Efficient Federated Learning with Adaptive Model Pruning and Quantization over Wireless Networks

Efficient Federated Learning with Adaptive Model Pruning and Quantization over Wireless Networks

Xiaodong Li, Yulong Gao, *Senior Member, IEEE*, Carla Fabiana Chiasserini, *Fellow, IEEE*, and Yansha Deng, *Senior Member, IEEE*

***Abstract*—Federated learning (FL) enables multiple devices to collaboratively train a global model without sharing local data. However, due to limited local computing capability and communication bandwidth, FL suffers from high learning latency, especially when the model size is large. To address these issues, we propose APQ-FL, an adaptive model pruning and quantization method for wireless FL, to reduce the neural network size and improve communication efficiency. Moreover, device selection and wireless resource allocation are also integrated. We first present a convergence analysis of FL with model pruning and quantized transmission, and then jointly optimize the pruning ratio, quantization bit width, device selection, and wireless bandwidth allocation to minimize the convergence upper bound under latency and bandwidth constraints. We prove that the optimized quantization bit width can be obtained via binary search, and derive the closed-form solutions for the optimal pruning ratio and bandwidth allocation. Subsequently, we propose an efficient device selection strategy and further introduce its fairness-aware extension, APQ-FL-Fair. Experiments show that APQ-FL and APQ-FL-Fair improve test accuracy by 4.96%–15.33% while reducing 23.14%–74.45% communication overhead compared to other methods, and exhibit stable and superior performance even under stringent latency constraints.**

***Index Terms*—Wireless federated learning, model pruning, quantization schemes, communication and computation latency.**

I. Introduction

With the advancement of the Internet of Things (IoT) and mobile communication technologies, an increasing number of wireless devices are being deployed in various wireless systems and continuously generating massive amounts of data [1]. Traditional machine learning (ML) follows a centralized training paradigm, where devices are required to upload their local data to a cloud server. However, transmitting all data to the cloud is unrealistic due to high communication cost and serious privacy concerns. To address these issues, federated learning (FL) [2] has emerged as a distributed paradigm that enables multiple devices to collaboratively train ML models in a privacy-preserving manner. Instead of sharing local data, FL exchanges model parameters or updates between devices and the server for model updating. Compared with the traditional centralized paradigm, FL offers improved privacy protection and reduced communication load, and has shown promising results, e.g., in edge networks [3]–[5], internet of vehicles [6], intelligent transportation [7].

However, there are still several challenges in deploying FL within wireless networks. Firstly, the wireless resources are limited, which can lead to communication bottlenecks and considerable latency between devices and the server. Secondly, the computational capabilities of devices are also constrained, which may result in high computation latency, especially when the model size is large. Moreover, due to wireless fading, wireless channels are often unreliable, which further affects the transmission of model parameters. Therefore, it is essential to achieve efficient FL under resource-constrained scenarios.

A. Related Works

To tackle the problem of limited wireless resources, many methods have been proposed for device scheduling and wireless resource allocation. For instance, [8] jointly optimized device scheduling and resource allocation under latency and bandwidth constraints to maximize model accuracy. The study in [9] further considered energy consumption and jointly optimized device selection, transmit power, and resource allocation to minimize training loss under latency and energy constraints. In [10], a communication time minimization problem is formulated and solved through an optimized probabilistic device scheduling policy. Likewise, [1] optimized device scheduling and bandwidth allocation to minimize a weighted sum of training latency and energy consumption. Although these methods can effectively alleviate communication pressure for wireless FL, transmitting the entire model still poses challenges for devices with poor communication capabilities. Moreover, they cannot mitigate the issue of high computational latency caused by the constrained computational resources.

To address the above problems, federated dropout and model pruning have been introduced into FL to adjust model size, thereby reducing computation and communication overhead and minimizing the learning latency. In [11], a federated dropout scheme was proposed based on the classical dropout mechanism for random model pruning, where the server generates subnetworks for devices according to channel conditions. [12] developed an adaptive federated dropout method to reduce communication and computation costs by allowing each device to train a subset of the global model. The authors of [13] en-

This work was supported in part by the National Natural Science Foundation of China under Grant No. 62171163.

Xiaodong Li and Yulong Gao are with the School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China (e-mail: lixd@stu.hit.edu.cn; ylgao@hit.edu.cn).

Carla Fabiana Chiasserini is with the Politecnico di Torino and the CNR-IEIIT, Torino, Italy, with the CNIT, Parma, Italy, and with the Chalmers University of Technology, Sweden (e-mail: carla.chiasserini@polito.it).

Yansha Deng is with the Department of Engineering, King's College London, London WC2R 2LS, U.K. (e-mail: yansha.deng@kcl.ac.uk).

abled each device to generate a submodel through controllable weight dropout to reduce computation and communication overhead. In [14], the authors proposed FedLoDrop, which jointly optimizes the dropout rate and resource allocation under latency and per-device energy constraints to minimize an upper bound of the generalization error. In addition to these methods, model pruning typically reduces the model size by removing less important weights. The study in [15] proposed PruneFL, which applies model pruning to minimize training time while maintaining accuracy comparable to the original model. However, it adopts a uniform pruning ratio for each device and does not optimize it under limited resources. To overcome this limitation, [16] and [17] optimized the pruning ratio and generated pruned sub-models for each device based on their computation and communication capabilities. [18] further jointly optimized the pruning ratio, device selection, and wireless resource allocation under latency and bandwidth constraints to prune the global model based on the importance of model weights. Building on this, [19] took into account the energy and optimized device selection, pruning ratio, and bandwidth allocation by minimizing a weighted sum of the convergence bound and energy cost. In [20] and [21], model pruning was further extended to hierarchical FL, where the pruning ratio and resource allocation were jointly optimized by minimizing the convergence upper bound.

However, all the above methods overlooked the optimization of quantization schemes and assumed a fixed quantization bit width, which limits communication efficiency under resource constraints and fails to adapt to varying wireless conditions. Note that the quantization here is different from that in ML model compression, which focuses on reducing model storage and simplifying inference computation after the training is completed. In contrast, the former aims to improve communication efficiency and has to be carried out in every round.

Currently, several works have explored the optimization of quantization schemes in FL. [22] provided a detailed discussion on quantization schemes in FL and proposed two quantization strategies based on nearest rounding and stochastic rounding. Building on this, [23] optimized the quantization bit width and wireless resource allocation for each device under transmission latency and outage constraints. Similarly, [24] adopted the stochastic quantization in [22] and jointly optimized the uplink quantization bit width and wireless resource allocation by minimizing the convergence time of FL under energy and quantization error constraints. In [25], the quantization bit width and device selection are jointly optimized by minimizing the training loss under latency constraints. However, these methods still require training and transmitting the whole model between devices and the server, which cannot reduce computation latency and may still lead to communication bottlenecks under limited resources.

Thus, jointly considering model pruning and quantization schemes in resource-constrained FL is necessary for improving both computation and communication efficiency. A few recent studies have applied them simultaneously in FL. In [26] and [27], model pruning and quantization were used to reduce computation and communication overhead and improve the efficiency of FL. Similarly, [28] introduced them in federated split learning to achieve overhead reduction.

However, none of these works considers the joint optimization of model pruning and quantization, instead focusing primarily on their separate and straightforward application. Moreover, they do not incorporate device selection or wireless resource allocation strategies to further enhance FL efficiency. To fill this gap, we focus on jointly optimizing model pruning, quantization schemes, device selection, and wireless resource allocation under limited resources, aiming to improve FL efficiency while preserving model performance.

B. Challenges and Contributions

Based on the above analysis, several critical issues need to be addressed. First, the pruning ratio and quantization bit width are inherently interdependent. Intuitively, a higher pruning ratio will result in fewer model weights to be transmitted, which in turn allows for a larger quantization bit width to reduce quantization error, and vice versa. However, such empirical observations are insufficient for determining the optimal pruning ratio and quantization bit width. It remains theoretically unclear how they influence each other and jointly affect model performance. Thus, theoretical analysis is needed to analyze their impact on convergence and guide the design of joint optimization strategies. Second, device selection and wireless resource allocation should also be considered. Motivated by the above, we first analyze the convergence error introduced by model pruning and quantization schemes. Then, we propose APQ-FL, an adaptive joint optimization method for pruning ratio, quantization bit width, device selection, and bandwidth allocation under latency and bandwidth constraints. Moreover, we introduce a fairness-aware device selection extension of APQ-FL, termed APQ-FL-Fair, to promote participation fairness while accounting for system efficiency. Our main contributions are summarized as follows.

- To cope with the limited resources and dynamic wireless conditions, we propose APQ-FL with adaptive model pruning and quantization. To the best of our knowledge, this is the first work to jointly optimize pruning and quantization in resource-constrained FL.
- We analyze the convergence upper bound of FL with adaptive model pruning and quantization schemes, and theoretically demonstrate the interdependent relationship between the pruning ratio and quantization bit width under limited resources. Then, the pruning ratio, quantization bit width, device selection, and bandwidth allocation are jointly optimized to minimize the upper bound under latency and bandwidth constraints.
- By decoupling the optimization problem, we first prove that the optimal quantization bit width can be obtained through binary search and derive the closed-form solutions for the optimal pruning ratio and bandwidth allocation under given device selection. Then, we propose an efficient device selection strategy by removing devices with high pruning ratios and low quantization bit widths, and introduce its fairness-aware extension, APQ-FL-Fair.
- Extensive experiments show that APQ-FL and APQ-FL-Fair consistently outperform state-of-the-art methods that

TABLE I
MAIN NOTATIONS

Notation	Definition
K	Total number of devices
T	Number of communication rounds
$\boldsymbol{w}_t$	Global model in round t
$\hat{\boldsymbol{w}}_{t,r}^k$	Pruned local model of device k after r iterations
$F_k(\hat{\boldsymbol{w}}_{t,r}^k)$	Local objective function of device k
$\Delta \boldsymbol{w}_t^k$	Local update of device k in round t
ρ_t^k	Pruning ratio for device k in round t
q_t^k	Quantization bit width of device k in round t
l_t^k	Wireless bandwidth allocation for device k in round t
a_t^k	Selection indicator for device k in round t
$\boldsymbol{m}_t^k$	Pruning mask for device k in round t
$\mathcal{K}_t$	Set of selected devices in round t
$\mathcal{K}_t^j$	Subset of $\mathcal{K}_t$ that retains the j-th weight after pruning
J	Number of model weights
B	Total wireless bandwidth
$T^{\max}$	Latency constraint in each round

do not jointly consider model pruning and quantization, improving test accuracy by 4.96%–15.33% while reducing communication overhead by 23.14%–74.45%. This further demonstrates the necessity for joint optimization.

The rest of this paper is organized as follows. Section II introduces the system model. Section III presents the convergence analysis and problem formulation. Section IV proposes our APQ-FL. Section V shows the experimental results, and Section VI presents the conclusion.

II. SYSTEM MODEL

Consider a synchronous cross-device FL system consisting of a server and K devices. Each device $k \in \{1, \ldots, K\}$ owns a local dataset $\mathcal{D}_k = \{(\boldsymbol{x}_i^k, y_i^k)\}_{i=1}^{|\mathcal{D}_k|}$, where $\boldsymbol{x}_i^k$ denotes the sample features and y_i^k represents the corresponding label. FL iteratively trains a global model through multiple communication rounds between the server and devices. Following the standard FL process [2], in each round, the server first sends the current global model to the selected devices. Subsequently, each device performs local training and uploads local updates. Then, the server waits for and aggregates the local updates to update the global model. This process continues until convergence. The main notations are listed in Table I.

A. Model Pruning

Growing neural network scale can induce high computation and communication latency in FL. To address these issues, model pruning can be employed to prune unimportant neurons or weights, thereby reducing model size while causing only a small performance loss. The model accuracy degrades dramatically only when the pruning ratio is very high [18].

