Efficient Wireless Federated Learning with Partial Model Aggregation

Zhixiong Chen, *Graduate Student Member, IEEE*, Wenqiang Yi, *Member, IEEE*,
Hyundong Shin, *Fellow, IEEE*, Arumugam Nallanathan, *Fellow, IEEE*, and Geoffrey Ye Li, *Fellow, IEEE*

***Abstract*—The data heterogeneity across clients and the limited communication resources, e.g., bandwidth and energy, are two of the main bottlenecks for wireless federated learning (FL). To tackle these challenges, we first devise a novel FL framework with partial model aggregation (PMA). This approach aggregates the lower layers of neural networks, responsible for feature extraction, at the parameter server while keeping the upper layers, responsible for complex pattern recognition, at clients for personalization. The proposed PMA-FL is able to address the data heterogeneity and reduce the transmitted information in wireless channels. Then, we derive a convergence bound of the framework under a non-convex loss function setting to reveal the role of unbalanced data size in the learning performance. On this basis, we maximize the scheduled data size to minimize the global loss function through jointly optimize the client selection, bandwidth allocation, computation and communication time division policies with the assistance of Lyapunov optimization. Our analysis reveals that the optimal time division is achieved when the communication and computation parts of PMA-FL have the same power. We also develop a bisection method to solve the optimal bandwidth allocation policy and use the set expansion algorithm to address the client scheduling policy. Compared with the benchmark schemes, the proposed PMA-FL improves 3.13% and 11.8% absolute accuracy on two typical datasets with heterogeneous data distribution settings, i.e., MINIST and CIFAR-10, respectively. In addition, the proposed joint dynamic client selection and resource management approach achieve slightly higher accuracy than the considered benchmarks, but they provide a satisfactory energy and time reduction: 29% energy or 20% time reduction on the MNIST; and 25% energy or 12.5% time reduction on the CIFAR-10.**

***Index Terms*—Client selection, federated Learning, Lyapunov optimization, resource management**

Part of this work has been presented at the IEEE International Conference on Communications (ICC), 2023 [1]. (Corresponding author: Arumugam Nallanathan and Hyundong Shin)

Zhixiong Chen is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, U.K. (email: zhixiong.chen@qmul.ac.uk).

Wenqiang Yi is with the School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, U.K. (email: wy23627@essex.ac.uk).

Arumugam Nallanathan is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, U.K and with the Department of Electronics and Information Convergence Engineering, Kyung Hee University, Yongin-si, Gyeonggido 17104, Republic of Korea (email: a.nallanathan@qmul.ac.uk).

Hyundong Shin is with the Department of Electronics and Information Convergence Engineering, Kyung Hee University, Yongin-si, Gyeonggido 17104, Republic of Korea (e-mail: hshin@khu.ac.kr).

Geoffrey Ye Li is with the Faculty of Engineering, Department of Electrical and Electronic Engineering, Imperial College London, England (e-mail: geoffrey.li@imperial.ac.uk).

I. Introduction

Federated learning (FL) is a promising distributed learning approach for protecting data privacy. In FL, edge clients collaboratively train a model under the orchestration of a parameter server (PS), which only requires clients to upload local models instead of local private data [2]. FL operations can be divided into two parts, namely the communication part and the computation part. For the communication part, the learning performance is constrained by the limited communication resources, e.g., bandwidth and energy. The inadequate wireless resources hinder more clients devoted to the FL process, thus negatively affecting the convergence speed and learning accuracy [3]. For the computation part, the model accuracy is degraded by non-independent and identically distributed (non-IID) data samples. Since the PS aggregates models learned from the different clients, the data heterogeneity presented on different clients may lead to weak generalization ability of the trained global model, even resulting in an unstable training process of FL [2]. Therefore, FL needs well-designed solutions to address these two challenges.

A. Related Works

From the communication perspective, efficient resource management and client selection schemes can enable additional clients to participate in the FL process and thus enhancing learning performance. To this end, existing works focus on resource optimization [4]–[6], client selection [7]–[10], and alternating direction method of multipliers to reduce the communication rounds of training [11]. The energy-efficient workload partitioning scheme in [4] balances the computation between the central processing unit and graphics processing unit in the FL system. The time-sharing-based transmission scheme in [5] can improve the communication efficiency of FL. The work in [6] introduced an energy-efficient transmission and computation resource allocation approach for energy consumption minimization of FL system under a latency constraint. The joint client scheduling and resource allocation policy in [7] maximizes the model accuracy in latency-constrained FL. The joint client selection and resource optimization approach in [8] maximizes the seleted client number while adhering to clients' energy restrictions. In [9], a gradient norm approximation method can assist the client scheduling for boosting the training performance of FL. A joint learning and resource allocation problem has been investigated in [10] to minimize an FL loss function. Although these works have devised different client selection

and resource management policies to facilitate FL, the joint optimization of communication and computation in FL has been rarely explored.

From the computation perspective, the emerging personalized FL techniques are promising to tackle the data heterogeneity-related challenges, which adapt the collaboratively learned global model for individual clients [12]. Existing works toward this direction utilize various techniques to implement model personalization, including multi-task learning [13], [14], meta-learning [15], model regularization [16], and model interpolation [17]. More specifically, it has been shown in [13] that multi-task learning is a natural choice for building personalized federated models. The work in [14] proposed a personalized federated multi-task learning over wireless fading channels and analyzed the convergence behaviour based on the bilevel optimization method. However, the multi-task FL heavily relies on the full participation of clients in each round. The federated meta-learning algorithm in [15] can improve the model accuracy of FL, which maps the meta-training to the federated training process and meta-testing to FL personalization. A proximal term is introduced in [16] to limit the impact of local updates, achieving convergence stability and improving model generalization. The adaptive personalized FL algorithm in [17] can find the optimal combination of global and local models in a communication efficient manner. However, the above techniques require more computation or memory resources than the conventional FL algorithms that solely train a global model, e.g., Federated Averaging (FedAvg) [18]. In addition to the above works, the partial model aggregation approach in [19]–[21] effectively address the data heterogeneity in FL, in which the learning model is decoupled into two parts, i.e., feature extractor and predictor parts, and clients learn a shared feature extractor and unique local predictors. However, the approaches in [19], [20] requires to separately train the feature extractor and predictor in the local training process or full client participation in the learning process. The separate training may waste computation resources since both the feature extractor and predictor update require complete computation on the whole learning model, and the full client participation may restrict their practical implementation in wireless networks due to limited wireless resources. In addition, [20] did not analyze the convergence behaviour, while the convergence analysis in [19] is based on the linear model and quadratic loss functions that are not satisfied by most practical machine learning models. The work in [21] investigated both the separate and parallel training schemes among feature extractor and predictor and analyzed the convergence bound based on general machine learning models. However, it presupposed a balanced data size distribution among clients. It is worth mentioning that some existing works, e.g., [22]–[24], investigated another partial model aggregation mechanism, which allows devices to train a portion of the global model to fit their communication and computation capabilities. However, these approaches mainly focused on learning a shared global model for all devices. In practical wireless networks, the data distributions among devices are generally heterogeneous, and the shared global model may show lower performance on specific devices than personalized local models. Thus, this work mainly focused on personalized federated learning to mitigate the data heterogeneity issues.

B. Motivations and Contributions

Although the resource allocation and client selection schemes in [4]–[11] effectively alleviate the communication burden for FL in wireless networks, they are all operated by averaging local models for global aggregation and are hard to cope with the data heterogeneity nature of FL. In addition, the personalized FL algorithms in [13], [15]–[17] require more computation or memory resources than the conventional weight averaging-based FL algorithms. Motivated by this, this work aims to devise an efficient FL approach that simultaneously tackles data heterogeneity and communication resource limitations for FL in wireless networks. Inspired by the success of centralized learning, different learning tasks often share the lower layers of neural networks responsible for feature extraction while the heterogeneity mainly focuses on the upper layers corresponding to complex pattern recognition [25], [26]. We propose a novel FL framework that partially aggregates local model parameters of the clients in the learning process to learn a shared feature extractor, while the label predictor part are localized at clients for personalization. Note that, unlike [19]–[21], this work trains the feature extractor and predictor simultaneously in the local training process. We analyze the convergence bound of the proposed method under a general non-convex loss function, unbalanced data size setting, and partial client participation scenario. In addition, in view of the clients' limited wireless resources and energy budget, we devise a joint client selection, wireless bandwidth, and computation resources allocation scheme to improve the learning performance of FL in practical wireless networks. The main contributions of this paper are summarized as follows:

- To tackle the data heterogeneity across clients in the FL system, we devise a novel federated learning framework, namely partial model aggregation-FL (PMA-FL), in which clients only collaboratively train the lower layers of the neural networks while the upper layers are individually trained by each client for personalization. This design significantly improves the learning performance in data heterogeneity scenarios. Note that, unlike the existing personalized FL works, e.g., [12], [13], [15]–[17], PMA-FL does not necessitate additional computational or memory resources than conventional Federated Learning algorithms that solely train a global model, such as FedAvg.
- To facilitate efficient FL in practical networks, we minimize the global loss function while simultaneously considering clients' long-term energy budget, bandwidth limitation, and latency constraints. However, it is intractable to minimize the global loss due to its inexplicit form. To this end, we theoretically analyze the convergence behaviour of PMA-FL, finding a new objective function, termed scheduled data sample volume, which is in an explicit form for the client selection decision. The minimum global loss function can be obtained by maximizing this metric.

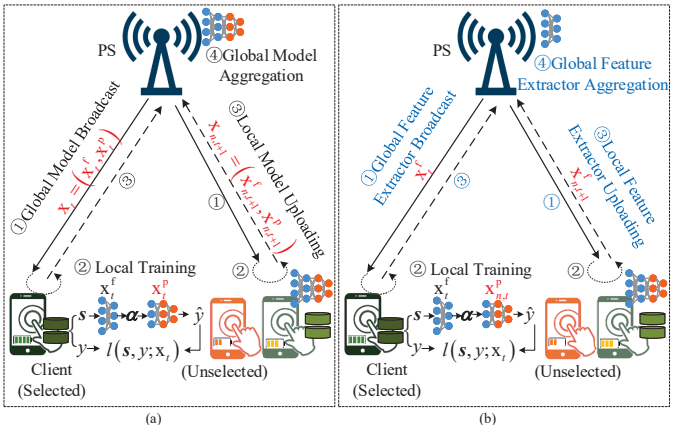


Fig. 1. Illustrating the federated learning system and mechanism: (a) shows the traditional federated learning mechanism which trains a global model (including feature extractor and predictor); and (b) presents the federated learning mechanism with collaboratively train a feature extractor while the predictor is trained by each client itself for personalization.

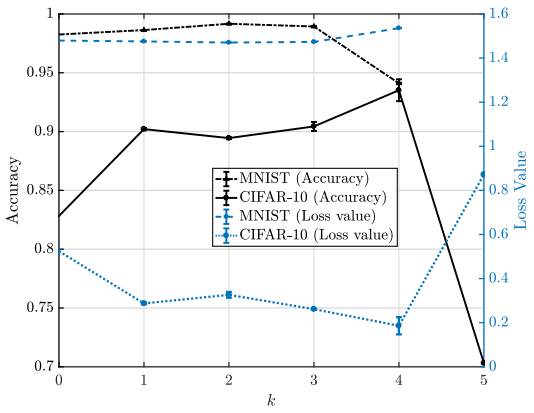


Fig. 2. The test accuracy and global loss versus the number of layers of $\mathbf{x}_t^{\mathrm{f}}$ (i.e., k): from without sharing parameters (each client solely train the model by using its own dataset) to sharing entire model parameters (FedAvg).

- To maximize the scheduled data sample volume, we formulate a joint client selection, wireless bandwidth allocation, and computation-communication-time division optimization problem, which is a mixed-integer nonlinear programming problem and is challenging to solve. We first decouple the long-term stochastic problem into a deterministic one by using the Lyapunov optimization framework. Next, we solve the optimal solution for time division policies through convex optimization techniques, develop a bisection method to address the optimal bandwidth allocation policy, and use the set expansion algorithm to achieve the client scheduling policy.
- Experiments on the MNIST and CIFAR-10 datasets show that the proposed PMA approach converges faster than the benchmark schemes, and improves 3.13% and 11.8% accuracy on these two datasets, respectively. Moreover, our developed joint client scheduling and resource management scheme can reduce around 29% energy budget or 20% time budget and is able to achieve higher accuracies than the considered benchmarks under the MNIST dataset. For CIFAR-10, the proposed algorithm can obtain slightly higher accuracies than the benchmark schemes and reduce the 25% energy budget or 12.5% time budget.

C. Organization

The remainder of this work is organized as follows: Section II introduces the FL system and problem formulation. In Section III, we analyze the convergence of PMA-FL and transform the original problem into a tractable one. Section IV provides the solution for joint time division, client selection, and bandwidth allocation problem. Section V evaluates the proposed approach by simulations. Section VI concludes this work.

II. System Model

A. Federated Learning System

In this work, we consider a typical FL system consisting of one PS and N clients indexed by $\mathcal{N} = \{1, 2, \cdots, N\}$. The client n ($n \in \mathcal{N}$) holds its local dataset $\mathcal{D}_n$ with $D_n = |\mathcal{D}_n|$ data samples. The whole dataset, $\mathcal{D} = \cup\{\mathcal{D}_n\}_{n=1}^N$, is with total number of samples $D = \sum_{n=1}^N D_n$.

