FedD2D: Device Pairing and Scheduling in D2D-Assisted Federated Edge Learning

Chongyu Bao, Yunwen Qiu, Tong Liu, Wenchao Xia(✉), and Haitao Zhao

Nanjing University of Posts and Telecommunications, Nanjing, China
{p21000108,q22010301,1222014021,xiawenchao,zhaoht}@njupt.edu.cn

Abstract. The proliferation of Internet of Things (IoT) applications has significantly increased edge devices and data volume at the edge of network, making them ideal candidates for federated edge learning (FEEL). However, the limited spectral resources of edge base stations (BSs) restrict device participation, exacerbating the impact of data heterogeneity, negatively affecting model convergence, even leading to model drift. To mitigate the impact of BS access restrictions on the performance of FEEL, this paper introduces a novel FEEL architecture that is empowered by a mode of direct inter-device communication, Device-to-Device (D2D) communication. Considering the negative influence of data heterogeneity and the potential packet error rate (PER) under this architecture, a nonlinear integer programming problem is formulated. Subsequently, we proposed an elegant algorithm termed FedD2D that jointly optimizes device pairing and scheduling in D2D communications while incorporating fairness constraints to counteract the negative consequences of above factors. Ultimately, the experimental results demonstrate the superiority of proposed, specifically reflected in the increased device participation and the reduction of the impact of communication unreliability, thereby enhancing the performance of FEEL.

Keywords: Federated learning · device-to-device communication · device scheduling

1 Introduction

The recent upsurge in the Internet of Things (IoT) [1] applications results in an exponentially increasing number of IoT devices, which in turn has generated a

This work was supported in part by the National Natural Science Foundation of China under Grant 62371250 and Grant 62201285; in part by the Jiangsu Natural Science Foundation for Distinguished Young Scholars under Grant BK20220054; and in part by the Natural Science Foundation on Frontier Leading Technology Basic Research Project of Jiangsu under Grant BK20212001.

H.-H. Chen and W. Meng (Eds.): WiSATS 2024, LNICST 606, pp. 109–119, 2025.
https://doi.org/10.1007/978-3-031-86203-8_9

massive volume of data at the edge of the wireless network. To leverage these valuable data resources, traditional machine learning (ML) algorithms upload them to a centralized server for training. This, however, is impractical for edge intelligent systems due to privacy concerns. With these considerations in mind, a type of wireless federated learning (FL) system deployed at the network edge, namely federated edge learning (FEEL) [2], has been proposed to overcome this bottleneck, which allows participating edge devices to build a global model collaboratively while maintaining data locally.

Unfortunately, FEEL deployment primarily faces two practical challenges: limited spectral resources and data heterogeneity. Specifically, limited by the concurrent processing capabilities and spectral resource constraints, edge base stations (BSs) cannot support transmission service of massive IoT devices simultaneously [3]. Consequently, only a portion of devices can access to edge BSs at once, which restricts the degree of device participation in the model training process and negatively impacts convergence performance. In addition, local datasets of massively distributed devices [4] are usually statistically heterogeneous and highly redundant, which may lead to slow model convergence and even model drift especially when only a limited number of devices can participate in each round of training. Recent works have responded to address the aforementioned challenges. [5,6] highlighted the significance of optimizing device selection and resource allocation within cellular FEEL environments to mitigate the impact of constrained communication resources. Meanwhile, efficient broadband analog transmission schemes and distributed approximate Newton-type algorithms with fast convergence were proposed by [7,8] as solutions to alleviate FEEL resource constraints. Additionally, [9] reduced the impact of imperfect communication capabilities in FEEL.

Different from the above-mentioned work and inspired by [10], this paper introduce Device-to-Device (D2D) communication to mitigate the access constraint of BSs. More specifically, we propose a novel D2D-enabled FEEL architecture, aiming to enhance the convergence performance of FEEL. Our proposed architecture enables devices not only to communicate directly with BSs (D2B) but also to pair with each other via D2D communication, which substantially increases the number of devices participating in each FL iteration. Compared to existing FL architectures that utilize D2D [10,11], we primarily focus on the pairing methods for D2D pairs and the scheduling strategies for entities, which include both single devices and D2D pairs. Particular attention is given to the communication packet loss rate (PER) and data heterogeneity. Our approach aims to circumvent the device access limitation issues caused by the limited spectrum resources at the BS. By establishing and incorporating D2D devices, we increase the number of devices participating in each FL round. This strategy effectively reduces the negative impact of model bias, thereby enhancing the model's robustness and convergence performance. Motivated by these considerations, we formulate the challenge as a nonlinear integer programming problem. To optimize this problem, an elegant algorithm is designed, which improves convergence performance of FEEL efficiently by conducting D2D pairing and device scheduling, mitigating the impacts of communication unreliability and data heterogeneity. Finally, simulation results demonstrate that significant enhancements in the training efficiency of our proposed FEEL system, while ensuring reliable communication among devices.

