Joint Device and Training Scheduling for Wireless Federated Learning

Xiaobing Chen, Xiangwei Zhou, Hongchao Zhang, Mingxuan Sun, and Taibiao Zhao

***Abstract*—The advent of ubiquitous computing devices in the Internet of Things (IoT) has resulted in an explosion of data. Traditional centralized machine learning models face challenges including limited bandwidth in wireless environments and privacy concerns due to their data aggregation approach. Federated learning addresses these challenges via decentralizing model training across numerous devices, leveraging model updates to enhance privacy and reduce communication overhead. To improve its cost efficiency, current research focuses on minimizing either time or energy costs but rarely both, and does not jointly optimize the parameters of device and training scheduling in the presence of system and data heterogeneity inherent in IoT networks. In our paper, we first introduce a multi-group transmission scheme and propose a comprehensive device scheduling framework, Group Scheduling on Orthogonal Frequency-Division Multiple Access (GS-OFDMA), to address time bottlenecks. Then we formulate a joint optimization problem for device and training scheduling that minimizes the total cost of training while ensuring model convergence. To tackle the resulting mixed integer nonlinear programming problem, we develop an iterative algorithm. Experimental results show that our approach significantly reduces the total cost by at least 35% across various real-world datasets and data distributions in comparison with random participant selection. The proposed GS-OFDMA protocol also exhibits higher time efficiency over other device scheduling schemes.**

***Index Terms*—Wireless federated learning, scheduling, participant selection, optimization, Internet of Things.**

I. Introduction

With the rapid expansion of the Internet of Things (IoT), vast networks of interconnected devices—including smartphones, wearables, and IoT sensors—generate an ever-growing volume of data. Traditional machine learning models face challenges due to impractical large-scale data aggregation and data privacy concerns. Federated learning offers a decentralized framework for model training across multiple IoT devices without collecting raw data, thus reducing communication overhead and avoiding data leakage [2].

Since IoT networks often operate under constrained bandwidth and varying wireless system conditions, communication overhead management during the federated training is critical for overall cost efficiency. In addition, as IoT devices commonly have limited computational resources and battery life, the practical deployment of federated learning systems in IoT settings demands cost-effective solutions that optimize both time and energy consumption [3]. To improve training efficiency, empirical methods such as [4, 5] track client training metrics, such as loss and time cost, to guide participant selection. However, these heuristic approaches often lack theoretical guarantees for convergence.

Research in federated learning increasingly focuses on cost minimization, aiming to reduce time [6, 7] or energy costs [8, 9] through various control variables and constraints. These control variables fall into two categories: device scheduling [6, 7, 10] and training scheduling [11–13]. Device scheduling involves selecting participants for each round and organizing model transmissions in IoT networks, to enhance model convergence through data diversity and reduce costs by excluding inefficient participants. Training scheduling involves configuring the training process, such as the number of participants per round [12], the number of local iterations [11], and the number of communication rounds between clients and the server. Without proper training scheduling, issues such as client shifts can compromise model convergence and increase resource usage.

However, current studies on optimizing the cost efficiency of federated learning systems have notable limitations. First, many studies focus on reducing either time cost [6, 10, 14] or energy consumption [8, 9], but not both. Second, comprehensive optimization that jointly considers both device scheduling and training scheduling is often absent [11, 12]. Third, problem formulations frequently fail to jointly consider system and data heterogeneity [14, 15]. Lastly, in terms of the wireless communication model, existing work using frequency-sharing protocols typically only considers single-group transmission, limiting the number of participants to the number of subchannels within the frequency band [3, 9, 16].

To address the limitations inherent in current research, we present a novel joint optimization problem on both device and training scheduling to minimize the training cost of federated learning with a convergence guarantee. The main contributions of our paper are as follows:

1) We introduce a Multi-Group Transmission (MGT) scheme on Orthogonal Frequency-Division Multiple Ac-

This paper was supported in part by the National Science Foundation under Grants 1943486, 1946231, 2110722, 2246757, 2309549, 2315612, and 2332011. *(Corresponding author: Xiangwei Zhou.)*

X. Chen and X. Zhou are with the Division of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, LA, 70803, USA (e-mail: {xchen87, xwzhou}@lsu.edu).

H. Zhang is with the Department of Mathematics, Louisiana State University, Baton Rouge, LA, 70803, USA (e-mail: zhc@lsu.edu).

M. Sun and T. Zhao is with the Division of Computer Science and Engineering, Louisiana State University, Baton Rouge, LA, 70803, USA (e-mail: msun@csc.lsu.edu, tzhao3@lsu.edu).

This paper has been presented in part at the IEEE International Conference on Communications (ICC), June 2024. [1]

Copyright (c) 20xx IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

cess (OFDMA) to increase the number of participants per training round, reduce stochastic variance, and speed up model convergence. To address device scheduling challenges in the MGT scheme, we formulate it as a Hybrid Flow Shops (HFS) problem and propose a group-based Johnson's rule. Additionally, we develop the Group Scheduling on OFDMA (GS-OFDMA) protocol to address various time bottlenecks during training, optimizing the number of participants per round by ordering and grouping them based on their time costs.

2) We formulate a new cost-minimization problem to jointly optimize the parameters of device scheduling and training scheduling. Control variables include the participant selection probability, the number of participants, and the number of local iterations. To make the problem practical, both system and data heterogeneity are factored in. We develop an iterative algorithm integrating coordinate descent and polyhedral active set algorithm (PASA) to solve the challenging mixed integer nonlinear programming (MINLP) problem.
3) Through experiments on real datasets, we evaluate the total cost and convergence speed of the model training configured based on the solution to our optimization. Experiment results demonstrate that our optimal participant selection probability reduces the total cost by at least 35% across different datasets and data distributions in comparison with random selection.

II. Related Work

Reducing the resource cost of federated learning training, including time and energy, is crucial for practical implementation in resource-constrained environments such as IoT networks. FedAvg [2] addresses this by scheduling a subset of clients for model training in each round and allowing multiple local iterations, which significantly reduces communication overhead while maintaining good training performance.

However, due to data heterogeneity among clients, with unique and non-I.I.D. private data, FedAvg's random client selection leads to sub-optimal model performance and resource utilization [17]. To address this, various device scheduling methods based on importance sampling have been proposed. In [14], a probabilistic device scheduling method is proposed, which measures the importance of clients by the gradient norms of local updates and assigns higher participant probabilities to those with larger gradient norms. Other metrics to measure the data importance have also been studied, including the volume of local dataset [18], training loss divergence [19], and model divergence [20].

While these methods improve model convergence and reduce communication rounds, they overlook a crucial factor: system heterogeneity. Differences in computational power, memory, battery life, and network bandwidth among devices can lead to varying training times and energy consumption. Ignoring this can increase training time and inefficient resource use, reducing federated learning's cost efficiency. Heuristic methods that consider both data and system heterogeneity have been proposed. Oort [4] calculates both statistical utility (data importance) and system utility (training efficiency) to rank and select high-utility clients, enhancing time-to-accuracy performance. PyramidFL [5] further fine-tunes statistical and system utilities to allow non-straggler participants to train in more iterations and submit partial updates. Conversely, to promote fairness, FCFL [21] loosens the network capacity constraints to allow clients with poor networks to be selected and schedule participants with high movement relevance to the global model.

Another approach to improving cost efficiency in federated learning is to formulate a cost optimization problem with control variables. The objective function of the optimization is to minimize the time cost [6, 7, 10, 15, 16, 22], energy cost [8, 9], or both [11, 12]. The main control variables are device scheduling and training scheduling.

Device scheduling involves selecting participants and determining transmission policies for local models. For example, [6] discusses round robin and proportional fair schemes. CMFL [10] uses feedback to schedule only highly relevant clients. An optimal probabilistic client selection scheme in [7] minimizes the weighted sum of training time and loss.

Training scheduling optimizes parameters such as the number of global rounds, local iterations, and participants per round. In [11], an optimization problem minimizes training loss within time and energy budgets by adaptively choosing optimal local iterations and communication rounds. Similarly, [12] finds optimal values for local iterations and participants to minimize costs while ensuring model convergence. In hierarchical federated learning, [13] optimizes training scheduling through joint resource allocation and edge association.

Joint optimization of training and device scheduling is less explored. [22] presents an online non-linear program to minimize resource usage by jointly controlling participant selection and local training iterations, though it only addresses binary selection and ignores data heterogeneity. To address both system and data heterogeneity, we propose a new cost-minimization problem that jointly optimizes device and training scheduling with model convergence guarantee.

III. System Model

In this section, we first introduce federated learning with probabilistic participant selection. Then we present a new multi-group transmission scheme to schedule the model updates.

A. Federated Learning with Client-Selection Probability

Assume that there are one server and N clients with index set $\mathcal{N} = \{1, 2, ..., N\}$ in a federated learning system. Client i possesses a unique and private dataset $\mathcal{D}_i = \{\xi_j^i \mid j = 1, 2, ..., |\mathcal{D}_i|\}$ of size $|\mathcal{D}i|$, where ξ_j^i denotes the j-th data sample of client i. The composite dataset across all clients is represented as $\mathcal{D} = \bigcup_{i\in\mathcal{N}} \mathcal{D}_i$ and of size $|\mathcal{D}|$. The local loss function on dataset $\mathcal{D}_i$ is defined by

$$F_i(x) \coloneqq \frac{1}{|\mathcal{D}_i|} \sum_{\xi_j^i \in \mathcal{D}_i} F_i(x; \xi_j^i), \tag{1}$$

where x represents the model parameters.

The objective of a federated learning system is to find an

optimal model parameter x to minimize a global loss function $f(x)$ on all the distributed datasets, defined by

$$\min_x f(x) \coloneqq \sum_{i=1}^{N} d_i F_i(x), \tag{2}$$

where $d_i = |\mathcal{D}_i|/|\mathcal{D}|$ denotes the ratio of data volume at client i and $\sum_{i=1}^{N} d_i = 1$.