Given a data sample $(\boldsymbol{s}, y) \in \mathcal{D}$, where $\boldsymbol{s} \in \mathbb{R}^d$ represents the input vector, and $y \in \mathbb{R}$ is the corresponding label. Let $\boldsymbol{\alpha} \in \mathbb{R}^p$ be the latent feature of $\boldsymbol{s}$. The machine learning model parameterized by $\mathbf{x} = [\mathbf{x}^{\mathrm{f}}, \mathbf{x}^{\mathrm{p}}]$ consists of two components: a feature extractor $\boldsymbol{s} \to \boldsymbol{\alpha}$ parameterized by $\mathbf{x}^{\mathrm{f}}$ and a predictor $\boldsymbol{\alpha} \to \hat{y}$ parameterized by $\mathbf{x}^{\mathrm{p}}$. Denote $l(\boldsymbol{s}, y; \mathbf{x})$ as the sample-wise loss function. The local loss function of client n ($n \in \mathcal{N}$) is

$$\mathcal{L}_n(\mathbf{x}_n) = \mathcal{L}_n(\mathbf{x}_n^{\mathrm{f}}, \mathbf{x}_n^{\mathrm{p}}) = \frac{1}{D_n} \sum\nolimits_{(\boldsymbol{s},y)\in\mathcal{D}_n} l(\boldsymbol{s}, y; \mathbf{x}_n), \quad (1)$$

where $\mathbf{x}_n$ is client n's local model; $\mathbf{x}_n^{\mathrm{f}}$ and $\mathbf{x}_n^{\mathrm{p}}$ correspond to the feature extractor and predictor, respectively. Accordingly, the global loss function is

$$\mathcal{L}(\mathbf{x}_1, \cdots, \mathbf{x}_N) = \frac{1}{D} \sum\nolimits_{n=1}^N D_n \mathcal{L}_n(\mathbf{x}_n). \quad (2)$$

The federated learning process is done by solving the following problem

$$\min_{(\mathbf{x}_1,\cdots,\mathbf{x}_N)} \Big(\mathcal{L}(\mathbf{x}_1, \cdots, \mathbf{x}_N) = \frac{1}{D} \sum\nolimits_{n=1}^N D_n \mathcal{L}_n(\mathbf{x}_n)\Big). \quad (3)$$

Note that the formulation in (3) is widely used to characterize the personalized FL problem [20]. The main objective of the typical federated learning algorithms, such as the FedAvg [18], is to find an optimal shared global model $\mathbf{x}^* = \mathbf{x}_n^*$ ($\forall n \in \mathcal{N}$) that minimizes the global loss function $\mathcal{L}(\mathbf{x}_1, \cdots, \mathbf{x}_N)$, as shown in Fig. 1(a). However, the data distributions among different clients in real-world FL systems are often heterogeneous. In this presence, the local optimal models may drift significantly from each other, and thus solely optimizing for the global model's accuracy leads to a poor generalization of each client. To tackle this issue, this work designs a personalized FL approach for solving problem (3), in which each client learns a specific local model to fit its local dataset and capture the knowledge from other clients.

B. Federated Learning with Partial Model Aggregation

The success of centralized learning in training multiple tasks or learning multiple classes simultaneously has shown that data often shares a global feature representation (i.e., $\mathbf{x}^{\mathrm{f}}$), while the statistical heterogeneity across clients or tasks is mainly located at the labels' predictor (i.e., $\mathbf{x}^{\mathrm{p}}$) [25], [26]. Thus, this work proposes PMA in the FL training process, in which clients collaboratively learn a shared global feature extractor $\mathbf{x}^{\mathrm{f}}$ and each client n ($n \in \mathcal{N}$) learns a personalized label predictor $\mathbf{x}_n^{\mathrm{p}}$ in the training process, as shown in Fig. 1(b). For the sake of analysis, we use $\boldsymbol{X}^{\mathrm{p}} = (\mathbf{x}_1^{\mathrm{p}}, \mathbf{x}_2^{\mathrm{p}}, \cdots, \mathbf{x}_N^{\mathrm{p}})$ denotes all the clients' predictors throughout this paper. Consequently, the global loss function defined in (2) can be written as $\mathcal{L}(\mathbf{x}_1, \cdots, \mathbf{x}_N) = \mathcal{L}(\mathbf{x}^{\mathrm{f}}, \boldsymbol{X}^{\mathrm{p}})$. The learning process consists of T rounds, and each round includes the following steps:

- **Client Selection**: The PS determines the set of selected clients in the current round, denoted by $\boldsymbol{S}_t$. Let $\beta_{n,t} \in \{0, 1\}$ represent the selecting indicator of client n in round t, where $\beta_{n,t} = 1$ represents that client n is selected, $\beta_{n,t} = 0$ otherwise. Thus, $\boldsymbol{S}_t = \{n : \beta_{n,t} = 1, \forall n \in \mathcal{N}\}$.
- **Global Feature Extractor Broadcasting**: The PS broadcasts its current global feature extractor $\mathbf{x}_t^{\mathrm{f}}$ to the clients in $\boldsymbol{S}_t$.
- **Local Model Training**: Each client n ($n \in \boldsymbol{S}_t$) updates its local model after receiving the global feature extractor $\mathbf{x}_t^{\mathrm{f}}$. Specifically, its local feature extractor is updated as

$$\mathbf{x}_{n,t+1}^{\mathrm{f}} = \mathbf{x}_t^{\mathrm{f}} - \beta_{n,t}\eta_{\mathrm{f}}\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \tag{4}$$

and the corresponding predictor updates its parameters via

$$\mathbf{x}_{n,t+1}^{\mathrm{p}} = \mathbf{x}_{n,t}^{\mathrm{p}} - \beta_{n,t}\eta_{\mathrm{p}}\nabla_{\mathrm{p}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \tag{5}$$

where η_f and η_p represent the learning rate of $\mathbf{x}^{\mathrm{f}}$ and $\mathbf{x}^{\mathrm{p}}$, respectively.

- **Global Feature Extractor Aggregation**: After finishing the local training, all scheduled clients upload their updated local feature extractors to the PS for aggregation. Specifically, the PS computes the global shared feature extractor as follows:

$$\mathbf{x}_{t+1}^{\mathrm{f}} = \frac{\sum_{n=1}^{N} \beta_{n,t} D_n \mathbf{x}_{n,t+1}^{\mathrm{f}}}{\sum_{n=1}^{N} \beta_{n,t} D_n}. \tag{6}$$

To better illustrate the benefits of partial sharing the model parameters, we provide an experiment on both MNIST and CIFAR-10 datasets in Fig. 2, where the data distribution of each client is non-IID. Specifically, each client has no more than two classes of data and participates in the training in each round. The MNIST dataset is trained by a 4-layers multi-layer perceptron (MLP) model, and the CIFAR-10 dataset is trained by a 5-layers convolutional neural network (CNN) model. The detailed configurations are shown in the experimental setting part in Section V. Fig. 2 shows that the MLP with only sharing the first two layers ($k = 2$) in the training process obtains the highest accuracy and the CNN achieves the highest accuracy by aggregating the first four layers ($k = 4$). One interesting result is that for both MLP and CNN, the global trained model ($k = 4$ for the MLP and $k = 5$ for the CNN) is less accurate than the local models of clients ($k = 0$ for both the MLP and CNN) trained by their local datasets. Thus, aggregating the feature extractor with sufficient feature extracting ability in the training process is an efficient method to obtain better performance in the non-IID data distribution scenarios, instead of aggregating the entire model or solely training models on clients' local datasets.

C. Computation Cost

In each round t, the selected clients will update its local model after receiving the global feature extractor, $\mathbf{x}_t^{\mathrm{f}}$, then uploading the trained local feature extractor parameters, $\mathbf{x}_{n,t+1}^{\mathrm{f}}$ ($\forall n \in \boldsymbol{S}_t$), to the PS for aggregation. Let $f_{n,t}$ denote the CPU frequency of client n. Based on [27], the local computation energy consumption of each client n is proportional to the square of its CPU frequency. Employing dynamic voltage and frequency scaling techniques [28], client n can reduce the energy consumption of computation by reducing the CPU frequency. Denote $f_{n,\max}$ the maximum CPU frequency of client n. For any given machine learning model, the number of floating-point operations for one sample to calculate gradient can be estimated, denoted by G [29]. Let ζ_n denote the number of CPU cycles required to process one floating-point operation, which depends on the CPU. Thus, the required CPU cycles to process one data sample at client n is represented by $C_n = \zeta_n G$. Given the computation time restriction, $T_{n,t}^{\mathrm{L}}$, the most energy efficient CPU frequency is $f_{n,t} = \frac{\tau D_n C_n}{T_{n,t}^{\mathrm{L}}}$, where τ is the the number of local iterations. The corresponding energy consumption of client n to perform local training is

$$\mathcal{E}_{n,t}^{\mathrm{L}} = \kappa\tau D_n C_n f_{n,t}^2 = \frac{\kappa\tau^3 D_n^3 C_n^3}{(T_{n,t}^{\mathrm{L}})^2}, \tag{7}$$

where κ denotes the clients' energy coefficient that hinges on chip architecture. Since the CPU frequency of client n is restricted by $f_{n,\max}$, the computation time should satisfy

$$T_{n,t}^{\mathrm{L}} \geq \frac{\tau D_n C_n}{f_{n,\max}}. \tag{8}$$

It is worth noting that, we have ignored the global feature extractor aggregation cost, because the PS usually has strong computation capability with negligible aggregation delay.

D. Communication Cost

Similar to [4], [6]–[8], this work uses the frequency-division multiple access (FDMA) techniques with B Hz bandwidth for clients to upload local feature extractors $\mathbf{x}_{n,t}^{\mathrm{f}}$. Let $\theta_{n,t}$ ($0 \leq \theta_{n,t} \leq 1$) represent the ratio of bandwidth allocated to client n and $p_{n,t}$ denote the transmit power of client n ($n \in \mathcal{N}$). We assume that the channel gain, $h_{n,t}$, between client n and the PS remains unchanged within one round but varies independently and identically over rounds [6]–[8]. Consequently, the achievable uplink rate for client n in round t is $r_{n,t} = \theta_{n,t} B \log(1 + \frac{p_{n,t} h_{n,t}}{\theta_{n,t} B N_0})$, where N_0 is the power density of noise. Denote Q by the number of parameters of feature extractor ($\mathbf{x}_{n,t}^{\mathrm{f}}, \forall n \in \mathcal{N}, \forall t$), where each parameter is quantized by q bits. Given the maximum communication time $T_{n,t}^{\mathrm{U}}$, the most energy efficient transmission method is $r_{n,t} = \frac{Qq}{T_{n,t}^{\mathrm{U}}}$ [30]. Thus, the transmit power is

$$p_{n,t} = \frac{\theta_{n,t} B N_0}{h_{n,t}} \left(2^{\frac{Qq}{\theta_{n,t} B T_{n,t}^{\mathrm{U}}}} - 1\right). \tag{9}$$

The corresponding energy consumption is $\mathcal{E}_{n,t}^{\mathrm{U}} = p_{n,t} T_{n,t}^{\mathrm{U}}$. Thus, the total energy consumption of client n in round t for both computation and communication is $\mathcal{E}_{n,t} = \mathcal{E}_{n,t}^{\mathrm{L}} + \mathcal{E}_{n,t}^{\mathrm{U}}$. Let $p_{n,\max}$ denote the maximum transmit power of client n, then $0 \leq p_{n,t} \leq p_{n,\max}$. Thus, the communication time of client n should satisfy

$$T_{n,t}^{\mathrm{U}} \geq \frac{Qq}{\theta_{n,t} B \log\left(1 + \frac{p_{n,\max} h_{n,t}}{\theta_{n,t} B N_0}\right)}. \tag{10}$$

Similar to many existing works, e.g., [5]–[9], we ignore the global feature extractor broadcasting cost and mainly focus on the performance bottleneck of the battery and communication-constrained edge clients because the PS usually supplied by the grid is energy-sufficient. Moreover, the whole bandwidth can be used for broadcasting and the transmit power of the PS is usually large, the transmission delay is negligible. In addition, our proposed solution in Section IV can be directly generalized to the case of considering broadcasting delay by simply subtracting broadcasting delay from the latency constraint $T_{\max}$ in problem (11), since the global broadcasting delay is a constant and uniform for the selected clients in each round.

E. Problem Formulation

The objective of this work is to minimize the expected global loss after T rounds of training, i.e., $\mathbb{E}[\mathcal{L}(\mathbf{x}_T^{\mathrm{f}}, \boldsymbol{X}_T^{\mathrm{p}})]$, where $\mathbf{x}_T^{\mathrm{f}}$ is the global feature extractor after T rounds of training and $\boldsymbol{X}_T^{\mathrm{p}} = (\mathbf{x}_{1,T}^{\mathrm{p}}, \cdots, \mathbf{x}_{N,T}^{\mathrm{p}})$ is the label predictors of all clients. The expectation $\mathbb{E}[\mathcal{L}(\mathbf{x}_T^{\mathrm{f}}, \boldsymbol{X}_T^{\mathrm{p}}]$ is taken over the randomness of channel noise and client scheduling in each round. To this end, we jointly optimize the client scheduling, bandwidth allocation, computation time, and communication time allocation policy. Denote $\boldsymbol{\theta}_t = (\theta_{1,t}, \theta_{2,t}, \cdots, \theta_{N,t})$ as the proportions of the bandwidth for different clients in round t. Let $\boldsymbol{T}_t^{\mathrm{L}} = (T_{1,t}^{\mathrm{L}}, T_{2,t}^{\mathrm{L}}, \cdots, T_{N,t}^{\mathrm{L}})$ and $\boldsymbol{T}_t^{\mathrm{U}} = (T_{1,t}^{\mathrm{U}}, T_{2,t}^{\mathrm{U}}, \cdots, T_{N,t}^{\mathrm{U}})$ denote the computation time and communication time for all clients in round t, respectively. We formulate the problem as follows:

$$\mathcal{P}: \min_{\left\{\boldsymbol{S}_t, \boldsymbol{\theta}_t, \boldsymbol{T}_t^{\mathrm{L}}, \boldsymbol{T}_t^{\mathrm{U}},\right\}_{t=0}^{T-1}} \mathbb{E}\left[\mathcal{L}(\mathbf{x}_T^{\mathrm{f}}, \boldsymbol{X}_T^{\mathrm{p}})\right] \tag{11}$$

$$\text{s. t.} \quad (8), (10), \tag{11a}$$

$$\sum\nolimits_{t=0}^{T-1} \mathcal{E}_{n,t} \leq \mathcal{E}_n, \forall n \in \mathcal{N}, \tag{11b}$$

$$\beta_{n,t} \in \{0,1\}, \forall n \in \mathcal{N}, \forall t, \tag{11c}$$

$$\sum\nolimits_{n=1}^{N} \theta_{n,t} \leq 1, \forall t, \tag{11d}$$

$$0 \leq \theta_{n,t} \leq 1, \forall n \in \mathcal{N}, \forall t, \tag{11e}$$

$$T_{n,t}^{\mathrm{L}} + T_{n,t}^{\mathrm{U}} \leq T_{\max}, \forall n \in \mathcal{N}, \forall t. \tag{11f}$$

In problem $\mathcal{P}$, (11a) restricts the computation and communication time. (11b) indicates that for each client, the total energy consumption for both computation and communication over T global rounds cannot exceed its given budget. (11c) is the client scheduling policy restriction. (11d) and (11e) impose restrictions on the wireless bandwidth resource allocated to all clients and each client. (11f) stipulates that the completion time for the participating clients in one round cannot exceed its maximum allowable delay $T_{\max}$.