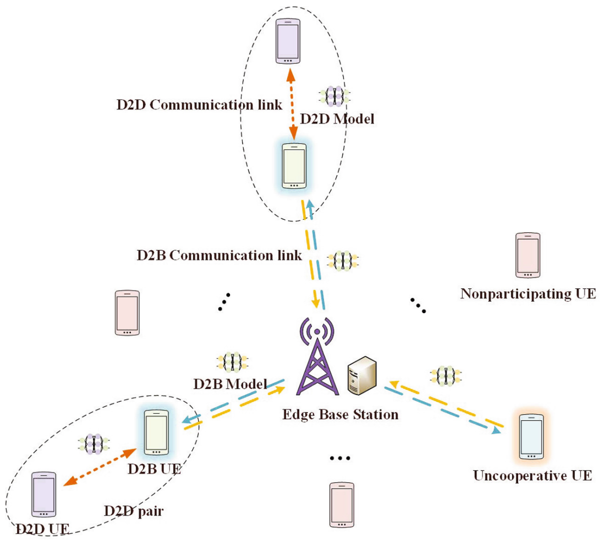


Fig. 1. D2D-enabled FEEL architecture

2 System Model

2.1 Learning Model

As shown in Fig. 1, we consider a wireless network comprising an edge BS and a substantial number of user devices (UEs), denoted by a set $\mathcal{C} = \{1, \ldots, C\}$ with $C = |\mathcal{C}|$, aiming to train a FL model cooperatively. It is assumed that the BS can build connections with at most N UEs simultaneously in each iteration of FL because of limited wireless resources. These UEs can participate in the training process by communicating directly with the BS (i.e., D2B communication) or with neighboring UEs (i.e., D2D communication). Denote by $\mathcal{S}_t$ the set of single UEs that engage exclusively in D2B communication without participating in D2D pairing in round t. Moreover, we refer to a pair of two UEs as a virtual UE and denote the set of virtual UEs by $\mathcal{P}_t$. For any virtual UE i, we denote $B_{i,t}$ as the UE transmitting to the BS, and $D_{i,t}$ as the UE transmitting to $B_{i,t}$.

We define an indicator vector $\boldsymbol{a}_t$ of UE scheduling, whose i-element is $a_{i,t}$. If UE i is selected by BS in round t, $a_{i,t} = 1$, otherwise $a_{i,t} = 0$. In each round, BS only selects a fraction of UEs from $\mathcal{S}_t \bigcup \mathcal{P}_t$ to participate in the training process.

Each selected UE updates the locally sent model and returns it to the BS. Then, BS updates the global model and broadcasts the global model to the selected UEs. The BS conducts FL training aiming at minimizing a weighted global loss function i.e.,

$$\min_{\mathbf{w}} \tfrac{1}{K} \textstyle\sum_{i\in\mathcal{C}} \sum_{k=1}^{K_i} \mathcal{L}\left(\mathbf{w}_i\right), \tag{1}$$

where K_i is the local dataset size of UE i, $K = \sum_{i\in\mathcal{C}} K_i$ is the size of total dataset, $\mathbf{w}$ is the global model, $\mathbf{w}_i$ and $\mathcal{L}(\mathbf{w}_i)$ represent the local model and loss function of UE i, respectively. The update rule of the global model $\mathbf{w}$ is given as [12]

$$\mathbf{w}_{t+1} = \frac{\sum_{i\in\mathcal{C}} K_i \mathbf{w}_{i,t}}{K}, \tag{2}$$

where $\mathbf{w}_{t+1}$ represents the updated global model at the end of training round t, and $\mathbf{w}_{i,t}$ denotes the model parameters contributed by UE $i \in \mathcal{C}$ based on its local data.