In this paper, we study federated learning with probabilistic participant selection, shown in Algorithm 1. Suppose there are in total T rounds of communications between participants and the server to achieve model convergence, and the stopping criterion is given by

$$\mathbb{E}(f(x_T)) - f^* \leq \epsilon, \tag{3}$$

where f^* denotes the minimum global loss and $\epsilon > 0$.

Algorithm 1: Generalized FedAvg with Probabilistic Participation

Input: $x_0, \gamma, I, M, T, \mathbf{p}$
Output: x_T

for $t = 0, 1, ..., T-1$ **do**
  Select M participants according to $\mathbf{p}$ with replacement to form $\mathcal{M}^{(t)}$
  for $i \in \mathcal{M}^{(t)}$ *in parallel* **do**
    $y^i_{t,0} \leftarrow x_t$
    for $j = 0, 1, ..., I-1$ **do**
      $y^i_{t,j+1} \leftarrow y^i_{t,j} - \gamma \nabla F_i(y^i_{t,j}, \xi^i_j)$
    end
  end
  $x_{t+1} \leftarrow \sum_{i \in \mathcal{M}^{(t)}} \frac{d_i}{M p_i} y^i_{t,I}$
end

In each round, a training process consists of model broadcasting, local training, model transmission, and aggregation. Initially, the server chooses M clients to participate in the t-th training round, forming participant set $\mathcal{M}^{(t)}$ based on the participant selection probability $\mathbf{p} = [p_1, p_2, ..., p_N]$, and sends out the global model parameters x_t to these participants. Each participant then synchronizes its local model with the global one, trains the model using the stochastic gradient descent (SGD) algorithm, updates the model over I iterations, and transmits the updated model parameters back to the server. Finally, the server aggregates all received updates to renew the global model by

$$x_{t+1} = \sum_{i \in \mathcal{M}^{(t)}} \frac{d_i}{M p_i} y^i_{t,I}. \tag{4}$$

B. Multi-Group Transmission

Wireless transmission models in federated learning mainly consist of time-sharing [3, 12] and frequency-sharing protocols [9, 23]. Time-sharing protocols, such as Time-Division Multiple Access (TDMA), schedule the model submission of participants in distinct time slots where each participant occupies all the frequency resources in the assigned time slot. In contrast, frequency-sharing protocols subdivide the available bandwidth into multiple frequency sub-carriers, which allows participants

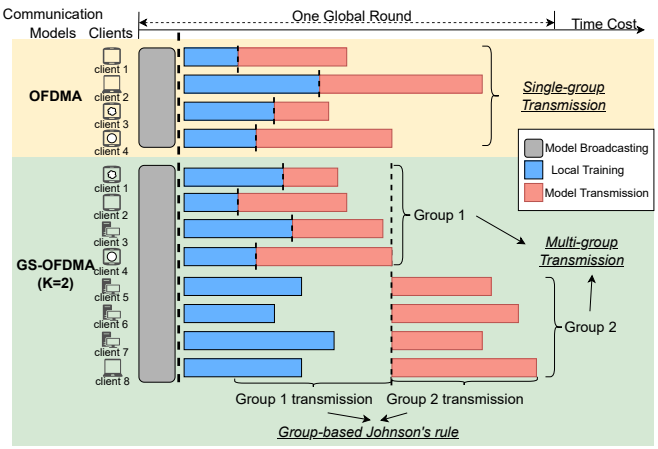


Fig. 1: Comparison of device scheduling methods. Unlike OFDMA, GS-OFDMA adopts multi-group transmission, allowing more participants in one global round. To reduce the time cost of multi-group transmission, GS-OFDMA employs the group-based Johnson's rule to group and order the model transmission.

to submit their model updates simultaneously. In terms of transmission efficiency, it has been shown that frequency-sharing protocols, such as Orthogonal Frequency-Division Multiple Access (OFDMA), achieve higher data rates than time-sharing protocols [9, 24]. However, existing work using frequency-sharing models [3, 9, 16] only considers Single-Group Transmission (SGT), i.e., the number of participants per round equals the number of sub-channels of the frequency bandwidth.

Existing research indicates that increasing the number of participants in each round can accelerate model convergence, leading to potential cost savings [2]. In light of this, we introduce a novel Multi-Group Transmission (MGT) scheme, leveraging OFDMA to reduce the stochastic variance and speed up the model convergence. Our approach significantly differs from the existing SGT methods, as shown in Figure 1. Specifically, the number of participants in MGT can greatly exceed the number of available sub-channels. Participants will be grouped and MGT in one global round will be implemented, which is organized through a specific scheduling process. We will elaborate on the scheduling details in Section IV-A.

We provide a formal description of the MGT scheme. Consider an OFDMA system where the total bandwidth is partitioned into S sub-channels, where each sub-channel is exclusively used by one participant. Unlike SGT methods where $M = S$, our MGT selects $M = KS$ participants according to participation probability $\mathbf{p}$, where K denotes the number of groups and $1 \leq K \leq \lceil \frac{N}{S} \rceil, K \in \mathbb{Z}^+$.

IV. Multi-Group Device Scheduling and Cost-efficient Federated Learning

In this section, we first propose a group-based Johnson's rule to address the challenges of device scheduling in the MGT scheme and introduce a comprehensive multi-group device scheduling framework. Then we formulate a joint device and training scheduling optimization problem to minimize the total

cost of the training process.

A. Group-based Johnson's Rule

With the introduction of the MGT scheme, a critical question emerges: how to optimally group participants and schedule group transmissions to minimize the time cost of a single-round training. Addressing this issue is vital for developing cost-efficient federated learning since various participants could have significantly different local training times and model transmission times.

We reframe the scheduling challenge in the MGT scheme as a Hybrid Flow Shops (HFS) problem [25]. In this model, each participant's task is conceptualized as a job with two stages, resulting in a total of M such jobs. Stage 1 is to train the model locally, and Stage 2 is to transmit the model updates. The processing times of Stage 1 and Stage 2 could be different, but Stage 2 can only start after Stage 1 is finished. For the local training stage (Stage 1), there are M identical and parallel machines, each capable of processing one job, with all machines initiating simultaneously. It is worth noting that "machines" here simply refer to abstract entities that process these jobs in Stage 1 and Stage 2 and do not indicate that clients in federated learning have identical hardware or capabilities. In the model transmission stage (Stage 2), the number of available machines is limited to S, corresponding to the S sub-channels for transmission. The objective of the HFS problem is to minimize the makespan, which is the duration from the start of local training to the completion of model transmission. This is achieved by strategically scheduling the M participants into K ordered groups.

Solving the HFS problem presents a significant challenge, particularly due to its computational complexity in scenarios with large numbers of participants (M) and groups (K). An exhaustive search approach becomes impractical in such contexts. Specifically, there are in total $\frac{M!}{(S!)^K}$ possible ways to partition M participants into K ordered groups, which implies a factorial time complexity. Moreover, the complexity is compounded in our specific HFS problem, which involves two distinct processing stages, each hosting multiple machines with no allowance for preemption. This problem has been established as NP-complete [26], underscoring the considerable computational effort required to find an optimal solution.

Branch and bound, while effective for some HFS problems, struggles with the large-scale job counts typical in federated learning, often exceeding 1,000 jobs [27]. In federated learning environments, where job counts can surpass tens of thousands [28], the time and computational expense of branch and bound make it impractical for efficient training processes that require quick scheduling decisions.

Johnson's rule is a heuristic algorithm that schedules jobs based on their ranking scores. The scheduling method provides a minimum makespan of a special HFS problem when there are two stages and only one machine at each stage [29]. Johnson's rule states that if a job's Stage-1 time is shorter than its Stage-2 time, it should be scheduled earlier; if the job's Stage-2 time is shorter, it should be scheduled later. It exhibits good performance in the case with multiple machines in Stage 1 and only one machine in Stage 2 [26].

To adapt Johnson's rule to solving our HFS problem, where Stage 2 comprises S machines, we propose a group-based Johnson's rule. Let $\delta_{i,l}$ denote the one-iteration training time of client i in each round and $\delta_{i,a}^{(t)}$ be the model transmission time of client i in round t. Then $\delta_{i,l}I$ and $\delta_{i,a}^{(t)}$ correspond to the processing times of job i in Stage 1 and Stage 2, respectively. Accordingly, the group-based Johnson's rule can be described in two steps: calculating the ranking scores of participants and modifying the rank. Details of group-based Johnson's rule are given in Algorithm 2. With the formula in Step 2 of Algorithm 2, a smaller $g_i^{(t)}$ naturally indicates that the i-th client should be scheduled earlier for the model transmission if its local training is faster, while a larger $g_i^{(t)}$ suggests that transmission is shorter, and it should be scheduled later. Algorithm 2 guarantees that the first group achieves the minimum local training time. This is a significant modification from the standard Johnson's rule, which typically adjusts the position of only one job.