At the beginning of each round t, the server applies different pruning strategies to prune the global model for each selected device, thereby generating pruned models tailored to their own computational and communication conditions. According to [29], the importance of a weight can be quantified by the objective function error induced by removing it. Let w_t^j denote the j-th model weight of $\boldsymbol{w}_t$. Then, the importance of w_t^j is

$$I_t^j = [F(\boldsymbol{w}_t) - F(\boldsymbol{w}_t | w_t^j = 0)]^2, \tag{1}$$

where $F(\boldsymbol{w}_t)$ is the global objective function. The larger the error, the more important the weight. However, calculating I_t^j for all model weights is time-consuming. To reduce the computational complexity, we adopt the first-order Taylor approximation of (1) used in [18], [29]:

$$\hat{I}_t^j = (g_t^j w_t^j)^2, \tag{2}$$

where g_t^j is the corresponding global gradient of the weight j, which can be easily obtained after receiving the local updates in the previous round. Note that this metric can be replaced by other importance criteria without altering the subsequent optimization design.

Let ρ_t^k denote the pruning ratio for device k in round t, i.e., the ratio of the number of pruned weights to the model size J. Then, for each selected device k, the server generates a pruning mask matrix $\boldsymbol{m}_t^k$ based on ρ_t^k to prune the model $\boldsymbol{w}_t$ to $\hat{\boldsymbol{w}}_t^k$ by setting the less important weights to zero [20], [29], [30]. Subsequently, $\hat{\boldsymbol{w}}_t^k$ and $\boldsymbol{m}_t^k$ are sent to the corresponding devices for their local training.

B. Model Quantization

We focus on the optimization of uplink quantization bit width for devices, and this is for two reasons. First, the uplink transmission from the device side dominates the communication bottleneck in FL [17]. Second, unified quantization schemes for each device are not well-suited to adapting to their sparse local updates induced by model pruning. According to [23], it is more efficient to transmit and aggregate model updates $\Delta \boldsymbol{w}_t^k$ (i.e., the difference between the local model before and after local training) rather than the model itself. Thus, each device uploads a quantized version of its local updates to the server. We adopt the stochastic quantization method in [22], which is designed for practical FL systems and has been widely used and proven to be effective [24]. Following that, for each model weight indexed by j, its local update $\Delta w_t^{k,j}$ is bounded such that $|\Delta w_t^{k,j}| \in [\Delta \underline{w}_{t,k}, \Delta \overline{w}_{t,k}]$, where $\Delta \underline{w}_{t,k}$ and $\Delta \overline{w}_{t,k}$ are set empirically or heuristically. Let q_t^k denote the quantization bit width of device k in round t. Then, the device k can divide the interval $[\Delta \underline{w}_{t,k}, \Delta \overline{w}_{t,k}]$ into ζ small intervals $[c_0, c_1], [c_1, c_2], \ldots, [c_{\zeta-1}, c_\zeta]$, where $\zeta = 2^{q_t^k} - 1$ and

$$c_u = \Delta \underline{w}_{t,k} + u \frac{\Delta \overline{w}_{t,k} - \Delta \underline{w}_{t,k}}{2^{q_t^k} - 1}, u = 0, \ldots, 2^{q_t^k} - 1. \tag{3}$$

If the local update $\Delta w_t^{k,j}$ of weight j falls in $[c_u, c_{u+1}]$, it can be quantized as

$$\mathcal{Q}(\Delta w_t^{k,j}) = \begin{cases} \text{sign}(\Delta w_t^{k,j}) \cdot c_u, & \text{w.p. } \dfrac{c_{u+1} - |\Delta w_t^{k,j}|}{c_{u+1} - c_u}, \\ \text{sign}(\Delta w_t^{k,j}) \cdot c_{u+1}, & \text{w.p. } \dfrac{|\Delta w_t^{k,j}| - c_u}{c_{u+1} - c_u}, \end{cases} \tag{4}$$

where 'w.p.' stands for 'with probability'. Meanwhile, for each device k in round t, the bit size of quantized local updates for the unpruned model weights is given by

$$s_t^k = J(1 - \rho_t^k)(q_t^k + 1) + z, \tag{5}$$

where J is the total number of model weights, the $+1$ accounts for the sign bit, and z specifies the values of $\Delta\underline{w}_{t,k}$ and $\Delta\overline{w}_{t,k}$.

C. Learning Process

FL aims to learn the global model $\boldsymbol{w}$ through iterative communication rounds to minimize the following objective:

$$\min_{\boldsymbol{w}} F(\boldsymbol{w}) = \frac{1}{K}\sum_{k=1}^{K} F_k(\boldsymbol{w}). \tag{6}$$

Let $a_t^k \in \{0,1\}$ denote the selection indicator for device k, where $a_t^k = 1$ indicates that device k is selected, and $a_t^k = 0$ otherwise. In each round t, the server first selects a subset of devices $\mathcal{K}_t$, and then generates the pruning mask $\boldsymbol{m}_t^k$ according to (2) and prunes the global model $\boldsymbol{w}_t$ to $\hat{\boldsymbol{w}}_t^k$ for each device $k \in \mathcal{K}_t$. Subsequently, the server sends each pruned model to the corresponding device. Each device k initializes local model $\hat{\boldsymbol{w}}_{t,0}^k$ with $\hat{\boldsymbol{w}}_t^k$ and performs τ steps of mini-batch stochastic gradient descent (SGD) to update $\hat{\boldsymbol{w}}_{t,0}^k$ to $\hat{\boldsymbol{w}}_{t,\tau}^k$, which can be expressed as

$$\hat{\boldsymbol{w}}_{t,r+1}^k = \hat{\boldsymbol{w}}_{t,r}^k - \eta \nabla F_k(\hat{\boldsymbol{w}}_{t,r}^k, \xi_{t,r}^k) \odot \boldsymbol{m}_t^k, \tag{7}$$

where $\hat{\boldsymbol{w}}_{t,r}^k$ denotes the local model in r-th iteration, η is the learning rate, $\nabla F_k(\hat{\boldsymbol{w}}_{t,r}^k, \xi_{t,r}^k)$ denotes the local gradient on a random mini-batch $\xi_{t,r}^k$ of local data, and $\odot$ represents element-wise multiplication. In $\boldsymbol{m}_t^k$, $m_t^{k,j} = 0$ indicates that the j-th weight is pruned, while $m_t^{k,j} = 1$ means it is retained. Then, the local update $\Delta\boldsymbol{w}_t^k$ of device k can be written as

$$\Delta\boldsymbol{w}_t^k = -\eta \sum_{r=0}^{\tau-1} \nabla F_k(\hat{\boldsymbol{w}}_{t,r}^k, \xi_{t,r}^k) \odot \boldsymbol{m}_t^k. \tag{8}$$

After local training, each device k quantizes $\Delta\boldsymbol{w}_t^k$ using (4) and uploads it to the server. Note that due to model pruning, the local updates of different devices involve different sets of model weights. Let $\mathcal{K}_t^j$ denote the subset of $\mathcal{K}_t$ that retain the j-th weight in round t, and $|\mathcal{K}_t^j|$ is the number of devices. Then, the j-th global model weight is updated by aggregating the corresponding local updates from devices in $\mathcal{K}_t^j$, i.e.,

$$w_{t+1}^j = w_t^j + \frac{1}{|\mathcal{K}_t^j|}\sum_{k\in\mathcal{K}_t^j} \mathcal{Q}(\Delta w_t^{k,j}), \tag{9}$$

where $\Delta w_t^{k,j}$ denotes the local update of j-th model weight of device k in round t. If certain weights are pruned for all selected devices, they will not be updated in this round and retain their values from the previous round. For simplicity, we adopt the average scheme in (9). The subsequent analysis and algorithms can be easily extended to scenarios where aggregation is based on dataset size.

D. Computation and Communication Model

As mentioned earlier, our focus lies on mitigating the communication bottleneck caused by the uplink transmission from the device side. Similar to [17], [23], we assume that the server can transmit the global model with an approximated small constant delay, enabled by its ability to vary the data rate. Thus, the downlink latency is considered negligible compared to that of the uplink [1], [31]. Moreover, since the server is typically more computationally powerful than devices, its computation latency is ignored [24]. Let f_k denote the CPU frequency on device k, b_k represent the local mini-batch size, and c_k be the number of CPU cycles required to process one sample. Then, following [18], [29], the latency of local iterations on device k in round t is modeled as

$$T_{t,k}^{\text{cmp}} = \frac{\tau c_k b_k (1-\rho_t^k)}{f_k}. \tag{10}$$

The computation latency of local quantization is ignored, as its computational complexity is much lower compared to the forward and backward propagation in local training [20], [24].

We adopt the orthogonal frequency-division multiple access (OFDMA) scheme. According to [20], [32], [33], the achievable transmission rate of device k in round t is denoted as

$$R_t^k = l_t^k B \log_2\!\left(1 + \frac{h_t^k p_t^k}{N_0}\right), \tag{11}$$

where B represents the total bandwidth, $l_t^k \in [0,1]$ denotes the bandwidth fraction allocated to device k, h_t^k is the uplink channel gain, p_t^k is the transmission power, and N_0 is the noise power. We model large-scale fading using distance-dependent path loss and small-scale fading using Rayleigh fading, with the channel conditions varying across rounds. Then, based on (5), the communication latency of device k in round t is written as

$$T_{t,k}^{\text{com}} = \frac{s_t^k}{R_t^k} = \frac{J(1-\rho_t^k)(q_t^k+1)+z}{R_t^k}. \tag{12}$$

Therefore, the total latency of device k in round t is

$$\begin{aligned} T_{t,k}^{\text{total}} &= T_{t,k}^{\text{cmp}} + T_{t,k}^{\text{com}} \\ &= \frac{\tau c_k b_k (1-\rho_t^k)}{f_k} + \frac{J(1-\rho_t^k)(q_t^k+1)+z}{R_t^k}. \end{aligned} \tag{13}$$

As a result, the overall latency of FL in round t is determined by the slowest device, which is written as

$$T_t^{\text{total}} = \max_k \{T_{t,k}^{\text{total}}\}. \tag{14}$$

III. Convergence Analysis and Problem Formulation

In this section, we first analyze the convergence of FL with model pruning and quantized transmission, and then formulate an optimization problem to minimize the convergence bound.

A. Convergence Analysis

Before the analysis, we make the following assumptions.

Assumption 1 (L-smoothness)**.** *The objective function $F(\boldsymbol{w})$ is L-smooth: there exists a constant $L > 0$ such that $\|\nabla F(\boldsymbol{w}_1) - \nabla F(\boldsymbol{w}_2)\| \le L\|\boldsymbol{w}_1 - \boldsymbol{w}_2\|$ for any $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$.*

Assumption 2 (Unbiased Gradient and Bounded Variance)**.** *Let $\xi_{t,r}^k$ be a random mini-batch data sampled from $\mathcal{D}_k$. For each device k, the stochastic gradient is unbiased, i.e., $\mathbb{E}\big[\nabla F_k(\hat{\boldsymbol{w}}_{t,r}^k, \xi_{t,r}^k)\big] = \nabla F_k(\hat{\boldsymbol{w}}_{t,r}^k)$, and the variance is bounded by $\mathbb{E}\big[\|\nabla F_k(\hat{\boldsymbol{w}}_{t,r}^k, \xi_{t,r}^k) - \nabla F_k(\hat{\boldsymbol{w}}_{t,r}^k)\|^2\big] \le \sigma^2, \forall t, r$.*

Assumption 3 (Bounded Gradient and Model)**.** *The second moments of the stochastic gradients and weights are bounded [18], [20], i.e.,* $\mathbb{E}\big[\|\nabla F_k(\boldsymbol{w}_{t,r}^k, \xi_{t,r}^k)\|^2\big] \leq G^2$ *and* $\mathbb{E}\big[\|\boldsymbol{w}_{t,r}^k\|^2\big] \leq D^2$*, where* G *and* D *are positive constants.*

These assumptions are commonly used in the theoretical analysis of FL [16], [34]. Our analysis does not rely on strong convexity, making it applicable to non-convex loss functions. To facilitate the convergence analysis, we further denote $\mathcal{I}_t = \{j \mid |\mathcal{K}_t^j| \geq 1\}$ as the set of model weights that are involved in the global update at round t, and define the per-round minimal occurrence ratio $p_t^* = \min_{j\in\mathcal{I}_t} \frac{|\mathcal{K}_t^j|}{|\mathcal{K}_t|}$, which measures the minimum occurrence ratio of any weight in $\mathcal{I}_t$. Note that p_t^* is defined for weights that are not pruned by all devices, since these weights contribute to the global update of the current round. If a weight is pruned for all selected devices, it will not participate in the global update during model aggregation and retain its value from the previous round. Thus, conditions based on the occurrence ratio are essential for ensuring the convergence of FL. We do not consider the case where all weights are pruned by all selected devices in a round, because such an event is highly uncommon. If it occurs, the round would be restarted. Therefore, p_t^* naturally admits a positive lower bound p^*, leading to the following assumption.

