m, \cdots, M\}$. It's noticeable that there exist overlapping areas among ENs. In each epoch, ENs will aggregate edge models from local models uploaded by associated IDs.
- ID layer: It consists of uniformly distributed IDs (e.g., computers, smartphones, wearable devices and intelligent terminals) with limited computing and communication capabilities. The set of IDs is denoted as $\mathcal{N} = \{1, \cdots, n, \cdots, N\}$. During each epoch, each ID trains the local model using its own data, and uploads it to the associated EN.

In addition, some main notations used in this paper are summarized in Table I.

TABLE I: Main Notations

Notation	Description
$\mathcal{Z}, Z$	Set of classes in the FL task, size of Z
$\mathcal{M}, M$	Set of edge nodes, size of M
$\mathcal{N}, N$	Set of IoT devices, size of N
$x_{n,m}$	Association indicator between ID n and EN m
$\mathcal{G}(\mathcal{F}_w(\mathcal{X}), \mathcal{Y})$	Loss function between $\mathcal{F}_w(\mathcal{X})$ and $\mathcal{Y}$
$\mathcal{N}_m$	Set of IDs located in the coverage of EN m
w_n^{id}, w_m^{en}	Local model of ID n, edge model of EN m
k	Number of training epoch
$\mathcal{D}_n$	Dataset of ID n used for training
$u(x, q)$	Total data distance
t_n^{cl}, t_n^{up}	Training and uploading latency of ID n
f_n, p_n	Computing resource and ransmission power of ID n
R_n	Number of CPU cycles required for one local training
h_n	Channel gain of ID n
E_n^{cl}	Training energy consumption of ID n
E_n^{up}	Uploading energy consumption of ID n
$d_{n,m}^D$	Data distance between ID n and EN m
$d_{n,m}^G$	Communication distance between ID n and EN m
$Y_{n,m}$	Dual-distance between ID n and EN m
α	Weight coefficient
$\boldsymbol{b}_n, \boldsymbol{g}_m$	Location vector of ID n and EN m, respectively
$\boldsymbol{c}_m$	The vector of class proportion in EN m
e_m	The average communication distance of IDs in EN m

B. Working Process of the Blockchain-enabled HFL Network

The working process of the blockchain-enabled HFL network is shown in Fig. 2, and the details are described by the following steps

1) Task Publication: The cloud server signs a smart contract to release the task of the publisher. Besides, the task information, such as the initial global model structure, the task training configurations, and the model accuracy requirement would be stated in the smart contract. Then the smart contract creates the genesis transaction of the main-chain with this information. Meanwhile, each EN signs this smart contract and obtains a Software Development Kit (SDK) certificate to become a consensus node.

2) Leader EN selection: Consensus ENs select a leader through the Raft consensus [28]. Then, the leader EN locks the main-chain transactions through simplified payment verification (SPV). Meanwhile, all ENs download the global model from the main-chain and broadcast it to IDs. After that, subsequent transactions will be stored in the side-chain.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2023.3279983

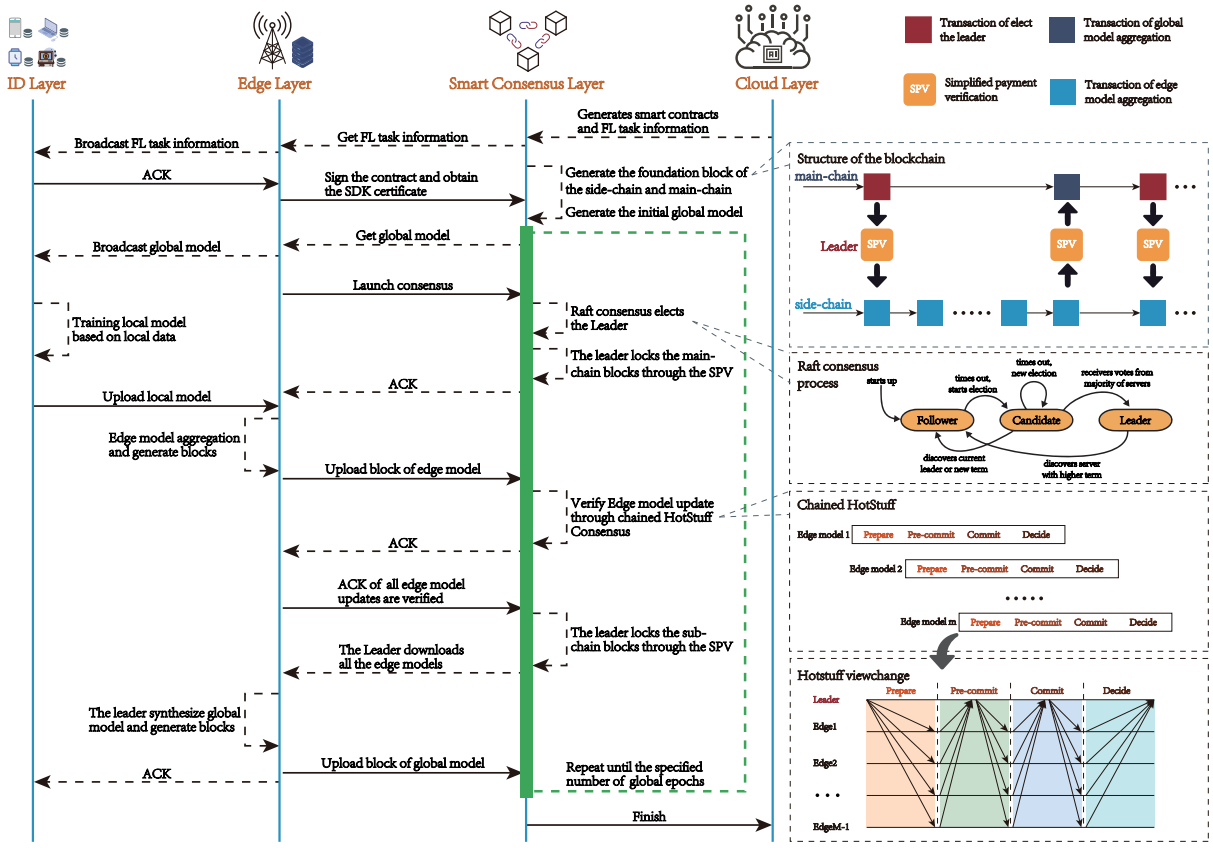


Fig. 2: Working process of the blockchain-enabled HFL network

3) Local Model Training: In each epoch, IDs could obtain the broadcasted global model from ENs and train locally. The aim of the local model training is to find an optimal function $\mathcal{F}_w : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X}$ is the local training data, and $\mathcal{Y}$ is the real label [29]. The difference between local model output $\mathcal{F}_w(\mathcal{X})$ and $\mathcal{Y}$ would reduce during the training process, then the optimal model w^* could be obtained by

$$w^* = \arg\min \mathcal{G}(\mathcal{F}_w(\mathcal{X}), \mathcal{Y}), \tag{1}$$

where $\mathcal{G}(\mathcal{F}_w(\mathcal{X}), \mathcal{Y})$ is the cross-entropy loss function that evaluates the difference between $\mathcal{F}_w(\mathcal{X})$ and $\mathcal{Y}$ [30].

Based on the stochastic gradient descent (SGD) algorithm, the local model of ID n at the k-th epoch denoted as $w_n^{id}(k)$, would be updated by

$$w_n^{id}(k+1) = w_n^{id}(k) - \eta \sum_{Z} q_n(z) \nabla \mathcal{G}(\mathcal{F}_{w_n^{id}(k)}(\mathcal{X}), \mathcal{Y}), \tag{2}$$

where η is the learning rate, $q_n(z)$ is the data distribution of ID n on class z, and $\nabla \mathcal{G}(\mathcal{F}_{w_n^{id}(k)}(\mathcal{X}), \mathcal{Y})$ is the gradient of the loss function of ID n at k-th epoch. IDs continue training the local models until they meet the model accuracy requirements. In this paper, the skewed distribution of labels is adopted to evaluate the data distribution of ID n, which is defined as the proportion of each class in the training dataset $\mathcal{D}_n$ [10], given by

$$q_n(z) = \frac{d_n^z}{D_n}, \tag{3}$$

where D_n and d_n^z is the quantity of $\mathcal{D}_n$ and the quantity of class z in $\mathcal{D}_n$, respectively.

4) Edge Model Aggregation: The local model of ID n is uploaded to the associated EN m. Then, EN m will aggregate the received local models to generate the edge model, which can be written as

$$w_m^{en}(k+1) = \frac{1}{|\mathcal{N}_m|} \sum_{n \in \mathcal{N}_m} x_{n,m} w_n^{id}(k+1), \tag{4}$$

where $w_m^{en}(k+1)$ denotes the edge model of EN m at $(k+1)$-th epoch, $\mathcal{N}_m$ denotes the set of IDs located in the coverage of EN m, $|\mathcal{N}_m|$ is the number of IDs in set $\mathcal{N}_m$ and $x_{n,m}$ is the association indicator between ID n and EN m.

5) Edge model Validation: EN m packages the transaction of the edge model to generate a block and submit it to the leader. Meanwhile, the leader launches validation transaction requests to all ENs according to the proposal from EN m. The chained HotStuff consensus is adopted in the side-chain, which includes four phases, namely, prepare phase, pre-commit phase, commit phase and decide phase [31]. The leader downloads all edge models from the side-chain after the model validation, then lock the side-chain transactions through the SPV. After that, subsequent transactions will be stored in the main-chain.