Algorithm 2: Group-based Johnson's Rule

Input: Processing times of participants $\{(\delta_{i,l}I, \delta_{i,a}^{(t)}), i \in \mathcal{M}^{(t)}\}$, number of sub-channels S

1 **for** *each participant* i **do**

2 Calculate ranking score $g_i^{(t)}$ using the formula:

$$g_i^{(t)} = \frac{\text{sign}(\delta_{i,l}I - \delta_{i,a}^{(t)})}{\min(\delta_{i,l}I, \delta_{i,a}^{(t)})};$$

3 **end**

4 Sort participants based on $g_i^{(t)}$ in ascending order;

5 Identify S participants with the smallest $\delta_{i,l}I$;

6 Prioritize these S participants at the top of the ranking;

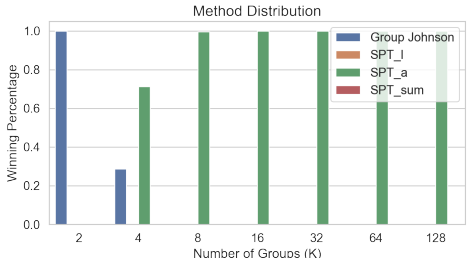
B. Group Scheduling on OFDMA (GS-OFDMA)

In practical federated learning scenarios, the time bottleneck varies and depends on the particular application. We identify three cases based on the relative lengths of local training time (Stage 1 processing time) and model transmission time (Stage 2 processing time):

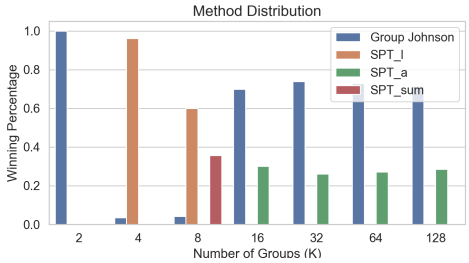
1) Stages 1 and 2 Comparable.
2) Stage 1 Dominant: local training time significantly exceeds model transmission time.
3) Stage 2 Dominant: model transmission time is substantially longer than local training time.

To evaluate the effectiveness of our group-based Johnson's rule across these cases, we conduct simulations of a two-stage HFS problem. With a fixed group size S, we generate random processing times for Stage 1 and Stage 2 for each job in one trial and the best method is identified across thousands of independent trials. The objective is to identify the most efficient heuristic scheduling method that minimizes the makespan.

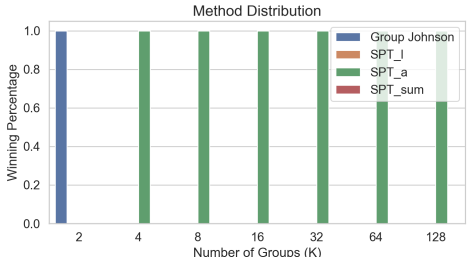
In addition to our proposed group-based Johnson's rule, we investigate the Shortest Processing Time first (SPT) rule, a notable heuristic algorithm in single-machine environments



(a) Stages 1 and 2 Comparable.



(b) Stage 1 Dominant.



(c) Stage 2 Dominant.

Fig. 2: Simulation results of four heuristic scheduling methods in three scenarios of time bottlenecks. SPT_a achieves minimum makespan in the Comparable and Stage 2 Dominant cases while group-based Johnson's rule is the best in the Stage 1 Dominant case. Some bars do not appear in the plot as their winning percentages are zero.

known for minimizing weighted completion time [25]. The SPT rule prioritizes jobs based on their shortest processing times. In our two-stage HFS problem, we define three variants of the SPT rule: SPT_l, SPT_a, and SPT_{sum}, which apply the SPT rule to $\delta_{i,l}I$, $\delta_{i,a}^{(t)}$, and the sum $(\delta_{i,l}I + \delta_{i,a}^{(t)})$, respectively.

Our simulations reveal that the best heuristic scheduling method for our HFS problem depends on the specific case. As shown in Figure 2, with a fixed number of sub-channels S, a distinct method emerges when the number of groups K exceeds a certain threshold, i.e., $K > 8$ in the simulations. Specifically, in the Comparable and Stage 2 Dominant cases, SPT_a achieves the minimum makespan in all the trials. However, in the Stage 1 Dominant case, the proposed group-based Johnson's rule outperforms other methods in 70% of the trials. It is also noteworthy that when there are only two groups ($K = 2$), the group-based Johnson's rule is the most effective scheduling method across all cases.

Therefore, combining our MGT scheme with the group-based Johnson's rule, we propose a comprehensive device scheduling framework on OFDMA, named Group Scheduling on OFDMA protocol (GS-OFDMA). This framework is designed to effectively address all three time bottleneck cases in federated learning environments. GS-OFDMA is described by the following protocol.

1) The server estimates local training times $\{\delta_{i,l} : i \in \mathcal{N}\}$.
2) In round t, the server estimates the model transmission times of participants $\{\delta_{i,a}^{(t)} : i \in \mathcal{M}^{(t)}\}$.
3) The server identifies the time bottleneck. For Stage 1 Dominant case, the server sorts the participants according to the group-based Johnson's rule. For Comparable and Stage 2 Dominant cases, the SPT_a rule is applied.
4) Sorted participants are divided into K groups according to $\{\delta_i^{(t)}\}$, i.e., $G_i = \{(i-1)S+1, (i-1)S+2, ..., iS\}, i = 1, 2, ..., K$.
5) Participants in the first group submit their updates immediately after finishing local training. Model transmission proceeds from G_1 to G_K, and group G_i, $i \geq 2$, starts transmitting only if all the updates of group G_{i-1} are received by the server. All the sub-channels are occupied only by one group.
6) Model transmission of this round ends after K groups finish uploading updates.

The detailed methodology of GS-OFDMA is outlined in Algorithm 3.

Algorithm 3: Group Scheduling on OFDMA (GS-OFDMA)

Input: $M, T, S, \mathbf{p}$
Server estimates local training time $\{\delta_{i,l} : i \in \mathcal{N}\}$;
for $t = 0, 1, ..., T-1$ **do**
  Server selects M participants according to $\mathbf{p}$ with replacement to form $\mathcal{M}^{(t)}$;
  Server estimates transmission time $\{\delta_{i,a}^{(t)} : i \in \mathcal{M}^{(t)}\}$;
  if $\sum_{i\in\mathcal{M}^{(t)}} \delta_{i,l} \gg \sum_{i\in\mathcal{M}^{(t)}} \delta_{i,a}^{(t)}$ **then**
    Sever sorts participants $\mathcal{M}^{(t)}$ by group-based Johnson's rule in Algorithm 2;
  end
  else
    Sever sorts participants $\mathcal{M}^{(t)}$ by SPT_a rule;
  end
  Server groups participants such that the index set $G_i = \{(i-1)S+1, (i-1)S+2, ..., iS\}, i = 1, 2, ..., K$;
  Groups are scheduled to submit updates in order of $G_1, G_2, ..., G_K$;
  Server updates the global model using (4);
end

C. Cost Evaluation of Federated Learning

Local training and model transmission are two major sources of cost in federated learning. In terms of cost, time and energy costs are crucial for efficient federated learning system design. Similar to existing work in wireless federated learning [12], we assume that each mobile device has relatively stable computation capacity but faces dynamic communication environments, i.e., the cost of local training is the same across

different rounds while the cost of model transmission is time-varying.

1) Time Cost

Let $\delta_{i,l}$ denote the one-iteration training time of client i in each round and $\delta_{i,a}^{(t)}$ be the model transmission time of client i in round t. Then the time cost of client i in round t is

$$\delta_i^{(t)} = \delta_{i,l} I + \delta_{i,a}^{(t)}, \tag{5}$$

where I is the number of iterations in the local training.

Suppose $\Delta_k^{(t)}$ is the total time cost after the first k groups finish the local training and model transmission in round t. For the first group ($K = 1$), $\Delta_1^{(t)} = \delta_S^{(t)}$ is the time cost of the straggler in G_1. For the following groups ($K \geq 2$), they need to wait for the previous group to finish the model transmission before sending their model updates to the server. Therefore, we can calculate $\Delta_k^{(t)}$ by

$$\Delta_k^{(t)} = \max\{\Delta_{(k-1)}^{(t)}, \delta_{kS,l} I\} + \delta_{kS,a}^{(t)}, k \in \{2, ..., K\}, \tag{6}$$

where $\delta_{kS,l} = \max\{\delta_{i,l}\}$ and $\delta_{kS,a}^{(t)} = \max\{\delta_{i,a}^{(t)}\}$, $i \in G_k$, are the largest time costs for local training per iteration and model transmission, respectively.

As in [12], we assume that the local training time of the current group is less than the makespan of previous groups. Then we have

$$\max\{\Delta_{(k-1)}^{(t)}, \delta_{kS,l} I\} = \Delta_{(k-1)}^{(t)}, k \in \{2, ..., K\}.$$

Therefore, after the first group finishes the local training, the communication channel will be occupied until all the participants finish the model transmission. The time cost for round t can be calculated as the summation of the longest computation time in the first group and the longest communication times of all groups:

$$\Delta_K^{(t)} = \delta_{S,l} I + \sum_{k=1}^{K} \delta_{kS,a}^{(t)}. \tag{7}$$

As a result, the total time cost of T global rounds can be calculated as

$$\Delta = \sum_{t=1}^{T} (\delta_{S,l} I + \sum_{k=1}^{K} \delta_{kS,a}^{(t)}). \tag{8}$$

2) Energy Cost

Similar to the time cost, let $e_{i,l}$ be the one-iteration normalized energy cost of client i in each round and $e_{i,a}^{(t)}$ be the normalized energy cost of model transmission in round t. The energy cost of client i in round t is given by

$$e_i^{(t)} = e_{i,l} I + e_{i,a}^{(t)}. \tag{9}$$

Since device scheduling schemes do not change the total energy cost, the cumulative energy cost of T rounds can be expressed as

$$e = \sum_{t=1}^{T} \sum_{i \in \mathcal{M}^{(t)}} e_i^{(t)}. \tag{10}$$

D. Cost Minimization Problem

Cost evaluation reveals that the efficiency is influenced by several key factors: the participation-selection probability $\mathbf{p}$, number of global rounds T, number of scheduled groups K in GS-OFDMA, and number of local iterations I. Here, $\mathbf{p}$ is the parameter for device scheduling, and (K, I, T) are for training scheduling. Moreover, the optimal solution to minimize time cost generally does not render the minimum energy cost, and vice versa. Therefore, we propose to integrate the time and energy costs of federated learning into one adjusted cost C_T, given by

$$C_T = \alpha \Delta + (1 - \alpha) e, \tag{11}$$

where $0 \leq \alpha \leq 1$ represents the relative importance of each cost and provides flexibility to the cost metric. Increasing α encourages time efficiency while decreasing α weights more on energy efficiency.

