# Energy Efficient Federated Learning over Heterogeneous Mobile Devices via Joint Design of Weight Quantization and Wireless Transmission

Rui Chen, *Student Member, IEEE*, Liang Li, *Member, IEEE*, Kaiping Xue, *Senior Member, IEEE*, Chi Zhang, *Member, IEEE*, Miao Pan, *Senior Member, IEEE*, and Yuguang Fang, *Fellow, IEEE*

**Abstract**—Federated learning (FL) is a popular collaborative distributed machine learning paradigm across mobile devices. However, practical FL over resource constrained mobile devices confronts multiple challenges, e.g., the local on-device training and model updates in FL are power hungry and radio resource intensive for mobile devices. To address these challenges, in this paper, we attempt to take FL into the design of future wireless networks and develop a novel joint design of wireless transmission and weight quantization for energy efficient FL over mobile devices. Specifically, we develop flexible weight quantization schemes to facilitate on-device local training over heterogeneous mobile devices. Based on the observation that the energy consumption of local computing is comparable to that of model updates, we formulate the energy efficient FL problem into a mixed-integer programming problem where the quantization and spectrum resource allocation strategies are jointly determined for heterogeneous mobile devices to minimize the overall FL energy consumption (computation + transmissions) while guaranteeing model performance and training latency. Since the optimization variables of the problem are strongly coupled, an efficient iterative algorithm is proposed, where the bandwidth allocation and weight quantization levels are derived. Extensive simulations are conducted to verify the effectiveness of the proposed scheme.

**Index Terms**—Federated learning over mobile devices, Weight quantization, Device heterogeneity

## 1 Introduction

Due to the incredible surge of mobile data and the growing computing capabilities of mobile devices, it becomes a trend to apply deep learning (DL) on these devices to support fast responsive and customized intelligent applications. Recently, federated learning (FL) has been regarded as a promising DL solution to providing an efficient, flexible, and privacy-preserving learning framework over a large number of mobile devices. Under the FL framework [1], each mobile device executes model training locally and then transmits the model updates, instead of raw data, to an FL server. The server will then aggregate the local models to obtain the global model and broadcast it to the participating devices. Its potential has prompted wide applications in various domains such as keyboard predictions [2], physical hazards detection in smart home [3], health event detection [4], and vehicular networks [5]. Unfortunately, it also faces many significant challenges when deploying FL over mobile devices in practice. First, although mobile devices are gradually equipped with artificial intelligence (AI) computing capabilities, the limited resources (e.g., battery power, computing and storage capacity) restrain them from training deep and complicated learning models at scale. Second, it is unclear how to establish an effective wireless network architecture to support FL over mobile devices. Finally, the power-hungry local computing and wireless communications during iterations in FL may be too much for the power-constrained mobile devices to afford.

intermediate results and broadcast the updated model

The mismatch between the computing and storage requirements of DL models and the limited resources of mobile devices becomes even more challenging due to the increasing complexity of the state-of-art DL models. To address this issue, one of the most popular solutions is to compress trained DL models [6]–[8]. Han et al. [7] successfully applied multiple compression methods, e.g., pruning and quantization, to several large-scale neural networks (e.g., AlexNet and VGG-16). These compression techniques help reduce model complexity by multiple orders of magnitude and speed up model inference on mobile devices. However, on-device training is less explored and more complicated than its inference counterpart. Some pioneering works [9], [10] have made efforts on quantizing the model parameters to make it possible to conduct computationally efficient on-device training. Nevertheless, most existing compressed on-device learning frameworks and the associated convergence analysis for the potential on-device training only consider the case of a single mobile device. A few works, such as [11], have considered quantized on-device training in distributed learning settings. However, they assign the same quantization strategy for different mobile devices.

- *R. Chen and M. Pan are with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX, 77204 (e-mail: rchen19@uh.edu, mpan2@uh.edu).*
- *L. Li is with the School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, P. R. China (e-mail: liliang1127@bupt.edu.cn).*
- *K. Xue is with the School of Cyber Security, and C. Zhang is with the School of Information Science and Technology, University of Science and Technology of China, China, Hefei, 230027, P. R. China (e-mail: kpxue@ustc.edu.cn, chizhang@ustc.edu.cn).*
- *Y. Fang is with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, China (e-mail: my.fang@cityu.edu.hk).*

.

In practice, FL may encompass massively distributed mobile devices that are highly heterogeneous in computing capability and communication conditions. Thus, it is in dire need to develop a flexible quantization scheme catering to the heterogeneous devices and investigate the impacts of such heterogeneity on learning performance.

Besides the on-device training for local computing, the energy consumption for FL over mobile devices also includes the wireless communications for the intermediate model updates. Particularly, with the advance of computing hardware and future wireless communication techniques, like 5G and beyond (5G+) [12], we have observed that the energy consumption for local computing in FL is comparable to that for the wireless transmissions on mobile devices. For instance, the energy consumption of local computing (e.g., 42.75J for one Tesla P100 GPU of one training iteration for Alexnet with batch size of 128) is comparable to that of today's wireless communications (e.g., 38.4J for transmitting 240MB Alexnet model parameters at 100 Mbps data rate [13]). Thus, a viable design of the energy efficient FL over mobile devices has to consider the energy consumption of both "working" (i.e., local computing) and "talking" (i.e., wireless communications). However, most existing works in wireless communities have mainly conducted the radio resource allocation under the FL convergence constraints [14]–[16], while neglecting the energy consumption during learning. Moreover, among the previous works, the targeted learning models are either relatively simple (i.e., with convex loss functions) or shallow networks [14]–[17], which is inconsistent with the current trend of the overparameterized DL models. On the other hand, most efforts in the machine learning communities have focused on communication efficient FL algorithmic designs, such as compressing the size of the model updates or reducing the update frequency during the training phase. The basic assumption is that the wireless transmission is slow, which results in the bottleneck to support complicated learning models over mobile devices. Therefore, the goal of such designs is to reduce the number of communications in model updates without considering the advance of wireless transmissions.