Solving problem $\mathcal{P}$ confronts two main challenges as follows: 1) **Inexplicit form of the objective function**: Since the evolutions of the feature extractor $\mathbf{x}_t^{\mathrm{f}}$ and predictors $\mathbf{x}_{n,t}^{\mathrm{p}}$ are complex in the training process, it is intractable to solve the close-form expression of $\mathbb{E}\left[F(\mathbf{x}_T^{\mathrm{f}}, \boldsymbol{X}_T^{\mathrm{p}})\right]$. 2) **Unknown future information**: The optimal solution of $\mathcal{P}$ requires exact channel state and clients' energy status information in the entire learning process at the beginning of FL, which is impractical in real-world systems. To tackle these challenges, we first analyze the convergence bound of the considered PMA-FL algorithm and transform problem $\mathcal{P}$ into optimizing the convergence bound in Section III.

III. Convergence Analysis and Problem Transformation

This section investigates the convergence behaviour of PMA-FL, which shows that the per-round scheduled data volume is a key factor in the learning performance of PMA-FL. Based on this, we introduce a new objective function, i.e., the scheduled data volumes, to instruct the design of client scheduling, bandwidth allocation, computation time, and communication time allocation. Then, we transform the inexplicit optimization problem $\mathcal{P}$ into maximizing this new objective function for global loss minimization. In addition, to address the challenge brought by the long-term energy constraint, we further transform the problem into a deterministic problem in each round with the assistance of the Lyapunov optimization framework.

A. Convergence Anaysis

In this subsection, we analyze the convergence bound of PMA-FL. To this end, we introduce some assumptions on the loss functions $\mathcal{L}(\cdot)$ as follows:

Assumption 1. For all loss function $\mathcal{L}_n(\mathbf{x}^{\mathrm{f}}, \mathbf{x}_n^{\mathrm{p}})$, there exist constants ℓ_{f}, ℓ_{p}, ℓ_{fp}, and ℓ_{pf} such that:

- $\nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}^{\mathrm{f}}, \mathbf{x}_n^{\mathrm{p}})$ is ℓ_{f}-Lipschitz continuous with $\mathbf{x}^{\mathrm{f}}$ and ℓ_{fp}-Lipschitz continuous with $\mathbf{x}_n^{\mathrm{p}}$, that is,

$$\left\|\nabla_{\mathrm{f}} \mathcal{L}_n(\bar{\mathbf{x}}^{\mathrm{f}}, \mathbf{x}_n^{\mathrm{p}}) - \nabla_{\mathrm{f}} \mathcal{L}_n(\hat{\mathbf{x}}^{\mathrm{f}}, \mathbf{x}_n^{\mathrm{p}})\right\| \leq \ell_{\mathrm{f}} \left\|\bar{\mathbf{x}}^{\mathrm{f}} - \hat{\mathbf{x}}^{\mathrm{f}}\right\|, \tag{12}$$

  and

$$\left\|\nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}^{\mathrm{f}}, \bar{\mathbf{x}}_n^{\mathrm{p}}) - \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}^{\mathrm{f}}, \hat{\mathbf{x}}_n^{\mathrm{p}})\right\| \leq \ell_{\mathrm{fp}} \left\|\bar{\mathbf{x}}_n^{\mathrm{p}} - \hat{\mathbf{x}}_n^{\mathrm{p}}\right\|. \tag{13}$$

- $\nabla_{\mathrm{p}} \mathcal{L}_n(\mathbf{x}^{\mathrm{f}}, \mathbf{x}_n^{\mathrm{p}})$ is ℓ_{p}-Lipschitz continuous with $\mathbf{x}_n^{\mathrm{p}}$ and ℓ_{pf}-Lipschitz continuous with $\mathbf{x}^{\mathrm{f}}$.

Assumption 2. There exist $\delta \geq 0$ and $\rho \geq 0$, for all $\mathbf{x}^{\mathrm{f}}$ and $\boldsymbol{V}$, we have

$$\left\|\nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}^{\mathrm{f}}, \mathbf{x}_n^{\mathrm{p}})\right\|^2 \leq \delta^2 + \rho^2 \left\|\nabla_{\mathrm{f}} \mathcal{L}(\mathbf{x}^{\mathrm{f}}, \boldsymbol{X}^{\mathrm{p}})\right\|^2. \tag{14}$$

Assumption 1 is not stringent, which is satisfied by most deep neural networks [31]–[33]. Assumption 2 is commonly used for the FL convergence analysis, e.g., [10], [15]. We first provide a key lemma in the following, proved in Appendix A.

Lemma 1. *Let Assumption* 1 *holds, we have*

$$\begin{aligned}&\mathcal{L}(\mathbf{x}_{t+1}^{\mathrm{f}}, \boldsymbol{X}_{t+1}^{\mathrm{p}}) - \mathcal{L}(\mathbf{x}_{t}^{\mathrm{f}}, \boldsymbol{X}_{t}^{\mathrm{p}}) \\ &\leq \left\langle \nabla_{\mathrm{f}} \mathcal{L}(\mathbf{x}_{t}^{\mathrm{f}}, \boldsymbol{X}_{t}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_{t}^{\mathrm{f}} \right\rangle + \frac{1+\chi}{2} \ell_{\mathrm{f}} \left\| \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_{t}^{\mathrm{f}} \right\|^2 \\ &\quad + \frac{1}{D} \sum\nolimits_{n=1}^{N} D_n \Big(\left\langle \nabla_{\mathrm{p}} \mathcal{L}_n(\mathbf{x}_{t}^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\rangle \\ &\qquad + \frac{1+\chi}{2} \ell_{\mathrm{p}} \left\| \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\|^2 \Big), \end{aligned} \tag{15}$$

where $\chi = \max\{\ell_{\mathrm{fp}}, \ell_{\mathrm{pf}}\} / \sqrt{\ell_{\mathrm{f}} \ell_{\mathrm{p}}}$.

Based on Lemma 1, the one-round global loss reduction bound is derived in Appendix B, which is summarized in the following Lemma.

Lemma 2. *Let the above two assumptions hold, and the learning rate satisfy* $\eta_{\mathrm{f}} \leq \frac{1}{(\chi+1)\ell_{\mathrm{f}}}$, $\eta_{\mathrm{p}} \leq \frac{2}{(\chi+1)\ell_{\mathrm{p}}}$, *we have*

$$\begin{aligned}&\mathbb{E}\left[\mathcal{L}\left(\mathbf{x}_{t+1}^{\mathrm{f}}, \boldsymbol{X}_{t+1}^{\mathrm{p}}\right) - \mathcal{L}(\mathbf{x}_{t}^{\mathrm{f}}, \boldsymbol{X}_{t}^{\mathrm{p}})\right] \\ &\leq \frac{1}{2}\eta_{\mathrm{f}} \Big(\frac{4}{D^2} (D - \sum_{n=1}^{N} \beta_{n,t} D_n)^2 \rho^2 - 1 \Big) \mathbb{E}\left\| \nabla_{\mathrm{f}} \mathcal{L}(\mathbf{x}_{t}^{\mathrm{f}}, \boldsymbol{X}_{t}^{\mathrm{p}}) \right\|^2 \\ &\qquad + 2\eta_{\mathrm{f}} \frac{\left(D - \sum_{n=1}^{N} \beta_{n,t} D_n \right)^2 \delta^2}{D^2}. \end{aligned} \tag{16}$$

According to Lemma 2, the number of data samples scheduled in each round, i.e., $\sum_{n=1}^{N} \beta_{n,t} D_n$, is the main factor that affects the learning performance. In the following theorem, we derive the T-rounds convergence bound of PMA-FL, proved in Appendix C.

Theorem 1. *Let Assumption* 1 *and Assumption* 2 *hold. The learning rate satisfy* $\eta_{\mathrm{f}} \leq \frac{1}{(\chi+1)\ell_{\mathrm{f}}}$, $\eta_{\mathrm{p}} \leq \frac{2}{(\chi+1)\ell_{\mathrm{p}}}$, *the convergence bound in the* T*-th global round is given by*

$$\begin{aligned}&\mathbb{E}\left[\mathcal{L}(\mathbf{x}_{T}^{\mathrm{f}}, \boldsymbol{X}_{T}^{\mathrm{p}}) - \mathcal{L}(\mathbf{x}^{\mathrm{f},*}, \boldsymbol{X}^{\mathrm{p},*})\right] \\ &\quad \leq \left(\mathbb{E}\left[\mathcal{L}(\mathbf{x}_{0}^{\mathrm{f}}, \boldsymbol{X}_{0}^{\mathrm{p}}) - \mathcal{L}(\mathbf{x}^{\mathrm{f},*}, \boldsymbol{X}^{\mathrm{p},*})\right] \right) \prod\nolimits_{t=0}^{T-1} A_t \\ &+ \sum\nolimits_{t=0}^{T-1} \frac{2\eta_{\mathrm{f}}\delta^2}{D^2} \Big(D - \sum\nolimits_{n=1}^{N} \beta_{n,t} D_n \Big)^2 \prod\nolimits_{j=t+1}^{T-1} A_j, \end{aligned} \tag{17}$$

where $A_t = 1 + \eta_{\mathrm{f}}\ell_{\mathrm{f}}(\frac{4}{D^2}(D - \sum_{n=1}^{N} \beta_{n,t} D_n)^2 \rho^2 - 1)$.

From Theorem 1, we can conclude when t trends to infinity with $0 < \rho < \frac{1}{2}$ (i.e., $0 < A_t < 1$): 1) The FL training converges since $\prod_{t=0}^{T-1} A_t$ turns to 0 as T increases, resulting in the first term in the right-hand side (RHS) of (17) converges to zero and the second term in the RHS of (17) approaches to be fixed. 2) A gap, i.e., the second term in the RHS of (17), exists between $\mathcal{L}(\mathbf{x}_{T}^{\mathrm{f}}, \boldsymbol{X}_{T}^{\mathrm{p}})$ and $\mathcal{L}(\mathbf{x}^{\mathrm{f},*}, \boldsymbol{X}^{\mathrm{p},*})$. Particularly, A_t and the second term in the RHS of (17) affect the convergence speed and learning accuracy, respectively. A small A_t induces a fast learning speed, and a small $\sum_{t=0}^{T-1} \frac{2\eta_{\mathrm{f}}\delta^2}{D^2} (D - \sum_{n=1}^{N} \beta_{n,t} D_n)^2 \prod_{j=t+1}^{T-1} A_j$ results in a small loss function and high learning accuracy. Increasing $\sum_{n=1}^{N} \beta_{n,t} D_n$ in each round helps $\prod_{t=0}^{T-1} A_t$ approach 0 faster and decreases the second term in the RHS of (17). These observations motivate us to maximize $\sum_{n=1}^{N} \beta_{n,t} D_n$ in each round to improve the learning performance of PMA-FL. Note that, Theorem 1 reveals the impact of unbalanced data on the convergence performance of PMA-FL and builds the bridge between the scheduled data samples maximization and the global loss minimization from a theoretical perspective.

B. Problem Transformation

Motivated by Theorem 1, we maximize the overall scheduled data size, i.e., $\sum_{t=0}^{T-1} \sum_{n=1}^{N} \beta_{n,t} D_n$, for the global loss function minimization. Thus, similar to [9], [10], we transform problem $\mathcal{P}$ into maximizing $\sum_{t=0}^{T-1} \sum_{n=1}^{N} \beta_{n,t} D_n$ as follows:

$$\begin{aligned}\mathcal{P}_1: \quad &\max_{\left\{\boldsymbol{S}_t, \boldsymbol{T}_t^{\mathrm{L}}, \boldsymbol{T}_t^{\mathrm{U}}, \boldsymbol{\theta}_t\right\}_{t=0}^{T-1}} \quad \sum\nolimits_{t=0}^{T-1} \sum\nolimits_{n=1}^{N} \beta_{n,t} D_n \\ &\text{s. t.} \quad (8), (10), (11\text{b}), (11\text{c}), (11\text{d}), (11\text{e}), (11\text{f}). \end{aligned} \tag{18}$$

Directly solving problem $\mathcal{P}_1$ requires exact channel state and clients' energy status information of the entire learning process at the start of FL, which is impractical. To enable online dynamic scheduling for clients only based on the current-round information of clients, we construct a virtual queue $q_{n,t}$ for each client n to indicate the gap between the cumulative energy consumption till round t and the budget, evolving according to

$$q_{n,t+1} = \max\left\{ q_{n,t} + \beta_{n,t} \mathcal{E}_{n,t} - \frac{\mathcal{E}_n}{T}, 0 \right\}, \tag{19}$$

with an initial value $q_{n,0} = 0$ for all clients. According to the virtual queues of clients, the long-term energy constraint (11b) and the objective function (18) can be transformed into Lyapunov drift-plus-penalty ratio function based on the Lyapunov drift-plus-penalty algorithm [34]. Then, problem $\mathcal{P}_1$ is transformed into minimizing the drift-plus-penalty ratio function as follows:

$$\begin{aligned}\mathcal{P}_2: \quad &\min_{\boldsymbol{S}_t, \boldsymbol{T}_t^{\mathrm{L}}, \boldsymbol{T}_t^{\mathrm{U}}, \boldsymbol{\theta}_t} \quad -\mathcal{V} \sum_{n=1}^{N} \beta_{n,t} D_n + \sum_{n=1}^{N} q_{n,t} \beta_{n,t} \mathcal{E}_{n,t} \\ &\text{s. t.} \quad (8), (10), (11\text{c}), (11\text{d}), (11\text{e}), (11\text{f}). \end{aligned} \tag{20}$$

where $\mathcal{V} \geq 0$ is an adjustable weight parameter to balance scheduled data size and energy consumption. A large $\mathcal{V}$ indicates that the optimization objective emphasizes more on the scheduled data size for improving the learning performance and less on energy consumption minimization, and vice versa.