2.2 Packet Error Rates

This paper considers the PER for the D2B and D2D communications[1]. The orthogonal frequency is adopted for the D2D and D2B communications, thus no inter-user interference exists between D2D and D2B communications.

In D2B communication, each UE, in particular, encapsulates its model parameters into a single data packet for transmission. Upon arrival at the BS, the data packet integrity is verified through Cyclic Redundancy Check (CRC). If the CRC indicates corruption or loss, the packet is discarded. Then, the PER for the D2B communication is given by

$$q_{i,\mathrm{B}} = \mathbb{E}_{h_{i,\mathrm{B}}} \left(1 - \exp\left(-\frac{\tau W_{\mathrm{B}} N_0}{P_{i,\mathrm{B}} h_{i,\mathrm{B}}}\right)\right), \tag{3}$$

where $q_{i,\mathrm{B}}$ represents the PER for UE i. $P_{i,\mathrm{B}}$ denotes the D2B transmit power of UE i, $h_{i,\mathrm{B}}$ is the D2B channel gain, W_{B} is the allocated uplink bandwidth, N_0 is the noise power spectral density, and τ is a system-specific threshold parameter [13].

In addition, similar to the D2B communication. Then, the PER for D2D communication is given as

$$q_{i,\mathrm{D}} = \mathbb{E}_{h_{i,\mathrm{D}}} \left(1 - \exp\left(-\frac{\tau' W_{\mathrm{D}} N_0}{P_{i,\mathrm{D}} h_{i,\mathrm{D}}}\right)\right), \tag{4}$$

where $h_{i,\mathrm{D}}$ is the D2D channel gain, W_{D} is the dedicated bandwidth for D2D communications, $P_{i,\mathrm{D}}$ denotes the D2D transmit power, and τ' is a threshold parameter related to the inverse coding gain specific to D2D transmission.

[1] Note that for B2D communication, since the broadcast power of the BS is relatively high, the downlink PER is negligible.

2.3 Availability Constraint

During the FL process, certain UEs may become temporarily unavailable due to factors such as poor channel conditions or exhausted computing capabilities [14]. In this paper, we focus primarily on UE dropouts caused solely by PER and introduce a D2B model transmitting indicator labeled as $C_{i,\mathrm{B}} \in \{0,1\}$, where $C_{i,\mathrm{B}} = 1$ signifies that the packet (i.e., the local model) from UE i is successfully transmitted, otherwise, $C_{i,\mathrm{B}} = 0$. The indicator $C_{i,\mathrm{B}}$ is given by

$$C_{i,\mathrm{B}} = \begin{cases} 1, & \text{with prob. } 1 - q_{i,\mathrm{B}}, \\ 0, & \text{with prob. } q_{i,\mathrm{B}}. \end{cases} \tag{5}$$

Similarly, we introduce an indicator $C_{i,\mathrm{D}} \in \{0,1\}$ for each UE $i \in \mathcal{P}_t$, where $C_{i,\mathrm{D}} = 1$ indicates that the D2D transmission is successful, otherwise, $C_{i,\mathrm{D}} = 0$. The indicator $C_{i,\mathrm{D}}$ can be expressed as

$$C_{i,\mathrm{D}} = \begin{cases} 1, & \text{with prob. } (1 - q_{i,\mathrm{D}})^2, \\ 0, & \text{with prob. } 1 - (1 - q_{i,\mathrm{D}})^2, \end{cases} \tag{6}$$

where the square operation is introduced because $D_{i,t}$ should first receive the model parameters from $B_{i,t}$ and then send the updated version to $B_{i,t}$ after local training.

Finally, based on the above settings, the model aggregation formula (2) can be reformulated as

$$\mathbf{w}_{t+1} = \frac{\sum_{i \in \mathcal{P}_t} (K_{i,\mathrm{B}} + K_{i,\mathrm{D}} C_{i,\mathrm{D}}) \mathbf{w}_{i,t} a_{i,t} C_{i,\mathrm{B}} + \sum_{j \in \mathcal{S}_t} K_j \mathbf{w}_{j,t} a_{j,t} C_{j,\mathrm{B}}}{\sum_{i \in \mathcal{P}_t} (K_{i,\mathrm{B}} + K_{i,\mathrm{D}} C_{i,\mathrm{D}}) a_{i,t} C_{i,\mathrm{B}} + \sum_{j \in \mathcal{S}_t} K_j a_{j,t} C_{j,\mathrm{B}}}, \tag{7}$$

where $K_{i,\mathrm{B}}$ and $K_{i,\mathrm{D}}$ specify the size of the local datasets of $B_{i,t}$ and $D_{i,t}$ for $i \in \mathcal{P}_t$, respectively.