Fortunately, the future wireless transmissions (e.g., 5G/6G cellular, WiFi-6 or future version of WiFi), featured by very high data rate (1 Gbps or more [12]) with ultra low latency of 1 ms or less for massive number of devices, can be leveraged to relieve the communication bottleneck with proper design. Furthermore, the multi-access edge computing in the future networks enhances the computing capabilities at the edge, and hence provides an ideal architecture to support viable FL.

Motivated by the aforementioned challenges (i.e., inefficient on-device training and large overall energy consumption in FL training), in this paper, we develop a wireless transmission and on-device weight quantization co-design for energy efficient FL over heterogeneous mobile devices. We aim to 1) facilitate efficient on-device training on heterogeneous local devices via a flexible quantization scheme, and 2) minimize the overall energy consumption during the FL learning process by considering the learning performance and training latency. Based on the derived convergence analysis, we formulate the energy minimization problem to determine the optimal strategy in term of local iterations, quantization levels, and bandwidth allocations. Our major contributions are summarized as follows.

- We propose a novel efficient FL scheme over mobile devices to reduce the overall energy consumption in communication and computing. Briefly, subject to their current computing capacities, the participating mobile devices are allowed to compress the model and compute the gradients of the compressed version of the models. Meanwhile, for a given training time threshold, the network resource allocation is to minimize the total computing and communication energy cost during FL training.
- To facilitate on-device training for FL over heterogeneous mobile devices, weight quantization is adapted to meet the resource demands while maintaining the model performance by representing model parameters with lower bit-widths. We further provide the theoretical analysis of the convergence rate of FL with quantization and obtain a closed-form expression for the novel convergence bound in order to explore the relationship between the weight quantization error, and the performance of the FL algorithm.
- Based on the obtained theoretical convergence bound, the energy minimization during FL training is formulated as a mixed-integer nonlinear problem to balance the computing and communication costs by jointly determining the bandwidth allocation and weight quantization levels for each mobile device. An efficient iterative algorithm is proposed with low complexity, in which we derive new closed-form solutions to determining the bandwidth allocation and weight quantization levels.
- We evaluate the performance of our proposed solution via extensive simulations using various open datasets and models to verify the effectiveness of our proposed scheme. Compared with existing schemes, our proposed method shows significant superiority in terms of energy efficiency for FL over heterogeneous devices.

The rest of this paper is organized as follows. The related work is discussed in Section 2. In Section 3, a detailed description of the system model is presented and the convergence analysis of the proposed FL with weight quantization is also discussed. The energy minimization, joint quantization selection, and bandwidth allocation algorithm are presented in Section 4. In Section 5, the feasible solutions from the real datasets are analyzed. The paper is concluded in Section 6.

## 2 Related Work

### 2.1 Cost-efficient design for FL over wireless networks

Recognizing that training large-scale FL models over mobile devices can be both time and energy consuming, several research efforts have been made on decreasing these costs via device scheduling [18], network optimization [17] and resource utilization optimization [15], [19]–[24]. In particular, the resource allocation for optimizing overall FL energy efficiency was studied in [21]–[24]. Mo et al. in [23] have designed the computing and communication resources allocation to minimize the energy consumption while only considering the CPU models for mobile devices. Zeng et al. [21] proposed to partition the computing workload between CPU-GPU to improve the computing energy efficiency. However, their resource allocation strategies are for particular (non-optimal) model parameters (i.e., weight quantization levels in this paper). Thus, they overlook the opportunities to first reduce the costs in learning (i.e., model quantization in this paper) before utilizing the available resources. Close to our

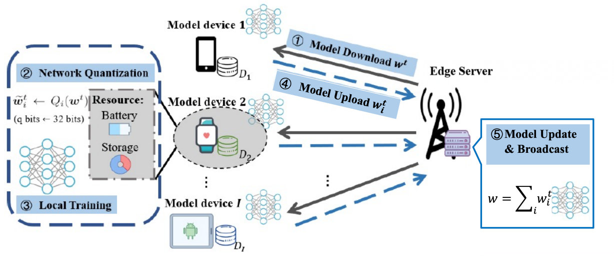


Fig. 1. Federated learning framework with weight quantization.

work, Li et al. [24] considered to sparsify the model size before transmission to improve communication efficiency and determine heterogeneity-aware gradient sparsification strategies. However, they neglect the mismatch between the computing/storage requirement for on-device training and the limited computing resources on mobile devices. Based on the example illustrated in Section 1, on-device computing consumes more energy than model update transmission. Hence, different from [24], this paper leverages the quantization method for on-device training instead of wireless transmission only.

### 2.2 On-device training with low precision

Various works have been developed for on-device learning to reduce the model complexities via low precision operation and storage requirements [25]. In the extreme case, the weights and activations are represented in one bit, called Binary Neural Networks (BNN) [26], while the performance degrades significantly in large DNNs. For weight quantization, the prior work such as "LQ-Net" in [9] quantized weights and activations such that the inner products can be computed efficiently with bit-wise operations, performing in the case of single machine computation. Similar to our work, Fu et al. [10] considered the weight quantization for local devices in the distributed learning setting and proposed to quantize activations via estimating Weibull distributions. However, they did not consider optimization for energy efficiency during FL training. Besides, they assigned the same quantization level on all participating devices, which limited the performance when facing the challenges of device heterogeneity. How the flexible quantization impact the learning model accuracy remains an open problem, which is addressed in this work. Unlike the existing works, a mobile-compatible FL algorithm with flexible weight quantization is introduced in our proposed model. By jointly considering the heterogeneous computing and communication conditions, we formulate the overall FL energy (computing + transmissions) minimization to seek for the optimal strategy in term of local iterations, quantization levels, and bandwidth allocations.