IV. Energy-Efficient Dynamic Client Selection and Resource Management

In this section, we solve the deterministic combinatorial problems $\mathcal{P}_2$ in each communication round. Note that the developed client selection and resource allocation solutions in this section can be readily generalized to other FL systems, such as the case of aggregating the entire model, by replacing the objective function of $\mathcal{P}_2$ and the communication model detailed in Section II-D with their respective convergence bounds and communication cost models. We first exploit the dependences among $\boldsymbol{S}_t$, $\boldsymbol{\theta}_t$, $\boldsymbol{T}_t^{\mathrm{L}}$, and $\boldsymbol{T}_t^{\mathrm{U}}$ in problem $\mathcal{P}_2$ and transform it into an equivalent problem that joint optimizing $\boldsymbol{S}_t$, $\boldsymbol{\theta}_t$, and $\boldsymbol{T}_t^{\mathrm{L}}$. Then we decompose it into three sub-problems and deploy an alternative optimization technique to obtain its

optimal solution. For the convenience of analysis, we rewrite the local feature extractor uploading energy consumption as

$$\mathcal{E}^{\mathrm{U}}_{n,t} = p_{n,t}T^{\mathrm{U}}_{n,t} = \frac{\theta_{n,t}BN_0T^{\mathrm{U}}_{n,t}}{h_{n,t}}\Big(2^{\frac{Qq}{\theta_{n,t}BT^{\mathrm{U}}_{n,t}}}-1\Big), \quad (21)$$

which is a non-increasing function with respect to $T^{\mathrm{U}}_{n,t}$. Thus, by taking into account the constraint (11f), the optimal communication time satisfies $T^{\mathrm{U}}_{n,t} = T_{\max} - T^{\mathrm{L}}_{n,t}$. Based on this, we can simplify problem $\mathcal{P}_2$ as the following equivalent problem,

$$\mathcal{P}_3: \min_{\{\boldsymbol{S}_t,\boldsymbol{\theta}_t,\boldsymbol{T}^{\mathrm{L}}_t\}} -\mathcal{V}\sum_{n=1}^{N}\beta_{n,t}D_n + \sum_{n=1}^{N}q_{n,t}\beta_{n,t}\mathcal{E}_{n,t} \quad (22)$$
$$\text{s. t. } (11\text{c}), (11\text{d}), (11\text{e}),$$
$$\frac{\tau D_nC_n}{f_{n,\max}} \le T^{\mathrm{L}}_{n,t} \le T_{\max} - \frac{Qq}{r^{\max}_{n,t}(\theta_{n,t})}, \quad (22a)$$

where

$$r^{\max}_{n,t}(\theta_{n,t}) = \theta_{n,t}B\log\Big(1+\frac{p_{n,\max}h_{n,t}}{\theta_{n,t}BN_0}\Big). \quad (23)$$

However, problem $\mathcal{P}_3$ is a mixed integer non-linear programming, which is still difficult to solve. Below we decompose it into three sub-problems and solve them one by one.

A. Local Training Time Allocation

For fixed client scheduling policy $\boldsymbol{S}_t$ and bandwidth allocation strategy $\boldsymbol{\theta}_t$, we can decompose the computation time allocation problem as follows,

$$\mathcal{P}_4: \min_{\boldsymbol{T}^{\mathrm{L}}_t} \quad \sum\nolimits_{n\in\boldsymbol{S}_t} q_{n,t}\mathcal{E}_{n,t} \quad (24)$$
$$\text{s. t.} \quad (22\text{a}).$$

Problem $\mathcal{P}_4$ is convex, its optimal solution is summarized in Lemma 3, proved in Appendix D.

Lemma 3. *Problem $\mathcal{P}_4$ is a convex problem and its optimal solution is given as*

$$T^{\mathrm{L},*}_{n,t} = \begin{cases} \frac{\tau D_nC_n}{f_{n,\max}}, & T^{\mathrm{L},0}_{n,t} \le \frac{\tau D_nC_n}{f_{n,\max}}, \\ T_{\max} - \frac{Qq}{r^{\max}_{n,t}(\theta_{n,t})}, & T^{\mathrm{L},0}_{n,t} \ge T_{\max} - \frac{Qq}{r^{\max}_{n,t}(\theta_{n,t})}, \\ T^{\mathrm{L},0}_{n,t}, & \text{otherwise.} \end{cases} \quad (25)$$

where $T^{\mathrm{L},0}_{n,t}$ satisfies the equality $\frac{\partial\mathcal{E}_{n,t}}{\partial T^{\mathrm{L},0}_{n,t}} = 0$.

In fact, constraint (22a) imposes restrictions on the maximum frequency and transmit power and is usually inactive in practical system design because this usually can be satisfied by modifying the minimum required latency constraint, $T_{\max}$, and bandwidth B. Thus, we have the following remark.

Remark 1. In general, the optimal computation time satisfy $T^{\mathrm{L},*}_{n,t} = T^{\mathrm{L},0}_{n,t}$, which is equivalent to $\frac{\partial\mathcal{E}^{\mathrm{L}}_{n,t}}{\partial T^{\mathrm{L}}_{n,t}} = \frac{\partial\mathcal{E}^{\mathrm{U}}_{n,t}}{\partial T^{\mathrm{U}}_{n}}$. In other words, the computation time allocation policy is optimal when the power of local training equals that of wireless communication.

B. Wireless Bandwidth Allocation

For any given computation time allocation decision $\boldsymbol{T}^{\mathrm{L}}_t$ and client scheduling policy $\boldsymbol{S}_t$, the bandwidth allocation problem can be separated from problem $\mathcal{P}_3$ as follows:

$$\mathcal{P}_5: \min_{\boldsymbol{\theta}_t} \quad \sum\nolimits_{n\in\boldsymbol{S}_t} q_{n,t}\mathcal{E}_{n,t} \quad (26)$$
$$\text{s. t.} \quad (11\text{d}), (11\text{e}),$$
$$Qq/(T_{\max} - T^{\mathrm{L}}_{n,t}) \le r^{\max}_{n,t}(\theta_{n,t}), \quad (26\text{a})$$

For the sake of analysis, we introduce an auxiliary function for each client n ($n \in \mathcal{N}$) as follows:

$$g_n(\theta_{n,t}) = \exp\left(\frac{Qq\ln 2}{\theta_{n,t}B(T_{\max} - T^{\mathrm{L}}_{n,t})}\right) - 1. \quad (27)$$

By removing the constant terms in the objective function (26), the objective function of bandwidth allocation problem can be written as

$$h(\boldsymbol{\theta}_t) = \sum\nolimits_{n\in\boldsymbol{S}_t}\theta_{n,t}\frac{N_0Bq_{n,t}(T_{\max} - T^{\mathrm{L}}_{n,t})}{h_{n,t}}g_n(\theta_{n,t}). \quad (28)$$

Consequently, we reformulate the wireless bandwidth allocation problem as follows:

$$\widehat{\mathcal{P}}_5: \min_{\boldsymbol{\theta}_t} \quad h(\boldsymbol{\theta}_t) \quad (29)$$
$$\text{s. t.} \quad (11\text{d}), (11\text{e}), (26\text{a}).$$

Problem $\widehat{\mathcal{P}}_5$ is a standard convex optimization problem, its proof is similar to that for Lemma 3 and thus omitted for brevity. Applying Karush-Kuhn-Tucker condition, the optimal solution for $\boldsymbol{\theta}_t$ satisfies

$$\frac{\partial h(\boldsymbol{\theta}_t)}{\partial\theta_{n,t}} = -\lambda^*, \forall n \in \boldsymbol{S}_t, \quad (30)$$

where λ^* is the optimal Lagrange multiply and $\sum_{n\in\boldsymbol{S}_t}\theta_{n,t} = 1$. Thus, for each client n, we have

$$g_k(\theta_{n,t}) + \theta_{n,t}g'_k(\theta_{n,t}) = \frac{-\lambda^*h_{n,t}}{q_{n,t}N_0B(T_{\max} - T^{\mathrm{L}}_{n,t})}, \quad (31)$$

its inverse function is

$$\theta_{n,t}(\lambda^*) = \frac{Qq\ln 2}{B(T_{\max} - T^{\mathrm{L}}_{n,t})\left(\mathcal{F}(\frac{\lambda^*h_{n,t}}{q_{n,t}N_0B(T_{\max} - T^{\mathrm{L}}_{n,t})e} - \frac{1}{e}) + 1\right)}, \quad (32)$$

where $\mathcal{F}$ refers to the principal branch of Lambert function, which is the solution of $\mathcal{F}(a)e^{\mathcal{F}(a)} = a$.

In (32), there still exists an unknown variable λ^*. The value of λ^* satisfies $\sum_{n=1}^{N}\theta_{n,t}(\lambda^*) = 1$. Since the expression of $\theta_{n,t}(\lambda^*)$ is complicated, it is difficult to solve the optimal λ^*. Below we propose a bisection search method to solve $\sum_{n=1}^{N}\theta_{n,t}(\lambda^*) = 1$. To proceed, we have the following Proposition.

Proposition 1. *$\theta_{n,t}(\lambda)$ is a monotonically decreasing function with respect to λ.*

Proof. Since the Lagrange multiply $\lambda > 0$, we have $\frac{\lambda h_{n,t}}{eq_{n,t}N_0B(T_{\max} - T^{\mathrm{L}}_{n,t})} - \frac{1}{e} > -\frac{1}{e}$. Since $\mathcal{F}(a)$ is positively correlated to a when $a \ge -\frac{1}{e}$, $\theta_{n,t}(\lambda)$ is monotonically

decreasing with λ. □

Based on Proposition 1, the bisection search method is employed to solve the equation. In the following, we derive the bisection search upper and lower bound on λ. Since $\lambda > 0$, the lower bound of λ is $\lambda_{\text{LB}} = 0$. For deriving the upper bound, we have $\max_{n\in\boldsymbol{S}_t}\{\theta_{n,t}(\lambda)\} \geq 1/|\boldsymbol{S}_t|$, thus

$$\mathcal{F}\left(\frac{\lambda h_{n,t}}{q_{n,t}N_0 B(T_{\max}-T^{\text{L}}_{n,t})e}-\frac{1}{e}\right) \leq \frac{|\boldsymbol{S}_t|Qq\ln 2}{B(T_{\max}-T^{\text{L}}_{n,t})}-1. \tag{33}$$

Let $\varphi_k = \frac{|\boldsymbol{S}_t|Qq\ln 2}{B(T_{\max}-T^{\text{L}}_{n,t})}$, from the definition of Lambert $\mathcal{F}$ function, we have

$$\lambda_{\text{UB}} = \max_{n\in\boldsymbol{S}_t}\left\{\frac{q_{n,t}N_0 B(T_{\max}-T^{\text{L}}_{n,t})\left((\varphi_k-1)e^{\varphi_k}+1\right)}{h_{n,t}}\right\}. \tag{34}$$

According to λ_{LB} and λ_{UB}, we can find the optimal Lagrange multiply, λ^*, through dichotomy method. Furthermore, we can find the optimal bandwidth allocation policy $\boldsymbol{\theta}_t$ based on (32).

Remark 2. According to (30), when the bandwidth allocation policy is optimal, all clients' energy consumption-bandwidth rates (i.e., $\frac{\partial h(\boldsymbol{\theta}_t)}{\partial \theta_{n,t}}$) are equal. This actual achieves the energy consumption balance between clients. Moreover, similar to the proof of Proposition 1, it can be proved that the optimal bandwidth form in (32) is monotonically decreasing with $h_{n,t}$ and increasing with $q_{n,t}$. Thus, more bandwidth should be allocated to the clients with weaker channels (smaller $h_{n,t}$) and less remaining energy budgets (larger $q_{n,t}$).

C. Client Selection Policy

Until now, for any given $\boldsymbol{S}_t$, the computation time or wireless bandwidth allocation policies can be solved if one of them is fixed. Below we solve the joint computation time and wireless bandwidth allocation policy. For clarity, we formulate the joint computation time and bandwidth allocation problem under given client scheduling decision $\boldsymbol{S}_t$ as follows:

$$\begin{aligned}\mathcal{P}_6: \min_{\{\boldsymbol{\theta}_t, \boldsymbol{T}^{\text{L}}_t\}} & \sum\nolimits_{n\in\boldsymbol{S}_t} q_{n,t}\mathcal{E}_{n,t} \\ \text{s. t.} \quad & (11\text{d}), (11\text{e}), (22\text{a}),\end{aligned} \tag{35}$$

which is a combination problem of $\mathcal{P}_4$ and $\mathcal{P}_5$. Building on the preceding results, the computation time allocation problem, $\mathcal{P}_4$, and the bandwidth allocation problem, $\mathcal{P}_5$, are both convex optimization problems, problem $\mathcal{P}_6$ is also a convex optimization problem. Thus, we solve the joint computation time and wireless bandwidth allocation policies by using the block-coordinate descent method [35], which iterates between problem $\mathcal{P}_4$ and problem $\mathcal{P}_5$. Each iteration consists of two steps: (1) solving the optimal solution of problem $\mathcal{P}_5$ for given $\boldsymbol{T}^{\text{L}}_t$; (2) solving the computation time allocation policy $\boldsymbol{T}^{\text{L}}_t$ based on the obtained bandwidth allocation solution $\boldsymbol{\theta}_t$. The two steps are iterated until convergence. We summarize the details on joint optimization of computation time and wireless bandwidth in Algorithm 1. Note that the block-coordinate descent method has been proven to be convergent in solving the convex optimization problem [36]. Thus, the convergence of Algorithm 1 to solve the convex optimization problem $\mathcal{P}_6$ can be guaranteed. Based on the complexity analysis results in [35], the time complexity of Algorithm 1 is $\mathcal{O}(2N^{3.5})$.