6) Global Model Aggregation: The leader updates the global model $w^G(k+1)$ at $(k+1)$-th epoch by synthesizing edge models based on the federated averaging algorithm [32], which can be expressed as

$$w^G(k+1) = \frac{1}{\sum_m |\mathcal{N}_m|} \sum_m |\mathcal{N}_m| w_m^{en}(k+1). \tag{5}$$

In addition, when the model accuracy satisfies the task requirement, the cloud server sends stop training information to ENs, and ENs broadcast the information to the covered IDs. However, during the above working process, there exist several problems threatening the efficiency of the HFL network. On the one hand, heterogeneity among IDs causes Non-IID local data, resulting in low model accuracy. On the other hand, the resource limited network lead to increased learning latency. All these concerns need to be carefully considered and well addressed in the design of the HFL network, which will be discussed in the next section.

III. Performance Analysis

In this section, we analyze the performance of the HFL network, which includes model accuracy and learning latency.

A. Model Accuracy Analysis

The model accuracy is inversely proportional to the model loss function, thus it could be represented by the model error. Define the model error as the difference between the global model of the HFL network with Non-IID data and the central model of the central learning method with IID data [10], which could be expressed as $\|w^G - w^C\|$, where w^C is the central model could be derived by the SGD algorithm

$$w^C(k+1) = w^C(k) - \eta \sum_{Z} q(z) \nabla \mathcal{G}(\mathcal{F}_{w^C(k)}(\mathcal{X}), \mathcal{Y}), \quad (6)$$

where $q(z)$ denotes the data distribution of the central learning method on class z. Each class is uniformly distributed, i.e., $q(z) = \frac{1}{Z}, l \in \mathcal{Z}$. Then, to improve the model accuracy with Non-IID data, the model error should be minimized.

1) Upper Bound of The Model Error: The following conditions are satisfied by all IDs, which has been widely used in the existing works [12], [16], [33].

- $\mathcal{G}(\mathcal{F}_w(\mathcal{X}), \mathcal{Y})$ is ρ-smooth, i.e., $\|\nabla \mathcal{G}(\mathcal{F}_{w_1}(\mathcal{X}), \mathcal{Y}) - \nabla \mathcal{G}(\mathcal{F}_{w_2}(\mathcal{X}), \mathcal{Y})\| \leq \rho \|w_1 - w_2\|$ for any w_1,w_2.
- There exists an upper bound of $\nabla \mathcal{G}(\mathcal{F}_w(\mathcal{X}), \mathcal{Y})$, i.e., $\|\nabla \mathcal{G}(\mathcal{F}_w(\mathcal{X}), \mathcal{Y})\| \leq Q$ for any w.

Then, the upper bound of the model error is provided by the following theorem.

Theorem 1. *The upper bound of the model error can be expressed as*

$$\|w^G(k+1) - w^C(k+1)\| \leq (1+\eta\rho)\|w^G(k) - w^C(k)\| + \frac{\eta Q}{N} \sum_m \Big\| \sum_{n \in \mathcal{N}_m} x_{n,m} \big(\boldsymbol{q_n} - \boldsymbol{q}\big) \Big\|. \quad (7)$$

Proof: Please refer to Appendix A. ∎

From the Theorem 1, it could be seen that the upper bound of the model error is determined by multiple parameters. ρ and Q depend on the model itself. Define the total data distance $u(\boldsymbol{x}, \boldsymbol{q})$ as

$$u(\boldsymbol{x}, \boldsymbol{q}) = \sum_m \Big\| \sum_{n \in \mathcal{N}_m} x_{n,m} \big(\boldsymbol{q_n} - \boldsymbol{q}\big) \Big\|, \quad (8)$$

which depend on the ID association strategy and the local data distribution.

2) The ID association strategy versus the total data distance: Define the class set of ID n and EN m as $\mathcal{Z}_n$ and $\mathcal{Z}_m$ respectively, where $\mathcal{Z}_m = \bigcup_{n \in \mathcal{N}_m} \mathcal{Z}_n$. Then (7) can be rewritten as

$$u = \sum_{m=1}^{M} \Big[\sqrt{\sum_{z \in \mathcal{Z}_m} \big(\sum_{n \in \mathcal{N}_m} \frac{x_{n,m} d_n^z}{D_n} - \frac{|\mathcal{N}_m|}{Z} \big)^2} + \sum_{l \notin \mathcal{Z}_m} \frac{|\mathcal{N}_m|}{Z} \Big], \quad (9)$$

where $\sum_{n \in \mathcal{N}_m} \frac{x_{n,m} d_n^z}{D_n}$ means the proportion on class z in EN m. Thus, the ID association strategy could affect the number and proportion of class z in EN m.

The data distribution could be divided into two situations, i.e., quantity-based data imbalance and proportion-based data imbalance [34]. For the quantity-based data imbalance case, each EN with a fixed number of classes but the proportion is different. For the proportion-based data imbalance, each EN with a uniform proportion of classes but the number of the class is different.

Based on the above discussions, the following lemmas could be obtained.

Lemma 1. *The total data distance is a decreased function of the number of classes in ENs.*

Proof: Assume there are two ID association strategies $\boldsymbol{x}^1$ and $\boldsymbol{x}^2$ with the uniform proportion of classes across all ENs. Then, the total data distance difference between $\boldsymbol{x}^1$ and $\boldsymbol{x}^2$ could be written as

$$\begin{aligned} & u(\boldsymbol{x}^2, \boldsymbol{q}) - u(\boldsymbol{x}^1, \boldsymbol{q}) \\ &= \sum_{m=1}^{M} \Big[\frac{|\mathcal{N}_m^2|(L - |\mathcal{Z}_m^2|)}{Z} - \frac{|\mathcal{N}_m^1|(Z - |\mathcal{Z}_m^1|)}{Z} \Big] \\ &= \sum_{m=1}^{M} \Big[\frac{Z(|\mathcal{N}_m^2| - |\mathcal{N}_m^1|) + |\mathcal{Z}_m^1||\mathcal{N}_m^1| - |\mathcal{Z}_m^2||\mathcal{N}_m^2|}{Z} \Big], \end{aligned} \quad (10)$$

where $\mathcal{N}_m^1$ and $\mathcal{Z}_m^1$ are the set of IDs and the number of classes in EN m based on the ID association strategy $\boldsymbol{x}^1$. The same as $\mathcal{N}_m^2$ and $\mathcal{Z}_m^2$. Notice that $|\mathcal{Z}_m^1| \geq |\mathcal{Z}_m^2|$, then the following in-equation can be derived

$$|\mathcal{N}_m^2| - \frac{|\mathcal{Z}_m^1|}{|\mathcal{Z}_m^2|} |\mathcal{N}_m^1| \leq |\mathcal{N}_m^2| - |\mathcal{N}_m^1|. \quad (11)$$

With the constraint of $0 \leq |\mathcal{L}_m^1| \leq Z$ and $0 \leq |\mathcal{Z}_m^2| \leq L$, we have

$$Z - |\mathcal{Z}_m^2| \frac{|\mathcal{N}_m^2| - \frac{|\mathcal{Z}_m^1|}{|\mathcal{Z}_m^2|} |\mathcal{N}_m^1|}{|\mathcal{N}_m^2| - |\mathcal{N}_m^1|} \geq 0, \quad (12)$$

which equals to

$$Z(|\mathcal{N}_m^2| - |\mathcal{N}_m^1|) + |\mathcal{Z}_m^1||\mathcal{N}_m^1| - |\mathcal{Z}_m^2||\mathcal{N}_m^2| \geq 0. \quad (13)$$

Then, we can infer that $u(\boldsymbol{x}^1, \boldsymbol{q}) - u(\boldsymbol{x}^2, \boldsymbol{q}) \leq 0$, which ends the proof. ∎

Lemma 2. *The total data distance is a decreasing function of the difference between the proportion of classes in ENs with Non-IID data and IID data.*

Proof: Assume there are two ID association strategies x_1 and x_2 but with a fixed number of classes of all ENs. Then, the

difference between $u(\boldsymbol{x}^1, \boldsymbol{q})$ and $u(\boldsymbol{x}^2, \boldsymbol{q})$ could be expressed as

$$\begin{aligned} u(\boldsymbol{x}^1, \boldsymbol{q}) - u(\boldsymbol{x}^2, \boldsymbol{q}) &= \sum_{m=1}^{M} \Big[\Big\| \sum_{n \in \mathcal{N}_m^1} x_{n,m}^1 \boldsymbol{q_n} - \frac{|\mathcal{N}_m^1|}{Z} \Big\| \\ &\quad - \Big\| \sum_{n \in \mathcal{N}_m^2} x_{n,m}^2 \boldsymbol{q_n} - \frac{|\mathcal{N}_m^2|}{Z} \Big\| \Big] \\ &= \sum_{m=1}^{M} \sqrt{\sum_{z \in \mathcal{Z}} \Big(\sum_{n \in \mathcal{N}_m^1} \frac{x_{n,m}^1 d_n^z}{D_n} - \frac{|\mathcal{N}_m^1|}{Z} \Big)^2} \\ &\quad - \sum_{m=1}^{M} \sqrt{\sum_{z \in \mathcal{Z}} \Big(\sum_{n \in \mathcal{N}_m^2} \frac{x_{n,m}^2 d_n^z}{D_n} - \frac{|\mathcal{N}_m^2|}{Z} \Big)^2}. \end{aligned} \tag{14}$$

Notice that the proportion of class difference with strategy x_1 is smaller than with strategy x_2. Then, the following inequation could be obtained.

$$\begin{aligned} &\sum_{z \in \mathcal{Z}} \Big(\sum_{n \in \mathcal{N}_m^1} \frac{x_{n,m}^1 d_n^z}{D_n} - \frac{|\mathcal{N}_m^1|}{Z} \Big)^2 \\ &- \sum_{z \in \mathcal{Z}} \Big(\sum_{n \in \mathcal{N}_m^2} \frac{x_{n,m}^2 d_n^z}{D_n} - \frac{|\mathcal{N}_m^2|}{Z} \Big)^2 < 0, \quad \forall m. \end{aligned} \tag{15}$$

Thus, $u(\boldsymbol{x}^1, \boldsymbol{q}) - u(\boldsymbol{x}^2, \boldsymbol{q}) < 0$ could be derived, which ends the proof. ∎

To sum up, optimizing the ID association could reduce the total data distance and alleviate the effect of Non-IID local data.