Assumption 4 (Bounded Minimal Occurrence Ratio)**.** *There is a constant* $p^* > 0$ *such that* $p_t^* \geq p^*$ *holds for all rounds* t*.*

Subsequently, the convergence bound of FL with model pruning and quantized transmission is presented as follows.

Theorem 1. *Under the above assumptions, let the learning rate* $\eta \leq \frac{1}{2\tau L}$*, then the convergence bound is*

$$\begin{aligned}
&\frac{1}{TJ}\sum_{t=1}^{T}\sum_{j=1}^{J}\mathbb{E}\left\|\nabla F^j(\boldsymbol{w}_t)\right\|^2 \\
&\leq \frac{2(F_0 - F_*)}{TJ\eta\tau} + 2\eta\tau L\sigma^2 + \frac{\eta^2 G^2 L^2(\tau-1)(2\tau-1)}{3p^*} \\
&\quad + H\underbrace{\sum_{t=1}^{T}\frac{1}{|\mathcal{K}_t|}\sum_{k\in\mathcal{K}_t}\left(\rho_t^k + \beta\frac{\delta_{t,k}^2}{(2^{q_t^k}-1)^2}\right)}_{\text{caused by model pruning and quantization}}
\end{aligned} \tag{15}$$

where $H = \frac{2L^2D^2}{Tp^*}$*,* $\beta = \frac{1}{8\eta\tau LD^2}$*,* $\delta_{t,k} = \Delta\overline{w}_{t,k} - \Delta\underline{w}_{t,k}$*,* $F_0 = F(\boldsymbol{w}_0)$*, and* $F_* = F(\boldsymbol{w}_*)$ *is the theoretical optimum.*

Proof: See Appendix A.

It can be observed that the pruning ratio ρ_t^k and quantization bit width q_t^k jointly influence the convergence bound in Theorem 1. A larger ρ_t^k and a smaller q_t^k will result in a higher convergence error. Therefore, arbitrarily increasing ρ_t^k or decreasing q_t^k to improve computation and communication efficiency may lead to poor convergence performance. Since the derivation does not rely on the IID assumption, Theorem 1 is also applicable to non-IID settings. The impact of non-IID data is primarily reflected in the gradient-related terms in the convergence bound. Greater data heterogeneity makes these terms more conservative, resulting in a more conservative convergence bound. However, it does not alter the mathematical forms of the error term induced by model pruning and quantized transmission in Theorem 1 and the optimization problem formulated in the following subsection.

B. Problem Formulation

Our goal is to obtain the optimal pruning ratio ρ_t^k, quantization bit width q_t^k, device selection a_t^k, and wireless resource allocation l_t^k for each device k and round t to enhance the efficiency of FL while preserving model accuracy. This leads to a problem of minimizing the objective function in (6) under the latency and wireless bandwidth constraints. However, directly analyzing the minimization of (6) theoretically before model training is difficult. Thus, we turn to minimizing the upper bound in Theorem 1 to guide the optimization design, an approach that has been proven effective and widely applied in ML theory [8], [35], [36]. Since the wireless environment varies across training rounds, the channel state information (CSI) of each round t cannot be known in advance. Therefore, we formulate the following optimization problem for each round t to minimize the convergence upper bound.

$$\mathbf{P1}\ \min_{\rho_t^k, q_t^k, l_t^k, a_t^k}\ \frac{1}{|\mathcal{K}_t|}\sum_{k\in\mathcal{K}_t}\left(\rho_t^k + \beta\frac{\delta_{t,k}^2}{(2^{q_t^k}-1)^2}\right) \tag{16}$$

$$\text{s.t.}\quad T_t^{total} \leq T^{max}, \tag{16a}$$

$$\sum_{k\in\mathcal{K}_t} l_t^k \leq 1, \tag{16b}$$

$$0 \leq l_t^k \leq 1, \quad \forall k \in \mathcal{K}_t, \tag{16c}$$

$$0 \leq \rho_t^k \leq \rho_{t,k}^{max}, \quad \forall k \in \mathcal{K}_t, \tag{16d}$$

$$q_t^k \in \mathbb{Z}^+, \quad \forall k \in \mathcal{K}_t, \tag{16e}$$

$$a_t^k \in \{0,1\}, \quad \forall k \in \{1,\ldots,K\}, \tag{16f}$$

where $T^{\max}$ in (16a) denotes the latency constraint in round t, (16b) and (16c) represent the wireless resource constraints, $\rho_{t,k}^{\max}$ is to limit ρ_t^k since the learning performance degrades significantly when it becomes too large, and (16d)-(16f) specify the constraints on the pruning ratio, quantization bit width, and device selection, respectively.

The optimization problem $\mathbf{P1}$ is a mixed-integer nonlinear programming (MINLP) problem, which is non-convex and NP-hard [37], making it difficult to solve directly. Therefore, we tackle $\mathbf{P1}$ by decoupling it into several sub-problems.

IV. Optimized Pruning Ratio, Quantization Bit, Device Selection, and Bandwidth Allocation

In light of $\mathbf{P1}$'s complexity, we decouple the optimization problem $\mathbf{P1}$ into two sub-problems: one for optimizing the pruning ratio, quantization bit width, and bandwidth allocation, and the other for determining the device selection strategy.

A. Optimal Pruning ratio, Quantization Bit Width, and Bandwidth Allocation

Given the device selection set $\mathcal{K}_t$, the optimization problem $\mathbf{P1}$ can be transformed into

$$\mathbf{P2}\ \min_{\rho_t^k, q_t^k, l_t^k}\ \sum_{k\in\mathcal{K}_t}\left(\rho_t^k + \beta\frac{\delta_{t,k}^2}{(2^{q_t^k}-1)^2}\right), \tag{17}$$

subject to (16a)-(16e). According to constraints (16a), (16d), and (16e), the pruning ratio ρ_t^k and quantization bit q_t^k of each device k must satisfy

$$\left(1-\frac{T^{\max}-\frac{z}{R_t^k}}{\frac{\tau c_k b_k}{f_k}+\frac{J(q_t^k+1)}{R_t^k}}\right)^+ \le \rho_t^k \le \rho_{t,k}^{\max}, \tag{18}$$

and

$$1 \le q_t^k \le \frac{R_t^k\left(T^{\max}-\frac{\tau c_k b_k(1-\rho_t^k)}{f_k}\right)-z}{J(1-\rho_t^k)}-1, \tag{19}$$

where $(x)^+=\max\{x,0\}$, and $q_t^k \in \mathbb{Z}^+$. To minimize (17) in $\mathbf{P2}$, the pruning ratio should take the minimum feasible value. Therefore, based on (18), the optimal pruning ratio for device k in round t is

$$\rho_{t,k}^* = \left(1-\frac{T^{\max}-\frac{z}{R_t^k}}{\frac{\tau c_k b_k}{f_k}+\frac{J(q_t^k+1)}{R_t^k}}\right)^+. \tag{20}$$

Then, by substituting (20), the problem $\mathbf{P2}$ can be transformed into the following form with respect to q_t^k and l_t^k:

$$\mathbf{P3}\ \min_{q_t^k, l_t^k} \underbrace{\sum_{k\in\mathcal{K}_t}\left(1-\frac{T^{\max}-\frac{z}{R_t^k}}{\frac{\tau c_k b_k}{f_k}+\frac{J(q_t^k+1)}{R_t^k}}+\beta\frac{\delta_{t,k}^2}{(2^{q_t^k}-1)^2}\right)}_{\triangleq\,\Phi(\boldsymbol{q}_t,\boldsymbol{l}_t)} \tag{21}$$

subject to (16b), (16c), (16e), and (19), where $\boldsymbol{q}_t=\{q_t^k\}_{k\in\mathcal{K}_t}$ and $\boldsymbol{l}_t=\{l_t^k\}_{k\in\mathcal{K}_t}$. At this point, it suffices to obtain the optimal q_t^k and l_t^k for $\mathbf{P3}$, and then the optimal ρ_t^k can be calculated using (20). However, $\mathbf{P3}$ is still a MINLP problem. To address this issue, we further decouple it into two subproblems and optimize q_t^k and l_t^k alternately.

1) Optimal Quantization Bit Width: Based on (19), each device k can achieve the maximum quantization bit width $q_{t,k}^{\max}$ when $\rho_t^k=\rho_{t,k}^{max}$. Since q_t^k is a positive integer, $q_{t,k}^{\max}$ can be expressed as

$$q_{t,k}^{max}=\left\lfloor\frac{R_t^k\left(T^{\max}-\frac{\tau c_k b_k(1-\rho_{t,k}^{max})}{f_k}\right)-z}{J(1-\rho_{t,k}^{\max})}-1\right\rfloor, \tag{22}$$

where $\lfloor x\rfloor$ is the floor of x. Thus, the quantization bit width q_t^k of device k satisfies $q_t^k\in\{1,\dots,q_{t,k}^{\max}\}$. Then, the theorem below is presented to guide the optimization of q_t^k.

Theorem 2. *Given the bandwidth allocation $\boldsymbol{l}_t=\{l_t^k\}_{k\in\mathcal{K}_t}$, the optimization objective $\Phi(\boldsymbol{q}_t,\boldsymbol{l}_t)$ in (21) exhibits one of the following properties as each q_t^k increases:*

1) *strictly monotonic: increasing or decreasing;*
2) *discretely valley-shaped: $\exists\, q_{t,k}^L\in\{1,\dots,q_{t,k}^{\max}-1\}$ that*

$$\begin{cases}\Phi(q_t^k=q+1)<\Phi(q_t^k=q),\ 1\le q<q_{t,k}^L,\\ \Phi(q_t^k=q+1)\ge\Phi(q_t^k=q),\ q=q_{t,k}^L,\\ \Phi(q_t^k=q+1)>\Phi(q_t^k=q),\ q_{t,k}^L<q<q_{t,k}^{\max},\end{cases} \tag{23}$$

   where $\Phi(q_t^k=q)$ denotes $\Phi(\boldsymbol{q}_t,\boldsymbol{l}_t|q_t^k=q)$.
3) *discretely unimodal: the reverse inequalities of case 2).*

These cases are determined by system parameters such as β, c_k, f_k, and $T^{\max}$. Thus, the optimal quantization bit width $q_{t,k}^*$ for each device k must lie at the boundary points 1, $q_{t,k}^{\max}$, or the valley bottom point $q_{t,k}^L$.

Proof: See Appendix B.

Remark 1. Theorem 2 demonstrates the trend of the optimization objective in (21) with respect to q_t^k for each device k under the given bandwidth allocation. If the trend is strictly monotonic, the optimal q_t^k must lie at the boundary points 1 or $q_{t,k}^{\max}$. In the discretely valley-shaped case, the objective function first decreases and then increases, forming a valley bottom at either a single point $q_{t,k}^L$ or a flat minimum over two adjacent points $q_{t,k}^L$ and $q_{t,k}^L+1$. Nevertheless, in both conditions, $q_{t,k}^L$ is the optimal solution, as a smaller quantization bit width is preferred for communication efficiency. In the discretely unimodal case, the situation is opposite to that of the valley case. Under this scenario, the optimal q_t^k must occur at one of the boundary points 1 or $q_{t,k}^{\max}$.