Algorithm 1 Computation Time and Bandwidth Allocation

1: Input $\boldsymbol{S}_t$, the computation time as $\widetilde{T}^{\text{L}}_t$, and bandwidth allocation policy $\boldsymbol{\theta}_t$, the tolerant error $\Upsilon > 0$
2: Calculate the objective function value (20), denote as $\mathcal{B}_0$
3: **repeat**
4: Calculate the upper limit of the Lagrange multiply λ_{UB} based on (34), and let $\lambda_{\text{LB}} = 0$
5: Utilize the bisection search method to solve the optimal bandwidth allocation policy $\boldsymbol{\theta}_t$
6: Solve the computation time allocation policy based on the obtained $\boldsymbol{\theta}_t$ by using (25), update $\widetilde{T}^{\text{L}}_t$
7: Calculate the objective function value (35) of $\boldsymbol{S}_t$ by substituting the obtained $\widetilde{T}^{\text{L}}_t$ and $\boldsymbol{\theta}_t$, denote as $\mathcal{B}_1$
8: $\Delta = \mathcal{B}_0 - \mathcal{B}_1$, update $\mathcal{B}_0 = \mathcal{B}_1$
9: **until** $\Delta \leq \Upsilon$
10: **return** The computation time allocation policy $\widetilde{T}^{\text{L}}_t$ and bandwidth allocation policy $\boldsymbol{\theta}_t$

Through the above analysis, we can solve the optimal value of the objective function in (22) for any given client scheduling decision $\boldsymbol{S}_t$. An intuitive approach for finding the client selection solution is to solve the objective function value of all the possible client scheduling decisions first and then select the one with the minimum objective function value. However, this method has exponential time complexity $\mathcal{O}(N^{3.5}\times 2^{N+1})$ since there are total $\sum_{l=0}^{N} C_N^l = 2^N$ possible client scheduling decisions. To tackle this challenge, we have the following designs.

Based on (22), it is desired to schedule clients with small $q_{n,t}$ and $\mathcal{E}_{n,t}$. The small $\mathcal{E}_{n,t}$ can be achieved by strong channels or/and high computation efficiencies. To identify such clients, we first perform equal bandwidth allocation over all clients and then evaluate the resulting energy consumption of each client $\bar{\mathcal{E}}_{n,t}$. Specifically, each client n is allocated the same portion, $\theta_{n,t} = \frac{1}{N}$, of the total bandwidth B, and then solve problem $\mathcal{P}_4$ to obtain the computation time allocation policy $\boldsymbol{T}^{\text{L}}_t$. Then, by substituting $\theta_{n,t} = \frac{1}{N}$ and $\boldsymbol{T}^{\text{L}}_t$ into the (7) and (21), the estimated energy consumption is $\bar{\mathcal{E}}_{n,t} = \mathcal{E}^{\text{U}}_{n,t} + \mathcal{E}^{\text{L}}_{n,t}$.

Based on $\bar{\mathcal{E}}_{n,t}$, we sort $\mathcal{C}_{n,t} = q_{n,t}\bar{\mathcal{E}}_{n,t}$ in the ascending order, and then use the set expansion algorithm [8] to find the client selection decision. Firstly, the clients with $q_{n,t} = 0$ are all added into client set Δ, denote this client set by Δ_0. Next, we gradually add the clients with $q_{n,t} > 0$ into Δ based on the ascending order of $\mathcal{C}_{n,t}$. For each possible client scheduling set Δ, we use Algorithm 1 to compute the computation time and bandwidth allocation decisions. Let $\mathcal{R}^*(\Delta) = (\theta^*(\Delta), T^*(\Delta))$ denote the time and wireless bandwidth decision and $\mathcal{Z}(\Delta)$ represent the corresponding objective function value of Δ, respectively. Let $\mathcal{G}$ denote the set encompassing all possible client scheduling sets Δ.

It is worth mentioning that $\mathcal{Z}(\Delta_0) = -\mathcal{V}\sum_{k\in\Delta_0} D_n$ due to $q_{n,t} = 0$ for all $k \in \Delta_0$. As the energy consumption of users in Δ_0 has no impact on the objective function, we allocate the minimum required bandwidth to these clients and conserve bandwidth for users belonging to $(\Delta - \Delta_0)$. Moreover, we add the users with $q_{n,t} > 0$ one by one into Δ and solve the $\mathcal{R}^*(\Delta)$ and $\mathcal{Z}(\Delta)$. For Δ, if its optimal computation time and wireless bandwidth allocation policy results in $-\mathcal{V}D_n + q_{n,t}\mathcal{E}_{n,t} > 0$ for the last added client n, we stop adding clients into Δ and remove the last added client. Then, the client scheduling decision is the set $\Delta \in \mathcal{G}$ with minimal objective function value, i.e., $\boldsymbol{S}_t^* = \arg\min_{\Delta\in\mathcal{G}} \mathcal{Z}(\Delta)$. The computation time and optimal bandwidth allocation policy correspond to $T_t^*(\boldsymbol{S}_t^*)$ and $\theta_t^*(\boldsymbol{S}_t^*)$. The details involved in client scheduling are summarized in Algorithm 2, which obtains the client scheduling solution of problem $\mathcal{P}_1$ by solving at most N times convex problem $\mathcal{P}_6$. Thus, the time complexity of Algorithm 2 is $\mathcal{O}(2N^{4.5})$, which is smaller than $O(N^{3.5} \times 2^{N+1})$ when $N > 1$.

Algorithm 2 Client Selection Algorithm

1: Input $q_{n,t}$ ($n \in \mathcal{N}$), and $\mathcal{V}$
2: Sort $\mathcal{C}_{n,t}$ in ascending order.
3: Set $\Delta_0 = \{n : q_{n,t} = 0\}$, $\Delta = \Delta_0$ and $\mathcal{G} = \{\Delta_0\}$
4: **for** $k = |\Delta_0|+1, \cdots, K$ **do**
5: Update $\Delta = \Delta \cup \{n\}$
6: Apply Algorithm 1 for allocate bandwidth and computation time, i.e., $\mathcal{R}(\Delta) = (\boldsymbol{T}_t^{\mathrm{L}}, \boldsymbol{\theta}_t)$.
7: **if** $-\mathcal{V}D_n + q_{n,t}\mathcal{E}_{n,t} \leq 0$ **then**
8: $\mathcal{G} = \mathcal{G} \cup \Delta$
9: **else**
10: Quit the circulation
11: **end if**
12: **end for**
13: Compute the client selection policy as $\boldsymbol{S}_t^* = \arg\min_{\Delta\in\mathcal{G}} \mathcal{Z}(\Delta)$
14: **return** The optimal client selection decision $\boldsymbol{S}_t^*$, computation time $\boldsymbol{T}_t^{\mathrm{L}}$ and wireless bandwidth allocation $\boldsymbol{\theta}_t$

TABLE I
SIMULATION SETTINGS

Parameter	Value	Parameter	Value
N	100	B	10MHz
$f_{n,\max}$	1GHz	N_0	-174dBm/Hz
v	2	$p_{n,\max}$	30mW
κ	5×10^{-27}	$\eta_{\mathrm{f}}, \eta_{\mathrm{p}}$	0.05
τ	5	q	32
h_0	-30dB	ζ_n	4
Q(MLP)	533248	C_n(MLP)	137,586
$\bar{\mathcal{E}}_n$(MLP)	0.1J	$T_{\max}$(MLP)	2s
Q(CNN)	307192	C_n(CNN)	7051728.5
$\bar{\mathcal{E}}_n$(CNN)	2J	$T_{\max}$(CNN)	14s
Q(VGG-11)	9220480	C_n(VGG-11)	85858649
$\bar{\mathcal{E}}_n$(VGG-11)	30J	$T_{\max}$(VGG-11)	200s

V. NUMERICAL RESULTS

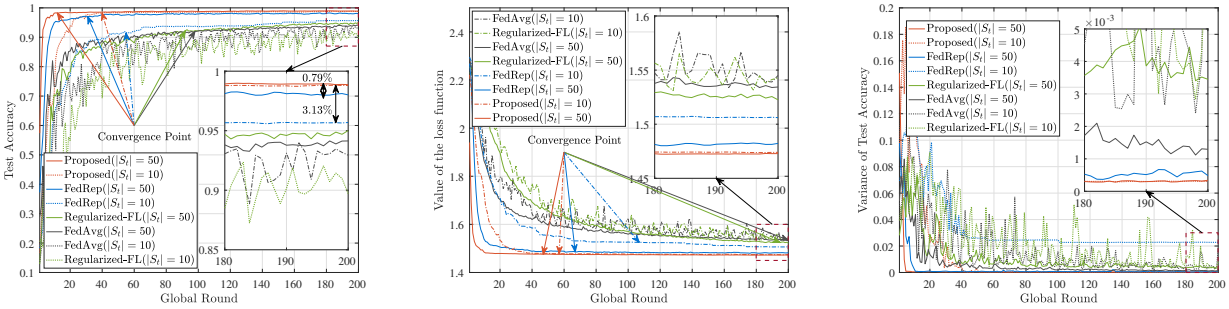
A. Experimental Setting

We consider that N clients are randomly distributed within a 500m $\times$ 500m single cell, and the PS is located in the cell's centre. According to the real measurement result in [27], we set the energy coefficient $\kappa = 5 \times 10^{-27}$. The channel gain is modeled as $h_{n,t} = h_0\rho_{n,t}d_n^{-v}$ [37], where h_0 is the path loss constant; d_n is the distance from client n to the PS; $\rho_{n,t} \sim \mathrm{Exp}(1)$ is the small-scale fading channel gain; v is the path loss factor. We evaluate the proposed algorithm for image classification tasks using MNIST, CIFAR-10, and CIFAR-100 datasets. Similar to [9], [32], [33], we sort the data samples by their labels and distribute a disjoint subset of data with two labels to each client, i.e., each client has at most two classes of data. Similar to [9], [10], for the MNIST dataset, we train a four-layer MLP consisting of 784, 512, 256, 64, and 10 neurons. In our proposed PMA-FL, clients only share parameters of the first two layers, which accounts for 96.7% of the entire model parameters. For CIFAR-10, we train a CNN consisting of two 5×5 convolution layers, each of which has 64 channels and a 2×2 max-pooling layer. Then, three fully connected layers have 120, 64, and 10 units, respectively [9]. The PMA-FL only share the first 4 layers in the training process, which has 99.7% of the total number of model parameters. For CIFAR-100, we train a VGG-11 model [38]. Note that the original VGG-11 model has 1000 output units. To adapt VGG-11 to the CIFAR-100 dataset, we replace its full-connected layers with the following structure: a 512-unit hidden layer, a 256-unit hidden layer, and a 100-unit output layer. The PMA-FL only share the first 8 layers in the training process, accounting for 77% of the entire model parameters. The cross entropy is adopted as the loss function for the above illustrated three models. In the simulations, we use $\bar{\mathcal{E}}_n$ denote the energy budget per round of client n, and the total energy budget of client n is $\mathcal{E}_n = T\bar{\mathcal{E}}_n$. The default experiment settings are summarized in Table I unless specified otherwise.

Note that, similar to many existing personalized FL works, e.g., [15], [19], [32], the test accuracy and loss value reported in the simulations is the average accuracy and average loss of all clients' model on their local datasets, respectively.

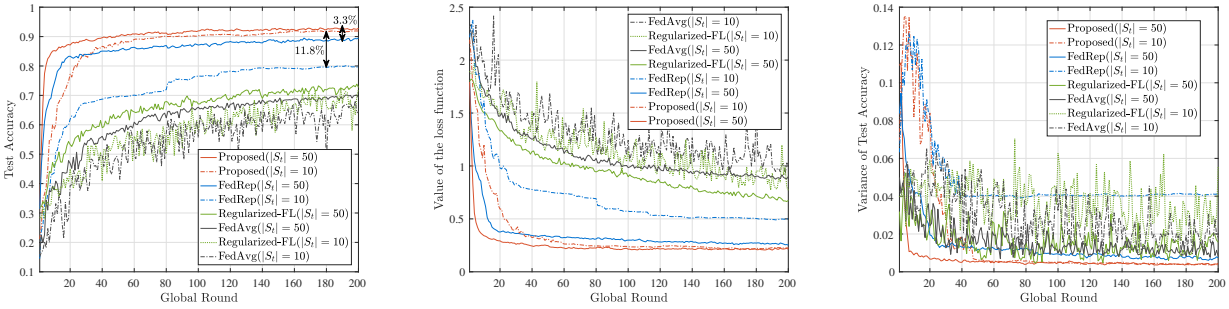
B. Performance of Partial Model Parameters Aggregation

To verify the advantages of the proposed PMA-FL algorithm, we compare its performance with three benchmarks. 1) *Regularized FL* [16]: Regularized FL uses a proximal term to regularize each local loss function for tackling the data heterogeneity. 2) *FedAvg* [18]: The selected clients upload the entire model to the PS for aggregation in each round. 3) *FedRep* [19]: In each round, the selected clients sequentially train the feature extractor and predictor. Then, the selected clients upload their feature extractors for aggregation. Actually, Regularized FL and FedAvg requires more computation and bandwidth resources than the proposed approach. Note that this subsection mainly evaluates the effectiveness of the proposed partial model aggregation approach. Therefore, energy and bandwidth limits are ignored in this subsection. The overall effectiveness of the proposed joint learning and wireless network design will be evaluated in Section V-C by considering all the energy and wireless resource limitations. In addition, we evaluate the fairness among clients through the variance of test accuracy of clients. Specifically, let ACC_n



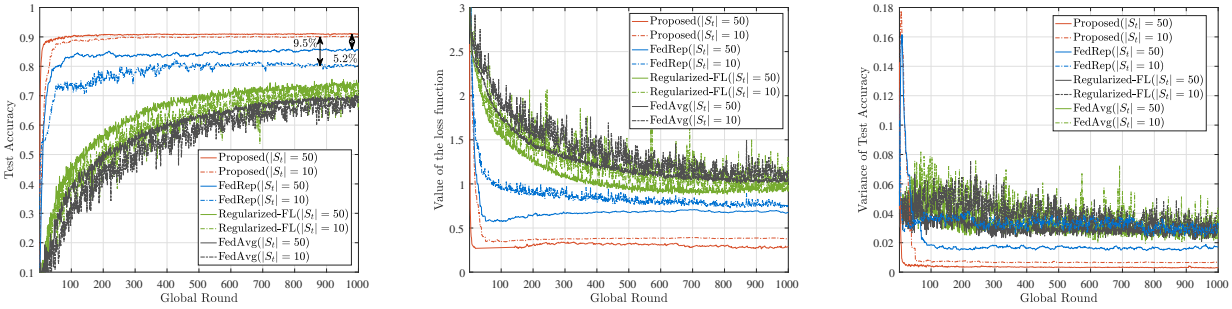
(a) (b) (c)

Fig. 3. Comparison between the proposed PMA approach and benchmarks on MNIST dataset: (a) test accuracy; (b) loss value; (c) variance of test accuracy.