B. Learning Latency Analysis

In the HFL network, the learning latency comprises of several factors, including local model training latency, local model upload latency, edge model aggregation latency, edge model validation latency and global model latency. Since ENs are typically equipped with ample computation and communication resources [12], the edge and global model aggregation steps contribute less to the FL learning steps. As a result, the global model aggregation latency, edge model aggregation latency, and edge model validation latency are less significant in the learning latency. Therefore, in our analysis, we focus on the local model training latency and the local model upload latency, while ignoring the other three latencies.

1) local model training latency: The local model training latency of ID n could be calculated by

$$t_n^{cl} = \beta \log_2 \left(\frac{1}{\epsilon} \right) \frac{R_n D_n}{f_n}, \tag{16}$$

where β is a constant determined by the desired global model accuracy [35], ϵ is the expectant local model accuracy. $\beta \log_2(1/\epsilon)$ represents the necessary rounds of the local training to achieve ϵ-accuray [36]. R_n is the number of CPU cycles required for one local training, D_n is the quantity of dataset $\mathcal{D}_n$ of ID n, and f_n is the computing resource of ID n measured by the CPU frequency.

2) local model upload latency: Orthogonal frequency-division multiple access (OFDMA) is adopted in uplink transmission, and the transmission rate of ID n is given as

$$r_n = B \log_2 \Big(1 + \frac{p_n h_n}{N_0 B} \Big), \tag{17}$$

where B is the bandwidth, which is equal for each ID. p_n is the transmission power of ID n, h_n is the channel gain of ID n, and N_0 is the noise power. Denote the size of the local model as S. Then, the local model upload latency from ID n to EN m could be expressed as

$$t_n^{up} = \frac{S}{r_n} = \frac{S}{B \log_2 \Big(1 + \frac{p_n h_n}{N_0 B} \Big)}. \tag{18}$$

Additionally, based on the synchronous aggregation mechanism, the learning latency is decided by the maximum latency among all IDs, which could be expressed as

$$t(\boldsymbol{f}, \boldsymbol{p}) = \max_{n \in \mathcal{N}} \{t_n\} = \max_{n \in \mathcal{N}} \{t_n^{cl} + t_n^{up}\}. \tag{19}$$

C. Problem Formulation

According to the above analysis, the total data distance and the learning latency are affected by the ID association and resource allocation. Therefore, the learning utility U, which is the weighted sum of the total data distance and the learning latency could be adopted to evaluate the learning performance, given by

$$U(\boldsymbol{x}, \boldsymbol{f}, \boldsymbol{p}) = \alpha u + (1 - \alpha) t, \tag{20}$$

where α is the weight coefficient determined by the requirement of the task.

The energy consumption for training the local model of ID n could be written as

$$E_n^{cl} = t_n^{cl} f_n \times \gamma f_n^2 = \gamma \rho \log_2 \left(\frac{1}{\epsilon} \right) R_n D_n f_n^2, \tag{21}$$

where γ is a constant that depends on the effective switched capacitance of ID n [37]. Moreover, the energy consumption for uploading local model from ID n to EN m is given as

$$E_n^{up} = p_n \times t_n^{up} = \frac{S p_n}{r_n}. \tag{22}$$

To improve the learning performance, we jointly optimize the ID association, computing and communication resource allocation under multiple constraints. The optimization problem could be formulated as follows

$$\begin{aligned} \boldsymbol{P1}: \min_{\boldsymbol{x}, \boldsymbol{f}, \boldsymbol{p}} \ & \alpha \sum_{m} \Big\| \sum_{n \in \mathcal{N}_m} x_{n,m} (\boldsymbol{q_n} - \boldsymbol{q}) \Big\| \\ & + (1 - \alpha) \max_{n \in \mathcal{N}} \Big\{ \beta \log_2 \left(\frac{1}{\epsilon} \right) \frac{R_n D_n}{f_n} + \frac{S}{r_n} \Big\}, \end{aligned} \tag{23}$$

$$\text{s.t.} \ \sum_{n=1}^{N} \Big[\gamma \rho \log_2 \left(\frac{1}{\epsilon} \right) R_n D_n f_n^2 + \frac{S p_n}{r_n} \Big] \leq E, \tag{24}$$

$$r_{\min} \leq r_n, \tag{25}$$

$$0 \leq f_n \leq f_n^{\max}, \tag{26}$$

$$0 \leq p_n \leq p_n^{\max}, \tag{27}$$

$$x_{n,m} \in \{0, 1\}, \tag{28}$$

$$\sum_{m \in \mathcal{M}} x_{n,m} = 1, \quad \forall n, \tag{29}$$

where E is the maximum energy consumption of the network, (24) is the energy consumption constraint, (25) is the minimum required transmission rate of ID n, (26) and (27) are the computing frequency and transmission power constraints of ID n, and (28) and (29) are the ID association constraints. The optimization problem (23) is a mixed integer nonlinear programming problem, which is difficult to obtain the optimal solution directly. Thus, this problem could be decomposed into two subproblems by separating integer variables from continuous variables.

IV. Distance-Aware Hierarchical Federated Learning Algorithm

In this section, we design the DAHFL algorithm, which optimizes the ID association strategy based on dual-distance between IDs and resource allocation to improve the learning performance of the HFL network.

A. Dual-distance based ID association strategy

In the scenario, we assume that ENs could obtain both the location information and the data distribution of IDs. Based on the analysis of the model error, the learning latency and the total data distance of the HFL network could affect the learning performance. To minimize the learning utility, ID n should connect EN m with the smallest communication and data distance.

Denote $\boldsymbol{b}_n$ as the location vector of ID n. $\boldsymbol{c}_m$, $\boldsymbol{g}_m$ and e_m are the vector of class proportion in EN m, the location vector of EN m and the average communication distance of IDs in EN m, respectively. To unify the scale of two distances, we normalize $\boldsymbol{q}_n$,$\boldsymbol{b}_n$ and $\boldsymbol{g}_m$ by the following function

$$\boldsymbol{X} = \frac{\boldsymbol{X} - \min \boldsymbol{X}}{\max \boldsymbol{X} - \min \boldsymbol{X}}, \tag{30}$$

where $\boldsymbol{X}$ indicates $\boldsymbol{q}_n$,$\boldsymbol{b}_n$ or $\boldsymbol{g}_m$. Then, the details of the dual-distance based ID association strategy are as follows

Step 1. Initialize the association between EN m and ID n to generate the $\boldsymbol{c}_m^{(0)}$ and $e_m^{(0)}$ based on (35) and (36).

Step 2. Calculate the data distance $d_{n,m}^D$, communication distance $d_{n,m}^G$ and dual-distance $Y_{n,m}$ between ID n and EN m, given as

$$d_{n,m}^D = \|\boldsymbol{q}_n + \boldsymbol{c}_m^{(\tau)} - \frac{|\mathcal{N}_m^{(\tau)}| + 1}{Z}\|, \tag{31}$$

$$d_{n,m}^G = \|\boldsymbol{b}_n - \boldsymbol{g}_m\|, \tag{32}$$

$$Y_{n,m} = \alpha d_{n,m}^D + (1-\alpha) d_{n,m}^G, \tag{33}$$

where τ denotes the iteration number.

*Step 3.*ID n could connect to EN m^* with the minimum dual-distance, written as

$$m^* = \arg\min Y_{n,m}. \tag{34}$$

Then, let $x_{n,m^*}^{(\tau)} = 1$, $x_{n,m}^{(\tau)} = 0$ $(m \neq m^*)$ and $\mathcal{N}_{m^*}^{(\tau)} = \mathcal{N}_{m^*}^{(\tau)} \bigcup n$.

Step 4. Update the class proportion vectors and average communication distance of IDs in EN m by the following

$$\boldsymbol{c}_m^{(\tau+1)} = \frac{1}{|\mathcal{N}_m^{(\tau)}|}\Big(\sum_{n \in \mathcal{N}_m^{(\tau)}} \boldsymbol{q}_n - \frac{|\mathcal{N}_m^{(\tau)}|}{Z}\Big), \tag{35}$$

$$e_m^{(\tau+1)} = \frac{1}{|\mathcal{N}_m^{(\tau)}|} \sum_{n \in \mathcal{N}_m^{(\tau)}} \|\boldsymbol{b}_n - \boldsymbol{g}_m\|. \tag{36}$$

Repeat Step 2 to Step 4 until $\tau \leq \tau_{\max}$. Then, the output is the optimal ID association strategy.