Based on Theorem 2, binary search can be employed to find the optimal q_t^k for each device k under the given bandwidth allocation. Regardless of the case in Theorem 2, binary search can identify a candidate for optimal q_t^k, which may correspond to either $q_{t,k}^L$ or one of the boundary points. Then, we compare this candidate with the boundary points 1 and $q_{t,k}^{\max}$, and select the one with the minimum objective function in (21) as the optimal quantization bit width $q_{t,k}^*$.

2) Optimal Bandwidth Allocation: According to the following lemma, we first prove that the optimization problem $\mathbf{P3}$ with respect to l_t^k is convex under the given q_t^k.

Lemma 1. *Given $\boldsymbol{q}_t=\{q_t^k\}_{k\in\mathcal{K}_t}$, the optimization problem $\mathbf{P3}$ with respect to l_t^k is convex.*

Proof: See Appendix C.

Therefore, according to the Lagrange multiplier method and Karush-Kuhn-Tucker (KKT) conditions, the optimal wireless resource allocation $l_{t,k}^*$ for each device $k\in\mathcal{K}_t$ in round t can be obtained from the theorem below.

Theorem 3. *Given the quantization bit width q_t^k, the optimal bandwidth allocated to device k is*

$$l_{t,k}^*=\left(\frac{\sqrt{\frac{\left(T^{\max}W_t^k+V_k z\right)B\log_2\left(1+\frac{h_t^k p_t^k}{N_0}\right)}{\lambda^*}}-W_t^k}{V_k B\log_2\left(1+\frac{h_t^k p_t^k}{N_0}\right)}\right)^+, \tag{24}$$

where $W_t^k=J(q_t^k+1)$, $V_k=\frac{\tau c_k b_k}{f_k}$, and λ^* is the optimal Lagrange multiplier given by

$$\lambda^*=\left(\frac{\sum_{k\in\mathcal{K}_t}\sqrt{\frac{T^{\max}J(q_t^k+1)+V_k z}{V_k^2 W_t^k}}}{1+\sum_{k\in\mathcal{K}_t}\frac{J(q_t^k+1)}{V_k W_t^k}}\right)^2. \tag{25}$$

Proof: See Appendix D.

From the above analysis, the optimized q_t^k and l_t^k for each device k can be obtained through alternating optimization.

Algorithm 1 Optimized Pruning Ratio, Quantization Bit Width and Wireless Bandwidth Allocation

Input: Selected devices $\mathcal{K}_t$.
Initialize the wireless bandwidth allocation l_t^k uniformly for all selected devices $k \in \mathcal{K}_t$;
repeat
  ▷ **Binary Search for Quantization Bit Width:**
  for each device $k \in \mathcal{K}_t$ **do**
    Initialize $q_{t,k}^L = 1$ and $q_{t,k}^R = q_{t,k}^{\max}$ using (22);
    while $q_{t,k}^L < q_{t,k}^R$ **do**
      $q_{t,k}^{mid} = (q_{t,k}^L + q_{t,k}^R)//2$;
      if $\Phi(q_t^k = q_{t,k}^{mid}) > \Phi(q_t^k = q_{t,k}^{mid} + 1)$ **then**
        $q_{t,k}^L = q_{t,k}^{mid} + 1$;
      else
        $q_{t,k}^R = q_{t,k}^{mid}$;
      end if
    end while
    $q_{t,k}^* = \arg\min_{q \in \{1, q_{t,k}^L, q_{t,k}^{\max}\}} \Phi(q_t^k = q)$;
  end for
  ▷ **Optimize Wireless Bandwidth Allocation:**
  Compute the optimized bandwidth allocation $l_{t,k}^*$ for each device $k \in \mathcal{K}_t$ based on Theorem 3;
until $q_{t,k}^*$ and $l_{t,k}^*$ no longer change;
▷ **Obtain Optimal Pruning Ratio:**
Compute the optimized pruning ratio $\rho_{t,k}^*$ for each device $k \in \mathcal{K}_t$ using (20);
return $\rho_{t,k}^*$, $q_{t,k}^*$ and $l_{t,k}^*$ for each device $k \in \mathcal{K}_t$.

Subsequently, the pruning ratio can be computed by substituting q_t^k and l_t^k into (20). The optimization algorithm for ρ_t^k, q_t^k, and l_t^k is given in Algorithm 1. We initialize the bandwidth allocation uniformly, which naturally satisfies the bandwidth nonnegativity and total bandwidth constraints without requiring prior information or additional calculations. Then, q_t^k and ρ_t^k can be obtained during the optimization without preset initial values. Moreover, the alternating optimization in Algorithm 1 can also be viewed as a form of block coordinate descent. Since each subproblem is solved optimally and the objective value is monotonically non-increasing, the obtained solution is block-coordinate optimal [38].

B. Device Selection Strategy

As discussed earlier, the optimized ρ_t^k, q_t^k, and l_t^k can be obtained under the given device selection. Thus, the most straightforward approach to determine the optimal device selection strategy for problem $\mathbf{P1}$ is the exhaustive method. However, this approach is impractical for real-world applications because of its high computational complexity. From the above analysis, both local computation power and channel condition have a significant impact on the convergence error and learning latency. Devices with low computing power or poor channel condition, referred to as stragglers, may exacerbate the convergence error and increase the latency. Therefore, we propose a heuristic device selection strategy that accounts for these two factors to identify such stragglers and prevent them from model training.

Algorithm 2 APQ-FL

Initialize the selected device set $\mathcal{K}_t$ to include all devices, i.e., $a_{t,k}^* = 1$ for $k = 1, \ldots, K$.
Obtain optimal pruning ratio $\rho_{t,k}^*$, quantization bit width $q_{t,k}^*$ and wireless bandwidth allocation $l_{t,k}^*$ for each device k through Algorithm 1;
Sort $s_t(k)$ defined in (26) in ascending order, and compute the current objective function in (16);
repeat
  Remove the device with the highest $s_t(k)$ from $\mathcal{K}_t$, and set the corresponding $a_{t,k}^* = 0$;
  Obtain $\rho_{t,k}^*$, $q_{t,k}^*$ and $l_{t,k}^*$ for each device $k \in \mathcal{K}_t$ through Algorithm 1;
  Compute the objective function in (16);
until the objective function in (16) no longer decreases.
return $a_{t,k}^*$, $\rho_{t,k}^*$, $q_{t,k}^*$ and $l_{t,k}^*$ for each device k.

As shown in Theorem 1, the convergence error increases with ρ_t^k and decreases with q_t^k. Moreover, according to (21) and Theorem 2, the pruning ratio and quantization bit width can quantify the device capability from both the computation and communication perspectives. The larger ρ_t^k and the smaller q_t^k are, the poorer the device's capabilities will be. Thus, based on the objective function (16) of problem $\mathbf{P1}$, the following $s_t(k)$ can be adopted as the criterion to identify stragglers:

$$s_t(k) = \rho_t^k + \beta \frac{\delta_{t,k}^2}{(2^{q_t^k} - 1)^2}. \tag{26}$$

Then, the device selection can be achieved by iteratively removing the device with the highest $s_t(k)$ until the objective function (16) no longer decreases. Based on Algorithm 1, the proposed APQ-FL is presented in Algorithm 2.

Note that the priority of the devices is the same for each iteration in Algorithm 2. This is because, according to (20), (24), and Theorem 2, the pruning ratio, quantization bit width, and bandwidth allocation are only related to the computation capabilities and channel conditions of devices, so that their order remains unchanged under different device selection. It helps ensure correctness while reducing computational complexity. Since the objective function in $\mathbf{P1}$ is the average of $s_t(k)$, the proposed method can effectively reduce the convergence error while achieving significantly lower computational cost compared with exhaustive search. The joint solution obtained by APQ-FL is not guaranteed to be globally optimal. However, APQ-FL can yield an effective solution with polynomial computational complexity, and its effectiveness is further validated by the experimental results.

Fairness-Aware Device Selection Extension: The above device selection strategy is aligned with the objective of $\mathbf{P1}$. Nevertheless, it primarily focuses on system efficiency, which may lead to long-term unfairness in device participation and exacerbate the impact of data heterogeneity. To improve device participation fairness, we further introduce a fairness-aware device selection extension that jointly considers participation fairness and system efficiency. Specifically, let r_t^k denote the

proportion of rounds in which device k has participated up to the end of round t, given by

$$r_t^k = \frac{1}{t}\sum_{\tau=1}^{t} a_\tau^k, \quad t \geq 1. \tag{27}$$

Then, the fairness-aware selection score for device k is

$$\tilde{s}_t(k) = \begin{cases} \frac{s_t(k)}{\sum_{n=1}^{K} s_t(n)}, & t = 1, \\ (1-\gamma)\frac{s_t(k)}{\sum_{n=1}^{K} s_t(n)} + \gamma \frac{r_{t-1}^k}{\sum_{n=1}^{K} r_{t-1}^n}, & t > 1. \end{cases} \tag{28}$$

At $t = 1$, no device has participated in training, and thus $\tilde{s}_t(k)$ is determined solely by system efficiency. For $t > 1$, the first and second terms represent the normalized system efficiency and historical participation, respectively, while $\gamma \in [0, 1]$ controls the trade-off between them. A larger γ places a greater emphasis on participation fairness. This extension can be incorporated into Algorithm 2 by replacing $s_t(k)$ with $\tilde{s}_t(k)$ for device ranking and removal, while keeping the remaining steps unchanged. Since devices with larger $\tilde{s}_t(k)$ are removed first, devices that have participated more frequently, i.e., those with larger r_{t-1}^k, are assigned a higher removal priority, whereas less frequently selected devices with smaller r_{t-1}^k are more likely to be retained. We refer to this extension as APQ-FL-Fair, and it reduces to APQ-FL when $\gamma = 0$.

C. Computational Complexity and Algorithm Deployment

In each iteration of Algorithm 1, the complexity of optimizing q_t^k using binary search for all selected devices is $\mathcal{O}(K \log_2 N)$, where N denotes the maximum feasible quantization bit width. The complexity of computing l_t^k using (24) for each device is $\mathcal{O}(1)$. Thus, the complexity of alternately optimizing q_t^k and l_t^k is $\mathcal{O}(IK \log_2 N)$, where I is the number of alternating iterations. Moreover, the complexity of computing the optimal ρ_t^k for each device is $\mathcal{O}(1)$. Therefore, the total complexity of Algorithm 1 is $\mathcal{O}(IK \log_2 N)$. In Algorithm 2, the complexity of initially sorting $s_t(k)$ is $\mathcal{O}(K \log_2 K)$. The number of device selection iterations is at most K. Therefore, the total complexity of the proposed method is $\mathcal{O}(K \log_2 K + IK^2 \log_2 N)$.

In each round, the server first obtains the estimated CSI and executes APQ-FL or APQ-FL-Fair to optimize ρ_t^k, q_t^k, a_t^k, and l_t^k for each device. Then, the server prunes the global model for each selected device and distributes the pruned model and the optimized strategies to the corresponding device. Subsequently, the training procedures are carried out as described in Section II. Note that model pruning is performed at the server, while the quantization of local updates before uploading is a necessary step for communication rather than an operation specific to APQ-FL or APQ-FL-Fair. Therefore, our methods do not introduce additional device-side compression overhead.

In practical systems, the CSI obtained using existing channel estimation methods [39], [40] may be biased due to estimation errors and delayed feedback, thereby affecting the optimization decisions and model performance. However, this does not invalidate the convergence analysis established in Theorem 1, and the proposed method remains applicable. This is because biased CSI does not alter the mathematical forms of local updates and model aggregation. An overestimated channel quality may lead to an underestimated transmission latency, such that some devices may fail to upload their local updates within the latency constraint. In this case, once the latency limit is reached, the server can aggregate the received local updates and directly proceed to the next round. The timed-out devices are similar to those not participating in the current round. When the channel quality is underestimated, the actual transmission rate is higher than the estimated rate, making the latency constraint easier to satisfy. Therefore, to reduce the timeout risk caused by CSI bias, a straightforward approach is to construct a conservative estimate of the channel gain based on the available CSI. If the actual channel gain is no lower than this conservative estimate, the actual upload latency can be guaranteed to satisfy the given latency constraint. Since CSI is treated as the environmental parameter, its processing does not require any modification to APQ-FL or APQ-FL-Fair.