(a) (b) (c)

Fig. 4. Comparison between the proposed PMA approach and benchmarks on CIFAR-10 dataset: (a) test accuracy; (b) loss value; (c) variance of test accuracy.



(a) (b) (c)

Fig. 5. Comparison between the proposed PMA approach and benchmarks on CIFAR-100 dataset: (a) test accuracy; (b) loss value; (c) variance of test accuracy.

denote the test accuracy of client n's local model on its test dataset, and then the variance of test accuracy is computed as $\frac{1}{N}\sum_{n=1}^{N}(ACC_n - \overline{ACC})^2$, where $\overline{ACC} = \frac{1}{N}\sum_{n=1}^{N} ACC_n$ is the average test accuracy of clients.

Fig. 3 compares the performance of the proposed approach with two benchmarks on the MNIST dataset. It is observed that the proposed approach outperforms the benchmarks. Specifically, the proposed approach boosts 3.13% when $|\boldsymbol{S}_t| = 10$ and 0.79% accuracy when $|\boldsymbol{S}_t| = 50$ compared with the benchmarks. Moreover, the proposed approach converges faster than the benchmarks. Note that the convergence point in 3(a) and 3(b) are defined as the first point that the variation of test accuracy and loss value is less than 10^{-6}, respectively. Additionally, compared with the three benchmarks, the proposed approach is less sensitive to the fraction of participating clients in each round. After 40 global rounds, the proposed approach with 10 clients participating in each round can obtain a similar performance as 50 clients participating in each round. The client participating ratio only affects the convergence speed and almost without reducing the final accuracy. However, the benchmarks are sensitive for the fraction of participating clients in each round, especially the training processes of Regularized FL and FedAvg are unstable when the participating ratio of clients is small, like 10 clients. In addition, Fig. 3(c) demonstrates that the proposed approach achieves the lowest variance in test accuracy among clients than the benchmarks. Thus, the proposed approach promotes fairness among clients more effectively than the benchmarks.

Fig. 4 compares the proposed approach to the benchmarks on CIFAR-10, drawing a similar conclusion with the experiments on the MNIST dataset. In particular, the proposed approach obtained a more distinct performance improvement on this more complicated dataset, boosting 11.8% and 3.3% accuracy than the benchmark schemes when $|\boldsymbol{S}_t| = 10$ and $|\boldsymbol{S}_t| = 50$, respectively. Fig. 4(c) shows that the proposed approach achieves the lowest variance in test accuracy among clients compared to the benchmarks. In addition, Fig. 5 evaluates the performance of the proposed approach on the CIFAR-100 dataset. Compared to the benchmarks, it is observed that the proposed approach improves the test accuracy by 9.5% and 5.2% when $|\boldsymbol{S}_t| = 10$ and $|\boldsymbol{S}_t| = 50$, respectively. It is also observed that the proposed approach has the lowest variance in test accuracy among clients, indicating that it is more beneficial for promoting fair learning among clients than the benchmarks.

C. Performance of Energy-Efficient Client Selection Algorithm

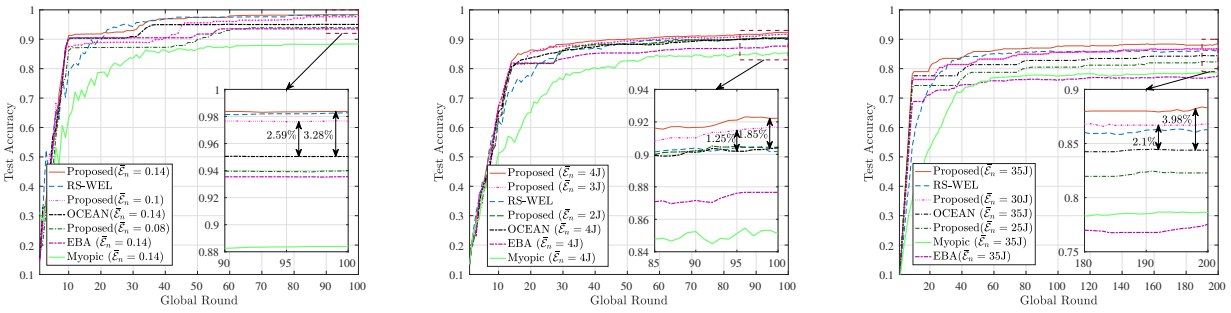
This subsection evaluates the proposed dynamic client scheduling approach through comparing it with the following client scheduling schemes. For fairness, we use these benchmark schemes to schedule clients for the proposed PMA-FL approach instead of their original FedAvg approach. Each curve is averaged over 50 runs in this subsection, respectively. 1) *OCEAN* [8]: In OCEAN, the spectral bandwidth is orthogonally allocated to the selected clients in each round to maximize the scheduled data samples, regardless of the heterogeneous local training and communication time among clients. 2) *Random scheduling without energy limitation (RS-WEL)*: Clients do not have energy limitation while the bandwidth and delay constraints exist. In each round, RS-WEL incrementally adds clients (randomly selected from all clients without replacement) into the scheduling set until violating the bandwidth constraint. 3) *Equal bandwidth allocation policy (EBA)* [10]: In each round, the client selection policy is determined by the proposed approach, while the bandwidth is equally allocated to each selected client. 4) *Myopic client selection policy (Myopic)* [7]: For each client n, the available energy in round t is given by the remaining energy divided by the remaining number of rounds, i.e., $\frac{\mathcal{E}_n - \sum_{i=0}^{t-1} \beta_{n,i} \mathcal{E}_{n,i}}{T-t-1}$.

Fig. 6 compares the performance of the proposed approach and the benchmarks under different clients' energy budgets, demonstrating the advantages of the joint optimization of client selection, bandwidth allocation, communication time, and computation time allocation policies. From the simulation results on the MNIST dataset in Fig. 6(a), it is observed that the proposed outperforms the benchmarks under the same clients' energy budgets. Specifically, given the same energy budget, i.e., $\bar{\mathcal{E}}_n = 0.14$J, the proposed algorithm achieves 3.28% test accuracy improvement compared to the OCEAN algorithm and higher accuracy improvement than the EBA and Myopic scheme. Moreover, the proposed algorithm is able to obtain better performance than the OCEAN algorithm with a smaller energy budget. Specifically, the proposed algorithm with energy budget $\bar{\mathcal{E}}_n = 0.1$J (71% of the energy budget of OCEAN) remains improving 2.59% accuracy compared to the OCEAN algorithm with energy budget $\bar{\mathcal{E}}_n = 0.14$J. That is, the proposed approach is able to obtain better learning performance than the benchmarks while saving 29% energy consumption. Compared with the RS-WEL scheme with unlimited energy budget, the proposed algorithm slightly improves accuracy when the energy budget is $\bar{\mathcal{E}}_n = 0.14$J.

Similar evaluations on the CIFAR-10 and CIFAR-100 datasets are shown in Fig. 6(b) and Fig. 6(c), respectively. For the simulation results on the CIFAR-10 dataset in Fig. 6(b), given energy budget $\bar{\mathcal{E}}_n = 4$J for both the proposed algorithm and the benchmark algorithms, the proposed algorithm achieves around a 1.85% accuracy boosts compared to the OCEAN algorithm and more notable improvements than the EBA and Myopic schemes. Moreover, the proposed algorithm under 75% energy budget ($\bar{\mathcal{E}}_n = 3$J) outperforms the OCEAN algorithm with an energy budget $\bar{\mathcal{E}}_n = 4$J, obtaining 1.25% accuracy gain. Additionally, the proposed algorithm with $\bar{\mathcal{E}}_n = 2$J obtains a similar performance as the OCEAN algorithm with $\bar{\mathcal{E}}_n = 4$J and the RS-WEL scheme. For the results on the CIFAR-100 dataset in Fig. 6(c), when clients' energy budgets are set to $\bar{\mathcal{E}}_n = 35$J, the proposed approach boosts at least 3.98% accuracy compared to the benchmarks. In addition, the proposed approach with $\bar{\mathcal{E}}_n = 30$J still outperforms than the benchmarks with $\bar{\mathcal{E}}_n = 35$J. The performance gain mainly comes from the joint optimization for both computation and wireless resources. In our proposed algorithm, the participating clients can get a trade-off between computation and communication energy consumption, achieving the most energy-efficient learning process. Specifically, the clients with poor channel conditions can boost their CPU frequency for reducing computation time and thus reserve more time for wireless communications. In contrast, clients with good channel conditions can lower the CPU frequency to balance computation and communication energy consumption.

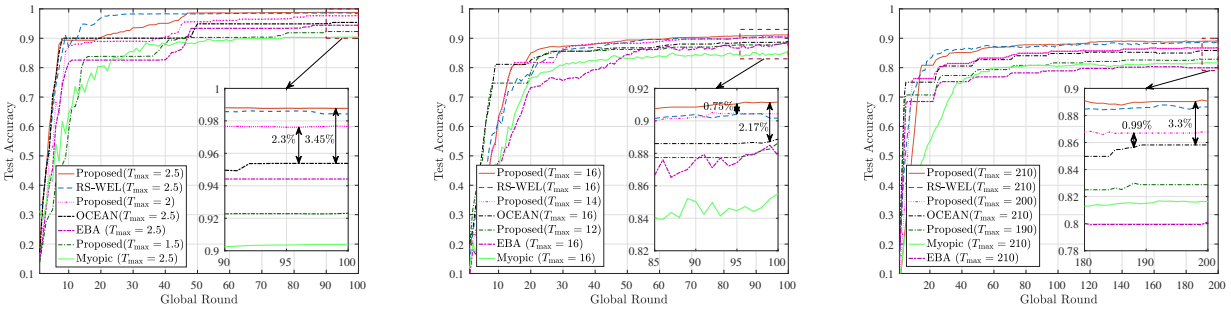
We compare our proposed client selection algorithm with the benchmarks under different latency constraints in Fig. 7. Clearly, as the latency constraint, $T_{\max}$, increases, the learning performance is improved. This is because a larger $T_{\max}$ helps save the computation and communication energy and thus more data samples are able to scheduled in each round. From the comparison results on MNIST dataset in Fig. 7(a), under the same latency constraints, i.e., $T_{\max} = 2.5$s, the proposed algorithm boosts 3.45% test accuracy compared with the OCEAN algorithm. Using the RS-WEL as the baseline, the proposed algorithm obtains a minor accuracy gain. One interesting phenomenon is that the proposed algorithm outperforms the OCEAN algorithm with a stricter delay restriction. Specifically, given time budget $T_{\max} = 2$s for the proposed algorithm, it obtains 2.3% accuracy gain than the OCEAN algorithm with $T_{\max} = 2.5$s. In other words, the proposed algorithm is able to obtain a better accuracy with a 20% time budget reduction.

Fig. 7(b) shows the impact of time budget on CIFAR-10 dataset, obtaining a similar conclusion as MNIST. Specifically, the proposed algorithm boosts 2.17% test accuracy with the OCEAN algorithm under same delay restriction $T_{\max} = 16$s. Compared to RS-WEL, the proposed algorithm gains 0.75% performance improvement with $T_{\max} = 16$s, and obtains a



(a) (b) (c)

Fig. 6. Performance of the proposed algorithm and benchmarks under different energy budget $\bar{\mathcal{E}}_n$: (a) on MNIST dataset; (b) on CIFAR-10 dataset; (c) on CIFAR-100 dataset.



(a) (b) (c)

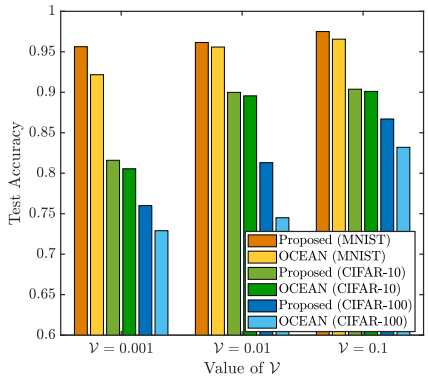
Fig. 7. Performance of the proposed algorithm and benchmarks under different delay constraint $T_{\max}$: (a) on MNIST dataset; (b) on CIFAR-10 dataset; (c) on CIFAR-100 dataset.

similar performance with $T_{\max} = 14$s. Moreover, under a stringent delay requirement, i.e., $T_{\max} = 14$s, the proposed algorithm achieves a better performance than the OCEAN algorithm with $T_{\max} = 16$s. That is, the proposed algorithm is able to get a better performance as OCEAN with 12.5% time budget reduction. The underlying reason is that the joint optimization of computation and communication achieves lower energy consumption than solely considering the optimal communication. Even with less time budget, the balance between computation and communication can also lower the overall energy consumption, enabling more clients to participate in the FL training process in a sustainable way. From the comparison results on the CIFAR-100 dataset in Fig. 7(c), under the same per-round time budget, i.e., $T_{\max} = 210$s, the proposed approach improves 3.3% accuracy compared to the OCEAN scheme. In addition, given a more stringent time constraint $T_{\max} = 200$s to the proposed approach, it remains to obtain a slight accuracy improvement in comparison with the OCEAN, EBA, and Myopic schemes.