2.4 Fairness Constraint

In addition to the availability constraint, the issue of data heterogeneity among UEs is particularly significant [15]. Specifically, the non-i.i.d. and unbalanced properties of local datasets across UEs can negatively impact the efficiency of model training. An effective solution is introducing fairness constraints for each client [16],

$$\lim_{T \to \infty} \inf \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[a_{i,t}] \geq c_i, \quad \forall i \in \mathcal{C}, \tag{8}$$

where $c_i \in [0,1)$ represents the minimum fraction of communication rounds required for UE i. We use the normalized product of the local dataset size and the upload success rate as a metric to describe the importance of UEs and thus set $c_i = \frac{V_i(1-q_{i,\mathrm{B}})}{\sum_{i\in\mathcal{C}} V_i(1-q_{i,\mathrm{B}})}$. To address the fairness constraint in (8), we introduce a virtual queue D_i for UE $i \in \mathcal{C}$, i.e.,

$$D_i(t) = \left[D_i(t-1) + c_i - a_{i,t-1}\right]^+ . \tag{9}$$

Let $[x]^+ = \max\{x, 0\}$, where $D_i(t)$ represents the queue length at the start of round t. We denote $\mathbf{D}(t) = [D_1(t), \ldots, D_K(t)]$ with an initial condition of $\mathbf{D}(1) = \mathbf{0}$.

3 Convergence Analysis and Problem Formulation

During the training process, all participating UEs employ the standard gradient descent for local training. The BS updates the global FL model by $\mathbf{w}_{t+1} = \mathbf{w}_t - \lambda\left(\nabla\mathcal{L}(\mathbf{w}_t) - \boldsymbol{\varepsilon}\right)$, where λ is the learning rate. Based on Eq. (7), $\boldsymbol{\varepsilon}$ can be represented as

$$\begin{aligned}\boldsymbol{\varepsilon} = {} & \nabla\mathcal{L}(\mathbf{w}_t) \\ & - \frac{\sum\limits_{i\in\mathcal{P}_t} \left(\sum_{l=1}^{K_{i,\mathrm{B}}} \nabla\ell(\mathbf{w}_t) + \sum_{l=1}^{K_{i,\mathrm{D}}} \nabla\ell(\mathbf{w}_t) C_{i,\mathrm{D}}\right) a_{i,t} C_{i,\mathrm{B}}}{\sum\limits_{i\in\mathcal{P}_t} (K_{i,\mathrm{B}} + K_{i,\mathrm{D}} C_{i,\mathrm{D}}) a_{i,t} C_{i,\mathrm{B}} + \sum\limits_{j\in\mathcal{S}_t} K_j a_{j,t} C_{j,\mathrm{B}}} \\ & - \frac{\sum\limits_{j\in\mathcal{S}_t} \sum_{l=1}^{K_j} \nabla\ell(\mathbf{w}_t) a_{j,t} C_{j,\mathrm{B}}}{\sum\limits_{i\in\mathcal{P}_t} (K_{i,\mathrm{B}} + K_{i,\mathrm{D}} C_{i,\mathrm{D}}) a_{i,t} C_{i,\mathrm{B}} + \sum\limits_{j\in\mathcal{S}_t} K_j a_{j,t} C_{j,\mathrm{B}}},\end{aligned} \tag{10}$$

where $\nabla\ell(\mathbf{w}_t)$ and $\nabla\mathcal{L}(\mathbf{w}_t)$ is the gradient of the local model of UEs and the updated global model at the end of training round t.