Note that the optimal value of β in (16) is difficult to obtain in practice, as it is influenced by the unknown parameters L and D. Thus, we empirically determine the value of β through experiments. This approach can be easily implemented and demonstrates superior performance in real-world scenarios.

V. Experimental Results

We consider $K = 20$ devices randomly distributed in a circular area with radius 600 m, where the server is located at the center. The total uplink bandwidth for wireless transmission is $B = 20$ MHz. The path loss model is $128.1 + 37.6 \log_{10}(d)$, where d denotes the distance (in km) between the device and the server, and the small-scale fading is modeled as Rayleigh fading. The noise power spectral density is -174 dBm/Hz. The downlink quantization bit width is set to 16. We employ ResNet18 [41] on two benchmark datasets, CIFAR10 and CINIC10, and adopt the commonly used Dirichlet partition to generate non-IID data across devices. The Dirichlet parameter α is set to 0.5 by default. The transmit power of each device is randomly selected from the range [10, 26] dBm. The CPU frequency of each device is randomly chosen from the range [2, 3] GHz, and the number of CPU cycles required to process one data sample is set to 5×10^5. Moreover, the quantization limits $\Delta\underline{w}_{t,k}$ and $\Delta\overline{w}_{t,k}$ are set to 0 and 1, respectively, since the absolute value of model weight updates rarely exceeds 1 in practice. The coefficients β in (16) and γ in (28) are set to 0.8 and 0.4, respectively. The maximum pruning ratio for devices is set to 0.9. The number of local epochs is set to 5. The local optimizer is SGD with learning rate of 0.01 and momentum of 0.9. The local batch size is set to 64.

A. Performance Evaluation of APQ-FL and APQ-FL-Fair

We compare APQ-FL and APQ-FL-Fair with other baselines under different latency constraints. Since, to the best of our knowledge, no prior work jointly optimizes model pruning, quantization scheme, device selection, and bandwidth allocation under latency and bandwidth constraints, we select the following methods with similar tasks for comparison.

- **Baseline 1**: Standard FedAvg [2], which does not apply pruning, uses a fixed quantization bit width, allocates the bandwidth uniformly, and selects devices at random.

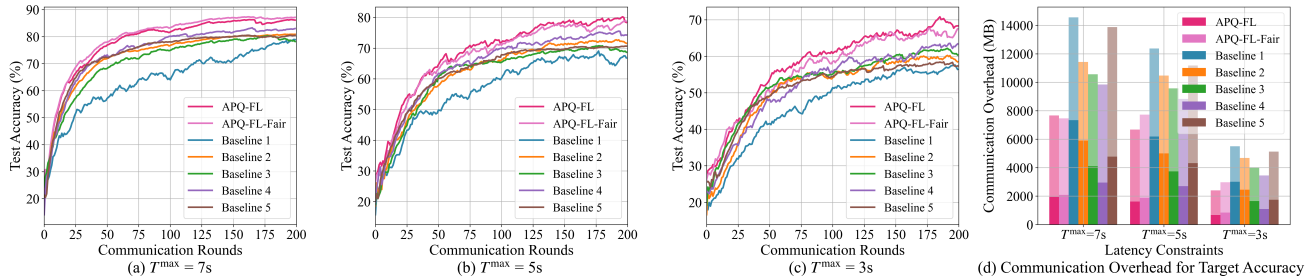


Fig. 1. Comparison of different methods under different latency constraints on the CIFAR10 dataset. (a)-(c) report the test accuracy for $T^{\max} = 7$, 5, and 3 s, respectively, and (d) presents the total communication overhead to reach the target accuracy with darker/lighter parts denoting uplink/downlink overhead.

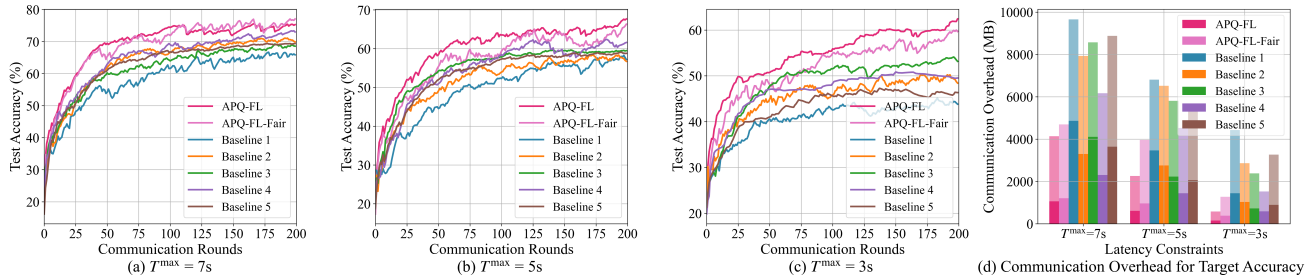


Fig. 2. Comparison of different methods under different latency constraints on the CINIC10 dataset. (a)-(c) show the test accuracy for $T^{\max} = 7$, 5, and 3 s, respectively, and (d) reports the total communication overhead to reach the target accuracy with darker/lighter parts denoting uplink/downlink overhead.

- **Baseline 2**: The method in [18], which optimizes pruning ratio, device selection, and resource allocation under latency constraints, but does not optimize quantization.
- **Baseline 3**: The method in [19], which also focuses on the pruning ratio, device selection, and bandwidth allocation under latency constraints using a different strategy.
- **Baseline 4**: The method in [24], which optimizes quantization bit width and wireless resource allocation but does not involve pruning or device selection.
- **Baseline 5**: The method in [23], which similarly optimizes quantization bit width and bandwidth allocation without considering pruning or device selection.

For the methods without device selection optimization (baselines 1, 4, and 5), we iteratively add devices to the selection set at random and apply their algorithms until the latency constraint is no longer satisfied. This ensures the maximum number of devices is selected under the constraint. For baselines 1, 2, and 3 with fixed quantization, we set the quantization bit width to 16 to minimize quantization error. These configurations enable all baselines to exhaust their latency budgets to achieve better possible performance.

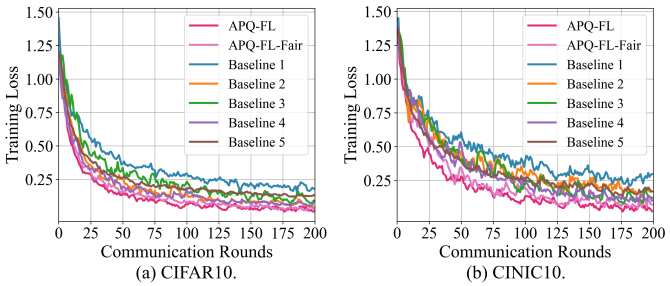
We first plot the test accuracy over communication rounds and the total communication overhead required to reach the target accuracy under latency thresholds of 7, 5, and 3 s, as shown in Figures 1 and 2. The total communication overhead refers to the cumulative uplink and downlink communication volume (in MB) up to the round when the FL system first achieves the target accuracy. Note that the downlink overhead is influenced by the number of selected devices in each round. The target accuracies are set to 75%, 70%, and 55% under latency constraints of 7, 5, and 3 s, respectively, for the CIFAR10 dataset, and 65%, 55%, and 45% for the CINIC10 dataset. These values are chosen such that all baselines can reach them. It can be observed that APQ-FL and APQ-FL-Fair significantly outperform these baselines in all cases, improving the test accuracy by 4.96%–15.33%. Meanwhile, APQ-FL and APQ-FL-Fair reduce the uplink communication overhead by 23.14%–74.45% and the total overhead by 12.47%–62.16% to reach the target accuracy. Moreover, when the latency constraint is relatively tight, APQ-FL performs better than APQ-FL-Fair. As $T^{\max}$ increases, APQ-FL-Fair can achieve better model performance by facilitating fairer device participation. These results reflect the trade-off between system efficiency and device participation fairness.

To further analyze APQ-FL and APQ-FL-Fair, we present their final test accuracy and device participation fairness on the CIFAR10 dataset, as shown in Table II. We employ Jain's fairness index [42] to evaluate participation fairness, defined as $\nu = \frac{\left(\sum_{k=1}^{K} n_k\right)^2}{K \sum_{k=1}^{K} n_k^2}$, where n_k denotes the number of times device k is selected over all communication rounds. The closer ν is to 1, the higher the participation fairness. When the latency constraint is relatively tight, system efficiency becomes the dominant factor. Devices generally need to adopt higher pruning ratios and lower quantization bit widths to satisfy the latency requirement, which introduces larger compression errors and degrades the quality of the uploaded updates. In this case, the effect of pruning and quantization errors outweighs the benefit of greater training data diversity. As the latency constraint is relaxed, the required degree of compression decreases, thereby reducing compression errors. Therefore, fairer device participation can alleviate the model bias caused by imbalanced device selection under non-IID data.

To validate the impact of the proposed method on model convergence, we report the training loss of different methods

TABLE II
COMPARISON OF APQ-FL AND APQ-FL-FAIR ON THE CIFAR10 DATASET UNDER DIFFERENT LATENCY CONSTRAINTS

Methods	$T^{\max} = 7$ s		$T^{\max} = 5$ s		$T^{\max} = 3$ s	
	Accuracy (%)	Fair Index	Accuracy (%)	Fair Index	Accuracy (%)	Fair Index
APQ-FL	86.21	0.87	79.43	0.83	68.19	0.54
APQ-FL-Fair	87.36	0.95	78.59	0.93	66.22	0.92



(a) CIFAR10. (b) CINIC10.

Fig. 3. Training loss of different methods on CIFAR10 and CINIC10 datasets.

over communication rounds on the CIFAR10 and CINIC10 datasets under $T^{\max} = 5$ s, as shown in Figure 3. It is observed that APQ-FL and APQ-FL-Fair can enable faster convergence and achieve a lower final loss, demonstrating their effectiveness in improving convergence performance.

We further tighten the latency constraints to 1 and 0.5 s, and evaluate these methods on the CIFAR10 dataset. Under these stricter constraints, we observe that only APQ-FL and APQ-FL-Fair can maintain stable and effective FL training. In contrast, all other baselines fail to yield solutions that satisfy the latency requirements, making FL training infeasible. This is because for such stricter constraints, solely optimizing pruning or quantization is insufficient due to the limited computation and communication capabilities of devices, which demonstrates the necessity of their joint optimization. As a result, only APQ-FL and APQ-FL-Fair can be evaluated under these constraints, as shown in Table III. We also present the average number of selected devices per round, as well as the average pruning ratio and bit width per device per round.

To evaluate the impact of data heterogeneity, we also compare the test accuracy of different methods on the CIFAR10 dataset with $\alpha = 0.1$ and $\alpha = 0.5$ under $T^{\max} = 5$ s, as shown in Figure 4. The results show that APQ-FL and APQ-FL-Fair consistently outperform the other baselines under different degrees of data heterogeneity. Moreover, APQ-FL achieves higher test accuracy than APQ-FL-Fair when $\alpha = 0.5$, whereas APQ-FL-Fair outperforms APQ-FL when $\alpha = 0.1$. This indicates that fair device participation becomes more beneficial as data heterogeneity increases and can help mitigate the impact of non-IID data.

We further conduct experiments with different numbers of devices K on the CIFAR10 dataset with $\alpha = 0.5$ under $T^{\max} = 5$ s to evaluate the scalability of the proposed methods. We set K to 20, 50, and 100, and report the test accuracy of each method and its average per-round wall-clock time for joint optimization in Table IV. It is observed that APQ-FL and APQ-FL-Fair outperform the baselines in all cases, while maintaining competitive execution time. Moreover, as K increases, the test accuracy of APQ-FL-Fair gradually shifts from below that of APQ-FL to above it. This indicates that when the number of devices is large, the efficiency-oriented selection strategy of APQ-FL may lead to more pronounced long-term participation bias, whereas APQ-FL-Fair can mitigate this bias by facilitating fairer device participation.