The objective is to minimize the adjusted cost of federated learning training while satisfying the model convergence requirement. Therefore, we formulate the joint device and training scheduling optimization problem as follows

$$\min_{\mathbf{p}, T, K, I} \quad \mathbb{E}(C_T) \tag{P1}$$

$$\text{s.t.} \quad \mathbb{E}(f(x_T)) - f^* \leq \epsilon, \tag{P1a}$$

$$\sum_{i=1}^{N} p_i = 1, \tag{P1b}$$

$$1 \leq K \leq \lceil \frac{N}{S} \rceil, K \in \mathbb{Z}^+, \tag{P1c}$$

$$1 \leq T \leq T_{max}, 1 \leq I \leq I_{max}, T, I \in \mathbb{Z}^+, \tag{P1d}$$

where T_{max} and I_{max} denote the maximum numbers of global rounds and local iterations, respectively.

V. Joint Device and Training Scheduling Optimization

In this section, we transform problem P1 into a more tractable form and develop an iterative algorithm to find the optimal solutions. First, we derive the analytical expression of $\mathbb{E}(C_T)$. After the approximation of constraint (P1a) with the convergence upper bound of Algorithm 1, we connect the objective and constraints with control variables $\mathbf{p}, K, T, I$ and formulate an alternative optimization problem, P2. Subsequently, we demonstrate that Problem P2 can be effectively solved following the estimation of the unknown parameters.

A. Analytical Expression of $\mathbb{E}(C_T)$

According to (11), we have

$$\mathbb{E}(C_T) = \alpha \mathbb{E}(\Delta) + (1 - \alpha) \mathbb{E}(e),$$

where analytical expressions of $\mathbb{E}(\Delta)$ and $\mathbb{E}(e)$ with respect to $\mathbf{p}$ and K are needed to make the problem tractable.

According to Algorithm 1, the probability of client i participating in round t can be given by

$$P(i \in \mathcal{M}^{(t)}) = 1 - (1 - p_i)^M \overset{(a)}{\approx} M p_i, \tag{12}$$

where (a) approximates $(1 - p_i)^M$ by its zero and first order terms.

Even though client i could be selected multiple times in one round, the energy cost will be counted only once (one-time local training and one-time model transmission). Therefore, we have

$$\mathbb{E}(e) = \mathbb{E}\left(\sum_{t=1}^{T} \sum_{i \in \mathcal{M}^{(t)}} e_i^{(t)} \right) = \sum_{t=1}^{T} \sum_{i=1}^{N} P(i \in \mathcal{M}^{(t)}) e_i^{(t)}$$

$$\approx \sum_{t=1}^{T}\sum_{i=1}^{N} M p_i e_i^{(t)} = KTS\sum_{i=1}^{N} p_i(e_{i,l}I + \bar{e}_{i,a}), \quad (13)$$

where the average energy cost of model transmission for client i is denoted by $\bar{e}_{i,a} = \frac{1}{T}\sum_{t=1}^{T} e_{i,a}^{(t)}$.

For $\mathbb{E}(\Delta)$, we have

$$\mathbb{E}(\Delta) = \mathbb{E}\left(\sum_{t=1}^{T}\left(\delta_{S,l}I + \sum_{k=1}^{K}\delta_{kS,a}^{(t)}\right)\right). \quad (14)$$

To find the expectation of the first term in (14) that denotes the longest local training time of the first group, we define q_i to be the probability of client i being selected in round t and with the longest training time of the first group, which is equivalent to that only clients $1, 2, ..., i$ are candidates in the first group. Therefore, we have

$$\begin{aligned} q_i &= \mathbb{P}(i \text{ is the straggler of group } G_1) \\ &= \sum_{m=1}^{S}\binom{S}{m} p_i^m \left(\sum_{j=1}^{i-1} p_j\right)^{S-m} = \left(\sum_{j=1}^{i} p_j\right)^S - \left(\sum_{j=1}^{i-1} p_j\right)^S, \end{aligned} \quad (15)$$

where the last equality uses the binomial theorem. Then the first term in (14) becomes

$$\mathbb{E}\left(\sum_{t=1}^{T}\delta_{S,l}I\right) = \sum_{t=1}^{T}\sum_{i=1}^{N} q_i\delta_{i,l}I. \quad (16)$$

The second term $\{\delta_{kS,a}^{(t)} : k = 1, ..., K\}$ in (14) denotes the largest model transmission time in each scheduling group and can vary in different rounds. It is difficult to enumerate all the combinations of $\{\delta_{kS,a}^{(t)}\}$ and their corresponding probabilities. Since the model transmission time of each group is independent, we approximate the transmission time of each group by that of the first group and have

$$\mathbb{E}\left(\sum_{t=1}^{T}\sum_{k=1}^{K}\delta_{kS,a}^{(t)}\right) = \sum_{t=1}^{T} K\sum_{i=1}^{N} q_i\delta_{i,a}^{(t)}. \quad (17)$$

With (16) and (17), we have

$$\mathbb{E}(\Delta) = T\sum_{i=1}^{N} q_i\left(\delta_{i,l}I + K\bar{\delta}_{i,a}\right), \quad (18)$$

where $\bar{\delta}_{i,a} = \frac{1}{T}\sum_{t=1}^{T}\delta_{i,a}^{(t)}$ denotes the average time cost of model transmission for client i.

The above formulation is still very hard to optimize because q_i includes the polynomial term of p_i with the order of S. For analytical tractability, we approximate q_i with $q_i \approx p_i$. Note that there are two cases where p_i is equivalent to q_i in terms of $\mathbb{E}(\Delta)$: 1) When $S = 1$, we can easily show that $q_i = p_i$. 2) When $\delta_{i,l} = \delta_{j,l}$ and $\bar{\delta}_{i,a} = \bar{\delta}_{j,a}$ for $i \neq j$, we have

$$\sum_{i=1}^{N} q_i\left(\delta_{i,l}I + K\bar{\delta}_{i,a}\right) = \sum_{i=1}^{N} p_i\left(\delta_{i,l}I + K\bar{\delta}_{i,a}\right).$$

Therefore, we formally define the approximated $\tilde{\mathbb{E}}(\Delta)$ as

$$\tilde{\mathbb{E}}(\Delta) = T\sum_{i=1}^{N} p_i\left(\delta_{i,l}I + K\bar{\delta}_{i,a}\right). \quad (19)$$

With the analytical expressions of $\mathbb{E}(e)$ and $\tilde{\mathbb{E}}(\Delta)$, the approximation of total cost $\tilde{\mathbb{E}}(C_T)$ is

$$\begin{aligned} \tilde{\mathbb{E}}(C_T) &= \alpha\tilde{\mathbb{E}}(\Delta) + (1-\alpha)\mathbb{E}(e) \\ &= T\sum_{i=1}^{N} p_i(Iw_{i,l}(K) + w_{i,a}(K)), \end{aligned} \quad (20)$$

where $w_{i,l}(K)$ and $w_{i,a}(K)$ are functions of K defined as follows:

$$\begin{aligned} w_{i,l}(K) &= \alpha\delta_{i,l} + (1-\alpha)KSe_{i,l}, \\ w_{i,a}(K) &= K\left(\alpha\bar{\delta}_{i,a} + (1-\alpha)S\bar{e}_{i,a}\right). \end{aligned} \quad (21)$$

Here, $w_{i,l}(K)$ and $w_{i,a}(K)$ can also be regarded as "pseudo costs" of local training and model transmission of client i, respectively.

B. Approximation Optimization Problem of P1

To approximate the ϵ-convergence constraint, we first make some commonly used assumptions about local loss functions $\{F_i\}$ [11, 30].

Assumption 1. *$F_i(x)$ is μ-strongly convex, i.e., $F_i(x) \geq F_i(y) + (x-y)^T\nabla F_i(y) + \frac{\mu}{2}\|x-y\|_2^2$ for all x and y.*

Remarks: While the μ-strong convexity assumption is restrictive and does not accurately characterize the loss landscapes of modern non-convex deep neural networks, it is widely used to derive interpretable convergence guarantees and enable tractable analysis [11, 30]. Moreover, in the experiments, we will empirically demonstrate that our method remains robust in practical non-convex settings.

Assumption 2. *The gradient of $F_i(x)$ is L-Lipschitz continuous: for any $x, y \in dom(F_i)$, we have $\|\nabla F_i(x) - \nabla F_i(y)\| \leq L\|x-y\|$.*

Assumption 3. *The variance of the stochastic gradient of $F_i(x)$ is bounded, i.e., $\mathbb{E}\left\|\nabla F_i(x,\xi_j^i) - \nabla F_i(x)\right\|^2 \leq \delta_i^2, \xi_j^i \in \mathcal{D}_i$.*

Assumption 4. *The expected second moment of $\nabla F_i(x)$ is bounded: for any data sample $\xi_j^i \in \mathcal{D}_i$ and when there exists a constant $G_i > 0$, we have $\mathbb{E}(\left\|\nabla F_i(x,\xi_j^i)\right\|^2) \leq G_i^2, \forall x \in dom(F_i)$.*

With the above assumptions, we have

Theorem 1. *Let Assumptions 1 to 4 hold, $\gamma = \max\{\frac{8L}{\mu}, E\}$, and decaying learning rate $\eta_t = \frac{2}{\mu(\gamma+t)}$, where t denotes the index of global rounds. Then federated learning with participation-selection probability* **p** *satisfies*

$$\mathbb{E}(f(x_T)) - f^* \leq \frac{1}{T}\left[AI\left(\frac{1}{KS}\sum_{i=1}^{N}\frac{C_i}{p_i} + D\right) + \frac{B}{I}\right], \quad (22)$$

where $A, B, \{C_i\}$, and D are defined as follows: $A = \frac{8L}{\mu^2}$, $C_i = d_i^2G_i^2$, $D = 2\sum_{i=1}^{N} d_iG_i^2$, $B = \frac{L}{\mu^2}\left(2\sum_{i=1}^{N} d_iG_i^2 + 12L\Gamma + 4L\mu\|x_0 - x^*\|^2\right)$ with $\Gamma = \left(f^* - \sum_{i=1}^{N} d_iF_i^*\right)$.