3.1 Convergence Analysis

Let the conditions of Lipschitz continuity and μ-strong convexity from [17] hold, where L and μ are the positive constants for Lipschitz continuity and strong convexity, respectively. Moreover, the global loss function $\mathcal{L}(\mathbf{w})$ is twice continuously differentiable, satisfying $\mu I \preceq \nabla^2\mathcal{L}(\mathbf{w}) \preceq LI$, and the local loss function satisfies $\|\nabla\ell(\mathbf{w}_t)\|^2 \le \zeta_1 + \zeta_2\|\nabla\mathcal{L}(\mathbf{w}_t)\|^2$, where ζ_1 and ζ_2 are positive constants. Finally, given the UE scheduling indicator matrix $\boldsymbol{a}_t$ and the optimal global FL model $\mathbf{w}^*$, we have following theorem.

Theorem 1. *With the learning rate $\lambda = \frac{1}{L}$, the upper bound of $\mathbb{E}[\mathcal{L}(\mathbf{w}_{t+1}) - \mathcal{L}(\mathbf{w}^*)]$ can be provided as follows*

$$\begin{aligned}\mathbb{E}\left[\mathcal{L}\left(\mathbf{w_{t+1}}\right) - \mathcal{L}\left(\mathbf{w}^*\right)\right] \le & \boldsymbol{\Gamma}^t \mathbb{E}\left[\mathcal{L}\left(\mathbf{w_0}\right) - \mathcal{L}\left(\mathbf{w}^*\right)\right] \\ & + \frac{2\xi_1}{LK}\left(\vec{A} + \vec{B}\right)\frac{1-\boldsymbol{\Gamma}^t}{1-\boldsymbol{\Gamma}}.\end{aligned} \tag{11}$$

From Eq. (11), $\boldsymbol{A} = \sum_{i \in \mathcal{P}_t} \left[1 - a_{i,t}\left(K_{i,\mathrm{B}} + K_{i,\mathrm{D}}(1-q_{i,\mathrm{D}})^2\right)(1-q_{i,\mathrm{B}})\right]$, represents impact of wireless factors on convergence from virtual UEs, $\boldsymbol{B} = \sum_{j \in \mathcal{S}_t} K_j\left(1 - a_{j,t} + a_{j,t}q_{j,\mathrm{B}}\right)$, represents impact of wireless factors on convergence from single UEs and $\boldsymbol{\Gamma} = 1 - \frac{\mu}{L} + \frac{4\mu\zeta_2}{LK}(\boldsymbol{A}+\boldsymbol{B})$, $\mathbf{w}_{t+1}$ is the global FL model that is generated based only on the local FL models of selected UEs $(a_{i,t}=1)$ at step $t+1$. $\mathbf{w}^*$ is the theoretically optimal FL model in this scenario.

Proof. Due to the space limitation, the proof is omitted.

3.2 Problem Formulation

From Theorem 1, to minimize the upper bound of Eq. (11), we need to only minimize the gap $\frac{2\xi_1}{LK}(\boldsymbol{A}+\boldsymbol{B})\frac{1-\boldsymbol{\Gamma}^t}{1-\boldsymbol{\Gamma}}$, in which $\boldsymbol{A}$ can be rewritten as $K_{\mathcal{P}_t} - \sum_{i \in \mathcal{P}_t} a_{i,t}\left[K_{i,\mathrm{B}} + K_{i,\mathrm{D}}(1-q_{i,\mathrm{D}})^2\right](1-q_{i,\mathrm{B}})$ and $\boldsymbol{B}$ can also be rewritten as $K_{\mathcal{S}_t} - \sum_{j \in \mathcal{S}_t} a_{j,t}K_j(1-q_{j,\mathrm{B}})$, where $K_{\mathcal{P}_t}$ and $K_{\mathcal{S}_t}$ are the total size of dataset for $\mathcal{P}_t$ and $\mathcal{S}_t$ in round t, respectively. Further, the optimization problem can be expressed as

$$\max_{\mathcal{S}_t,\mathcal{P}_t,\boldsymbol{a}_t}\left(\underbrace{\sum_{i \in \mathcal{P}_t} a_{i,t}[K_{i,\mathrm{B}} + K_{i,\mathrm{D}}(1-q_{i,\mathrm{D}})^2](1-q_{i,\mathrm{B}})}_{\text{total value of virtual UEs}} + \underbrace{\sum_{j \in \mathcal{S}_t} a_{j,t}K_j(1-q_{j,\mathrm{B}})}_{\text{total value of single UEs}}\right), \tag{12}$$