TABLE III
TEST ACCURACY (%) OF APQ-FL AND APQ-FL-FAIR UNDER DIFFERENT LATENCIES ON THE CIFAR10 DATASET, ALONG WITH THE AVERAGE NUMBER OF SELECTED DEVICES PER ROUND, AND THE AVERAGE PRUNING RATIO AND QUANTIZATION BIT WIDTH PER DEVICE PER ROUND.

Method	$T^{\max}$	Accuracy (%)	Average Number of Devices	Average Pruning Ratio	Average Bit Width
APQ-FL	1.0 s	66.76	4.1	0.582	4.3
	0.5 s	59.78	3.2	0.797	3.5
APQ-FL-Fair	1.0 s	63.29	3.9	0.724	2.7
	0.5 s	51.24	2.6	0.876	2.3

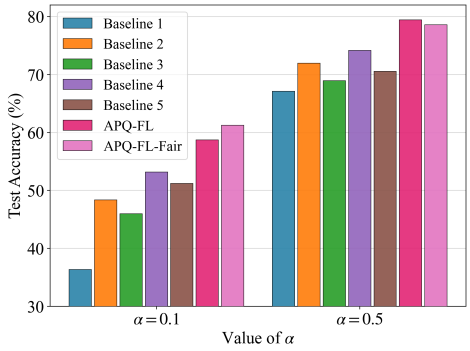


Fig. 4. Test accuracy of different methods on the CIFAR10 dataset with $\alpha = 0.1$ and $\alpha = 0.5$.

TABLE IV
TEST ACCURACY (%) AND AVERAGE PER-ROUND WALL-CLOCK TIME FOR JOINT OPTIMIZATION (MS) WITH VARYING K. "–" INDICATES THAT THE CORRESPONDING METHOD DOES NOT INVOLVE JOINT OPTIMIZATION.

Methods	$K = 20$		$K = 50$		$K = 100$	
	Accuracy (%)	Time (ms)	Accuracy (%)	Time (ms)	Accuracy (%)	Time (ms)
Baseline 1	67.13	–	59.89	–	39.51	–
Baseline 2	71.96	7.23	66.03	15.74	60.17	28.74
Baseline 3	68.94	12.93	65.61	22.17	61.56	53.26
Baseline 4	74.18	4.23	73.24	7.85	68.12	12.03
Baseline 5	70.57	4.57	68.27	8.02	64.74	13.11
APQ-FL	79.43	4.34	78.12	9.55	75.03	14.85
APQ-FL-Fair	78.59	4.32	77.96	8.31	76.76	15.35

B. Effect of Adaptive Model Pruning, Quantization Scheme, Device Selection, and Wireless Bandwidth Allocation

We evaluate the impact of adaptive model pruning, quantization scheme, device selection, and wireless bandwidth allocation in APQ-FL on the CIFAR10 dataset. Since APQ-FL-Fair differs from APQ-FL only in its fairness-aware device selection score, the effect of this extension has been evaluated

TABLE V
TEST ACCURACY (%) OF APQ-FL AND ITS VARIANTS ON THE CIFAR10 DATASET UNDER DIFFERENT LATENCY CONSTRAINTS.

	$T^{\max} = 7$s	$T^{\max} = 5$s	$T^{\max} = 3$s
NP	84.75	77.89	64.33
NQ	83.41	75.18	63.86
NB	83.64	76.03	65.42
ND	82.96	75.52	66.27
APQ-FL	86.21	79.43	68.19

in the preceding comparison with APQ-FL. Specifically, we compare APQ-FL with the following variants.

- **NP (No Pruning)**: APQ-FL is used to optimize quantization bit width, device selection, and bandwidth allocation without pruning.
- **NQ (No Quantization Optimization)**: The pruning ratio, device selection, and bandwidth allocation are optimized via APQ-FL with a fixed quantization bit width of 16.
- **NB (No Bandwidth Optimization)**: APQ-FL is applied to optimize the pruning ratio, quantization bit width, and device selection with uniform bandwidth allocation.
- **ND (No Device Selection Optimization)**: APQ-FL optimizes pruning ratio, quantization bit width, and bandwidth allocation while randomly selecting the maximum number of devices that meet the latency constraint.

To ensure variants remain feasible under the latency requirements and facilitate comparison with the aforementioned baselines, we set the latency thresholds to 7, 5, and 3 s, as shown in Table V. It is observed that removing any of the optimization factors will lead to a decline in test accuracy. These results illustrate the necessity of joint optimization of these factors. Moreover, these variants still outperform the baselines, indicating that our method can achieve superior model performance even when optimizing the same factors.

C. Experiments on the Coefficients β and γ

We evaluate the impact of different values of β on APQ-FL and different values of γ on APQ-FL-Fair under latency thresholds of 5 and 1 s using the CIFAR10 dataset, as shown in Tables VI and VII. As β and γ increase, the test accuracy of APQ-FL and APQ-FL-Fair first increases and then decreases. APQ-FL achieves its highest test accuracy at $\beta = 0.8$, whereas APQ-FL-Fair reaches its peak at $\gamma = 0.4$. This suggests that the most favorable trade-off between pruning and quantization errors in APQ-FL is achieved when $\beta = 0.8$, and the best balance between system efficiency and participation fairness is obtained at $\gamma = 0.4$. Despite these variations, APQ-FL and APQ-FL-Fair consistently provide superior and stable performance regardless of the value of β and γ.

TABLE VI
TEST ACCURACY (%) OF APQ-FL ON THE CIFAR10 DATASET WITH VARYING β

β	0.2	0.4	0.6	0.8	1.0
$T^{\max} = 5$ s	75.25	77.14	78.31	79.43	78.92
$T^{\max} = 1$ s	60.76	62.51	64.38	66.76	65.94

TABLE VII
TEST ACCURACY (%) OF APQ-FL-FAIR ON THE CIFAR10 DATASET WITH CONSTANT $\beta = 0.8$ AND VARYING γ

γ	0.2	0.4	0.6	0.8	1.0
$T^{\max} = 5$ s	78.28	78.59	77.85	77.02	75.38
$T^{\max} = 1$ s	62.91	63.29	62.26	61.17	59.94

VI. CONCLUSION

In this paper, we proposed APQ-FL to improve the efficiency of FL under resource-constrained scenarios while preserving model performance. We first provided the convergence analysis of FL with model pruning and quantized transmission, and revealed the interdependent relationship between the pruning ratio and quantization bit width. Then, the pruning ratio, quantization bit width, device selection, and bandwidth allocation were jointly optimized to minimize the convergence upper bound under latency and bandwidth constraints. Furthermore, we introduced a fairness-aware device selection extension for APQ-FL, termed APQ-FL-Fair. Experiments have shown that APQ-FL and APQ-FL-Fair consistently outperform state-of-the-art alternatives that do not jointly consider the optimization of model pruning and quantization.

APPENDIX A
PROOF OF THEOREM 1

First, we introduce the following lemma established in [22] on the stochastic quantization.

Lemma 2. *$\mathcal{Q}(\Delta w_t^{k,j})$ is an unbiased estimator of $\Delta w_t^{k,j}$, i.e., $\mathbb{E}[Q(\Delta w_t^{k,j})] = \Delta w_t^{k,j}$, while it also holds that*

$$\mathbb{E}[\|Q(\Delta w_t^{k,j}) - \Delta w_t^{k,j}\|^2] \leq \frac{\delta_{t,k}^2}{4(2^{q_t^k}-1)^2}, \tag{29}$$

where $\delta_{t,k} = \Delta\overline{w}_{t,k} - \Delta\underline{w}_{t,k}$.

We briefly summarize its proof as follows. From (4),

$$\begin{aligned}\mathbb{E}[\mathcal{Q}(\Delta w_t^{k,j})] &= c_u \frac{c_{u+1} - |\Delta w_t^{k,j}|}{c_{u+1} - c_u} + c_{u+1}\frac{|\Delta w_t^{k,j}| - c_u}{c_{u+1} - c_u} \\ &= \frac{\Delta w_t^{k,j}(c_{u+1} - c_u)}{c_{u+1} - c_u} = \Delta w_t^{k,j}. \end{aligned} \tag{30}$$

$$\begin{aligned}&\mathbb{E}[\|\mathcal{Q}(\Delta w_t^{k,j}) - \Delta w_t^{k,j}\|^2] = (c_u - |\Delta w_t^{k,j}|)^2 \frac{c_{u+1} - |\Delta w_t^{k,j}|}{c_{u+1} - c_u} \\ &\quad + (c_{u+1} - |\Delta w_t^{k,j}|)^2 \frac{|\Delta w_t^{k,j}| - c_u}{c_{u+1} - c_u} \\ &= (|\Delta w_t^{k,j}| - c_u)(c_{u+1} - |\Delta w_t^{k,j}|) \\ &\leq (\frac{c_{u+1} - c_u}{2})^2 = \left(\frac{\delta_{t,k}}{2(2^{q_t^k}-1)}\right)^2 = \frac{\delta_{t,k}^2}{4(2^{q_t^k}-1)^2}. \end{aligned} \tag{31}$$

We next prove Theorem 1. According to Assumption 1,

$$\begin{aligned}&\mathbb{E}\left[F(\boldsymbol{w}_{t+1})\right] - \mathbb{E}\left[F(\boldsymbol{w}_t)\right] \\ &\leq \underbrace{\mathbb{E}\left\langle \nabla F(\boldsymbol{w}_t), \boldsymbol{w}_{t+1} - \boldsymbol{w}_t \right\rangle}_{A_1} + \frac{L}{2}\underbrace{\mathbb{E}\left\|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\right\|^2}_{A_2}. \end{aligned} \tag{32}$$

Then, A_1 in (32) can be derived as

$$\begin{aligned}
A_1 &\overset{(a)}{=} \sum_{j\in\mathcal{I}_t} \mathbb{E}\left\langle \nabla F^j(\boldsymbol{w}_t), \boldsymbol{w}_{t+1}^j - \boldsymbol{w}_t^j \right\rangle \\
&\overset{(b)}{=} \sum_{j=1}^{J} \mathbb{E}\left\langle \nabla F^j(\boldsymbol{w}_t), \boldsymbol{w}_{t+1}^j - \boldsymbol{w}_t^j \right\rangle \\
&\overset{(c)}{=} \sum_{j=1}^{J} \mathbb{E}\left\langle \nabla F^j(\boldsymbol{w}_t), \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} Q(\Delta \boldsymbol{w}_t^{k,j}) \right\rangle \\
&\overset{(d)}{=} \sum_{j=1}^{J} \mathbb{E}\left\langle \nabla F^j(\boldsymbol{w}_t), \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \Delta \boldsymbol{w}_t^{k,j} \right\rangle \\
&\overset{(e)}{=} -\eta \frac{1}{\tau} \sum_{j=1}^{J} \mathbb{E}\left\langle \tau \nabla F^j(\boldsymbol{w}_t), \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right\rangle \\
&\overset{(f)}{=} -\frac{\eta\tau}{2} \sum_{j=1}^{J} \mathbb{E}\left\|\nabla F^j(\boldsymbol{w}_t)\right\|^2 \\
&\quad - \frac{\eta}{2\tau} \sum_{j=1}^{J} \mathbb{E}\left\| \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right\|^2 \\
&\quad + \frac{\eta}{2\tau} \sum_{j=1}^{J} \mathbb{E}\left\| \tau \nabla F^j(\boldsymbol{w}_t) - \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right\|^2,
\end{aligned} \tag{33}$$

where (a) considers the sum of inner products over the weights in $\mathcal{I}_t$, (b) follows from $|w_{t+1}^j - w_t^j| = 0$ for any $j \notin \mathcal{I}_t$, (c) applies (9), (d) uses Lemma 2, (e) utilizes Assumption 2, and (f) uses $\langle a, b\rangle = \frac{\|a\|^2 + \|b\|^2 - \|a-b\|^2}{2}$.