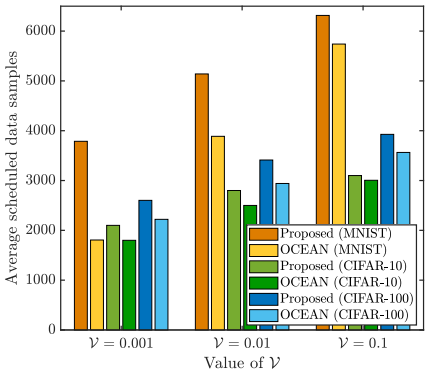
Fig. 8 verifies that the adjustable parameter $\mathcal{V}$ can balance the training performance and energy consumption of clients. Fig. 8(b) shows that as $\mathcal{V}$ increases, clients consume energy in a more aggressive manner, resulting in scheduling more data samples, thus obtaining accuracy improvement. From Fig. 8(a), the experiments on the MNIST dataset indicate that the proposed algorithm achieves 3.46%, 1.94%, and 0.63% test accuracy improvement compared with the OCEAN algorithm under $\mathcal{V} = 0.001$, $\mathcal{V} = 0.01$, and $\mathcal{V} = 0.1$, in each one respectively. Interestingly, the proposed algorithm with $\mathcal{V} = 0.001$ obtains a similar performance with the OCEAN algorithm with $\mathcal{V} = 0.01$. This further reveals that the proposed algorithm has the ability to obtain a similar performance as the OCEAN algorithm under a more rigid energy restriction. Similarly, on the CIFAR-10 dataset, the proposed algorithm boosts 1.05% and 1.23% accuracy in terms of $\mathcal{V} = 0.001$ and $\mathcal{V} = 0.01$, and obtains a slight accuracy improvement when $\mathcal{V} = 0.1$ compared with the OCEAN algorithm. In addition, the simulation results on the CIFAR-100 dataset also show that the proposed approach is able to schedule more data samples and achieve higher learning accuracy than the OCEAN scheme. The performance gain comes from the joint optimization of the client selection, bandwidth allocation, communication, and computation time allocation policies. Note that, if $\mathcal{V}$ is too large, the client scheduling algorithm would pay less attention for clients' energy consumption and try to schedule more clients. This may break the energy limitation for clients. Thus, the value of $\mathcal{V}$ should be judiciously adjusted to optimize the training performance while satisfying the energy constraints.

VI. Conclusion

In this work, we have proposed a novel PMA-FL algorithm, which only shares the feature extractor part of neural networks for global aggregation in the learning process while the predictor part of each client is localized for personalization. This design effectively improves the robustness and performance of the training process, overcoming the data heterogeneity



(a)



(b)

Fig. 8. Performance of the proposed algorithm and benchmarks under different weight parameter $\mathcal{V}$: (a) test accuracy on MNIST and CIFAR-10 datasets; (b) average scheduled data samples per round.

across clients. We have theoretically analyzed the convergence bound of PMA-FL in resource-limited wireless networks, which reveals that maximizing the scheduled data volumes in each round improves the training performance of PMA-FL. Based on this, we have devised a joint client scheduling, communication and computation resource allocation approach to improve the learning performance by achieving the energy consumption balance between communication and computation for each client and the energy consumption-bandwidth balance between clients. Experimental results show that the proposed PMA-FL and client scheduling algorithm outperform the benchmarks in improving learning performance, saving energy for clients, and reducing training latency. The proposed approach is convenient to implement and finds applicability in practical situations, such as healthcare monitoring and traffic prediction of vehicle networks.

APPENDIX

A. Proof of Lemma 1

Due to ℓ_{f}-smooth of $\mathcal{L}_n(\cdot, \mathbf{x}_n^{\mathrm{p}})$ and ℓ_{p}-smooth of $\mathcal{L}(\mathbf{x}^{\mathrm{f}}, \cdot)$, we have:

$$\begin{aligned}&\mathcal{L}_n(\mathbf{x}_{t+1}^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}) - \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}})\\ &\le \left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle + \frac{\ell_{\mathrm{f}}}{2} \left\| \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\|^2, \quad (36)\end{aligned}$$

and

$$\begin{aligned}&\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}) - \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}})\\ &\le \left\langle \nabla_{\mathrm{p}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\rangle\\ &+ \frac{\ell_{\mathrm{p}}}{2} \|\mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}}\|^2. \quad (37)\end{aligned}$$

Summarizing (36) and (37), we have

$$\begin{aligned}&\mathcal{L}_n(\mathbf{x}_{t+1}^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}) - \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}})\\ &\le \left\langle \nabla_{\mathrm{p}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\rangle + \frac{\ell_{\mathrm{p}}}{2} \left\| \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\|^2\\ &+ \left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle + \frac{\ell_{\mathrm{f}}}{2} \left\| \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\|^2. \quad (38)\end{aligned}$$

We now bound the inner product term $\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \rangle$ as follows:

$$\begin{aligned}&\left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle\\ &\overset{(a)}{=} \left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle\\ &+ \left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}) - \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle\\ &\overset{(b)}{\le} \left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle\\ &+ \left\| \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}) - \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}) \right\| \left\| \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\|\\ &\overset{(c)}{\le} \left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle\\ &+ \ell_{\mathrm{fp}} \left\| \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\| \left\| \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\|\\ &\overset{(d)}{\le} \left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle\\ &+ \sqrt{\chi \ell_{\mathrm{p}}} \left\| \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\| \sqrt{\chi \ell_{\mathrm{f}}} \left\| \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\|\\ &\overset{(e)}{\le} \left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle\\ &+ \frac{1}{2} \chi \ell_{\mathrm{p}} \left\| \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\|^2 + \frac{1}{2} \chi \ell_{\mathrm{f}} \left\| \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\|^2, \quad (39)\end{aligned}$$

where (a) is derived by adding and subtracting $\nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}})$, (b) following the Cauthy-Schwarz inequality, (c) is due to L-smooth of loss functions, (d) follows the definition of χ, (e) comes from the triangle-inequality. Substituting (39) into (38), we have

$$\begin{aligned}&\mathcal{L}_n(\mathbf{x}_{t+1}^{\mathrm{f}}, \mathbf{x}_{n,t+1}^{\mathrm{p}}) - \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}})\\ &\le \left\langle \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\rangle + \frac{1+\chi}{2} \ell_{\mathrm{f}} \left\| \mathbf{x}_{t+1}^{\mathrm{f}} - \mathbf{x}_t^{\mathrm{f}} \right\|^2\\ &+ \left\langle \nabla_{\mathrm{p}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}}), \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\rangle\\ &+ \frac{1+\chi}{2} \ell_{\mathrm{p}} \left\| \mathbf{x}_{n,t+1}^{\mathrm{p}} - \mathbf{x}_{n,t}^{\mathrm{p}} \right\|^2. \quad (40)\end{aligned}$$

Substituting the above equation into the global loss function (2), the proof completes.

B. Proof of Lemma 2

Firstly, we introduce an auxiliary variable as:

$$\boldsymbol{o}_t = \nabla_{\mathrm{f}} \mathcal{L}(\mathbf{x}_t^{\mathrm{f}}, \boldsymbol{X}_t^{\mathrm{p}}) - \frac{\sum_{n=1}^N \beta_{n,t} D_n \nabla_{\mathrm{f}} \mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}}, \mathbf{x}_{n,t}^{\mathrm{p}})}{\sum_{n=1}^N \beta_{n,t} D_n}. \quad (41)$$

Based on Lemma 1, we now bound the two terms on the RHS of (15). For the first term in the RHS of (15), we have

$$\begin{aligned}
&\mathbb{E}\Big[\left\langle\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}}),\mathbf{x}_{t+1}^{\mathrm{f}}-\mathbf{x}_t^{\mathrm{f}}\right\rangle+\frac{1+\chi}{2}\ell_{\mathrm{f}}\left\|\mathbf{x}_{t+1}^{\mathrm{f}}-\mathbf{x}_t^{\mathrm{f}}\right\|^2\Big]\\
&=\mathbb{E}\left[\left\langle\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}}),-\eta_{\mathrm{f}}\left(\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})-\boldsymbol{o}_t\right)\right\rangle\right]\\
&+\frac{1}{2}(\chi+1)\ell_{\mathrm{f}}\mathbb{E}\left\|\eta_{\mathrm{f}}(\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})-\boldsymbol{o}_t)\right\|^2\\
&=(\frac{(\chi+1)\ell_{\mathrm{f}}}{2}\eta_{\mathrm{f}}^2-\eta_{\mathrm{f}})\mathbb{E}\left\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})\right\|^2\\
&+\frac{(\chi+1)\ell_{\mathrm{f}}}{2}\eta_{\mathrm{f}}^2\mathbb{E}\|\boldsymbol{o}_t\|^2\\
&+\left(\eta_{\mathrm{f}}-(\chi+1)\ell_{\mathrm{f}}\eta_{\mathrm{f}}^2\right)\mathbb{E}\left[\left\langle\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}}),\boldsymbol{o}_t\right\rangle\right]\\
&\overset{(a)}{\leq}-\frac{1}{2}\eta_{\mathrm{f}}\mathbb{E}\left\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})\right\|^2+\frac{1}{2}\eta_{\mathrm{f}}\mathbb{E}\left\|\boldsymbol{o}_t\right\|^2,
\end{aligned}\tag{42}$$

where (a) is due to $\eta_{\mathrm{f}}\leq\frac{1}{(\chi+1)\ell_{\mathrm{f}}}$ and the triangle-inequality. For the second term in the RHS of (15), we have

$$\begin{aligned}
&\frac{1}{D}\sum_{n=1}^{N}D_n\Big(\mathbb{E}\left[\left\langle\nabla_{\mathrm{p}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}}),\mathbf{x}_{n,t+1}^{\mathrm{p}}-\mathbf{x}_{n,t}^{\mathrm{p}}\right\rangle\right]\\
&+\frac{1+\chi}{2}\ell_{\mathrm{p}}\mathbb{E}\left\|\mathbf{x}_{n,t+1}^{\mathrm{p}}-\mathbf{x}_{n,t}^{\mathrm{p}}\right\|^2\Big)\\
&=\frac{1}{D}\sum_{n=1}^{N}D_n\mathbb{E}\left[\left\langle\nabla_{\mathrm{p}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}}),-\beta_{n,t}\eta_{\mathrm{p}}\nabla_{\mathrm{p}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})\right\rangle\right]\\
&+\frac{1}{D}\sum\nolimits_{n=1}^{N}D_n\frac{1}{2}(1+\chi)\ell_{\mathrm{p}}\mathbb{E}\left\|\beta_{n,t}\eta_{\mathrm{p}}\nabla_{\mathrm{p}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})\right\|^2\\
&\overset{(a)}{=}\sum_{n=1}^{N}\beta_{n,t}\frac{D_n}{D}(\frac{(1+\chi)}{2}\ell_{\mathrm{p}}\eta_{\mathrm{p}}^2-\eta_{\mathrm{p}})\mathbb{E}\left\|\nabla_{\mathrm{p}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})\right\|^2,
\end{aligned}\tag{43}$$

where (a) follows the fact that $\beta_{n,t}\in\{0,1\}$, which induces $\beta_{n,t}^2=\beta_{n,t}$. Substituting (42) and (43) into (15), we have

$$\begin{aligned}
&\mathbb{E}\left[\mathcal{L}\left(\mathbf{x}_{t+1}^{\mathrm{f}},\boldsymbol{X}_{t+1}^{\mathrm{p}}\right)-\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})\right]\\
&\leq\frac{1}{D}\sum_{n=1}^{N}\beta_{n,t}D_n(-\eta_{\mathrm{p}}+\frac{1}{2}(1+\chi)\ell_{\mathrm{p}}\eta_{\mathrm{p}}^2)\mathbb{E}\left\|\nabla_{\mathrm{p}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})\right\|^2\\
&-\frac{1}{2}\eta_{\mathrm{f}}\mathbb{E}\left\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})\right\|^2+\frac{1}{2}\eta_{\mathrm{f}}\mathbb{E}\|\boldsymbol{o}_t\|^2.
\end{aligned}\tag{44}$$

Next, we focus on bounding $\|\boldsymbol{o}_t\|^2$.

$$\begin{aligned}
&\mathbb{E}\|\boldsymbol{o}_t\|^2=\mathbb{E}\Big\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})-\frac{\sum_{n=1}^{N}\beta_{n,t}D_n\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})}{\sum_{n=1}^{N}\beta_{n,t}D_n}\Big\|^2\\
&=\mathbb{E}\Big\|-\frac{(D-\sum_{n=1}^{N}\beta_{n,t}D_n)\sum_{n=1}^{N}\beta_{n,t}D_n\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})}{D\sum_{n=1}^{N}\beta_{n,t}D_n}\\
&\qquad+\frac{\sum_{n=1}^{N}(1-\beta_{n,t})D_n\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})}{D}\Big\|^2\\
&\leq\mathbb{E}\Big(\frac{(D-\sum_{n=1}^{N}\beta_{n,t}D_n)\sum_{n\in\mathcal{N}_{1,t}}D_n\left\|\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})\right\|}{D\sum_{n=1}^{N}\beta_{n,t}D_n}\\
&\qquad+\frac{\sum_{n\in\mathcal{N}_{2,t}}D_n\left\|\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})\right\|}{D}\Big)^2,
\end{aligned}\tag{45}$$

where $\mathcal{N}_{1,t}=\{\beta_{n,t}=1\,|n\in\mathcal{N}\}$ is the set of clients who participating the training process in round t, and $\mathcal{N}_{2,t}=\{\beta_{n,t}=0\,|n\in\mathcal{N}\}$ represents the set of clients that have not been selected in round t. The inequality in (45) follows the triangle-inequality. By using Assumption 2, we have

$$\begin{aligned}
&\sum\nolimits_{n\in\mathcal{N}_{1,t}}D_n\left\|\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})\right\|\\
&\leq\sum\nolimits_{n=1}^{N}\beta_{n,t}D_n\sqrt{\delta^2+\rho^2\left\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}})\right\|^2},
\end{aligned}\tag{46}$$

and

$$\begin{aligned}
&\sum\nolimits_{n\in\mathcal{N}_{2,t}}D_n\left\|\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}_t^{\mathrm{f}},\mathbf{x}_{n,t}^{\mathrm{p}})\right\|\\
&\leq(D-\sum\nolimits_{n=1}^{N}\beta_{n,t}D_n)\sqrt{\delta^2+\rho^2\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}})\|^2}.
\end{aligned}\tag{47}$$

Substituting (46) and (47) into (45), we have

$$\mathbb{E}\left\|\boldsymbol{o}_t\right\|^2\leq\frac{4(D-\sum\limits_{n=1}^{N}\beta_{n,t}D_n)^2(\delta^2+\rho^2\left\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}})\right\|^2)}{D^2}.\tag{48}$$

Now, by substituting (48) into (44) and let $\eta_{\mathrm{p}}\leq\frac{2}{(\chi+1)\ell_{\mathrm{p}}}$, the proof completes.