$$\text{s.t.} \quad a_{i,t} \in \{0,1\}, \forall i \in \mathcal{C}, \forall t \in \mathcal{T}, \tag{12a}$$

$$\textstyle\sum_{i \in \mathcal{S}_t \bigcup \mathcal{P}_t} a_{i,t} \le N, \forall t \in \mathcal{T}, \tag{12b}$$

$$|\mathcal{S}_t| + |\mathcal{P}_t| \le C, \forall t \in \mathcal{T}, \tag{12c}$$

where the term $[K_{i,\mathrm{B}} + K_{i,\mathrm{D}}(1-q_{i,\mathrm{D}})^2](1-q_{i,\mathrm{B}})$ represents the value of each virtual UE, while $K_j(1-q_{j,\mathrm{B}})$ corresponds to the value of each single UE and $\mathcal{T}$ is the set of total training rounds. A scheduling strategy $\boldsymbol{a}_t$ is optimized to select the UEs in $\mathcal{S}_t \bigcup \mathcal{P}_t$ with the maximum values in order to minimize the gap in (1), where constraint (12a) defines the D2B transmission association indicator as binary, (12b) limits the maximum number of UEs that BS can access at any given time, which is fewer than the total tally of UEs, and (12c) limits the total number of UEs of $\mathcal{S}_t \bigcup \mathcal{P}_t$.

In fact, above problem can be modeled as a graph theory problem. The UEs in the FEEL system can be regarded as vertices, the potential matchings between UEs can be regarded as edges, and the values associated with these matchings can be regarded as the weights of the edges. Therefore, the optimization problem can be viewed as finding a set of edges that do not share vertices and that maximize the total weight, subject to a constraint on the size of the set. Therefore, Eq. (12) can be can be reduced to a Maximum Weighted Independent Edge Set (MWIES) problem with constraints, which is known to be NP-hard.

4 Algorithm Design

As shown in Algorithm 1, for UE $i \in \mathcal{C}$, we consider it as a node $u_i \in U$ to construct the general graph $G(U)$. Based on Eq. (12), we define the vertex weight of single UE i,

$$V_{i \in \mathcal{S}_{\mathrm{t}}} = \beta K_i(1 - q_{i,\mathrm{B}}) + (1 - \beta) D_i(t), \tag{13}$$

where the term $K_i(1 - q_{i,\mathrm{B}})$ arises from Eq. (12), jointly considering both the size of its own dataset and PER, representing the value of UE i. $D_i(t)$ is the virtual queue of UE i, used to enforce fairness constraints, $\beta \in [0, 1]$ is a non-negative constant. Without loss of generality, egde weight of virtual UE i is given by

$$\begin{aligned} V_{i \in \mathcal{P}_{\mathrm{t}}} = & \beta(1 - q_{i,\mathrm{B}})(K_{i,\mathrm{B}} + K_{i,\mathrm{D}}(1 - q_{i,\mathrm{D}})^2) \\ & + (1 - \beta)(D_{i,\mathrm{B}}(t) + D_{i,\mathrm{D}}(t)), \end{aligned} \tag{14}$$

where $D_{i,\mathrm{B}}(t)$ indicate the virtual queue of $B_{i,t}$ while $D_{i,\mathrm{D}}(t)$ refer to $D_{i,t}$. Notice that the device with a lower $q_{i,\mathrm{B}}$ is prioritized as $B_{i,t}$ when the BS selects potential pairings.