## 3 FL with Flexible Weight Quantization

### 3.1 Preliminary of Weight Quantization

In this subsection, we introduce the related concepts about weight quantization for on-device training. Quantization is an attractive solution to implementing FL models on mobile devices efficiently. It represents model parameters, including the weights, feature maps, and even gradients, with low-precision arithmetic (e.g., 8-bit fixed-point numbers). When the model parameters are stored and computed with low-bitwidth, the computational units and memory storage to perform the operations during on-device training are much smaller than the full-precision counterparts, leading to energy reduction during on-device training.

To train the FL model in low precision, we define a quantization function $Q(\cdot)$ to convert a real number $w$ into a quantized version $\hat{w} = Q(w)$. We use the same notation for quantizing vectors since $Q$ acts on each dimension of the vector independently in the same manner. Moreover, we employ stochastic rounding (SR) [8] in our proposed model and analyze its convergence properties. SR, also known as unbiased rounding, possesses the important property: $\mathbb{E}[Q(w)] = w$. This property avoids the negative effect of quantization noise, which is useful for the theory of non-convex setting [27]. For each component $w_n$ of a vector $\mathbf{w}$, the function $Q(\cdot)$ converts the data type from 32-bit into $q$-bit, defined as:

$$Q(w_n) = s \cdot \mathrm{sgn}(w_n) \cdot \begin{cases} I_{a+1}, & w.p.\ \frac{|w_n|}{s\Delta_q} - \frac{I_a}{\Delta_q} \\ I_a, & w.p.\ \frac{I_{a+1}}{\Delta_q} - \frac{|w_n|}{s\Delta_q} \end{cases}, \qquad (1)$$

where $\mathrm{sgn}(\cdot)$ represents the sign function, $s = \|\mathbf{w}\|_\infty$ denotes the scaling factor, the index $k$ satisfies $I_a \leq \frac{|w_n|}{s} \leq I_{a+1}$, quantization set $\mathcal{S}_w = \{-I_A, \cdots, I_0, \cdots, I_A\}$ with $A = 2^{q-1} - 1, 0 = I_0 \leq I_1 \leq \cdots \leq I_A$ are uniformly spaced, and $\Delta_q$ denotes the quantization resolution as $\Delta_q = I_{a+1} - I_a = 1/(2^q - 1)$. According to the definition in (1), we have the following lemma [28].

**Lemma 1** (Weight quantization error in SR [28])**.** *For model weight $\mathbf{w} \in R^d$ satisfying $\|\mathbf{w}\|_\infty = s$, let $\mathbf{w}^{r+1} = \mathbf{w}^r - \eta\widetilde{f}(\mathbf{w}^r)$ be the SGD update for a single iteration $r$ and $Q(\mathbf{w}^{r+1}, q)$ be the stochastic quantization scheme of $\mathbf{w}^{r+1}$ in (1), quantization level $q$, and the learning rate $\eta$. The weight quantization error on each iteration can be bounded, in expectation, as follows,*

$$\mathbb{E}_Q\left[\left\|Q\left(\mathbf{w}^{r+1}, q\right) - \mathbf{w}^{r+1}\right\|_2^2\right] \leq \eta\sqrt{d}s\Delta_q\left\|\widetilde{f}(\mathbf{w}^r)\right\|_2. \qquad (2)$$

In the above, smaller resolution results in a smaller gap and keeps as much information as the original weight, while it has higher memory requirements. In practice, the bit-width for the weight quantization can be extremely small, like 2 or 3 bits without notable performance degradation. Other parameters, such as the weight gradient calculations and updates, are applied to capture accumulated small changes in stochastic gradient descent (SGD). In contrast, quantization makes them insensitive to such information and may impede convergence performance during training. Therefore, we keep a higher precision for the gradients than the weights and inputs so that the edge server aggregates the local gradients and updates the global model in full precision.

### 3.2 FL with flexible weight quantization

We consider a mobile edge network consisting of one edge server and a set $\mathcal{N} = \{1, 2, \cdots, N\}$ of distributed mobile devices, collaboratively training a DNN model through FL framework, which is depicted in Fig. 1. Each mobile device $i$ is equipped with a single antenna and has its own dataset $\mathcal{D}_i$ with data size $|\mathcal{D}_i|$. The data is collected locally by the mobile device $i$ itself. Generally, each learning model has a particular loss function $f_j(\mathbf{w})$ with the parameter vector $\mathbf{w}$ for each data sample $j$. The loss function represents the difference of the model prediction and groundtruth of the training data. Thus, the loss function on the local data of mobile devices $i$ is given as $F_i(\mathbf{w}) := \frac{1}{|\mathcal{D}_i|}\sum_{j=1}^{|\mathcal{D}_i|} f_j(\mathbf{w})$. The training objective of the shared

**Algorithm 1** Flexible Weight Quantizated FL (FWQ-FL)

```
Input: $\eta$ = learning rate; $Q(\cdot)$ = quantization function; initial $\bar{\mathbf{w}}^0$; a mini-batch size $M$; a number of local SGD iterations $H$; a number of training iterations $R$
Output: $\bar{\mathbf{w}}^R$
for $r = 0, \cdots, R-1$ do
    Edge server sends $\bar{\mathbf{w}}^r$ to the set of participating mobile devices $\mathcal{N}$
    for each mobile device $i \in \mathcal{N}$ in parallel do
        Sample a mini-batch of $M$ training data points from $\mathcal{D}_i$
        Compute the mini-batch stochastic gradient
        $\mathbf{g}_i^r = \frac{1}{M}\sum_{m=1}^{M} \triangledown f_m(\mathbf{w}_i^r)$
        Update the model parameters
        $\mathbf{w}_i^{r+1} \leftarrow Q\left(\mathbf{w}_i^r - \eta_i \mathbf{g}_i^r\right)$
        if $((r+1) \mod H) = 0$ then
            Send $\mathbf{w}_i^{r+1}$ to the FL server.
        end if
    end for
    Edge server updates the global model $\bar{\mathbf{w}}^{r+1}$ as follows
    if $((r+1) \mod H) = 0$ then
        $\bar{\mathbf{w}}^{r+1} \leftarrow \sum_{i=1}^{N} \pi_i \mathbf{w}_i^{r+1}$
    else
        $\bar{\mathbf{w}}^{r+1} \leftarrow \bar{\mathbf{w}}^r$
    end if
end for
```