C. Proof of Theorem 1

To prove Theorem 1, we first prove that $\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}})$ is ℓ_{f}- Lipschitz continuous with $\mathbf{x}^{\mathrm{f}}$.

$$\begin{aligned}
&\left\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}})-\nabla_{\mathrm{f}}\mathcal{L}(\hat{\mathbf{x}}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}})\right\|\\
&=\frac{1}{D}\Big\|\sum\nolimits_{n=1}^{N}D_n\Big(\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}^{\mathrm{f}},\mathbf{x}_n^{\mathrm{p}})-\nabla_{\mathrm{f}}\mathcal{L}_n(\hat{\mathbf{x}}^{\mathrm{f}},\mathbf{x}_n^{\mathrm{p}})\Big)\Big\|\\
&\overset{(a)}{\leq}\frac{1}{D}\sum\nolimits_{n=1}^{N}D_n\left\|\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}^{\mathrm{f}},\mathbf{x}_n^{\mathrm{p}})-\nabla_{\mathrm{f}}\mathcal{L}_n(\hat{\mathbf{x}}^{\mathrm{f}},\mathbf{x}_n^{\mathrm{p}})\right\|\\
&\overset{(b)}{\leq}\frac{1}{D}\sum\nolimits_{n=1}^{N}D_n\ell_{\mathrm{f}}\left\|\mathbf{x}^{\mathrm{f}}-\hat{\mathbf{x}}^{\mathrm{f}}\right\|=\ell_{\mathrm{f}}\left\|\mathbf{x}^{\mathrm{f}}-\hat{\mathbf{x}}^{\mathrm{f}}\right\|,
\end{aligned}\tag{49}$$

where (a) follows Cauchy-Schwarz inequality, (b) is due to the ℓ_{f}- Lipschitz continuity of $\nabla_{\mathrm{f}}\mathcal{L}_n(\mathbf{x}^{\mathrm{f}},\mathbf{x}_n^{\mathrm{p}})$ ($\forall n\in\mathcal{N}$). Thus, $\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}})$ is ℓ_{f}- Lipschitz continuous with $\mathbf{x}^{\mathrm{f}}$. According to Lemma 2, we add and subtract $\mathcal{L}(\mathbf{x}^{\mathrm{f},*},\boldsymbol{X}^{\mathrm{p},*})$ in the left-hand side of (16), and then rearrange it gives

$$\begin{aligned}
&\mathbb{E}\left[\mathcal{L}(\mathbf{x}_{t+1}^{\mathrm{f}},\boldsymbol{X}_{t+1}^{\mathrm{p}})-\mathcal{L}(\mathbf{x}^{\mathrm{f},*},\boldsymbol{X}^{\mathrm{p},*})\right]\\
&\leq\mathbb{E}\left[\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})-\mathcal{L}(\mathbf{x}^{\mathrm{f},*},\boldsymbol{X}^{\mathrm{p},*})\right]\\
&+\frac{\eta_{\mathrm{f}}}{2}\Big(\frac{4\rho^2}{D^2}(D-\sum_{n=1}^{N}\beta_{n,t}D_n)^2-1\Big)\mathbb{E}\left\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})\right\|^2\\
&+\frac{2\eta_{\mathrm{f}}}{D^2}\Big(D-\sum_{n=1}^{N}\beta_{n,t}D_n\Big)^2\delta^2.
\end{aligned}\tag{50}$$

By using the ℓ_{f}- Lipschitz continuity of $\nabla_{\mathrm{f}}F(\mathbf{x}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}})$ is ℓ_{f}, we have [39]

$$\left\|\nabla_{\mathrm{f}}\mathcal{L}(\mathbf{x}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}})\right\|^2\leq2\ell_{\mathrm{f}}\Big(\mathcal{L}\left(\mathbf{x}^{\mathrm{f}},\boldsymbol{X}^{\mathrm{p}}\right)-\mathcal{L}(\mathbf{x}^{\mathrm{f},*},\boldsymbol{X}^{\mathrm{p},*})\Big).\tag{51}$$

Substituting (51) into (50), we have

$$\begin{aligned}
&\mathbb{E}\left[\mathcal{L}(\mathbf{x}_{t+1}^{\mathrm{f}},\boldsymbol{X}_{t+1}^{\mathrm{p}})-\mathcal{L}(\mathbf{x}^{\mathrm{f},*},\boldsymbol{X}^{\mathrm{p},*})\right]\\
&\leq A_t\mathbb{E}\left[\mathcal{L}(\mathbf{x}_t^{\mathrm{f}},\boldsymbol{X}_t^{\mathrm{p}})-\mathcal{L}(\mathbf{x}^{\mathrm{f},*},\boldsymbol{X}^{\mathrm{p},*})\right]\\
&\qquad+\frac{2\eta_{\mathrm{f}}}{D^2}\Big(D-\sum\nolimits_{n=1}^{N}\beta_{n,t}D_n\Big)^2\delta^2,
\end{aligned}\tag{52}$$

where $A_t = \left(1+\eta_{\mathrm{f}}\ell_{\mathrm{f}}\left(\frac{4}{D^2}(D-\sum_{n=1}^{N}\beta_{n,t}D_n)^2\rho^2-1\right)\right)$. By telescoping the above inequality, we complete the proof.

D. Proof of Lemma 3

In problem $\mathcal{P}_3$, the communication time variable $T_{n,t}^{\mathrm{L}}$ ($\forall n \in \mathcal{N}$) is continuous real number variable. The first-order derivatives of the objective function is

$$\frac{\partial \mathcal{E}_{n,t}}{\partial T_{n,t}^{\mathrm{L}}} = -2\frac{\kappa\tau^3 D_n^3 C_n^3}{(T_{n,t}^{\mathrm{L}})^3} + \frac{\theta_{n,t}BN_0}{h_{n,t}} \times \left(\left(\frac{Q\ln 2}{\theta_{n,t}B(T_{\max}-T_{n,t}^{\mathrm{L}})}-1\right)2^{\frac{Q}{\theta_{n,t}B(T_{\max}-T_{n,t}^{\mathrm{L}})}}+1\right). \tag{53}$$

Furthermore, if $n \neq m$, we have $\frac{\partial^2 \mathcal{E}_{n,t}}{\partial T_{n,t}^{\mathrm{L}} \partial T_{m,t}^{\mathrm{C}}} = 0$. When $n = m$, the second-order derivatives is given by

$$\frac{\partial^2 \mathcal{E}_{n,t}}{\partial (T_{n,t}^{\mathrm{L}})^2} = 6\frac{\kappa\tau^3 D_n^3 C_n^3}{(T_{n,t}^{\mathrm{L}})^4} + \frac{Q^2 N_0 (\ln 2)^2 2^{\frac{Q}{\theta_{n,t}B(T_{\max}-T_{n,t}^{\mathrm{L}})}}}{\theta_{n,t}Bh_{n,t}(T_{\max}-T_{n,t}^{\mathrm{L}})^3}. \tag{54}$$

It is straightforward to see that $\frac{\partial^2 \mathcal{E}_{n,t}}{\partial (T_{n,t}^{\mathrm{L}})^2} \geq 0$. Thus, the Hessian matrixes of $\mathcal{E}_{n,t}$ with respect to $T_{n,t}^{\mathrm{L}}$ is a diagonal matrix and the elements on the diagonal are all non-negative. Consequently, the Hessian matrix of $\mathcal{E}_{n,t}$ is semi-positive matrix and $\mathcal{E}_{n,t}$ is a convex function with respect to $T_{n,t}^{\mathrm{L}}$. Moreover, the constraints in problem $\mathcal{P}_3$ are all linear with respect to $T_{n,t}^{\mathrm{L}}$. This implements that problem $\mathcal{P}_3$ is convex. By using Karush-Kuhn-Tucker condition, we have the optimal solution satisfy $\frac{\partial \mathcal{E}_{n,t}}{\partial T_{n,t}^{\mathrm{L}}} = 0$. In fact, $\frac{\partial \mathcal{E}_{n,t}}{\partial T_{n,t}^{\mathrm{L}}} = 0$ is equivalent to $\frac{\partial \mathcal{E}_{n,t}^{\mathrm{L}}}{\partial T_{n,t}^{\mathrm{L}}} = \frac{\partial \mathcal{E}_{n,t}^{\mathrm{U}}}{\partial T_{n,t}^{\mathrm{U}}}$.

REFERENCES

[1] Z. Chen, W. Yi, A. Nallanathan, and G. Y. Li, "Is partial model aggregation energy-efficient for federated learning enabled wireless networks?" in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2023, pp. 166–171.

[2] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for internet of things: Recent advances, taxonomy, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1759–1799, 2021.

[3] Z. Yang, M. Chen, K.-K. Wong, H. V. Poor, and S. Cui, "Federated learning for 6G: Applications, challenges, and opportunities," *Engineering*, vol. 8, pp. 33–41, 2022.

[4] Q. Zeng, Y. Du, K. Huang, and K. K. Leung, "Energy-efficient resource management for federated edge learning with CPU-GPU heterogeneous computing," *IEEE Trans. Wireless Commun.*, vol. 20, no. 12, pp. 7947–7962, 2021.

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

[6] 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.*, vol. 20, no. 3, pp. 1935–1949, 2021.

[7] W. Shi, S. Zhou, Z. Niu, M. Jiang, and L. Geng, "Joint device scheduling and resource allocation for latency constrained wireless federated learning," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 453–467, 2021.

[8] J. Xu and H. Wang, "Client selection and bandwidth allocation in wireless federated learning networks: A long-term perspective," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1188–1200, 2021.

[9] Y. Sun, S. Zhou, Z. Niu, and D. Gündüz, "Dynamic scheduling for over-the-air federated edge learning with energy constraints," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 227–242, 2022.

[10] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, 2021.

[11] S. Zhou and G. Y. Li, "Federated learning via inexact admm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 9699–9708, 2023.

[12] A. Z. Tan, H. Yu, L. Cui, and Q. Yang, "Towards personalized federated learning," *IEEE Trans. Neural Netw. and Learn. Sys.*, vol. 34, no. 12, pp. 9587–9603, 2023.

[13] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Proc. Neural Inf. Process. Sys. (NeurIPS)*, 2017.

[14] M. Mortaheb, C. Vahapoglu, and S. Ulukus, "Personalized federated multi-task learning over wireless fading channels," *Algorithms*, vol. 15, no. 11, 2022. [Online]. Available: https://www.mdpi.com/1999-4893/15/11/421

[15] S. Yue, J. Ren, J. Xin, D. Zhang, Y. Zhang, and W. Zhuang, "Efficient federated meta-learning over multi-access wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 5, pp. 1556–1570, 2022.

[16] V.-D. Nguyen, S. K. Sharma, T. X. Vu, S. Chatzinotas, and B. Ottersten, "Efficient federated learning algorithm for resource allocation in wireless iot networks," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3394–3409, 2021.

[17] Y. Deng, M. M. Kamani, and M. Mahdavi, "Adaptive personalized federated learning," *arXiv preprint arXiv:2003.13461*, 2020.

[18] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proc. AISTATS*, 20–22, Apr. 2017.

[19] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "Exploiting shared representations for personalized federated learning," in *Proc. Int. Conf. Mach. Learning (ICML)*, 18–24 Jul 2021.

[20] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary, "Federated learning with personalization layers," *arXiv preprint arXiv:1912.00818*, 2019.

[21] K. Pillutla, K. Malik, A.-R. Mohamed, M. Rabbat, M. Sanjabi, and L. Xiao, "Federated learning with partial model personalization," in *Proc. International Conference on Machine Learning*, vol. 162, 2022, pp. 17 716–17 758.

[22] Y. Jiang, S. Wang, V. Valls, B. J. Ko, W.-H. Lee, K. K. Leung, and L. Tassiulas, "Model pruning enables efficient federated learning on edge devices," *IEEE Trans. Neural Networks and Learning Systems*, vol. 34, no. 12, pp. 10 374–10 386, 2023.

[23] E. Diao, J. Ding, and V. Tarokh, "Heterofl: Computation and communication efficient federated learning for heterogeneous clients," *arXiv preprint arXiv:2010.01264*, 2020.

[24] Z. Chen, W. Yi, H. Shin, and A. Nallanathan, "Adaptive model pruning for communication and computation efficient wireless federated learning," *IEEE Trans. Wireless Commun.*, pp. 1–1, 2023.

[25] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, 2013.

[26] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[27] A. P. Miettinen and J. K. Nurminen, "Energy efficiency of mobile clients in cloud computing." *HotCloud*, vol. 10, pp. 1–7, 2010.

[28] S. Kaxiras and M. Martonosi, *Computer architecture techniques for power-efficiency*. Springer Nature, 2022.

[29] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[30] C. You, K. Huang, H. Chae, and B.-H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1397–1411, 2017.

[31] E. Abbasnejad, J. Shi, and A. van den Hengel, "Deep Lipschitz networks and dudley GANs," 2018. [Online]. Available: https://openreview.net/forum?id=rkw-jlb0W

[32] Z. Chen, W. Yi, Y. Liu, and A. Nallanathan, "Knowledge-aided federated learning for energy-limited wireless networks," *IEEE Trans. Commun.*, vol. 71, no. 6, pp. 3368–3386, 2023.

[33] Z. Chen, W. Yi, and A. Nallanathan, "Exploring representativity in device scheduling for wireless federated learning," *IEEE Trans. Wireless Commun.*, vol. 23, no. 1, pp. 720–735, 2024.

[34] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.

[35] I. Waldspurger, A. d'Aspremont, and S. Mallat, "Phase recovery, maxcut and complex semidefinite programming," *Mathematical Programming*, vol. 149, no. 1, pp. 47–81, 2015.

[36] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *Journal of optimization theory and applications*, vol. 109, pp. 475–494, 2001.
[37] Z. Chen, W. Yi, A. S. Alam, and A. Nallanathan, "Dynamic task software caching-assisted computation offloading for multi-access edge computing," *IEEE Trans. Commun.*, vol. 70, no. 10, pp. 6950–6965, 2022.
[38] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[39] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-iid data," in *Proc. Int. Conf. Learning Repr. (ICLR)*, 2020.