A, B, $\{C_i\}$, and D are constants related to the local loss functions and data heterogeneity among clients. The details of

convergence analysis can be found in [16].

To re-formulate Problem P1, we use (22) to replace the convergence constraint and (20) as the objective function. We have

$$\min_{\mathbf{p},K,T,I} \quad T\sum_{i=1}^{N} p_i(Iw_{i,l}(K)+w_{i,a}(K)) \tag{P2}$$

$$\text{s.t.} \quad \frac{1}{T}\left[AI\left(\frac{1}{KS}\sum_{i=1}^{N}\frac{C_i}{p_i}+D\right)+\frac{B}{I}\right]\le\epsilon, \tag{P2a}$$

$$\text{(P1b), (P1c), (P1d).}$$

Note that the feasible set of Problem P2 is smaller than that of Problem P1, i.e., any solution to P2 is also the solution to P1.

C. Solution to P2

With continuous ($\mathbf{p}$) and integer (K,T,I) variables, P2 is a Mixed Integer Nonlinear Programming (MINLP) problem, which is difficult to solve directly in general. We present an iterative algorithm to effectively solve Problem P2 by formulating two sub-problems P4 and P5 and iteratively updating (K,I) and $\mathbf{p}$, as shown in Algorithm 4.

Algorithm 4: Iterative Algorithm for Solving Problem P2

Input: $A, B, \{C_i\}, D, S, \{w_{i,l}\}, \{w_{i,a}\}$
Output: $\mathbf{p}^*, K^*, T^*, I^*$

1 Relax the variables K,T,I such that they are continuous: $K,T,I\in\mathbb{R}^+$;
2 Substitute T in the objective function of Problem P2 using Equation (23) to formulate Problem P3;
3 Initialize a feasible solution $(K_0, I_0, \mathbf{p}_0)$ of Problem P3;
4 Set iteration counter $l=0$;
5 **while** *Objective value of Problem P3 is not decreasing* **do**
6 For the given vector $\mathbf{p}_l$, solve Problem P4 to get (K_{l+1}, I_{l+1});
7 Using (K_{l+1}, I_{l+1}), solve Problem P5 to get $\mathbf{p}_{l+1}$;
8 Compute the optimal T_{l+1} using Equation (23);
9 Increment the iteration counter $l \leftarrow l+1$;
10 **end**

We first apply linear programming relaxation to convert integer variables K,T,I to be continuous variables, i.e., $K,T,I\in\mathbb{R}^+$. Observing T in the objective of P2 and constraint (P2a), we have that the optimal T^* minimizing the objective of P2 also maximizes the LHS of constraint (P2a). T^* satisfies

$$T^*=\frac{1}{\epsilon}\left[AI\left(\frac{1}{KS}\sum_{i=1}^{N}\frac{C_i}{p_i}+D\right)+\frac{B}{I}\right]. \tag{23}$$

Then Problem P2 is converted to

$$\min_{\mathbf{p},K,I} \quad \left(\frac{AI}{KS}\sum_{i=1}^{N}\frac{C_i}{p_i}+\frac{AI^2D+B}{I}\right) \cdot\left(\sum_{i=1}^{N}p_i(Iw_{i,l}(K)+w_{i,a}(K))\right) \tag{P3}$$

$$\text{s.t.} \quad \sum_{i=1}^{N}p_i=1 \tag{P3a}$$

$$1\le K\le N/S, \tag{P3b}$$

$$1\le I\le I_{max}. \tag{P3c}$$

Achieving the global minimizer of Problem P3 is challenging due to its non-convex nature. Notably, the variables $(K,I,\mathbf{p})$ are independent within the constraints, with $\mathbf{p}$ impacting only the coefficient terms of K and I. Therefore, we decompose Problem P3 into two sub-problems and effectively solve the sub-problems to obtain the suboptimal solution to Problem P3.

Given any $\mathbf{p}$, the first sub-problem with (K,I) is

$$\min_{K,I} \quad \left(\frac{a_0I}{K}+\frac{a_1I^2+B}{I}\right)(a_2KI+a_3K+a_4I) \tag{P4}$$

$$\text{(P3b), (P3c),}$$

where $a_0=\frac{A}{S}\sum_{i=1}^{N}\frac{C_i}{p_i}, a_1=AD, a_2=(1-\alpha)S\sum_{i=1}^{N}p_ie_{i,l}, a_3=\sum_{i=1}^{N}p_i(\alpha\bar{\delta}_{i,a}+(1-\alpha)S\bar{e}_{i,a})$ and $a_4=\alpha\sum_{i=1}^{N}p_i\delta_{i,l}$ are positive constants.

We present the following theorem for Problem P4.

Theorem 2. *Given positive constants a_0,a_1,a_2,a_3,a_4, suppose a set $H=[1,N/S]\times[1,I_{max}]$ and a function $h:H\to\mathbb{R}$ is defined as*

$$h(K,I)=\left(\frac{a_0I}{K}+\frac{a_1I^2+B}{I}\right)(a_2KI+a_3K+a_4I).$$

Then H is a biconvex set and h is a biconvex function on H. Therefore, Problem P4 is a biconvex optimization problem.

The proof can be found in Appendix A. Thanks to its biconvexity, we can use the coordinate descent algorithm to find the local minimum by alternatively updating K and I while fixing one of them and solving the corresponding convex optimization problem.

With Theorem 2, we find that the solution to Problem P4 exhibits several important properties that provide insights into how K and I affect the total cost $\tilde{\mathbb{E}}(C_T)$. We summarize some properties in Lemma 1.

Lemma 1. *Given the participation-selection probability $\mathbf{p}$, the following relationships between total cost $\tilde{\mathbb{E}}(C_T)$ and K,I hold:*

1) *If I is held constant and $\alpha=0$, the value of $\tilde{\mathbb{E}}(C_T)$ is monotonically increasing with K.*
2) *If I is held constant and $0<\alpha\le 1$, the value of $\tilde{\mathbb{E}}(C_T)$ initially decreases and subsequently increases as K increases.*
3) *If K is held constant and $0\le\alpha\le 1$, the value of $\tilde{\mathbb{E}}(C_T)$ initially decreases and subsequently increases as I increases.*

The proof can be found in Appendix B.

Given (K,I) from Problem P4, we have the second sub-problem given by

$$\min_{\mathbf{p}} \quad \left(b_0\sum_{i=1}^{N}\frac{C_i}{p_i}+b_1\right)\left(\sum_{i=1}^{N}p_i(Iw_{i,l}(K)+w_{i,a}(K))\right) \tag{P5}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} p_i = 1, \tag{P5a}$$

where $b_0 = \frac{AI}{KS}$ and $b_1 = AID + \frac{B}{I}$ are positive constants.

The challenge posed by Problem P5 is characterized by nonlinear optimization under polyhedral constraints. The Polyhedral Active Set Algorithm (PASA), as detailed by [31], is adopted to solve it. With $(K^*, I^*, \mathbf{p}^*)$, we update T by (23). The process continues by iteratively solving Problems P4 and P5 until the sequence of objective values of Problem P3 converges.

Regarding the computational complexity and scalability, each iteration in Algorithm 4 involves solving two subproblems: P4 and P5. Problem P4, a biconvex problem, is efficiently handled using a coordinate descent method with polynomial complexity per step. Problem P5 is solved with the highly effective Polyhedral Active Set Algorithm (PASA) [31] for polyhedral-constrained optimization. Both methods leverage matrix or vector sparsity in linear algebra computations, resulting in an overall computation cost that typically increases linearly (or at most polynomially) with the number of clients N. The algorithm's design ensures its scalability, with per-iteration complexity scaling linearly with N, making it feasible for large-scale federated learning scenarios.

D. Estimates of Unknown Parameters

As shown in Section V-C, there are multiple unknown parameters before solving the optimization problem, including time and energy costs $(w_{i,l}, w_{i,a})$, time cost adjustment term $\hat{\delta}$, model-relevant parameters (A, B), and data-relevant parameters $(\{C_i\}, D)$. We propose an empirical method to estimate those unknown parameters by running two trial experiments. The basic idea is to use different sets of parameters $(\mathbf{p}_{(a)}, K_{(a)}, I_{(a)}, \epsilon_{(a)})$ and $(\mathbf{p}_{(b)}, K_{(b)}, I_{(b)}, \epsilon_{(b)})$ to run two independent trial experiments and solve a system of equations using the convergence upper bound in (P2a).

Specifically, we run trial experiment a with $\mathbf{p}_{(a)} = [1/N, 1/N, ..., 1/N]$ corresponding to the random participation selection. We empirically choose $(K_{(a)}, I_{(a)}, \epsilon_{(a)})$ for the experimental settings and record $T_{(a)}$ when the global loss reduces to $\epsilon_{(a)}$. During the trial experiment, the participants are required to submit their model updates along with some statistics $(d_i, G_i, \delta_{i,l}, \bar{\delta}_{i,a}, e_{i,l}, \bar{e}_{i,a})$ to the server. Then the server can calculate C_i and D according to their definitions

$$C_i = d_i^2 G_i^2, \quad D = 2\sum_{i=1}^{N} d_i G_i^2. \tag{24}$$

For trial experiment b, we sample participants with $\mathbf{p}_{(b)} = [d_1, d_2, ..., d_N]$, choose $(K_{(b)}, I_{(b)}, \epsilon_{(b)})$ for the experimental settings, and record $T_{(b)}$. According to (P2a), we have

$$\begin{cases} AI_{(a)} \left(\frac{N \sum_{i=1}^{N} C_i}{K_{(a)} S} + D \right) + \frac{B}{I_{(a)}} = T_{(a)} \epsilon_{(a)}, \\ AI_{(b)} \left(\frac{\sum_{i=1}^{N} d_i G_i^2}{K_{(b)} S} + D \right) + \frac{B}{I_{(b)}} = T_{(b)} \epsilon_{(b)}. \end{cases} \tag{25}$$

We obtain A and B by solving the above system of equations.