model is to collaboratively learn from all the participating mobile devices, formulated as follows:

$$\min_{\mathbf{w} \in \mathbb{R}^d} F(\mathbf{w}) = \sum_{i=1}^{N} \pi_i F_i(\mathbf{w}), \tag{3}$$

where $d$ denotes the total number of the DNN model parameters and $\pi_i$ is the weight of the $n$-th device such that $\pi_i = |\mathcal{D}_i| / \sum_{i=1}^{N} |\mathcal{D}_i|$ and $\sum_{i=1}^{N} \pi_i = 1$.

Given the sensitive nature of the users' data, each mobile device keeps its data locally instead of uploading its data to the edge server. An FL framework [1] is adopted to solve problem (3), named FedAvg, that allows the users to update the model to the edge server periodically. Let $r$ be the $r$-th training iteration in FL. In FedAvg, the edge server first broadcasts the latest model $\bar{\mathbf{w}}^r$ to all the devices. Second, every device $i \in \mathcal{N}$ performs $H$ mini-batch SGD steps in parallel, obtains and transmits its intermediate local model $\mathbf{w}_i^{r+H}$ to the edge server. After that, the edge server will update the model based on aggregated results from the mobile devices, i.e., $\bar{\mathbf{w}}^{r+H} = \sum \pi_i \mathbf{w}_i^{r+H}$. This procedure repeats until FL converges.

Targeting at the energy-efficient FL training over mobile devices, we propose a flexible weight quantization (FWQ) scheme for heterogeneous mobile devices. After mobile devices receive the shard model from the edge server, they first quantize and store the model to satisfy their current storage budget. Unlike the prior works that maintain the same quantization strategy across all the participating devices, FWQ considers device heterogeneity and allows the mobile devices to perform weight quantization with different bit-widths of $q_i$ during on-device training and transmit the model updates in more bits. Note that the weights and gradients at the server side remain in full precision operations to avoid further model performance degradation. A pseudo-code of our FWQ algorithm is presented in Alg. 1.

### 3.3 Convergence Analysis of FL with FWQ

Before we discuss the convergence of Alg. 1, we make the following assumptions on the loss function, which are commonly used for the analysis of SGD approach under the distributed/federated learning settings [29], [30].

**Assumption 1.** *All the loss functions $f_j$ are differentiable and their gradients are L-Lipschitz continuous in the sense of $l_2$-norm: for any $x$ and $y \in \mathbb{R}^d$, $\|\triangledown f_j(x) - \triangledown f_j(y)\|_2 \le L\|x - y\|_2$.*

**Assumption 2.** *Assume that $\widetilde{f}_i$ is randomly sampled from the $i$-th mobile device local loss functions. For local device $i$, its stochastic gradient is an unbiased estimator and its variance: $\mathbb{E}||\triangledown \widetilde{f}_i(\mathbf{w}^r) - \triangledown F_i(\mathbf{w}^r)||_2^2 \le \tau_i^2$. Thus, the a mini-batch size $M$ of gradient variance is given as $\tau_i^2/M$ and its second moment is $\mathbb{E}\left\|\triangledown \widetilde{f}_i(\boldsymbol{w}^r)\right\|_2^2 \le G_i^2$, for any $i = 1, \cdots, N$ and define $G = \max_i\{G_i\}$.*

Assumption 1 indicates that the local loss functions $F_i$ and the aggregated loss function $F$ are also $L$-smooth. The unbiasedness and bounded variance of stochastic gradients in Assumption 2 are customary for non-convex analysis of SGD [29]–[33].

For the case of non-convex loss function $F_i$, the algorithm may have multiple stable fixed points. Hence, convergence to a global minimum cannot in general be guaranteed. A reasonable substitute is to study the convergence to local minima, or at the very least, to stationary points [32], [34]. Hence, similar to previous work [30], [35], we use the relationship between average expected squared gradient norm and the iteration number to characterize the convergence rate of FL with FWQ.

From the updating rule of Alg. 1, we use the following notation to denote the stochastic gradient used to update the local model and global model at the $r$-th iteration:

$$\mathbf{u}_i^{r+1} = Q_i\left(\mathbf{w}_i^r - \eta \mathbf{g}_i^r, q_i\right) = \mathbf{w}_i^r - \eta \mathbf{g}_i^r + \mathbf{e}_i^r, \tag{4}$$

$$\bar{\mathbf{w}}^{r+1} = \kappa_r \sum_{i=1}^{N} \pi_i \mathbf{u}_i^{r+1} + (1 - \kappa_r)\bar{\mathbf{w}}^r, \tag{5}$$

where $\mathbf{e}_i^r = Q_i\left(\mathbf{w}_i^r - \eta \mathbf{g}_i^r, q_i\right) - \left(\mathbf{w}_i^r - \eta \mathbf{g}_i^r\right)$ denotes the quantization error and the indicator $\kappa_r = 1$ if $(r+1) \mod H = 0$ and $\kappa_r = 0$ otherwise. $\mathbf{u}_i^{r+1}$ is introduced to represent the immediate result of one step SGD update with quantization from $\mathbf{w}_i^r$. We can access $\mathbf{w}^{r+1}$ only when $(r+1) \mod H = 0$. Thus, we have a virtual sequence $\bar{\mathbf{u}}^{r+1} = \bar{\mathbf{w}}^r - \eta \bar{\mathbf{g}}^r$ and $\mathbb{E}[\mathbf{g}_i^r] = \triangledown F_i(\mathbf{w}_i^r)$[1].