Algorithm 1. FedD2D: Joint optimization of D2D device pairing and scheduling

Require: N, d_{max}, $\mathcal{C}$, $\mathcal{S}_t \leftarrow \emptyset$, $\mathcal{P}_t \leftarrow \emptyset$, , $\boldsymbol{a}_t = \mathbf{0}$
Initialize $G(U, V, E)$, where $U \Rightarrow \mathcal{C}$, $V \Rightarrow \mathcal{C}$;
for $i \in \mathcal{C}$ **do**
  for $j \in \mathcal{C}$ where $d_{ij} \leq d_{max}$ **do**
    if $i = j$ **then**
      Calculate W_u based on (17);
      Add $e(i, j, W_u)$ to E;
    else if $i \neq j$ **then**
      Calculate W_e based on (18) for $e(i, j)$;
      Add $e(i, j, W_e)$ to E;
    end if
  end for
end for
Find $\mathcal{M}$ KM based on $G(U, V, E)$;
for each $e(i, j) \in \mathcal{M}$ **do**
  if $i = j$ **then**
    Assign i, W_u to $\mathcal{S}_t$;
  else if $i \neq j$ **then**
    Assign i, W_u to $\mathcal{P}_t$;
  end if
end for
Sort $\mathcal{U}_t = \mathcal{S}_t \bigcup \mathcal{P}_t$ in descending order based on assigned weights;
for $i \in \mathcal{U}_t[0 : N]$ **do**
  $a_{i,t} = 1$;
end for
return $\mathcal{S}_t$, $\mathcal{P}_t$, $\boldsymbol{a}_t$

The MWIES problem is hard to solve, so we restructure $G(U)$, through the execution of an elegant symmetrical operation upon the nodes contained within

U, a new collection of nodes denoted as V is acquired, thereby constructing the bipartite graph $G(U, V)$. Subsequently, we construct the edges. For $u_i \in U$ and $v_j \in V$, an edge $e(i, j)$ is established. When $i = j$, the edge $e(i, j)$ is considered a self-connecting edge, the weight of the edge is assigned as W_u according to (13). Conversely, when $i \neq j$, the edge $e(i, j)$ represents the combined value of the D2D pair (i, j), the edge weight is assigned as W_e according to (14). Then we use the Kuhn-Munkres algorithm to achieve the maximum matching $\mathcal{M}$, and to satisfy the constraint (12b), we sort $\mathcal{M}$ by descending edge weight and Top-N elements are taken.

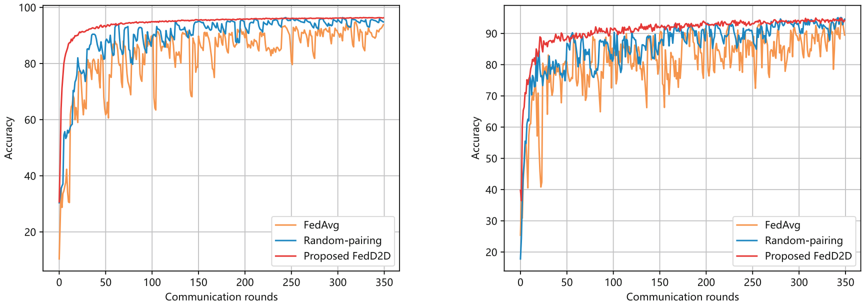
5 Simulation Results

Table 1. Experimental Parameter Settings

Parameter	Value
Radius	100 m
$P_{i,B}$	200 mW
$P_{i,D}$	100 mW
B_{D2B}	1 Mbps
B_{D2D}	1 Mbps
N_0	-173 dBm/Hz
Batch size	64
Learning rate	0.05

In preparing learning tasks, we experimented with a MLP for MNIST. The simulation settings are summarized in TABLE I. Assuming larger sample sizes equate to more comprehensive data distribution, we simulated non-i.i.d. and unbalanced local datasets by assigning random sample sizes to UEs. Larger samples had higher chances of being allocated two labels, while smaller samples likely received one label. We recorded the classification accuracy of the global model on the test dataset to evaluate FL training performance. We utilize the following baseline methods: a) The FedAvg algorithm, a typical FL algorithm [18]. b) a random D2D matching and selection algorithm ('Random-pairing'). Our proposed method is referred to as 'FedD2D'.

Figure 2 illustrates the convergence performance of our proposed scheme and two baseline schemes on the MNIST dataset, under both ideal conditions and conditions with PER. First, when the number of UE is limited, introducing D2D communication significantly improves the convergence efficiency, enhancing stability and the accuracy of the convergence curve. Compared to ideal conditions, all three algorithms experience a decline in accuracy and stability under conditions with PER. However, the accuracy curve of proposed FedD2D exhibits less jitter and better convergence than the other two algorithms.