A_2 in (32) is bounded by

$$\begin{aligned}
A_2 &\overset{(a)}{=} \sum_{j\in\mathcal{I}_t} \mathbb{E}\left\|\boldsymbol{w}_{t+1}^j - \boldsymbol{w}_t^j\right\|^2 = \sum_{j=1}^{J} \mathbb{E}\left\|\boldsymbol{w}_{t+1}^j - \boldsymbol{w}_t^j\right\|^2 \\
&\overset{(b)}{=} \sum_{j=1}^{J} \mathbb{E}\left\| \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \left(Q(\Delta \boldsymbol{w}_t^{k,j}) - \Delta \boldsymbol{w}_t^{k,j} \right) \right\|^2 \\
&\quad + \sum_{j=1}^{J} \mathbb{E}\left\| \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \Delta \boldsymbol{w}_t^{k,j} \right\|^2 \\
&\overset{(c)}{\le} \sum_{j=1}^{J} \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \mathbb{E}\left\| Q(\Delta \boldsymbol{w}_t^{k,j}) - \Delta \boldsymbol{w}_t^{k,j} \right\|^2 \\
&\quad + \sum_{j=1}^{J} \mathbb{E}\left\| - \frac{\eta}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k, \xi_k) \right\|^2 \\
&\le \frac{1}{p_t^* \left|\mathcal{K}_t\right|} \sum_{j=1}^{J} \sum_{k\in\mathcal{K}_t} \mathbb{E}\left\| Q(\Delta \boldsymbol{w}_t^{k,j}) - \Delta \boldsymbol{w}_t^{k,j} \right\|^2 \\
&\quad + 2\eta^2 \sum_{j=1}^{J} \mathbb{E}\left\| \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \left[\nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k, \xi_k) - \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right] \right\|^2 \\
&\quad + 2\eta^2 \sum_{j=1}^{J} \mathbb{E}\left\| \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right\|^2 \\
&\overset{(d)}{\le} \frac{J}{p^* \left|\mathcal{K}_t\right|} \sum_{k\in\mathcal{K}_t} \frac{\delta_{t,k}^2}{4(2^{q_t^k} - 1)^2} + 2\eta^2\tau^2\sigma^2 J \\
&\quad + 2\eta^2 \sum_{j=1}^{J} \mathbb{E}\left\| \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right\|^2,
\end{aligned} \tag{34}$$

where (a) decomposes the squared norm over weights, (b) follows from $\mathbb{E}[\|x\|^2] = \mathbb{E}[\|x - \mathbb{E}[x]\|^2] + \|\mathbb{E}[x]\|^2$, (c) uses Jensen's inequality, and (d) uses Assumptions 2, 4, and (29).

By substituting (33) and (34), (32) can be expressed as

$$\begin{aligned}
&\mathbb{E}\left[F(\boldsymbol{w}_{t+1})\right] - \mathbb{E}\left[F(\boldsymbol{w}_t)\right] \\
&\le -\frac{\eta\tau}{2} \sum_{j=1}^{J} \mathbb{E}\left\|\nabla F^j(\boldsymbol{w}_t)\right\|^2 \\
&\quad - \left(\frac{\eta}{2\tau} - \eta^2 L\right) \sum_{j=1}^{J} \mathbb{E}\left\| \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right\|^2 \\
&\quad + \frac{\eta}{2\tau} \sum_{j=1}^{J} \mathbb{E}\left\| \tau \nabla F^j(\boldsymbol{w}_t) - \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right\|^2 \\
&\quad + \frac{LJ}{2p^* \left|\mathcal{K}_t\right|} \sum_{k\in\mathcal{K}_t} \frac{\delta_{t,k}^2}{4(2^{q_t^k} - 1)^2} + \eta^2\tau^2\sigma^2 JL \\
&\overset{(a)}{\le} -\frac{\eta\tau}{2} \sum_{j=1}^{J} \mathbb{E}\left\|\nabla F^j(\boldsymbol{w}_t)\right\|^2 + \eta^2\tau^2\sigma^2 JL \\
&\quad + \frac{LJ}{2p^* \left|\mathcal{K}_t\right|} \sum_{k\in\mathcal{K}_t} \frac{\delta_{t,k}^2}{4(2^{q_t^k} - 1)^2} \\
&\quad + \frac{\eta}{2\tau} \underbrace{\sum_{j=1}^{J} \mathbb{E}\left\| \tau \nabla F^j(\boldsymbol{w}_t) - \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right\|^2}_{B}.
\end{aligned} \tag{35}$$

where (a) uses the fact that $\frac{\eta}{2\tau} - \eta^2 L \ge 0$ when $\eta \le \frac{1}{2\tau L}$.

B in (35) can be derived as

$$\begin{aligned}
B &= \mathbb{E}\left\| \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \sum_{r=0}^{\tau-1} \left[\nabla F_k^j(\boldsymbol{w}_t) - \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right] \right\|^2 \\
&\overset{(a)}{\le} \frac{1}{\left|\mathcal{K}_t^j\right|} \sum_{k\in\mathcal{K}_t^j} \tau \sum_{r=0}^{\tau-1} \mathbb{E}\left\| \nabla F_k^j(\boldsymbol{w}_t) - \nabla F_k^j(\hat{\boldsymbol{w}}_{t,r}^k) \right\|^2 \\
&\overset{(b)}{\le} \frac{\tau}{p_t^* \left|\mathcal{K}_t\right|} \sum_{k\in\mathcal{K}_t} \sum_{r=0}^{\tau-1} \mathbb{E}\left\| \nabla F_k(\boldsymbol{w}_t) - \nabla F_k(\hat{\boldsymbol{w}}_{t,r}^k) \right\|^2 \\
&\overset{(c)}{\le} \frac{\tau L^2}{p^* \left|\mathcal{K}_t\right|} \sum_{k\in\mathcal{K}_t} \sum_{r=0}^{\tau-1} \underbrace{\mathbb{E}\left\| \boldsymbol{w}_t - \hat{\boldsymbol{w}}_{t,r}^k \right\|^2}_{C},
\end{aligned} \tag{36}$$

where (a) uses Jensen's inequality. In step (b), we consider that l_2-gradient norm of a vector is no larger than the sum of norm of all sub-vectors, which allows us to consider ∇F_k rather than its sub-vectors. (c) utilizes Assumptions 1 and 4.

Then, C in (36) can be derived as

$$C = \mathbb{E}\left\| \hat{\boldsymbol{w}}_{t,r}^k - \hat{\boldsymbol{w}}_t^k + \hat{\boldsymbol{w}}_t^k - \boldsymbol{w}_t \right\|^2$$

$$\leq 2\mathbb{E}\left\|\hat{\boldsymbol{w}}_{t,r}^k-\hat{\boldsymbol{w}}_t^k\right\|^2+2\mathbb{E}\left\|\hat{\boldsymbol{w}}_t^k-\boldsymbol{w}_t\right\|^2$$
$$\overset{(a)}{\leq} 2\eta^2 r\sum_{s=0}^{r-1}\mathbb{E}\left\|\nabla F_k(\hat{\boldsymbol{w}}_{t,s}^k,\xi_k)\odot\boldsymbol{m}_t^k\right\|^2+2\rho_t^k\mathbb{E}\left\|\boldsymbol{w}_t\right\|^2$$
$$\overset{(b)}{\leq} 2\eta^2 r^2G^2+2\rho_t^kD^2, \tag{37}$$

where (a) follows from Jensen's inequality and that, as shown in [18], [43], the model error under pruning ratio ρ_t^k satisfies $\mathbb{E}\|\boldsymbol{w}_t-\hat{\boldsymbol{w}}_t^k\|^2\leq\rho_t^k\mathbb{E}\left\|\boldsymbol{w}_t\right\|^2$, and (b) uses Assumption 3.

By substituting (36) and (37) and taking the average expectation across all rounds, we obtain

$$\frac{1}{TJ}\sum_{t=1}^{T}\sum_{j=1}^{J}\mathbb{E}\left\|\nabla F^j(\boldsymbol{w}_t)\right\|^2$$
$$\leq\frac{2(F_0-F_*)}{TJ\eta\tau}+2\eta\tau L\sigma^2+\frac{\eta^2G^2L^2(\tau-1)(2\tau-1)}{3p^*}$$
$$+\frac{2L^2D^2}{Tp^*}\sum_{t=1}^{T}\frac{1}{|\mathcal{K}_t|}\sum_{k\in\mathcal{K}_t}\left(\rho_t^k+\frac{1}{8\eta\tau LD^2}\frac{\delta_{t,k}^2}{(2^{q_t^k}-1)^2}\right), \tag{38}$$

which proves the Theorem 1.

APPENDIX B
PROOF OF THEOREM 2

We first relax q_t^k to a continuous variable within the interval $[1,q_{t,k}^{\max}]$, and then present the following lemma.

Lemma 3. *After relaxing q_t^k to a continuous variable, the objective function* (21) *in* $\mathbf{P3}$ *has at most one point where the first-order derivative with respect to q_t^k is zero.*

Lemma 3 indicates that after relaxing q_t^k, the objective function (21) must fall into one of the following four cases: strictly increasing, strictly decreasing, increasing then decreasing, or decreasing then increasing. Therefore, Theorem 2 naturally holds when q_t^k is constrained back to integers.

We next prove Lemma 3. To simplify, we denote $A_{t,k}=T^{\max}-\frac{z}{R_t^k}$, $B_k=\frac{\tau c_k b_k}{f_k}$, and $C_{t,k}=\frac{J}{R_t^k}$. Then, the first-order derivative of the objective function $\Phi(\boldsymbol{q}_t^k,\boldsymbol{l}_t^k)$ in (21) with respect to q_t^k is

$$\frac{\partial\Phi}{\partial q_t^k}=\frac{A_{t,k}C_{t,k}}{\left(B_k+C_{t,k}(q_t^k+1)\right)^2}-\frac{2\beta\delta_{t,k}^2 2^{q_t^k}\ln 2}{(2^{q_t^k}-1)^3}. \tag{39}$$

Let

$$\begin{cases}L(q_t^k)=\dfrac{A_{t,k}C_{t,k}}{\left(B_k+C_{t,k}(q_t^k+1)\right)^2},\\ R(q_t^k)=\dfrac{2\beta\delta_{t,k}^2 2^{q_t^k}\ln 2}{(2^{q_t^k}-1)^3},\end{cases} \tag{40}$$

Then, (39) can be written as $\partial\Phi/\partial q_t^k=L(q_t^k)-R(q_t^k)$. Obviously, the first-order derivative of Φ with respect to q_t^k equals zero if and only if $L(q_t^k)=R(q_t^k)$. Consider

$$\varphi(q_t^k)=\ln\frac{L(q_t^k)}{R(q_t^k)}=\ln L(q_t^k)-\ln R(q_t^k). \tag{41}$$

At this point, to prove Lemma 3, we only need to prove that $\varphi(q_t^k)=0$ has at most one solution in $[1,q_{t,k}^{\max}]$. The first-order derivative of $\varphi(q_t^k)$ with respect to q_t^k is

$$\varphi'(q_t^k)=-\frac{2}{a_{t,k}+q_t^k+1}+\frac{2^{q_t^k+1}+1}{2^{q_t^k}-1}\ln 2, \tag{42}$$

where $a_{t,k}=\frac{B_k}{C_{t,k}}>0$. We next prove that $\varphi'(q_t^k)>0$ holds for all $q_t^k\in[1,q_{t,k}^{\max}]$, which directly implies that $\varphi(q_t^k)$ has at most one root in this interval. Since $a_{t,k}>0$, we have $\frac{2}{q_t^k+1}>\frac{2}{a_{t,k}+q_t^k+1}$. Therefore, based on (42), we only need to prove that

$$\frac{2^{q_t^k+1}+1}{2^{q_t^k}-1}\ln 2>\frac{2}{q_t^k+1},\quad\forall q_t^k\in[1,q_{t,k}^{\max}]. \tag{43}$$

Equation (43) can be rewritten as

$$(2^{q_t^k+1}+1)(q_t^k+1)\ln 2-2(2^{q_t^k}-1)>0, \tag{44}$$

Let $g(q_t^k)$ denote the left side of the above inequality. Then, $g(q_t^k)$ can be simplified as

$$g(q_t^k)=2^{q_t^k+1}\left((q_t^k+1)\ln 2-1\right)+(q_t^k+1)\ln 2+2. \tag{45}$$

Clearly, $g(q_t^k)$ attains its minimum at $q_t^k=1$, where $g(1)=4(2\ln 2-1)+2\ln 2+2>0$. Hence, $g(q_t^k)>0$ holds for all $q_t^k\in[1,q_{t,k}^{\max}]$. This means that $\varphi'(q_t^k)>0$ holds for all $q_t^k\in[1,q_{t,k}^{\max}]$, and thus $\varphi(q_t^k)$ is strictly increasing on this interval. Therefore, $\varphi(q_t^k)=0$ has at most one solution, which completes the proof of Lemma 3.