B. Communication and Computing Resource Allocation

Based on the solutions of the dual-distance based ID association strategy, $\boldsymbol{x}$ is obtained, and the original problem could be simplified to the resource allocation optimization problem of IDs, written as

$$\begin{aligned} \boldsymbol{P2}: \min_{f_n, p_n} \max_{n \in \mathcal{N}} & \left\{ \beta \log_2\left(\frac{1}{\epsilon}\right) \frac{R_n D_n}{f_n} + \frac{S}{r_n} \right\}, \\ \text{s.t. } & (24), (25), (26), (27). \end{aligned} \tag{37}$$

However, the problem ***P2*** remains non-convex since the non-convex constraint (24). To reduce the computation complexity, it could be decoupled into two subproblems, namely, transmission power optimization and computing resource optimization, and adopted the alternating optimization algorithm to approach sub-optimal solutions iteratively.

1) Transmission Power Optimization: Transmission power of ID n could be represented as $p_n = \frac{N_0 B}{h_n}(2^{\frac{r_n}{B}} - 1)$. With a fixed f_n, the optimization problem ***P2*** could be converted to

$$\begin{aligned} \boldsymbol{P3}: \min_{r_n} \quad & T(r_n), \\ \text{s.t. } & \sum_{n=1}^{N} \gamma C_n f_n^2 + \frac{S N_0 B}{r_n h_n}(2^{r_n B} - 1) \leq E, \\ & \frac{N_0 B}{h_n}(2^{\frac{r_n}{B}} - 1) \leq p_n^{\max}, \\ & \frac{C_n}{f_n} + \frac{S}{r_n} \leq T, \\ & r_n \leq r_{\min}, \qquad \forall n, \end{aligned} \tag{38}$$

where $C_n = \beta \log_2(\frac{1}{\epsilon}) R_n D_n$. It is convex with respect to r_n, which could be solved by the Lagrange dual method, given by

$$\begin{aligned} \mathcal{L}_{r_n}(\{r_n\}, \theta, \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) &= T + \sum_n \lambda_n \Big(\frac{C_n}{f_n} + \frac{S}{r_n} - T \Big) \\ &+ \sum_n \mu_n (r_n - r_{\min}) + \sum_n \nu_n [\frac{N_0 B}{h_n}(2^{\frac{r_n}{B}} - 1) - p_n^{\max}] \\ &+ \theta \Big\{ \sum_{n=1}^{N} \gamma C_n f_n^2 + \frac{S N_0 B}{r_n h_n}(2^{\frac{r_n}{B}} - 1) - E \Big\}, \end{aligned} \tag{39}$$

where λ_n, μ_n, θ and ν_n are non-negative Lagrange multipliers. According to Karush-Kuhn-Tucker (KKT) conditions, the necessary and sufficient conditions can be written as

$$\frac{\partial \mathcal{L}_{r_n}(\{r_n\},\theta,\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu})}{\partial r_n} = \mu_n - \frac{S\lambda_n}{r_n^2} + \frac{\nu_n N_0 \ln(2)}{h_n} 2^{\frac{r_n}{B}},$$
$$+ \theta\Big[\frac{N_0 S \ln(2) 2^{\frac{r_n}{B}}}{h_n r_n} - \frac{N_0 B S (2^{\frac{r_n}{B}}-1)}{h_n r_n^2}\Big] = 0, \tag{40}$$

$$\theta\Big\{\sum_{n=1}^{N} \gamma C_n f_n^2 + \frac{S N_0 B}{r_n h_n}(2^{\frac{r_n}{B}}-1) - E\Big\} = 0, \tag{41}$$

$$\nu_n[\frac{N_0 B}{h_n}(2^{\frac{r_n}{B}}-1) - p_n^{\max}] = 0, \tag{42}$$

$$\lambda_n\Big(\frac{C_n}{f_n} + \frac{S}{r_n} - T\Big) = 0, \tag{43}$$

$$\mu_n(r_n - r_{\min}) = 0, \tag{44}$$

$$\lambda_n, \mu_n, \nu_n, \theta \geq 0, \qquad \forall n. \tag{45}$$

The optimal value r_n^* could be derived by solving equation (40) using the bisection method. The subgradient search algorithm is used to calculate the non-negative Lagrange multipliers λ_n, μ_n, ν_n and θ, given by

$$\theta^{(i+1)} = \big[\theta^{(i)} + \varepsilon_1\big(\sum_{n=1}^{N} \gamma C_n f_n^2 + \frac{S N_0 B}{r_n h_n}(2^{\frac{r_n}{B}}-1) - E\big)\big]^+, \tag{46}$$

$$\nu_n^{(i+1)} = \big[\nu_n^{(i)} + \varepsilon_2\big(\frac{N_0 B}{h_n}(2^{\frac{r_n}{B}}-1) - p_n^{\max}\big)\big]^+, \tag{47}$$

$$\lambda_n^{(i+1)} = \big[\lambda_n^{(i)} + \varepsilon_3\big(\frac{C_n}{f_n} + \frac{S}{r_n} - T\big)\big]^+, \tag{48}$$

$$\mu_n^{(i+1)} = \big[\mu_n^{(i)} + \varepsilon_4(r_n - r_{\min})\big]^+, \tag{49}$$

where $[x]^+ \triangleq \max(0,x)$, ε_1, ε_2, ε_3 and ε_4 represent step sizes, i denotes the iteration number.

2) Computing Resource Optimization: Based on the optimal solution of the transmission power, the problem ***P2*** could be rewritten as the computing resource optimization problem, given as

$$\begin{aligned} \boldsymbol{P4}: \min_{f_n} \quad & T(f_n), \\ \text{s.t.} \quad & \sum_{n=1}^{N} \gamma C_n f_n^2 + \frac{S p_n^*}{r_n^*} \leq E, \\ & \frac{C_n}{f_n} + \frac{S}{r_n^*} \leq T, \\ & f_n \leq f_n^{\max}, \qquad \forall n, \end{aligned} \tag{50}$$

which is convex with respect to f_n. Then, the Lagrange dual method could be used to obtain the optimal solutions, written as

$$\mathcal{L}_{f_n}(\{f_n\},\pi,\boldsymbol{\xi},\boldsymbol{\sigma}) = T + \pi\Big(\sum_{n=1}^{N} \gamma C_n f_n^2 + \frac{S p_n^*}{r_n^*} - E\Big)$$
$$+ \sum_n \xi_n\Big(\frac{C_n}{f_n} + \frac{S}{r_n^*} - T\Big) + \sum_n \sigma_n(f_n - f_n^{\max}). \tag{51}$$

Similar to ***P3***, the optimal solution of f_n^* could be derived through KKT conditions, given as

$$f_n^* = \sqrt[3]{-\frac{\phi}{2} + \sqrt{\frac{\phi^2}{4} + \frac{\zeta^3}{27}}} + \sqrt[3]{-\frac{\phi}{2} - \sqrt{\frac{\phi^2}{4} + \frac{\zeta^3}{27}}}, \tag{52}$$

where $\phi = -\frac{\sigma_n^2}{12\theta^2 C_n^2 \gamma^2}$ and $\zeta = \frac{2\sigma_n^3 - 108\xi_n C_n^3 \pi^2 \gamma^2}{216\pi^3 C_n^3 \gamma^3}$. The Lagrange multipliers π, ξ_n and σ_n could be updated by

$$\pi^{(i+1)} = \big[\pi^{(i)} + \varepsilon_5\big(\sum_{n=1}^{N} \gamma C_n f_n^2 + \frac{S p_n^*}{r_n^*} - E\big)\big]^+, \tag{53}$$

$$\xi_n^{(i+1)} = \big[\xi_n^{(i)} + \varepsilon_6\big(\frac{C_n}{f_n} + \frac{S}{r_n^*} - T\big)\big]^+, \tag{54}$$

$$\sigma_n^{(i+1)} = \big[\sigma_n^{(i)} + \varepsilon_7(f_n - f_{\min})\big]^+, \tag{55}$$

where ε_5, ε_6 and ε_7 represent step sizes.

Algorithm 1 Distance-Aware Hierarchical Federated Learning (DAHFL) Algorithm.

Input: Weight coefficient α, class proportion vectors $\boldsymbol{q}$, location vectors $\boldsymbol{b}$ and $\boldsymbol{g}$, maximum iteration number $\tau_{\max}$ and I.