(a) Without PER

(b) With PER

Fig. 2. The PER comparison of performance on MNIST

6 Conclusion

In this paper, we enhanced the convergence performance of FL in edge computing environments and reduced the impact of communication unreliability by introducing a D2D-empowered FEEL architecture along with the FedD2D online algorithm. Experimental results have confirmed that our architecture can effectively utilize the limited spectrum resources of BS and achieve satisfactory model training outcomes. Moreover, in this work, we assume orthogonal D2B and D2D communications under the unified scheduling of BS. Future work will be dedicated to exploring more efficient coupling methods of D2D communication and FL, taking into account user willingness, as well as verifying model performance under more complex network conditions and data distributions.

References

1. Boyes, H., Hallaq, B., Cunningham, J., Watson, T.: The industrial internet of things (IIoT): an analysis framework. Comput. Ind. **101**, 1–12 (2018)
2. Tak, A., Cherkaoui, S.: Federated edge learning: design issues and challenges. IEEE Network **35**(2), 252–258 (2020)
3. Wang, Z., et al.: Asynchronous federated learning over wireless communication networks. IEEE Trans. Wireless Commun. **21**(9), 6961–6978 (2022)
4. Xia, W., Quek, T.Q.S., Guo, K., Wen, W., Yang, H.H., Zhu, H.: Multi-armed bandit-based client scheduling for federated learning. IEEE Trans. Wireless Commun. **19**(11), 7108–7123 (2020)
5. Lim, W.Y.B., et al.: Federated learning in mobile edge networks: a comprehensive survey. IEEE Commun. Surv. Tutorials **22**(3), 2031–2063 (2020)
6. Ren, J., He, Y., Wen, D., Yu, G., Huang, K., Guo, D.: Scheduling for cellular federated edge learning with importance and channel awareness. IEEE Trans. Wireless Commun. **19**(11), 7690–7703 (2020)
7. Liu, Y., Zhu, Y., Yu, J.J.: Resource-constrained federated edge learning with heterogeneous data: formulation and analysis. IEEE Trans. Network Sci. Eng. **9**(5), 3166–3178 (2022)

8. Zhu, G., Wang, Y., Huang, K.: Broadband analog aggregation for low-latency federated edge learning. IEEE Trans. Wireless Commun. **19**(1), 491–506 (2020)
9. Hosseinalipour, S., et al.: Multi-stage hybrid federated learning over large-scale D2D-enabled fog networks. IEEE/ACM Trans. Networking **30**(4), 1569–1584 (2022)
10. Lin, F.P.-C., Hosseinalipour, S., Azam, S.S., Brinton, C.G., Michelusi, N.: Semi-decentralized federated learning with cooperative D2D local model aggregations. IEEE J. Sel. Areas Commun. **39**(12), 3851–3869 (2021)
11. Zhang, X., Liu, Y., Liu, J., Argyriou, A., Han, Y.: D2D-assisted federated learning in mobile edge computing networks. In: IEEE Wireless Communications and Networking Conference (WCNC), pp. 1–7. IEEE (2021)
12. Chen, M., Yang, Z., Saad, W., Yin, C., Poor, H.V., Cui, S.: A joint learning and communications framework for federated learning over wireless networks. IEEE Trans. Wireless Commun. **20**(1), 269–283 (2020)
13. Xi, Y., Burr, A., Wei, J., Grace, D.: A general upper bound to evaluate packet error rate over quasi-static fading channels. IEEE Trans. Wireless Commun. **10**(5), 1373–1377 (2011)
14. Xia, W., Quek, T.Q.S., Guo, K., Wen, W., Yang, H.H., Zhu, H.: Multi-armed bandit-based client scheduling for federated learning. IEEE Trans. Wireless Commun. **19**(11), 7108–7123 (2020)
15. Abdelmoniem, A.M., Ho, C.-Y., Papageorgiou, P., Canini, M.: A comprehensive empirical study of heterogeneity in federated learning. IEEE Internet Things J. **10**(16), 14-071–14-083 (2023)
16. Xia, W., Wen, W., Wong, K.-K., Quek, T.Q., Zhang, J., Zhu, H.: Federated-learning-based client scheduling for low-latency wireless communications. IEEE Wirel. Commun. **28**(2), 32–38 (2021)
17. Yang, H.H., Liu, Z., Quek, T.Q., Poor, H.V.: Scheduling policies for federated learning in wireless networks. IEEE Trans. Commun. **68**(1), 317–333 (2019)
18. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. Artif. Intell. Stat., 1273–1282. PMLR (2017)