APPENDIX C
PROOF OF LEMMA 1

To simplify, we denote $B_k=\frac{\tau c_k b_k}{f_k}$ and $W_t^k=J(q_t^k+1)$. Then, the second-order derivative with respect to l_t^k is

$$\frac{\partial^2\Phi}{\partial(\boldsymbol{l}_t^k)^2}=\frac{2(T^{\max}W_t^k+B_kz)B_k(B_kR_t^k+W_t^k)}{(B_kR_t^k+W_t^k)^4}$$
$$\times\left[B\log_2\left(1+\frac{h_t^kp_t^k}{N_0}\right)\right]^2\geq 0. \tag{46}$$

Thus, the objective function in (21) is convex, as well as both the constraints in $\mathbf{P3}$ are convex. As a result, the optimization problem $\mathbf{P3}$ is convex, which proves the Lemma 1.

APPENDIX D
PROOF OF THEOREM 3

According to the objective function (21) and constraint (16b), the Lagrange function can be expressed as

$$\mathcal{L}(l_t^k,\lambda)=\sum_{k\in\mathcal{K}_t}\left(1-\frac{T^{\max}-\frac{z}{R_t^k}}{\frac{\tau c_k b_k}{f_k}+\frac{J(q_t^k+1)}{R_t^k}}+\beta\frac{\delta_{t,k}^2}{(2^{q_t^k}-1)^2}\right)$$
$$+\lambda\left(\sum_{k\in\mathcal{K}_t}l_t^k-1\right), \tag{47}$$

where λ is the Lagrange multiplier. Then, the KKT conditions can be written as

$$\frac{\partial\mathcal{L}}{\partial l_t^k}=\lambda-\frac{T^{\max}J(q_t^k+1)+\frac{\tau c_k b_k}{f_k}z}{R_t^k\frac{\tau c_k b_k}{f_k}+J(q_t^k+1)}B\log_2\left(1+\frac{h_t^kp_t^k}{N_0}\right)=0, \tag{48}$$

$$\lambda \left(\sum_{k \in \mathcal{K}_t} l_t^k - 1 \right) = 0, \quad \lambda \geq 0. \tag{49}$$

Based on the KKT conditions, the optimal bandwidth allocation can be derived by Theorem 3, which proves the thesis.

REFERENCES

[1] T. Zhang, K.-Y. Lam, J. Zhao, and J. Feng, "Joint device scheduling and bandwidth allocation for federated learning over wireless networks," *IEEE Trans. Wirel. Commun.*, vol. 24, no. 1, pp. 3–18, 2025.

[2] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proc. 20th Int. Conf. Artif. Intell. Statist. (AISTATS)*, 2017.

[3] L. Zhao, L. Cai, and W.-S. Lu, "Accelerating federated learning for edge intelligence using conjugated central acceleration with inexact global line search," *IEEE Trans. Cogn. Commun. Netw.*, vol. 11, no. 2, pp. 1244–1257, 2025.

[4] A. Ali and A. Arafa, "Delay sensitive hierarchical federated learning with stochastic local updates," *IEEE Trans. Cogn. Commun. Netw.*, vol. 11, no. 5, pp. 3412–3424, 2025.

[5] S. Liu, Y. Shen, J. Yuan, C. Wu, and R. Yin, "Storage-aware joint user scheduling and bandwidth allocation for federated edge learning," *IEEE Trans. Cogn. Commun. Netw.*, vol. 11, no. 1, pp. 581–593, 2025.

[6] J. Zhang, F. Huang, S. Zhu, and X. Xiao, "A resource allocation strategy in internet of vehicles based on multi-task federated learning and incentive mechanism," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–12, 2025.

[7] J. Chen, X. Zhu, C. Chakraborty, M. Guduri, A. Alharbi, A. Tolba, and K. Yu, "Dynamic optimization of vehicle production planning in transportation networks using federated reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–13, 2025.

[8] 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. Wirel. Commun.*, vol. 20, no. 1, pp. 453–467, 2021.

[9] 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. Wirel. Commun.*, vol. 20, no. 1, pp. 269–283, 2021.

[10] M. Zhang, G. Zhu, S. Wang, J. Jiang, Q. Liao, C. Zhong, and S. Cui, "Communication-efficient federated edge learning via optimal probabilistic device scheduling," *IEEE Trans. Wirel. Commun.*, vol. 21, no. 10, pp. 8536–8551, 2022.

[11] D. Wen, K.-J. Jeon, and K. Huang, "Federated dropout—a simple approach for enabling federated learning on resource constrained devices," *IEEE Wirel. Commun. Lett.*, vol. 11, no. 5, pp. 923–927, 2022.

[12] N. Bouacida, J. Hou, H. Zang, and X. Liu, "Adaptive federated dropout: Improving communication efficiency and generalization for federated learning," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, 2021, pp. 1–6.

[13] X. Jiao, G. Zhu, W. Jiang, L. Chen, W. Luo, and D. Wen, "Sensing–communication–computation integration for federated edge learning with controllable model dropout," *IEEE Internet Things J.*, vol. 12, no. 12, pp. 19 767–19 781, 2025.

[14] S. Xie, D. Wen, C. You, Q. Chen, M. Bennis, and K. Huang, "Fedlodrop: Federated lora with dropout for generalized llm fine-tuning," *IEEE J. Sel. Areas Commun.*, vol. 44, pp. 3541–3556, 2026.

[15] 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 Netw. Learn. Syst.*, vol. 34, no. 12, pp. 10 374–10 386, 2023.

[16] Z. Jiang, Y. Xu, H. Xu, Z. Wang, J. Liu, Q. Chen, and C. Qiao, "Computation and communication efficient federated learning with adaptive model pruning," *IEEE Trans. Mob. Comput.*, vol. 23, no. 3, pp. 2003–2021, 2024.

[17] Z. Chen, W. Yi, H. Shin, and A. Nallanathan, "Adaptive model pruning for communication and computation efficient wireless federated learning," *IEEE Trans. Wirel. Commun.*, vol. 23, no. 7, pp. 7582–7598, 2024.

[18] S. Liu, G. Yu, R. Yin, J. Yuan, L. Shen, and C. Liu, "Joint model pruning and device selection for communication-efficient federated edge learning," *IEEE Trans. Commun.*, vol. 70, no. 1, pp. 231–244, 2022.

[19] C. Chen, B. Jiang, S. Liu, C. Li, C. Wu, and R. Yin, "Efficient federated learning in resource-constrained edge intelligence networks using model compression," *IEEE Trans. Veh. Technol.*, vol. 73, no. 2, pp. 2643–2655, 2024.

[20] X. Liu, S. Wang, Y. Deng, and A. Nallanathan, "Adaptive federated pruning in hierarchical wireless networks," *IEEE Trans. Wirel. Commun.*, vol. 23, no. 6, pp. 5985–5999, 2024.

[21] M. F. Pervej, R. Jin, and H. Dai, "Hierarchical federated learning in wireless networks: Pruning tackles bandwidth scarcity and system heterogeneity," *IEEE Trans. Wirel. Commun.*, vol. 23, no. 9, pp. 11 417–11 432, 2024.

[22] S. Zheng, C. Shen, and X. Chen, "Design and analysis of uplink and downlink communications for federated learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 7, pp. 2150–2167, 2021.

[23] Y. Wang, Y. Xu, Q. Shi, and T.-H. Chang, "Quantized federated learning under transmission delay and outage constraints," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 323–341, 2022.

[24] P. S. Bouzinis, P. D. Diamantoulakis, and G. K. Karagiannidis, "Wireless quantized federated learning: A joint computation and communication design," *IEEE Trans. Commun.*, vol. 71, no. 5, pp. 2756–2770, 2023.

[25] S. Wang, M. Chen, C. G. Brinton, C. Yin, W. Saad, and S. Cui, "Performance optimization for variable bitwidth federated learning in wireless networks," *IEEE Trans. Wirel. Commun.*, vol. 23, no. 3, pp. 2340–2356, 2024.

[26] W. Xu, W. Fang, Y. Ding, M. Zou, and N. Xiong, "Accelerating federated learning for iot in big data analytics with pruning, quantization and selective updating," *IEEE Access*, vol. 9, pp. 38 457–38 466, 2021.

[27] P. Prakash, J. Ding, R. Chen, X. Qin, M. Shu, Q. Cui, Y. Guo, and M. Pan, "Iot device friendly and communication-efficient federated learning via joint model pruning and quantization," *IEEE Internet Things J.*, vol. 9, no. 15, pp. 13 638–13 650, 2022.

[28] J. Zhang, W. Ni, and D. Wang, "Federated split learning with model pruning and gradient quantization in wireless networks," *IEEE Trans. Veh. Technol.*, vol. 74, no. 4, pp. 6850–6855, 2025.

[29] P. Molchanov, A. Mallya, S. Tyree, I. Frosio, and J. Kautz, "Importance estimation for neural network pruning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 11 256–11 264.

[30] X. Liu, T. Ratnarajah, M. Sellathurai, and Y. C. Eldar, "Adaptive model pruning and personalization for federated learning over wireless networks," *IEEE Trans. Signal Process.*, vol. 72, pp. 4395–4411, 2024.

[31] X. Zhou, Y. Deng, H. Xia, S. Wu, and M. Bennis, "Time-triggered federated learning over wireless networks," *IEEE Trans. Wirel. Commun.*, vol. 21, no. 12, pp. 11 066–11 079, 2022.

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

[33] D. Wen, M. Bennis, and K. Huang, "Joint parameter-and-bandwidth allocation for improving the efficiency of partitioned edge learning," *IEEE Trans. Wirel. Commun.*, vol. 19, no. 12, pp. 8272–8286, 2020.

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

[35] R. Ye, M. Xu, J. Wang, C. Xu, S. Chen, and Y. Wang, "FedDisco: Federated learning with discrepancy-aware collaboration," in *Proc. 40th Int. Conf. Mach. Learn. (ICML)*, vol. 202. PMLR, 23–29 Jul 2023, pp. 39 879–39 902.

[36] B. Fehrman, B. Gess, and A. Jentzen, "Convergence rates for the stochastic gradient descent method for non-convex objective functions," *J. Mach. Learn. Res.*, vol. 21, no. 1, Jan. 2020.

[37] S. Burer and A. N. Letchford, "Non-convex mixed-integer nonlinear programming: A survey," *Surv. Oper. Res. Manage. Sci.*, vol. 17, no. 2, pp. 97–106, 2012.

[38] Y. Yang, M. Pesavento, Z.-Q. Luo, and B. Ottersten, "Inexact block coordinate descent algorithms for nonsmooth nonconvex optimization," *IEEE Trans. Signal Process.*, vol. 68, pp. 947–961, 2020.

[39] Z. Xiao, S. Cao, L. Zhu, Y. Liu, B. Ning, X.-G. Xia, and R. Zhang, "Channel estimation for movable antenna communication systems: A framework based on compressed sensing," *IEEE Trans. Wirel. Commun.*, vol. 23, no. 9, pp. 11 814–11 830, 2024.

[40] C. Ji, J. Dai, X.-Q. Jiang, and W. Xu, "Auxiliary bayesian learning approach for joint channel estimation and data detection with ris-assisted ofdm systems," *IEEE Trans. Wirel. Commun.*, pp. 1–1, 2025.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778.

[42] Y. Shi, Z. Liu, Z. Shi, and H. Yu, "Fairness-aware client selection for federated learning," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, 2023, pp. 324–329.

[43] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, "Sparsified sgd with memory," 2018.