Output: U^*, f_n^*, p_n^* and $x_{n,m}^*$, $n \in \mathcal{N}$, $m \in \mathcal{M}$

Normalize $\boldsymbol{q}_n$, $\boldsymbol{b}_n$ and $\boldsymbol{g}_m$ according to (30).
(I). ID association optimization
Random initialize the association between ENs and IDs.
for $\tau = 1 : \tau_{\max}$ **do**
  $x_{n,m} \to 0, \forall n \in \mathcal{N}, m \in \mathcal{M}$.
  for $n = 1 : N$ **do**
    for $m = 1 : M$ **do**
      $d_{n,m}^D = \|\boldsymbol{q}_n + \boldsymbol{c}_m^{(\tau)} - \frac{|\mathcal{N}_m^{(\tau)}|+1}{L}\|$.
      $d_{n,m}^G = \|\boldsymbol{b}_n - \boldsymbol{g}_m\|$.
      $Y_{n,m} = \alpha d_{n,m}^D + (1-\alpha) d_{n,m}^G$.
    end for
    $m^* = \arg\min Y_{n,m}$.
    $x_{n,m^*}^{(\tau)} = 1$, $x_{n,m}^{(\tau)} = 0$ $(m \neq m^*)$ and $\mathcal{N}_{m^*}^{(\tau)} = \mathcal{N}_{m^*}^{(\tau)} \bigcup n$.
  end for
  for $m = 1 : M$ **do**
    $\boldsymbol{c}_m^{(\tau+1)} = \frac{1}{|\mathcal{N}_m^{(\tau)}|}\Big(\sum_{n \in \mathcal{N}_m^{(\tau)}} \boldsymbol{q}_n - \frac{|\mathcal{N}_m^{(\tau)}|}{L}\Big)$.
    $e_m^{(\tau+1)} = \frac{1}{|\mathcal{N}_m^{(\tau)}|} \sum_{n \in \mathcal{N}_m^{(\tau)}} \|\boldsymbol{b}_n - \boldsymbol{g}_m\|$.
  end for
end for
(II). Resource allocation optimization
for $i = 1 : I$ **do**
  Calculate r_n^* by bisection method.
  Update Lagrange multipliers θ, ν_n, λ_n and μ_n by (46), (47), (48), and (49).
  Calculate f_n^* according to (52).
  Update Lagrange multipliers π, ξ_n and σ_n by (53), (54), and (55).
end for

To sum up, the DAHFL algorithm is summarized in Algorithm 1. For the dual-distance based ID association strategy, the total complexity of calculating $d_{n,m}^D$ and $d_{n,m}^G$ for IDs is

$\mathcal{O}(NM)$. In addition, $\boldsymbol{c}_m$ and e_m converge under $\tau_{\max}$ iterations. Thus, the complexity of the ID association optimization is $\mathcal{O}(NM\tau_{\max})$. For the resource allocation optimization, r_n^* and f_n^* could be derived after I iterations, the complexity is $\mathcal{O}(NI)$. In summary, the total complexity of the DAHFL algorithm is $\mathcal{O}(NM\tau_{\max} + NI)$.

V. Simulation Results

A. Simulation Parameters

In the simulations, we consider M ENs and N IDs randomly distributed within a rectangular area of size 1000m×800m. Multiple candidates ENs are available for each ID. All IDs locally train their models based on the global model at each epoch. Then the local model is uploaded to the associated EN for edge and global aggregations. Two classic neural networks, i.e., LeNet5 and LSTM, and two classic datasets, i.e., Traffic Signs Preprocessed (TSP)[1] (86989 training data and 12630 test data) and Fashion-MNIST (FMNIST)[2] (50000 training data and 10000 test data) are adopted to verify the efficiency of the DAHFL algorithm. Based on 3GPP TS. 38.901, the path loss model of the large-scale fading is given by $28.0 + 22\log_{10}(d_{3D}) + 20\log_{10}(f_c)$, where d_{3D} is the 3D separation distance between ENs and IDs in meters and f_c is the central frequency in GHz [38]. The main parameters adopted in simulations are listed in Table II.

TABLE II: Simulation Parameters

Parameters	Values
Bandwidth of the uplink channel B	20MHz
EN coverage radius	300m
Maximum transmission power $p^{\max}$ of IDs	23dBm
Noise power density	-154dBm/Hz
Switch capacitance coefficient γ	10^{-29}
Required number of local training $\beta\log_2(\frac{1}{\epsilon})$	$1 \times \log_2(\frac{1}{0.01})$
Maximum computing resource $f^{\max}$ of IDs	4GHz
CPU cycles for computing one sample data R_n	$\mathcal{U}(3,4) \times 10^4$
Learning rate η	0.001
Model size S	5×10^5Bytes
The central frequency of the path loss model f_c	6GHz

To create visual simulation results, we establish a blockchain simulation platform based on the open source projects of FISCO[3], WeCross[4] and WeBASE[5] developed by WeBankBlockchain team. FISCO is a reliable and efficient enterprise-level financial consortium blockchain platform for storing the main-chain and the side-chain transactions, WeCross is a cross-chain collaboration platform for the main-chain and the side-chain interactions, and WeBASE is a set of common components built between blockchain applications and FISCO nodes for visual management and presentation of multiple information in the FISCO. The interactive control of blockchain and FL is based on the Python SDK.

[1]https://www.kaggle.com/datasets/valentynsichkar/traffic-signs-preprocessed

[2]https://www.kaggle.com/datasets/zalando-research/fashionmnist

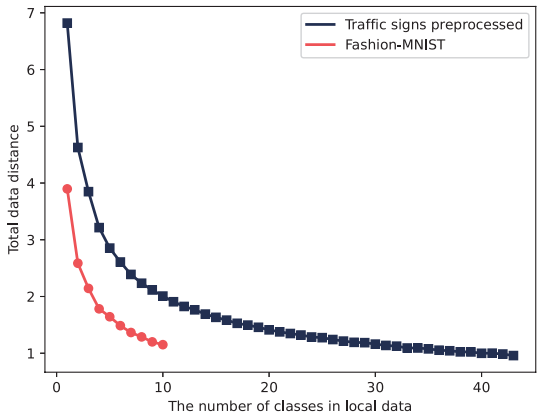
[3]https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest

[4]https://wecross.readthedocs.io/zh_CN/latest

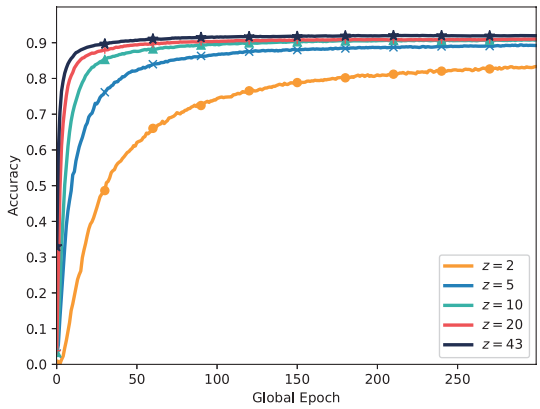
[5]https://webasedoc.readthedocs.io/zh_CN/latest

B. Simulation Results

Fig. 3 depicts the impact of the data distribution on the model accuracy with TSP, FMNIST and LeNet5. In the simulations, local data including z classes and IDs connect to ENs randomly. From Fig. 3a, the total data distance decreases with the increasing number of classes of local data in both datasets. It is noteworthy that the total data distance is not equal to zero, even IDs have all data classes, which is caused by the inappropriate ID association strategy. In Fig. 3b, the number of classes $z = [2, 5, 10, 20, 43]$, and the corresponding total data distance are $u = [4.623, 2.824, 2.013, 1.412, 0.925]$, respectively. It can be seen that the model accuracy is an increasing function of the number of classes. This finding could be attributed to the total data distance decreases as the number of classes increases, which could reduce the model error. Thus, optimizing the ID association is crucial in decreasing the model error.



(a) Number of classes in the local data versus the total data distance.



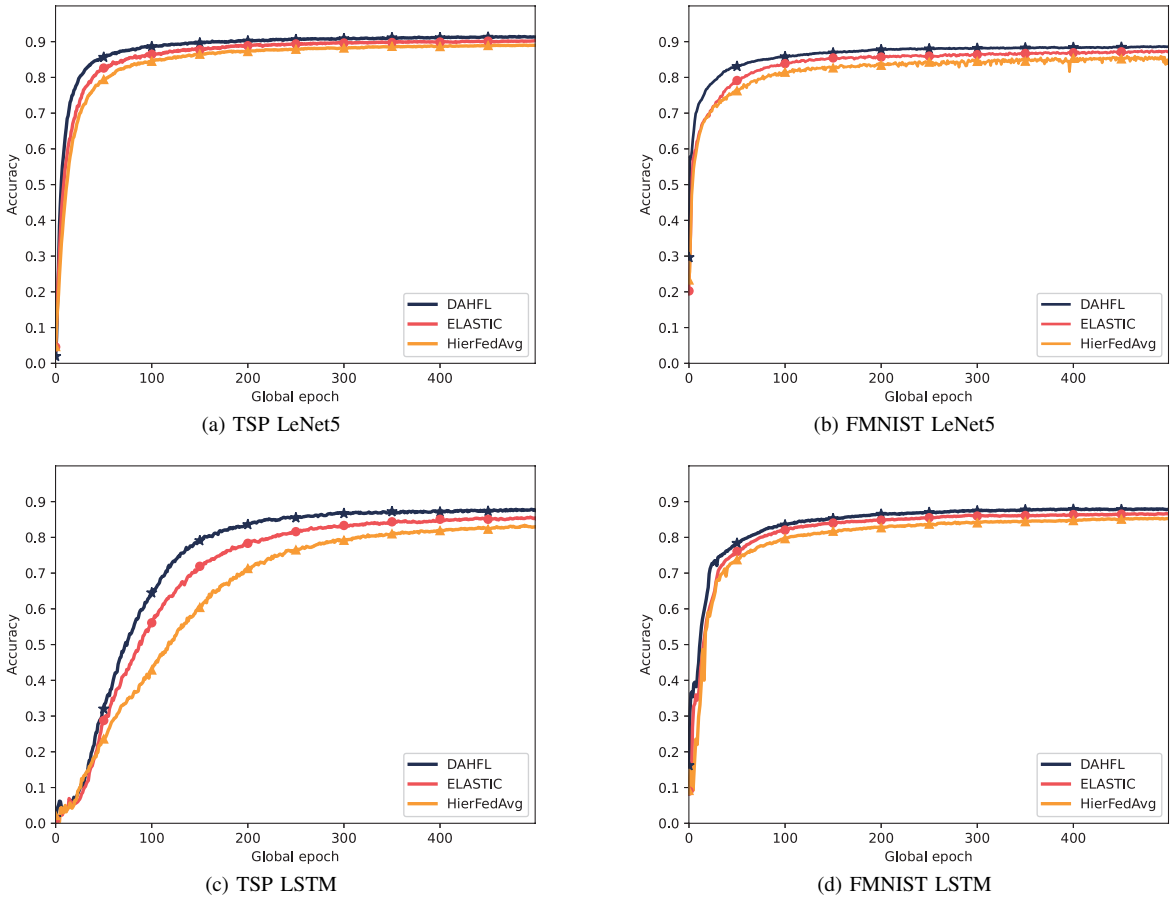
(b) Number of classes versus the model accuracy.