In the following, we establish the upper bound of the differential of loss values between two consecutive iterations. We first derive the upper bounds of model weight differential between two consecutive iterations and the model divergence in one iteration, shown as the following two lemmas.

**Lemma 2** (Bounding the model divergence)**.** *Let Assumption 1-2 hold and the learning rate $\eta$ satisfying $1 - 3\eta^2 L^2 H^2 > 0$, we have,*

$$\sum_{i=1}^{N} \pi_i^2 \mathbb{E}\left[\|\bar{\mathbf{w}}^r - \mathbf{w}_i^r\|_2^2\right] \le \frac{\eta H \sum_{i=1}^{N} \pi_i^2 \left(\eta H \tau_i^2/M + \sqrt{d} G \delta_i + 3\eta H \|\triangledown F(\bar{\mathbf{w}}^r)\|_2^2\right)}{1 - 3\eta^2 L^2 H^2}. \tag{6}$$

[1] For convenience, we define $\bar{\mathbf{u}}^r = \sum_{i=1}^{N} \pi_i \mathbf{u}_i^r$, $\bar{\mathbf{w}}^r = \sum_{i=1}^{N} \pi_i \mathbf{w}_i^r$, and $\bar{\mathbf{g}}^r = \sum_{i=1}^{N} \pi_i (\mathbf{g}_i^r - \mathbf{e}_i^r/\eta)$.

*Proof.* Please refer to the detailed proof in Appendix A in the separate supplemental file. □

**Lemma 3.** *If Assumptions 1 and 2 hold, then for any iteration $r$, we have*

$$\frac{L}{2}\mathbb{E}\left[\left\|\bar{\mathbf{w}}^{r+1}-\bar{\mathbf{w}}^{r}\right\|_2^2\right] \leq \frac{\eta L}{2}\sum_{i=1}^{N}\pi_i^2\left(\frac{\eta\tau_i^2}{M}+\sqrt{d}\delta_i G_i\right)+\frac{\eta^2 L}{2}\|\sum_{i=1}^{N}\pi_i\nabla F_i(\mathbf{w}_i^r)\|_2^2. \quad (7)$$

*Proof.* Please refer to the detailed proof in Appendix B in the separate supplemental file. □

Now we are ready to show the convergence property of FL with FWQ.

**Theorem 1.** *Let the learning rate $\eta$ be $\sqrt{\frac{M}{R}}$ and $\eta L \leq \frac{1}{3H}$. If Assumptions 1-2 hold, the average-squared gradient after $R$ iterations is bounded as follow,*

$$\frac{1}{R}\sum_{t=0}^{R-1}\mathbb{E}\left\|\nabla F(\bar{\mathbf{w}}^r)\right\|_2^2 \leq \frac{4(\mathbb{E}\left[F(\bar{\mathbf{w}}^0)\right]-F^\star)}{\sqrt{MR}}+\frac{4HL\tau}{\sqrt{MR}}+4\sqrt{d}LG\sum_{i=1}^{N}\pi_i^2\delta_i, \quad (8)$$

$$\leq \mathcal{O}(\frac{H+1}{\sqrt{MR}})+\mathcal{O}(\sqrt{d}\sum_{i=1}^{N}\pi_i^2\delta_i), \quad (9)$$

*where $\delta_i = s\Delta_{q_i}$, $\tau = \sum_{i=1}^N \pi_i^2\tau_i^2$, and $F^\star$ is the global minimum of F.*

*Proof.* According to the update rules in (5), we have

$$F\left(\bar{\mathbf{w}}^{r+1}\right)=F\left(\bar{\mathbf{w}}^{r}+\kappa_r\left(\bar{\mathbf{u}}^{r+1}-\bar{\mathbf{w}}^{r}\right)\right). \quad (10)$$

Under the Lipschitz gradient assumption on $F$, we have,

$$\begin{aligned}&\mathbb{E}\left[F(\bar{\mathbf{w}}^{r+1})\right]-\mathbb{E}\left[F(\bar{\mathbf{w}}^{r})\right]\\ &\leq \mathbb{E}\left[\langle\nabla F(\bar{\mathbf{w}}^r),\bar{\mathbf{w}}^{r+1}-\bar{\mathbf{w}}^r\rangle\right]+\frac{L}{2}\mathbb{E}\left[\left\|\bar{\mathbf{w}}^{r+1}-\bar{\mathbf{w}}^r\right\|_2^2\right]\\ &=\mathbb{E}\left[\langle\nabla F(\bar{\mathbf{w}}^r),-\eta\kappa_r\bar{\mathbf{g}}^r\rangle\right]+\frac{L}{2}\mathbb{E}\left[\left\|\bar{\mathbf{w}}^{r+1}-\bar{\mathbf{w}}^r\right\|_2^2\right]\\ &\overset{(a)}{=}\mathbb{E}\left[\langle\nabla F(\bar{\mathbf{w}}^r),-\sum_{i=1}^{N}\pi_i\nabla F_i(\mathbf{w}_i^r)\rangle+\frac{L}{2}\mathbb{E}\left[\left\|\bar{\mathbf{w}}^{r+1}-\bar{\mathbf{w}}^r\right\|_2^2\right]\right],\end{aligned} \quad (11)$$

s where in $(a)$ $\mathbb{E}_{\xi,Q}[\bar{\mathbf{g}}_i^r] = \mathbb{E}_{\xi,Q}[\sum_{i=1}^N \pi_i(\mathbf{g}_i^r - \mathbf{e}_i^r/\eta)] = \sum_{i=1}^N \pi_i\nabla F_i(\mathbf{w}_i^r)$ due to the unbiasedness of weight quantization scheme and SGD. The second term of (11) is bounded by Lemma 3. Now, we need to derive the expectation of the first term in (11).