Only a few rounds of training may be enough for the estimation process with large values of $\epsilon_{(a)}$ and $\epsilon_{(b)}$. Therefore, the computation cost and communication overhead for estimation can be low. The trained model in trial experiments can be reused as a good initial model in further experiments. Moreover, the estimation does not add much communication overhead, as only several statistics are added to clients' transmitted data.

VI. Experiments

In this section, we empirically evaluate the proposed cost-efficient federated learning algorithm with real-world datasets.

A. Experimental Settings

Testbed: Our testbed is built for general federated learning algorithms based upon the TensorFlow Federated infrastructure [32]. The experiments are run on an advanced computational cluster in the simulation environment.

Datasets and Predictive Model: We utilize the EMNIST_LETTERS [33] and FASHION_MNIST [34] datasets. EMNIST_LETTERS contains 26 lowercase English letter images, and FASHION_MNIST features ten different image classes. For our experiments, we split each dataset into private datasets for clients and a test dataset for evaluating our method. We employ LeNet-5 [35] as our classification model.

Data Heterogeneity: To simulate real-world data distributions in federated learning, we employ three data partitioning approaches: one I.I.D. and two non-I.I.D. configurations. *I.I.D.*: Training data is equally and randomly distributed among clients, ensuring a balanced class representation in each client's dataset. *Class*: Each client receives data from C randomly selected classes, where $C = 13$ for EMNIST_LETTERS and $C = 5$ for FASHION_MNIST. This approach represents 50% of classes in each dataset. *Dirichlet*: Data is distributed following a Dirichlet distribution with a parameter of 0.1, resulting in clients having varied data volumes and class distributions.

System Parameters: We set the total count of clients as $N = 10$ for FASHION_MNIST and $N = 40$ for EMNIST_LETTERS. During each iteration, participants are sampled according to the selection probability $\mathbf{p}$, following which every participant updates its local model across I iterations using Stochastic Gradient Descent (SGD) with a batch size of 256. We use the default Adam optimizer settings from TensorFlow.

For the communication system, we assume the Additive White Gaussian Noise (AWGN) channel with 2 MHz total bandwidth [9, 12]. The number of sub-channels is set as $S = 2$ for FASHION_MNIST and $S = 5$ for EMNIST_LETTERS, which leads to $K \in [1, 5]$ and $K \in [1, 8]$, respectively. The size of the transmitted model is 2 million bits. We simulate the energy costs of clients in millijoules (mJ) from a normal distribution. Specifically, $e_{i,l} \sim \mathcal{N}(10, 2)$ and $\bar{e}_{i,a} \sim \mathcal{N}(20, 4)$. To emulate heterogeneous time costs, we generate $\delta_{i,l}$ and $\bar{\delta}_{i,a}$ from different truncated normal distributions, given in milliseconds (ms). Specifically, for FASHION_MNIST, we generate $\delta_{i,l} \sim \mathcal{N}(5, 1)$ and $\bar{\delta}_{i,a} \sim \mathcal{N}(260, 100)$. For EMNIST_LETTERS, we have $\delta_{i,l} \sim \mathcal{N}(10, 2)$ and $\bar{\delta}_{i,a} \sim \mathcal{N}(560, 200)$. The parameter α defaults to 0.5 if not specified otherwise.

B. Experiment Results

1) Optimal Selection Probability

To evaluate our optimal solution of participant selection probability $\mathbf{p}*$, we fix (K, I) and solve Problem P3 to obtain $\mathbf{p}*$. Specifically, $(K, I) = (2, 120)$ for FASHION_MNIST and $(K, I) = (2, 40)$ for EMNIST_LETTERS. We adopt three commonly used baselines:

- *Uniform Selection* [2]. This scheme samples participants uniformly, i.e., $p_1 = p_2 = \cdots = p_N = 1/N$.
- *Norm Selection* [16]. Participants are chosen in accordance with p_i, where $p_i = \frac{d_i G_i}{\sum_{i=1}^{N} d_i G_i}$.
- *Ratio Selection* [17]. This technique selects participants with $p_i = d_i$.

In a broad range of data distributions and datasets, our scheduling method consistently outperforms the other methods in all metrics. As detailed in TABLE I, our approach achieves a minimum cost reduction of 35% across all the settings in comparison with the uniform method. Specifically, in terms of the total cost, our approach achieves a 40.97% to 61.92% reduction on the FASHION_MNIST dataset and even greater efficiency on the EMNIST_LETTERS dataset, reducing costs by 57.65% to 78.01%.

Notably, the robustness of our device scheduling stands in contrast to the setting-dependent performance of the norm and ratio methods. While norm and ratio methods show good performance in specific configurations, such as on the Dir data distribution, they can also incur higher costs than the uniform method under certain settings. In challenging scenarios characterized by high data heterogeneity (Dir) and a large number of clients (EMNIST_LETTERS), our method prevails, achieving nearly an 80% reduction in cost in comparison with the uniform method.

2) Convergence Performance

To evaluate the convergence performance of our solution (K^*, I^*), we demonstrate the training loss curves with respect to total cost C_T under *Dir* setting in Figures 3 and 4, where the optimal (K, I) by grid search and other (K, I) combinations are compared. Subfigures (a) in Figures 3 and 4 show the performance across different values of I. Subfigures (b) demonstrate the convergence behavior across different K.

As shown, the performance of our proposed solution (K^*, I^*) is notably distinct. Our solution exhibits a convergence trajectory that is on par with the optimal benchmark. Specially, in the stage characterized by a steep decline in loss, our solution (K^*, I^*) often reaches convergence quicker than the optimal grid search.

It is also noteworthy how various (K, I) combinations yield different convergence behaviors and overall total costs. Specifically, the number of participants, linear with K, compared with the number of local iterations I, appears to have a more significant influence on cost efficiency.

3) Optimal (K, I) versus Our Solutions

In Figure 5, the variation of total cost against different values of I is depicted for distinct settings of K. As observed, the total cost displays a non-monotonic trend with I: starting with an initial decrease, followed by an upward shift as I grows further. This behavior can be attributed to two primary factors.

At relatively smaller values of I, local data is not fully utilized, which results in local model under-fitting. This scenario requires a larger number of communication rounds to reach the target loss, thereby increasing the overall cost. Conversely, when I is set too high, there is a risk of over-utilizing the data. This is particularly noticeable for clients with limited data, where extensive local training can lead to overfitting. Consequently, this exhaustive local training increases both time and energy costs.

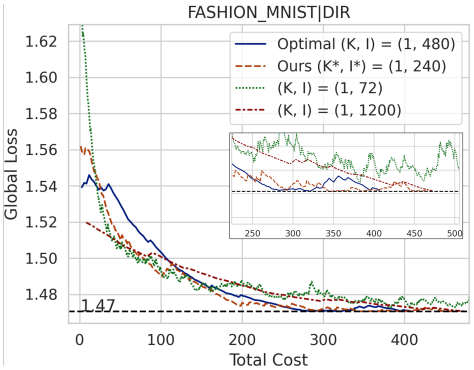
It is noteworthy that our proposed solution, denoted by (K^*, I^*), consistently achieves a total cost close to the optimal. Moreover, the sensitivity of the total cost with respect to I becomes especially obvious for larger values of K.

4) Ablation Study on K

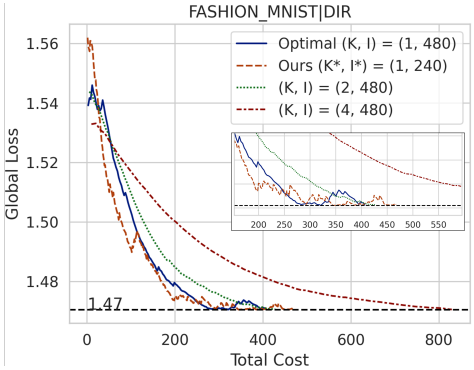
In Figure 6, the total cost is depicted with varying values of K, given fixed $I = 120$ for FASHION_MNIST and $I = 40$ for EMNIST_LETTERS. A pivotal observation, as underlined in Lemma 1, is the non-monotonic behavior of the total cost C_T with respect to K. The total cost C_T initially decreases and subsequently increases as K increases. This behavior is especially amplified under heterogeneous data distributions (Class and Dir settings), where an elevated K value correlates with a substantial surge in total cost. The findings suggest that

TABLE I: Cost evaluation for scheduling schemes (* denotes the cost reduced from the uniform method)

Data Dist.	Device Scheduling	FASHION_MNIST			EMNIST_LETTERS		
		Time Cost	Energy Cost	Total Cost	Time Cost	Energy Cost	Total Cost
IID	Uniform [2]	541.31	1870.07	1205.69	746.88	2007.96	1377.42
	Norm [16]	526.27	1816.45	1171.36	801.30	2160.90	1481.10
	Ratio [17]	556.83	1933.35	1245.09	560.12	1503.83	1031.98
	GS-OFDMA	**297.24 (-45.09%)***	**1040.79 (-44.34%)**	**669.02 (-44.51%)**	**312.56 (-58.15%)**	**854.11 (-57.46%)**	**583.34 (-57.65%)**
Class	Uniform [2]	458.33	1583.16	1020.75	961.03	2581.09	1771.06
	Norm [16]	598.55	2048.02	1323.29	722.63	1936.35	1329.49
	Ratio [17]	462.18	1606.59	1034.39	857.12	2296.36	1576.74
	GS-OFDMA	**296.71 (-35.26%)**	**908.33 (-42.63%)**	**602.52 (-40.97%)**	**345.83 (-64.01%)**	**938.78 (-63.63%)**	**642.31 (-63.73%)**
Dir	Uniform [2]	1178.36	4100.03	2639.20	782.95	2097.54	1440.25
	Norm [16]	523.56	1920.63	1222.10	462.23	1269.25	865.74
	Ratio [17]	767.22	2735.80	1751.51	655.98	1783.85	1219.92
	GS-OFDMA	**419.65 (-64.39%)**	**1590.41 (-61.21%)**	**1005.03 (-61.92%)**	**168.13 (-78.53%)**	**465.27 (-77.82%)**	**316.70 (-78.01%)**

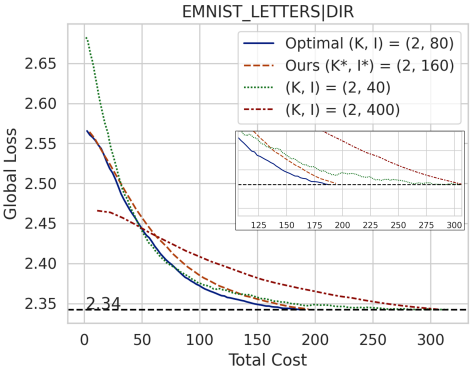


(a) Performance across different values of I.