Fig. 3: Impact of data distribution on the model accuracy.

In Fig. 4, to verify the efficiency of the DAHFL algorithm, two comparison algorithms, namely, the HierFedAvg and the ELASTIC are used in the simulation [39], [40] with two models (LeNet5 and LSTM) and two datasets (TSP and FMNIST), respectively.

- *HierFedAvg algorithm:* IDs would connect ENs with the highest uplink channel gain.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2023.3279983



(a) TSP LeNet5 (b) FMNIST LeNet5

(c) TSP LSTM (d) FMNIST LSTM

Fig. 4: Learning accuracy comparisons with different ID association strategies

- *ELASTIC algorithm:* If the training process of IDs is over time, IDs would not connect with any EN, which could decrease the energy consumption and the learning latency.

The number of class in local data of IDs is 5. From the results, the DAHFL algorithm achieves the highest model accuracy, while the ELASTIC and the HierFedAvg algorithm produce lower model accuracy. The reason is that the DAHFL optimizes the ID association to increase the number and balance the proportion of classes in ENs, which reduces the total data distance and the model error, thereby increases the model accuracy. The model accuracy of different algorithms at the 500-th epoch is shown in Table III, which demonstrates the efficiency of the DAHFL algorithm with Non-IID data.

TABLE III: Learning Accuracy of Different Algorithms

Algorithms	Dataset and Net			
	TSP		FMNIST	
	LeNet5	LSTM	LeNet5	LSTM
DAHFL	91.3%	87.8%	88.6%	87.8%
ELASTIC	89.9%	85.2%	87.3%	86.8%
HierFedAvg	89%	83%	85.3%	85.9%

Fig. 5 shows the effect of weight coefficient α on the model accuracy with LSTM and TSP, where weight coefficients are based on application scenarios, such as 0.9, 0.6, 0.4 and 0.1 corresponding to the medical image classification, the traffic forecasting in IoT, the steering angle forecasting and the failure detection in IoV, respectively. It is worth noting that we validate the universality of the DAHFL algorithm to different scenarios by different trade-offs between model accuracy and learning latency. However, personalized DAHFL algorithms, such as those combine meta-learning or knowledge distillation, should be developed to address the specific task requirements, training methods, and data characteristics in different scenarios. In the figure, the model accuracy of the DAHFL algorithm is higher with a larger weight coefficient,

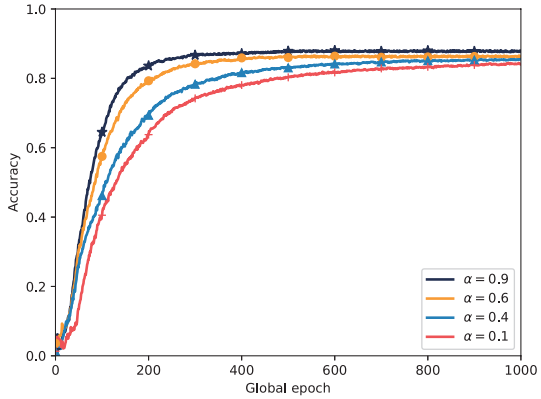


Fig. 5: Effect of weight coefficient α on the model accuracy

which are 87.9%, 86.2%, 85.5% and 84.1%, respectively. This is because a higher weight coefficient could result in a smaller total data distance but higher learning latency in the DAHFL algorithm, indicating that it could balance the total data distance and the learning latency effectively.

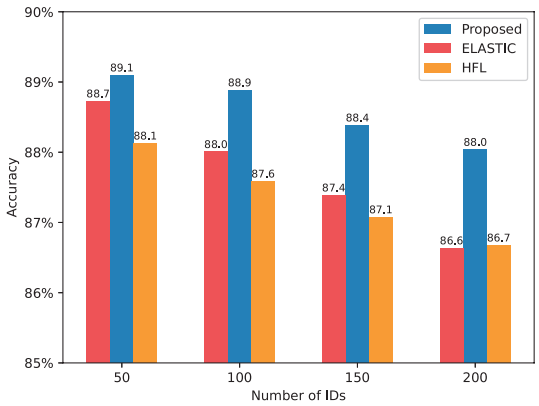


Fig. 6: Learning accuracy versus the number of IDs for different algorithms

Fig. 6 describes the model accuracy versus the number of IDs for different algorithms with LeNet5 and FEMNIST at the 300-th epoch. The training samples in the dataset are equally allocated to IDs, and the local data of each ID includes 2 classes. It is clear that the model accuracy of these algorithms decreases with the increase of IDs number. This could be attributed to the training samples of local data are decreasing and the total data distance is increasing with the increase of the number of IDs, resulting in a low accuracy of the local model and a high model error. Additionally, compared with the HFL and the ELASTIC algorithms, the DAHFL algorithm achieves better performance with the number of IDs increases, owing to the appropriate ID association strategy.

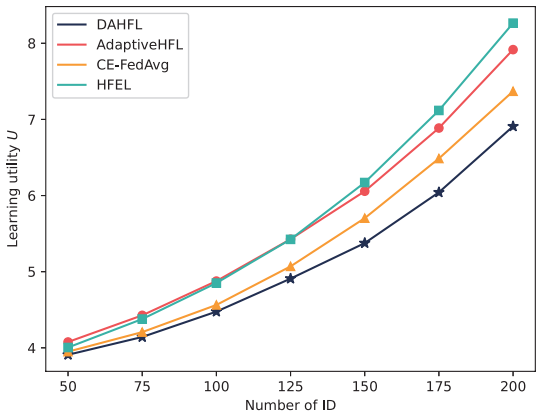


Fig. 7: Number of IDs versus learning utility U for different algorithms

In Fig. 7, we present the learning performance of various algorithms with a weight coefficient $\alpha = 0.8$. The comparison algorithms are from the existing works [11], [12] and [41], respectively.

- *HFEL:* Execute ID association with the Max-SNR strategy, and IDs allocate computing and communication resources equally.
- *AdaptiveHFL:* Optimize the interval of edge aggregation, and allocate computing and communication resources to have an equal energy consumption for all IDs.
- *CE-FedAvg:* Execute ID association randomly, and optimize computing and communication resource allocation.

It is obvious that the learning utility U is an increasing function of the number of IDs. When the number of IDs increases, the total data distance also increases, and the energy consumption constraints would result in increasing learning latency. In addition, compared with the other three algorithms, the DAHFL algorithm could achieve superior learning performance, which demonstrates that the ID association strategy and resource allocation algorithm will affect the learning performance.

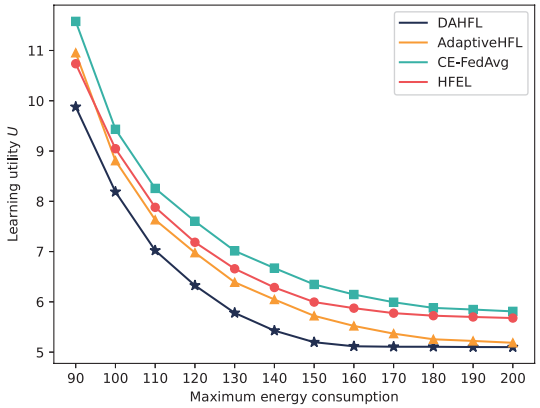
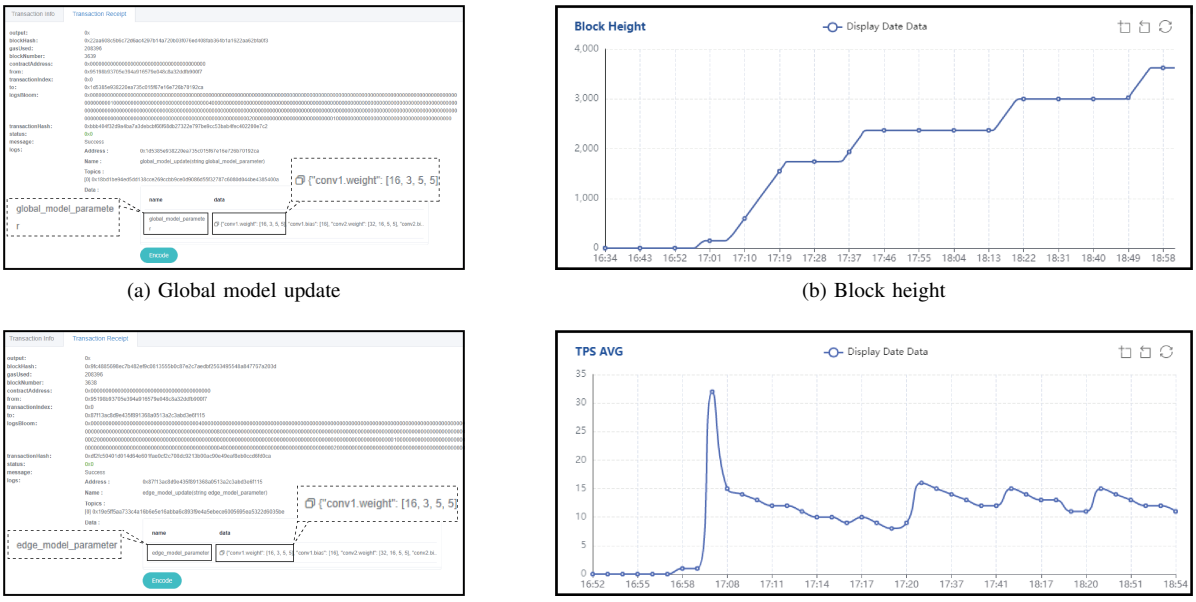


Fig. 8: Maximum energy consumption versus learning utility U for different algorithms

Fig. 8 depicts the maximum energy consumption versus learning utility U for different algorithms with the weight coefficient $\alpha = 0.8$. On the one hand, the DAHFL algorithm could achieve the highest learning performance since IDs could allocate a proper computing frequency and transmission power with different conditions to achieve a low learning utility. On the other hand, as the maximum energy consumption increases, IDs could allocate more computation capacity for local model training and higher transmission power for local model uploading. Finally, the learning utility becomes stable when E is larger than 160. In this case, each ID could process local training with maximum computing capacity and upload local models with maximum transmission power.