$$\begin{aligned}&\mathbb{E}\left[\left\langle\nabla F(\bar{\mathbf{w}}^r),-\sum_{i=1}^{N}\pi_i\nabla F_i(\mathbf{w}_i^r)\right\rangle\right]\\ &\overset{(a)}{=}-\frac{\eta}{2}\mathbb{E}\left[\|\nabla F(\bar{\mathbf{w}}^r)\|_2^2\right]-\frac{\eta}{2}\mathbb{E}\left[\left\|\sum_{i=1}^{N}\pi_i\nabla F_i(\mathbf{w}_i^r)\right\|_2^2\right]\\ &\quad+\frac{\eta}{2}\mathbb{E}\left[\left\|\nabla F(\bar{\mathbf{w}}^r)-\sum_{i=1}^{N}\pi_i\nabla F_i(\mathbf{w}_i^r)\right\|_2^2\right]\\ &\overset{(b)}{\leq}-\frac{\eta}{2}\mathbb{E}\left[\|\nabla F(\bar{\mathbf{w}}^r)\|_2^2\right]-\frac{\eta}{2}\mathbb{E}\left[\left\|\sum_{i=1}^{N}\pi_i\nabla F_i(\mathbf{w}_i^r)\right\|_2^2\right]\\ &\quad+\frac{\eta L^2}{2}\sum_{i=1}^{N}\pi_i^2\mathbb{E}_{\xi,Q}\left[\|\bar{\mathbf{w}}^r-\mathbf{w}_i^r\|_2^2\right],\end{aligned} \quad (12)$$

where $(a)$ is due to $2\langle a,b\rangle = ||a||^2+||b||^2-||a-b||^2$ and $(b)$ follows from $L$-smoothness assumption. The last part of (12) is bounded, as shown in Lemma 2.

By associating (11) and (12), the expectation of the objective change in one step is given below

$$\begin{aligned}&\mathbb{E}\left[F(\bar{\mathbf{w}}^{r+1})\right]-\mathbb{E}\left[F(\bar{\mathbf{w}}^{r})\right]\\ &\leq-\frac{\eta}{2}\mathbb{E}\left[\|\nabla F(\bar{\mathbf{w}}^r)\|_2^2\right]+\left(-\frac{\eta}{2}+\frac{\eta^2 L}{2}\right)\left\|\sum_{i=1}^{N}\pi_i\nabla F_i(\mathbf{w}_i^r)\right\|_2^2\\ &\quad+\frac{\eta L^2}{2}\sum_{i=1}^{N}\pi_i^2\mathbb{E}_{\xi,Q}\|\bar{\mathbf{w}}^r-\mathbf{w}_i^r\|_2^2+\frac{\eta L}{2}\sum_{i=1}^{N}\pi_i^2\left(\frac{\eta\tau_i}{M}+\sqrt{d}\delta_i G_i\right)\\ &\overset{(a)}{\leq}-\frac{\eta}{2}\mathbb{E}\left[\|\nabla F(\bar{\mathbf{w}}^r)\|_2^2\right]+\frac{\eta L^2}{2}\sum_{i=1}^{N}\pi_i^2\mathbb{E}_{\xi,Q}\left[\|\bar{\mathbf{w}}^r-\mathbf{w}_i^r\|_2^2\right]\\ &\quad+\frac{\eta L}{2}\sum_{i=1}^{N}\pi_i^2\left(\frac{\eta\tau_i}{M}+\sqrt{d}\delta_i G_i\right).\end{aligned} \quad (13)$$

By replacing $\sum_{i=1}^N \pi_i^2\mathbb{E}_{\xi,Q}\left[\|\bar{\mathbf{w}}^r-\mathbf{w}_i^r\|_2^2\right]$ with the bound derived in Lemma 2, we can get

$$\begin{aligned}&\mathbb{E}\left[F(\bar{\mathbf{w}}^{r+1})\right]-\mathbb{E}\left[F(\bar{\mathbf{w}}^{r})\right]\\ &\leq-(\frac{\eta}{2}-\frac{\eta L^2}{2}\frac{3\eta^2H^2\sum_{i=1}^N\pi_i^2}{1-3\eta^2L^2H^2})\mathbb{E}\left[\|\nabla F(\bar{\mathbf{w}}^r)\|_2^2\right]\\ &\quad+\left(\frac{\eta^2L}{2}+\frac{\eta L^2}{2}\frac{\eta^2H^2}{1-3\eta^2L^2H^2}\right)\sum_{i=1}^{N}\pi_i^2\frac{\tau_i^2}{M}\\ &\quad+\left(\frac{\eta L}{2}+\frac{\eta L^2}{2}\frac{\eta H}{1-3\eta^2L^2H^2}\right)\sqrt{d}\sum_{i=1}^{N}\pi_i^2\delta_i G_i.\end{aligned} \quad (14)$$

Here, (a) holds if the learning rate $\eta L \leq 1$. Summing up for all $R$ iterations, we have:

$$\begin{aligned}&\mathbb{E}\left[F(\bar{\mathbf{w}}^{R})\right]-\mathbb{E}\left[F(\bar{\mathbf{w}}^{0})\right]\\ &\leq-\frac{\eta}{2}(1-\frac{3\eta^2L^2H^2\sum_{i=1}^N\pi_i^2}{1-3\eta^2L^2H^2})\sum_{r=0}^{R-1}\mathbb{E}\left[\|\nabla F(\bar{\mathbf{w}}^r)\|_2^2\right]\\ &\quad+R\frac{\eta^2LC_1}{2M}\sum_{i=1}^{N}\pi_i^2\tau_i^2+R\frac{\eta LC_1\sqrt{d}}{2}\sum_{i=1}^{N}\pi_i^2\delta_i G_i,\end{aligned} \quad (15)$$

where $C_1 = \frac{1+\eta LH^2-3\eta^2L^2H^2}{1-3\eta^2L^2H^2}$ and $C_2 = \frac{1+\eta LH-3\eta^2L^2H^2}{1-3\eta^2L^2H^2}$and rearranging the terms, we have