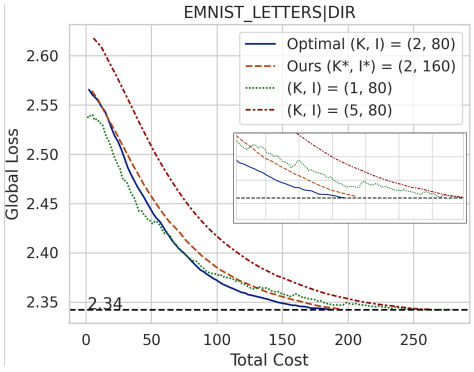


(b) Performance across different values of K.

Fig. 3: Convergence performance on FASHION_MNIST.

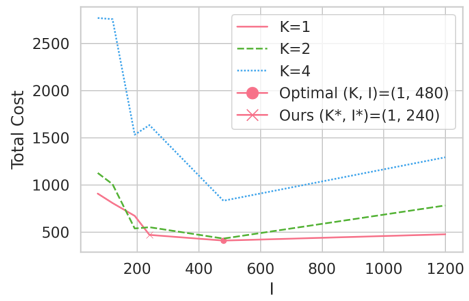


(a) Performance across different values of I.

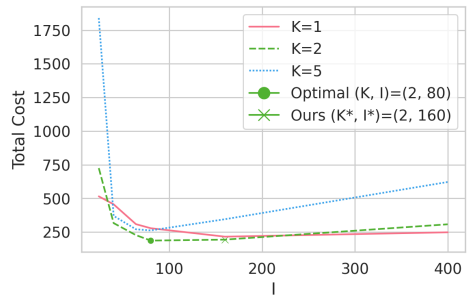


(b) Performance across different values of K.

Fig. 4: Convergence performance on EMNIST_LETTERS.

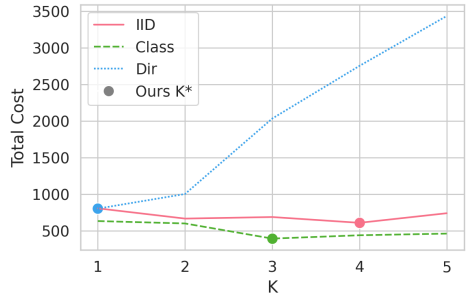


(a) FASHION_MNIST.

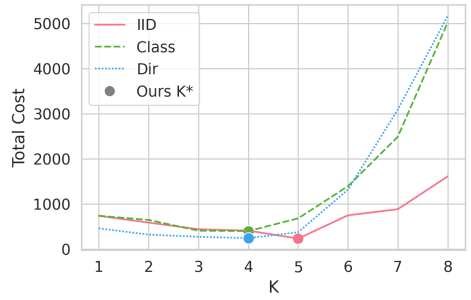


(b) EMNIST_LETTERS.

Fig. 5: Total cost with different values of I.



(a) FASHION_MNIST.



(b) EMNIST_LETTERS.

Fig. 6: Total cost with different values of K.

the advantages brought by data diversity can be overshadowed when an overly large K amplifies the straggler effect.

The proposed solution, represented by K^*, distinctly exhibits the optimal performance, delivering the minimal total cost across the tested K values. This shows the effectiveness of our approach and the importance of the correct K selection.

5) Evaluation of Device Scheduling Schemes

To evaluate the effectiveness of the proposed GS-OFDMA device scheduling method, we compare the overall time costs of the training with different device scheduling strategies:

- GS-OFDMA: Our proposed method with ordering participants and group scheduling.
- GS-wo: A variant of GS-OFDMA without ordering participants.
- OFDMA [7]: A specific instance of GS-OFDMA, constrained to a single group ($K = 1$).
- TDMA [8]: A conventional approach that schedules clients in a sequential order.

I is set to 120 and 40 for FASHION MNIST and EMNIST LETTERS, respectively, while K is the same with "Ours K^*" in Figure 6.

To enable a fair comparison of TDMA with frequency sharing methods, the communication time for TDMA has been reduced to $\frac{1}{S}$ of the time cost of the frequency-sharing methods. Results are presented in Figure 7.

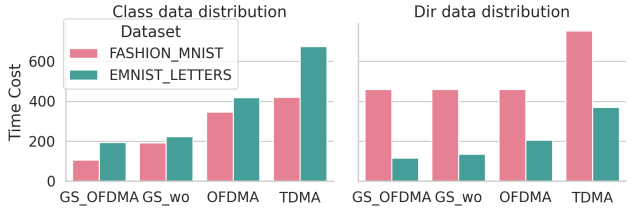


Fig. 7: Time costs with different device scheduling methods, datasets, and data distributions.

GS-OFDMA generally exhibits the lowest time costs in comparison with other methods across both datasets and data distributions. This suggests that the ordering of groups based on time costs and the group scheduling are beneficial in reducing the overall time cost of model transmission. It is particularly effective in the *Class* data distribution setting.

GS-wo incurs higher time costs than GS-OFDMA, highlighting the impact of participant ordering on transmission efficiency. Notably, in the special scenario where the optimal number of groups $K^* = 1$ for the FASHION_MNIST dataset under the *Dir* distribution, GS-OFDMA will be equivalent to GS-wo and OFDMA.

VII. Conclusion

In this paper, we have introduced a multi-group transmission scheme to schedule the model updates during the training process of wireless federated learning. To address the device scheduling challenges, we have proposed a group-based Johnson's rule and a comprehensive device scheduling framework, GS-OFDMA, which orders and groups participants for the model submissions. Then we have formulated a joint optimization problem on device and training scheduling upon the framework, incorporating practical considerations such as system and data heterogeneity. To solve the mixed integer nonlinear programming, we have developed an iterative algorithm. Empirical validations on real-world datasets showed that our approach significantly improves cost efficiency over standard federated learning algorithms across various datasets and data distributions. Our algorithm demonstrated near-optimal performance in terms of total cost and convergence speed.

Appendix A
Proof of Theorem 2

Proof. Let a function $h(K, I)$ be defined by

$$h(K,I) = \left(\frac{a_0 I}{K} + \frac{a_1 I^2 + B}{I}\right)(a_2 K I + a_3 K + a_4 I),$$

where $a_0, a_1, a_2, a_3, a_4 > 0$ are known, $1 \le K \le \frac{N}{S}$, and $1 \le I \le I_{max}$.

For any $1 \le K \le \frac{N}{S}$, we have the first-order derivative of $h(K, I)$ with respect to K as

$$\frac{\partial h(K,I)}{\partial K} = -\frac{a_0 a_4 I^2}{K^2} + \frac{(a_1 I^2 + B)(a_2 I + a_3)}{I}. \tag{26}$$

The second-order derivative of $h(K, I)$ with respect to K is

$$\frac{\partial^2 h(K,I)}{\partial K^2} = \frac{2 a_0 a_4 I^2}{K^3} \ge 0.$$

Therefore, with given I, $h(K, I)$ is convex with respect to K. Moreover, $h(K, I)$ first increases and then decreases as K increases. The optimal K^* can be explicitly computed by

$$\begin{aligned} K^* &= \arg \min_{1 \le K \le \frac{N}{S}} h(K,I) \\ &= \left(\frac{a_0 a_4 I^3}{(a_1 I^2 + B)(a_2 I + a_3)}\right)^{1/2}. \end{aligned} \tag{27}$$

Similarly, for $1 \le I \le I_{max}$, we have

$$\begin{aligned} \frac{\partial^2 h(K,I)}{\partial I^2} &= 2\left(a_0 a_2 + a_1 a_2 K + \frac{a_0 a_4}{K} + a_1 a_4\right) + \frac{2 a_3 K B}{I^3} \\ &\ge 0. \end{aligned}$$

Therefore, with given K, $h(K, I)$ is convex with respect to I, which makes $h(K, I)$ a biconvex function. Moreover, the feasible set $H = [1, N/S] \times [1, I_{\max}]$ is also biconvex. Therefore, Problem P4 is a biconvex problem [36]. □

Appendix B
Proof of Lemma 1

Proof. As shown in Appendix A, when $\alpha = 0$ and $a_4 = 0$ and for any $1 \le K \le \frac{N}{S}$, we have

$$\frac{\partial h(K,I)}{\partial K} > 0.$$

Therefore, Property 1 holds.

With $0 < \alpha \le 1$, $\frac{\partial h(K,I)}{\partial K} < 0$ when K is small and $\frac{\partial h(K,I)}{\partial K} > 0$ when K is large, which is also true for I. Therefore, Properties 2 and 3 hold. □

References

[1] X. Chen, X. Zhou, H. Zhang, M. Sun, and T. Zhao, "Cost-effective federated learning: A unified approach to

device and training scheduling," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2024, pp. 3488–3493.