Fig. 9a and Fig. 9c show the transactions of the global model and edge models stored in the FISCO blockchain, including the block height, the transaction object, and the parameters in the neural network model. In addition, the curve of the block height and the average transactions per second (TPS) are shown in Fig. 9b and Fig. 9d, respectively.

The visual display of the WeBASE platform is shown in Fig. 10. Fig. 10a is the overview of the WeBASE platform, which includes chain management, contract management, network management and monitoring, transaction audit, and other functions. Fig. 10b shows the data screen of the WeBASE



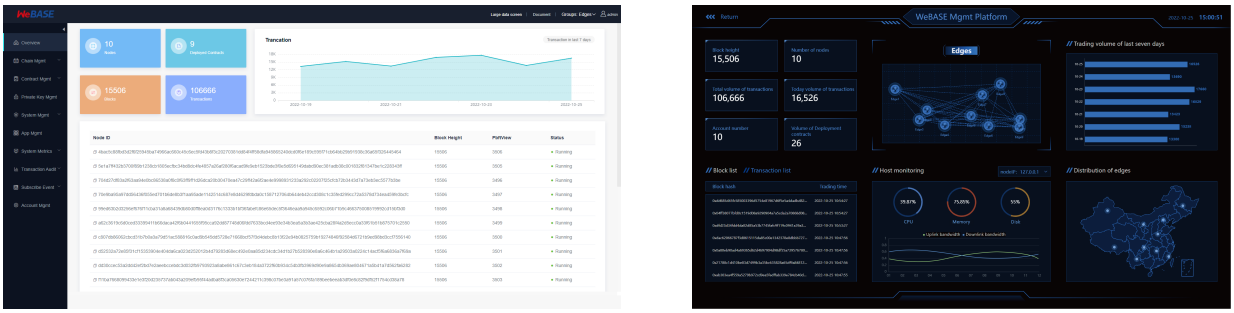
(a) Global model update

(b) Block height

(c) Edge model update

(d) Average TPS

Fig. 9: Block information and node metric



(a) WeBASE platform overview

(b) WeBASE platform data screen

Fig. 10: WeBASE platform exhibition

platform, which visually displays the detailed information of transactions and the basic blockchain information such as block height, the volume of transactions, account number, etc. In addition, block and transaction lists, host monitoring and node distribution are displayed on the screen.

VI. Conclusion

In this paper, we investigated a multi-layer blockchain-enabled HFL network while taking into account the imbalanced data distribution and resource allocations of IDs. Firstly, we derived the bottleneck of the model accuracy from the total data distance and analyzed the effect of the ID association strategy on the total data distance. Then, we designed the DAHFL algorithm for achieving the optimal ID association strategy, computing and communication resources allocation of IDs to improve the model accuracy and minimize the learning latency. Finally, the blockchain simulation platform exhibited the working process of the network and extensive simulation results demonstrated the superior performance of the DAHFL algorithm. Future work should investigate the utilization of asynchronous communication and learning in the proposed HFL network, as well as the integration of state-of-the-art technologies for applications in different scenarios.

Appendix A
Proof of Theorem1

According the definition of $w^G(k)$ and $w^C(k)$, it is facilitate to infer the expression of $w^G(k+1)$ and $w^C(k+1)$. Then, the model error can be further expressed as

$$\|w^G(k+1) - w^C(k+1)\| \leq$$
$$\Big\| \frac{1}{N} \sum_{m=1}^{M} \sum_{n \in \mathcal{N}_m} x_{n,m} \Big\{ w_n^{id}(k) - w^C(k) - \eta \cdot$$
$$\underbrace{\Big[\sum_{z=1}^{Z} q_n(z) \nabla \mathcal{G}(\mathcal{F}^k_{w_n^{id}}(\mathcal{X}), \mathcal{Y}) - \sum_{Z=1}^{Z} q(z) \nabla \mathcal{G}(\mathcal{F}^k_{w^C}(\mathcal{X}), \mathcal{Y}) \Big]}_{\kappa_1} \Big\} \Big\|. \tag{56}$$

Define $A(k+1) = \|w^G(k+1) - w^C(k+1)\|$, then (56) can be written as

$$A(k+1) \leq \Big\| A(k) - \frac{\eta}{N} \sum_{m=1}^{M} \sum_{n \in \mathcal{N}_m} x_{n,m} \kappa_1 \Big\|, \tag{57}$$

where κ_1 can be extended as

$$\begin{aligned}
\kappa_1 &= \sum_{Z} q_n(z) \nabla \mathcal{G}(\mathcal{F}^k_{w_n^{id}}(\mathcal{X}), \mathcal{Y}) - \sum_{Z} q(z) \nabla \mathcal{G}(\mathcal{F}^k_{w^C}(\mathcal{X}), \mathcal{Y}) \\
&+ \sum_{Z} q(z) \nabla \mathcal{G}(\mathcal{F}^k_{w_n^{id}}(\mathcal{X}), \mathcal{Y}) - \sum_{Z} q(z) \nabla \mathcal{G}(\mathcal{F}^k_{w^C}(\mathcal{X}), \mathcal{Y}) \\
&\overset{(a)}{\leq} Q\big[\sum_{Z} q_n(z) - \sum_{Z} q(z) \big] + \sum_{Z} q(z) \nabla \mathcal{G}(\mathcal{F}^k_{w_n^{id}}(\mathcal{X}), \mathcal{Y}) \\
&- \sum_{Z} q(z) \nabla \mathcal{G}(\mathcal{F}^k_{w^C}(\mathcal{X}), \mathcal{Y}) \\
&\overset{(b)}{\leq} Q\big[\sum_{Z} q^n(z) - \sum_{Z} q(z) \big] + \rho \sum_{Z} q(z) \|w_n^{id}(k) - w^C(k)\|,
\end{aligned} \tag{58}$$

where (a) and (b) can be derived based on first and second assumption, respectively. Therefore, (57) can be expressed as

$$\begin{aligned}
A(k+1) &\leq A(k) + \Big\| \frac{\eta}{N} \sum_{m} \sum_{n \in \mathcal{N}_m} x_{n,m} \kappa_1 \Big\| \\
&\leq A(k) + \Big\| \frac{\eta}{N} \sum_{m} \sum_{n \in \mathcal{N}_m} x_{n,m} \rho \sum_{Z} q(z) A(k) \Big\| \\
&+ \Big\| \frac{\eta}{N} \sum_{m} \sum_{n \in \mathcal{N}_m} x_{n,m} Q\big[\sum_{Z} q_n(z) - \sum_{Z} q(z) \big] \Big\| \\
&\leq \Big[1 + \eta\rho \sum_{Z} q(z) \Big] A(k) \\
&+ \frac{\eta Q}{N} \sum_{m} \Big\| \sum_{n \in \mathcal{N}_m} x_{n,m} \big(\boldsymbol{q_n} - \boldsymbol{q} \big) \Big\|,
\end{aligned} \tag{59}$$

where q_n and q denote class proportion vectors of the local data of ID n and the IID data in the central learning. Similarly, the upper bound on the model error of A^{kT} can be obtained. Finally, the model error at one global epoch can be written as (7), which ends the proof.

REFERENCES

[1] C. Chen, G. Yao, C. Wang, S. Goudos, and S. Wan, "Enhancing the robustness of object detection via 6g vehicular edge computing," *Digital Communications and Networks*, vol. 8, no. 6, pp. 923–931, 2022.

[2] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, "A survey of machine and deep learning methods for internet of things (iot) security," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1646–1685, 2020.

[3] K. Tan, D. Bremner, J. Le Kernec, L. Zhang, and M. Imran, "Machine learning in vehicular networking: An overview," *Digital Communications and Networks*, vol. 8, no. 1, pp. 18–24, 2022.

[4] J. Granjal, E. Monteiro, and J. Sá Silva, "Security for the internet of things: A survey of existing protocols and open research issues," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1294–1312, 2015.

[5] J. Konečný, H. Brendan McMahan, D. Ramage, and P. Richtárik, "Federated Optimization: Distributed Machine Learning for On-Device Intelligence," *arXiv e-prints*, p. arXiv:1610.02527, Oct. 2016.