$$\begin{aligned}&\frac{\eta}{2}C_1'\sum_{r=0}^{R-1}\|\nabla F(\bar{\mathbf{w}}^r)\|_2^2\\ &\leq\mathbb{E}\left[F(\bar{\mathbf{w}}^0)-F(\bar{\mathbf{w}}^R)\right]+R\frac{LC_1}{2}\sum_{i=1}^{N}\pi_i^2(\frac{\eta^2\tau_i^2}{M}+\eta\sqrt{d}\delta_i G_i),\end{aligned} \quad (16)$$

where $C_1' = 1 - \frac{3\eta^2 L^2 H^2 \sum_{i=1}^N \pi_i^2}{1-3\eta^2 L^2 H^2}$. If we set $\eta = \sqrt{\frac{M}{R}}$ and $\frac{3\eta^2 L^2 H^2}{1-3\eta^2 L^2 H^2} \leq \frac{1}{2}$, we can get $1/C_1' \leq 2$, $C_1/C_1' \leq 4H$, and $C_2/C_1' \leq 4$. Thus,

$$\begin{aligned}
&\frac{1}{R}\sum_{r=0}^{R-1} \|\nabla F(\bar{\mathbf{w}}^r)\|_2^2 \\
&\leq \frac{2}{\eta R C_1'}(\mathbb{E}\left[F(\bar{\mathbf{w}}^0)\right] - F^\star) + \frac{\eta L C_1}{C_1' M}\tau + \frac{\sqrt{d} L C_2}{C_1'}\sum_{i=1}^N \pi_i^2 \delta_i G_i \\
&\leq \frac{4(\mathbb{E}\left[F(\bar{\mathbf{w}}^0)\right] - F^\star)}{\sqrt{MR}} + \frac{4HL\tau}{\sqrt{MR}} + 4\sqrt{d} L G \sum_{i=1}^N \pi_i^2 \delta_i, \qquad (17)
\end{aligned}$$

where $\tau = \sum_{i=1}^N \pi_i^2 \tau_i^2$ and the proof is completed. □

From Theorem 1, we observe that the proposed model admits the same convergence rate as parallel SGD in the sense that both of them attain the asymptotic convergence rate $\mathcal{O}(\frac{1}{\sqrt{MR}})$. Weight quantization makes FL converge to the neighborhood of the optimal solution without affecting the convergence rate. The limit point of the iterates is related to the quantization noise $\delta_i$. If the quantization becomes more fine-grained (i.e., by increasing the number of bits), the model performance will approach the model with full precision floating point.

# 4 Optimization for Energy Efficient FWQ

Motivated by the above discussion, the quantization levels $\{q_i\}_{i=1}^N$ and the numbers of local SGD iterations, $H$, act as critical parameters of FL training performance (i.e., model convergence rate). Besides, these strategies also greatly impact the energy consumption of mobile devices since they affect the total communication rounds and computing workload per round. In this section, we formulate the energy efficient FWQ problem (EE-FWQ) under model convergence and training delay guarantee. We develop flexible weight quantization and bandwidth allocation to make the trade-off between computing and communication energy of mobile devices in FL training. We start with discussion on the computing and communication energy model, followed by problem formulation and solution.

## 4.1 Energy Model

### *4.1.1 Computing Model*

Here, we consider the GPU computing model instead of the CPU model, for two reasons. First, CPUs cannot support relatively large and complicated model training tasks. Second, GPUs are more energy efficient than CPUs for on-device training and are increasingly integrated into today's mobile devices (e.g., Google Pixel). The GPU based training makes computing energy consumption comparable to that of communications in FL. Noted that the local computing of mobile device $i$ involves the data fetching in GPU memory modules and the arithmetic in GPU core modules, where the voltage and frequency of each module are independent and configurable:

1) *GPU runtime power model* of mobile device $i$ is modeled as a function of the core/memory voltage/frequency [36],

$$p_i^{cp} = p_i^{G0} + \zeta_i^{mem} f_i^{mem} + \zeta_i^{core} (V_i^{core})^2 f_i^{core}, \qquad (18)$$

where $p_i^{G0}$ is the summation of the power consumption unrelated to the GPU voltage/frequency scaling; $V_i^{core}, f_i^{core}, f_i^{mem}$ denote the GPU core voltage, GPU core frequency, and GPU memory frequency, respectively; $\zeta_i^{mem}$ and $\zeta_i^{core}$ are the constant coefficients that depend on the hardware and arithmetic for one training iteration, respectively.

2) *GPU execution time model* of mobile device $i$ with quantization level $q_i$ is formulated as

$$T_i^{cp}(q_i) = t_i^0 + \frac{c_1(q_i)\theta_i^{mem}}{f_i^{mem}} + \frac{c_2(q_i)\theta_i^{core}}{f_i^{core}}, \qquad (19)$$

where $t_i^0$ represents the other component unrelated to training task; $\theta_i^{mem}$ and $\theta_i^{core}$ denote the number of cycles to access data from the memory and to compute one mini-batch size of data samples, respectively, which are measured on a platform-based experiment in this paper. Due to the weight quantization, the number of cycles for data fetching and computing are reduced with scaling $c_1(q_i)$ and $c_2(q_i)$, respectively. For simplicity, we assume that the number of cycles for data fetching and computing scales, $c_1(q_i)$ and $c_2(q_i)$, are linear functions of data bit-width $q_i$, respectively. This is reasonable since the quantization reduces the bit-widths, and the data size scales linearly to the bit representation [37].