[2] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, 2017, pp. 1273–1282.

[3] N. H. Tran, W. Bao, A. Zomaya, M. N. H. Nguyen, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *Proc. IEEE Conf. Computer Commun. (INFOCOM)*, 2019, pp. 1387–1395.

[4] F. Lai, X. Zhu, H. V. Madhyastha, and M. Chowdhury, "Oort: Efficient federated learning via guided participant selection." in *Proc. USENIX Symp. Operating Syst. Design Implement. (OSDI)*, 2021, pp. 19–35.

[5] C. Li, X. Zeng, M. Zhang, and Z. Cao, "PyramidFL: A fine-grained client selection framework for efficient federated learning," in *Proc. Annu. Int. Conf. Mob. Comput. Netw. (MobiCom)*, 2022, p. 158–171.

[6] H. H. Yang, Z. Liu, T. Q. S. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 317–333, 2020.

[7] M. Chen, H. V. Poor, W. Saad, and S. Cui, "Convergence time optimization for federated learning over wireless networks," *IEEE Trans. Wireless Commun. (TWC)*, vol. 20, no. 4, pp. 2457–2471, 2021.

[8] X. Mo and J. Xu, "Energy-efficient federated edge learning with joint communication and computation design," *J. Commn. Net*, vol. 6, no. 2, pp. 110–124, 2021.

[9] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, "Energy efficient federated learning over wireless communication networks," *IEEE Trans. Wireless Commun. (TWC)*, vol. 20, no. 3, pp. 1935–1949, 2021.

[10] L. WANG, W. WANG, and B. LI, "CMFL: Mitigating communication overhead for federated learning," in *Proc. IEEE Int. Conf. Distrib. Compt. Syst. (ICDCS)*, 2019, pp. 954–964.

[11] 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 J. Sel. Areas Commun. (JSAC)*, vol. 37, no. 6, pp. 1205–1221, 2019.

[12] B. Luo, X. Li, S. Wang, J. Huang, and L. Tassiulas, "Cost-effective federated learning in mobile edge networks," *IEEE J. Sel. Areas Commun. (JSAC)*, vol. 39, no. 12, pp. 3606–3621, 2021.

[13] 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 Trans. Wireless Commun. (TWC)*, vol. 19, no. 10, pp. 6535–6548, 2020.

[14] W. Chen, S. Horvath, and P. Richtarik, "Optimal client sampling for federated learning," *Trans. Machine Learning Research (TMLR)*, 2022.

[15] H. Ma, H. Guo, and V. K. N. Lau, "Communication-efficient federated multitask learning over wireless networks," *IEEE Internet Things J.*, vol. 10, no. 1, pp. 609–624, 2023.

[16] B. Luo, W. Xiao, S. Wang, J. Huang, and L. Tassiulas, "Tackling system and statistical heterogeneity for federated learning with adaptive client sampling," in *Proc. IEEE Conf. Computer Commun. (INFOCOM)*, 2022, pp. 1739–1748.

[17] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Conf. Machine Learning and Syst. (MLSys)*, vol. 2, 2020, pp. 429–450.

[18] X. Zeng, M. Yan, and M. Zhang, "Mercury: Efficient on-device distributed dnn training via stochastic importance sampling," in *Proc. ACM Conf. Embedded Networked Sensor Systems*, 2021, p. 29–41.

[19] H. Cao, Q. Pan, Y. Zhu, and J. Liu, "Birds of a feather help: Context-aware client selection for federated learning," in *Proc. Int. Workshop Trustable, Verifiable and Auditable Federated Learning with AAAI*, 2022.

[20] H. T. Nguyen, V. Sehwag, S. Hosseinalipour, C. G. Brinton, M. Chiang, and H. Vincent Poor, "Fast-convergent federated learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 201–218, 2021.

[21] P. Zhou, H. Xu, L. H. Lee, P. Fang, and P. Hui, "Are you left out? an efficient and fair federated learning for personalized profiles on wearable devices of inferior networking conditions," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. (IMWUT)*, vol. 6, no. 2, 2022.

[22] Y. Jin, L. Jiao, Z. Qian, S. Zhang, S. Lu, and X. Wang, "Resource-efficient and convergence-preserving online participant selection in federated learning," in *Proc. IEEE Int. Conf. Distrib. Compt. Syst. (ICDCS)*, 2020, pp. 606–616.

[23] Q. Zeng, Y. Du, K. Huang, and K. K. Leung, "Energy-efficient radio resource allocation for federated edge learning," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2020, pp. 1–6.

[24] H. Yin and S. Alamouti, "OFDMA: A broadband wireless access technology," in *Proc. IEEE Sarnoff Symp.*, 2006, pp. 1–4.

[25] M. L. Pinedo, *Scheduling: Theory, Algorithms, and Systems*, 3rd ed. Springer Publishing Company, Incorporated, 2008.

[26] J. N. D. Gupta, "Two-stage, hybrid flowshop scheduling problem," *The Journal of the Operational Research Society*, vol. 39, no. 4, pp. 359–364, 1988.

[27] R. Ruiz and J. A. Vázquez-Rodríguez, "The hybrid flow shop scheduling problem," *European Journal of Operational Research*, vol. 205, no. 1, pp. 1–18, 2010.

[28] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," *arXiv:1811.03604*, 2018.

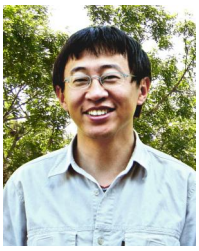
[29] S. M. Johnson, "Optimal two- and three-stage production schedules with setup times included," *Naval Research Logistics Quarterly*, vol. 1, no. 1, pp. 61–68, 1954.

[30] X. Chen, X. Zhou, H. Zhang, M. Sun, and H. Vincent Poor, "Client selection for wireless federated learn-

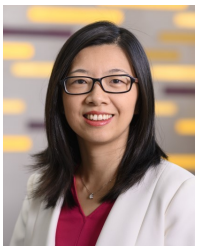
ing with data and latency heterogeneity," *IEEE Internet Things J.*, vol. 11, no. 19, pp. 32 183–32 196, 2024.
[31] W. W. Hager and H. Zhang, "An active set algorithm for nonlinear optimization with polyhedral constraints," *Science China Mathematics*, vol. 59, no. 8, pp. 1525–1542, 2016.
[32] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: a system for large-scale machine learning," in *Proc. USENIX Symp. Operating Syst. Design Implement. (OSDI)*, vol. 16, no. 19, 2016, pp. 265–283.
[33] G. Cohen, S. Afshar, J. Tapson, and A. Van Schaik, "EMNIST: Extending MNIST to handwritten letters," in *Proc. Int. Joint Conf. Neural Networks (IJCNN)*, 2017, pp. 2921–2926.
[34] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms," *arXiv:1708.07747*, 2017.
[35] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[36] J. Gorski, F. Pfeuffer, and K. Klamroth, "Biconvex sets and optimization with biconvex functions: a survey and extensions," *Mathematical Methods of Operations Research*, vol. 66, pp. 373–407, 2007.

Xiaobing Chen received the B.E. degree in electrical engineering and M.E. degree in control science and engineering from University of Electronic Science and Technology of China, Chengdu, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree in the Division of Electrical and Computer Engineering, Louisiana State University, where he joined as a Graduate Research Assistant in 2021. His research interests include federated learning, privacy in machine learning, and optimization theory.

Xiangwei Zhou received the B.S. degree in communication engineering from Nanjing University of Science and Technology, Nanjing, China, the M.S. degree in information and communication engineering from Zhejiang University, Hangzhou, China, and the Ph.D. degree in electrical and computer engineering from Georgia Institute of Technology, Atlanta, GA, USA, in 2005, 2007, and 2011, respectively. He is an Associate Professor with the Division of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, LA, USA. His research interests include wireless communications and statistical signal processing, with current emphasis on coexistence of wireless systems, Internet of Things, and machine learning for intelligent communications. He was the recipient of the Best Paper Award at the 2014 International Conference on Wireless Communications and Signal Processing and served as an Editor for the IEEE Transactions on Wireless Communications from 2013 to 2018.



Hongchao Zhang received the B.S. degree in Mathematics from Shandong University, China, in 1998, the M.Sc. degree from the Computing Center, Chinese Academy of Sciences, in 2001, and the Ph.D. degree in Mathematics from the University of Florida, Gainesville, FL, USA, in 2006. He is currently a Professor in the Department of Mathematics and the Center for Computation and Technology (CCT) at Louisiana State University, Baton Rouge, LA, USA. His research interests include nonlinear optimization theory and its applications, sparse matrix computing, graph partitioning, stochastic optimization algorithms and applications, and numerical linear algebra.



Mingxuan Sun received the B.S. degree in computer science and engineering from Zhejiang University, Hangzhou, China in 2004, the M.S. degree in computer science from University of Kentucky, Lexington, KY, USA in 2006, and the Ph.D. degree in computer science from Georgia Institute of Technology, Atlanta, GA, USA in 2012. She is an Associate Professor with the Division of Computer Science and Engineering, Louisiana State University, Baton Rouge, LA, USA. Her research interests include machine learning, information retrieval, and data mining. She is also interested in machine learning and AI applications in social informatics, security, and wireless communications. She has published research papers in leading journals and conferences including PAMI, JMLR, NIPS, AAAI, AISTATS, KDD, ICDM, WWW, WSDM, etc.

Taibiao Zhao received the B.E. degree in computer and information science from Northeastern University of China, Shenyang, China, in 2020. He is currently pursuing the Ph.D. degree in the Division of Electrical and Computer Engineering, Louisiana State University, where he joined as a Graduate Research Assistant in 2021. His current research interests include backdoor attacks, adversarial learning, and large language models.