[6] L. Liu, J. Zhang, S. Song, and K. B. Letaief, "Client-edge-cloud hierarchical federated learning," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.

[7] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[8] H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Blockchained on-device federated learning," *IEEE Communications Letters*, vol. 24, no. 6, pp. 1279–1283, 2020.

[9] Y. Yuan and F.-Y. Wang, "Blockchain and cryptocurrencies: Model, techniques, and applications," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 9, pp. 1421–1428, 2018.

[10] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1806.00582

[11] S. Luo, X. Chen, Q. Wu, Z. Zhou, and S. Yu, "Hfel: Joint edge association and resource allocation for cost-efficient hierarchical federated edge learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 6535–6548, 2020.

[12] B. Xu, W. Xia, W. Wen, P. Liu, H. Zhao, and H. Zhu, "Adaptive hierarchical federated learning over wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 2, pp. 2070–2083, 2022.

[13] Y. Gou, R. Wang, Z. Li, M. A. Imran, and L. Zhang, "Clustered hierarchical distributed federated learning," in *ICC 2022 - IEEE International Conference on Communications*. Seoul, Korea, Republic of: IEEE, May 2022, pp. 177–182.

[14] H. Zheng, M. Gao, Z. Chen, and X. Feng, "A distributed hierarchical deep computation model for federated learning in edge computing," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 7946–7956, Dec. 2021.

[15] W. Wen, Z. Chen, H. H. Yang, W. Xia, and T. Q. S. Quek, "Joint scheduling and resource allocation for hierarchical federated edge learning," *IEEE Transactions on Wireless Communications*, vol. 21, no. 8, pp. 5857–5872, Aug. 2022.

[16] Z. Zhao, C. Feng, W. Hong, J. Jiang, C. Jia, T. Q. S. Quek, and M. Peng, "Federated learning with non-iid data in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 21, no. 3, pp. 1927–1942, 2022.

[17] N. Mhaisen, A. A. Abdellatif, A. Mohamed, A. Erbad, and M. Guizani, "Optimal user-edge assignment in hierarchical federated learning based on statistical properties and network topology constraints," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 1, pp. 55–66, Jan. 2022.

[18] P. Tian, W. Liao, W. Yu, and E. Blasch, "Wscc: A weight-similarity-based client clustering approach for non-iid federated learning," *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 20 243–20 256, Oct. 2022.

[19] Z. He, L. Yang, W. Lin, and W. Wu, "Improving accuracy and convergence in group-based federated learning on non-iid data," *IEEE Transactions on Network Science and Engineering*, pp. 1–1, 2022.

[20] S. Liu, G. Yu, X. Chen, and M. Bennis, "Joint user association and resource allocation for wireless hierarchical federated learning with iid and non-iid data," *IEEE Transactions on Wireless Communications*, vol. 21, no. 10, pp. 7852–7866, 2022.

[21] Y. Qu, L. Gao, T. H. Luan, Y. Xiang, S. Yu, B. Li, and G. Zheng, "Decentralized privacy using blockchain-enabled federated learning in fog computing," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 5171–5183, Jun. 2020.

[22] L. Feng, Y. Zhao, S. Guo, X. Qiu, W. Li, and P. Yu, "Bafl: A blockchain-based asynchronous federated learning framework," *IEEE Transactions on Computers*, vol. 71, no. 5, pp. 1092–1103, May 2022.

[23] M. Cao, L. Zhang, and B. Cao, "Toward on-device federated learning: A direct acyclic graph-based blockchain approach," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2021.

[24] J. Qi, F. Lin, Z. Chen, C. Tang, R. Jia, and M. Li, "High-quality model aggregation for blockchain-based federated learning via reputation-motivated task participation," *IEEE Internet of Things Journal*, vol. 9, no. 19, pp. 18 378–18 391, Oct. 2022.

[25] Q. Hu, Z. Wang, M. Xu, and X. Cheng, "Blockchain and federated edge learning for privacy-preserving mobile crowdsensing," *IEEE Internet of Things Journal*, pp. 1–1, 2021.

[26] V. Mothukuri, R. M. Parizi, S. Pouriyeh, A. Dehghantanha, and K.-K. R. Choo, "Fabricfl: Blockchain-in-the-loop federated learning for trusted decentralized systems," *IEEE Systems Journal*, vol. 16, no. 3, pp. 3711–3722, Sep. 2022.

[27] K. Peter and et al, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[28] D. Ongaro and J. Ousterhout, "In search of an understandable consensus algorithm," *Proceedings of the 2014 USENIX Annual Technical Conference, USENIX ATC 2014*, pp. 305 – 319, 2014.

[29] X. Huang, Z. Chen, Q. Chen, and J. Zhang, "Federated learning based qos-aware caching decisions in fog-enabled internet of things networks," *Digital Communications and Networks*, vol. 9, no. 2, pp. 580–589, 2023.

[30] K. Hsieh, A. Phanishayee, O. Mutlu, and P. Gibbons, "The non-IID data quagmire of decentralized machine learning," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 4387–4398.

[31] M. Yin, D. Malkhi, M. K. Reiter, G. G. Gueta, and I. Abraham, "Hotstuff: Bft consensus with linearity and responsiveness," in *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing*. Toronto ON Canada: ACM, Jul. 2019, pp. 347–356.

[32] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Singh and J. Zhu, Eds., vol. 54. PMLR, 20–22 Apr 2017, pp. 1273–1282.

[33] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, 2019.

[34] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-iid data silos: An experimental study," Oct. 2021. [Online]. Available: https://doi.org/10.48550/arXiv.2102.02079

[35] Z. Yang, M. Chen, W. Saad, C. S. Hong, M. Shikh-Bahaei, H. V. Poor, and S. Cui, "Delay minimization for federated learning over wireless communication networks," Jul. 2020. [Online]. Available: https://doi.org/10.48550/arXiv.2007.03462

[36] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, "Energy efficient federated learning over wireless communication networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 1935–1949, 2021.

[37] Y. He, M. Yang, Z. He, and M. Guizani, "Resource allocation based on digital twin-enabled federated learning framework in heterogeneous cellular network," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 1, pp. 1149–1158, Jan. 2023.

[38] V. V. Díaz and D. Marcano Aviles, "A path loss simulator for the 3gpp 5g channel models," in *2018 IEEE XXV International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, 2018, pp. 1–4.

[39] Q. Li, Y. Diao, Q. Chen, and B. He, "Communication-efficient learning of deep networks from decentralized data," Feb. 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1602.05629

[40] L. Yu, R. Albelaihi, X. Sun, N. Ansari, and M. Devetsikiotis, "Jointly optimizing client selection and resource management in wireless federated learning for internet of things," *IEEE Internet of Things Journal*, vol. 9, no. 6, pp. 4385–4395, 2022.

[41] J. Mills, J. Hu, and G. Min, "Communication-efficient federated learning for wireless edge intelligence in iot," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 5986–5994, Jul. 2020.

Xiaoge Huang received her Ph. D. degree (with first honors) in the Group of Information and Communication Systems (GSIC), Institute of Robotics and Information & Communication Technologies (IRTIC) at the University of Valencia, Spain. In 2013, she joins the Group of Wireless Communication Technology, Chongqing University of Posts and telecommunications, as an Associate Professor. Her research interests include convex optimization, centralized and decentralized power allocation strategies, game theory, cognitive radio networks.

Yuhang Wu received the B.S. degree in Communication Engineering from Shanghai Maritime University, Shanghai, China, in 2021. He is currently working toward the M.S. degree in Information and Communication Engineering with Wireless Transmission Laboratory, Chongqing University of Posts and Telecommunications, Chongqing, China. His main research interests are federated learning, wireless communication, and edge computing network.

Chengchao Liang received the Ph.D. in electrical and computer engineering from Carleton University, Ottawa, ON, Canada, in 2017, where he was awarded the Senate Medal. He is currently a Full Professor with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China. Prior to this, he was a cross-appointed Postdoctoral Fellow with the Department of Systems and Computer Engineering, Carleton University as well as Huawei Ottawa Research & Development Centre from 2017 to 2019. He is with the Editorial Boards of EURASIP Journal on Wireless Commun. & Net. and Trans. Emerging Telecomm. Tech. He was the reviewer and the TPC Member of many IEEE journals and conferences. His research interests include wireless communications, satellite networks, Internet protocols and optimization theory.

Qianbin Chen received the Ph.D. degree in communication and information systems from the University of Electronic Science and Technology of China, Chengdu, China, in 2002. He is currently a professor with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, and the Director of the Chongqing Key Laboratory of Mobile Communication Technology. He has authored or co-authored more than 100 papers in journals and peer-reviewed conference proceedings, and has co-authored seven books. He holds 47 granted national patents.

Jie Zhang is currently a Full Professor and has been the Chair in Wireless Systems with the EEE Department, The University of Sheffield, Sheffield, U.K., since 2011. He was with Imperial College London, Oxford University, and the University of Bedfordshire. He and his students have piaered research in femto/small cell and Het-Nets and published some of the earliest and most widely cited publications on these topics (three of top ten most cited). He cofounded RANPLAN Wireless Network Design Ltd., which produces a suite of world leading in-building DAS, indoor-outdoor small cell/HetNet network design, and optimization tools iBuildNet-Professional, Tablet, DAS, and Manager. Since 2005, he has been awarded over 20 research projects by the EPSRC, the EC FP6/FP7/H2020, and industry, including some of the earliest projects on femtocell/HetNets.