With the above GPU power and performance model, the local energy consumed to pass a single mini-batch SGD with quantization strategy $q_i$ of the $i$-th mobile device is the product of the runtime power and the execution time, i.e.,

$$E_i^{comp}(q_i, H) = H \cdot p_i^{cp} \cdot T_i^{cp}(q_i). \qquad (20)$$

### *4.1.2 Communication Model*

We consider orthogonal frequency-division multiple access (OFDMA) protocol for devices to upload their local results to the edge server. The total channel bandwidth is bounded by $B_{\max}$ and $B_i$ is denoted as the bandwidth allocated to device $i$ where $B_i$ satisfies $\sum_{i=1}^N B_i \leq B_{\max}$. As a result, the achievable transmission rate (bit/s) of mobile device $i$ can be calculated as

$$\gamma_i = B_i \ln\left(1 + \frac{h_i p_i^{cm}}{N_0}\right), \qquad (21)$$

where $N_0$ represents the noise power, and $p_i^{cm}$ is the transmission power. Here, $h_i$ denotes the average channel gain of the mobile device $i$ to the edge server during the training task of FWQ-FL. The dimension of the gradient vector $g_i$ is fixed for a given model so that the overall data size to transmit the gradient vector is the same for all the mobile devices, which is denoted by $D_g$. Here, we only consider the energy consumption of uplink transmission[2]. Then, the communication time to transmit $D_g$ for mobile device $i$ is

$$T_i^{cm}(B_i) = \frac{D_g}{\gamma_i} = \frac{D_g}{B_i \ln\left(1 + \frac{h_i p_i^{cm}}{N_0}\right)}. \qquad (22)$$

Thus, the communication energy consumption at mobile device $i$ can be derived as

$$E_i^{comm}(B_i) = \frac{D_g p_i^{cm}}{B_i \ln\left(1 + \frac{h_i p_i^{cm}}{N_0}\right)}. \qquad (23)$$

[2] First, the downlink transmission energy at the edge server side is less concerned. Seconds, during the FL training process, the mobile edge server broadcasts the global model parameters to all devices in downlinks. In typical wireless transmission environments (e.g., cellular network), downlink bandwidth is much larger than uplink bandwidth [13]. The time of global model broadcasting using downlink is much shorter than that of local model updates using uplink. Besides, the receiving power is smaller than transmission power at mobile devices [38]. Hence, the downlink energy consumption is much smaller compared with the uplink one and can be neglected.

## 4.2 Problem Formulation

Considering the computing capabilities of different mobile devices vary, we formulate the problem as minimizing the total energy consumption during the training process as

$$\min_{\substack{H,K,\epsilon_q,\\ \mathbf{q},\mathbf{B}}} \quad \sum_{i=1}^{N} K\left(E_i^{comm}(B_i) + E_i^{comp}(q_i, H)\right) \tag{24a}$$

$$\text{s.t.} \quad c_3(q_i)U_i \leq C_i, \forall i \in \mathcal{N}, \tag{24b}$$

$$A_3 \sum_{i=1}^{N} \pi_i^2 \delta_i \leq \epsilon_q, \tag{24c}$$

$$\frac{A_1 H + A_2}{\sqrt{MHK}} + A_3 \sum_{i=1}^{N} \pi_i^2 \delta_i \leq \epsilon, \tag{24d}$$

$$\max_i \quad K\left(HT_i^{cp} + T_i^{comm}\right) \leq T_{\max}, \tag{24e}$$

$$\sum_{i=1}^{N} B_i \leq B_{\max}, \tag{24f}$$

$$B_i > 0, q_i \in \mathcal{Q}, \forall i \in \mathcal{N}, \tag{24g}$$

$$H \in \mathbb{Z}^+, 0 \leq \epsilon_q \leq \epsilon, \tag{24h}$$

where $K$ represents the total number of communication rounds, $U_i$, and $C_i$ represent the learning model size (MB) stored in full precision and the memory capacity in mobile device $i$, respectively. $c_3(q_i)$ is the ratio of the bit-width to full precision. $\mathbf{q} = [q_1, \cdots, q_N]$ and $\mathbf{B} = [B_1, \cdots, B_N]$ are the quantization and bandwidth allocation strategies of mobile devices, respectively. Constraint (24b) states the model size stored on mobile device $i$ does not exceed its storage capacity. The constraint (24c) controls the average quantization error over participating devices as small as possible. The constraints in (24e) ensures the entire training time can be completed within predefined deadline $T_{\max}$. In constraint (24f), the bandwidth allocation to the mobile devices must not exceed the channel bandwidth available to the edge server. Constraints (24g) and (24h) indicate that variables take the values from a set of non-negative numbers. Bit representation set $\mathcal{Q}$ is defined as a power of 2, ranging from 8 to 32 bits, which is a standard-setting and hardware friendly [39]. The number of communication rounds $K$ is determined by the FL model convergence. Based on the results in Theorem 1, we set upper bound to satisfy the convergence constraint as in (24d), where $A_1$, $A_2$ and $A_3$ are coefficients[3] used to approximate the big-$\mathcal{O}$ in Eqn. (8). Furthermore, given the constraint (24c), we can rewrite the (24d) as

$$\frac{A_1 H + A_2}{\sqrt{MHK}} + \epsilon_q \leq \epsilon. \tag{25}$$

For the relaxed problem, if any feasible solution $H$, $\epsilon_q$, and $K$ satisfies constraint (25) with inequality, we note that the objective function is a decreasing function of $K$. Thus, for optimal $K$, the constraint (25) is always satisfied with equality, and we can derive $K$ from this equality as

$$K(H, \epsilon_q) = \frac{(A_1 H + A_2)^2}{MH(\epsilon - \epsilon_q)^2}, \tag{26}$$

From (26), we observe that $K(H, \epsilon_q)$ is a function of $H$ that first decreases and then increases, which implies that too small and too large $H$ all lead to high communication cost and that there exists an optimal $H$. Besides, a large $\epsilon_q$, which results from aggressive quantization levels (small bit-widths), also hinders the learning efficiency since it requires more communication rounds to recover the learning accuracy. In light of this, local update and weight quantization levels should be carefully determined to minimize the overall energy consumption for FWQ-FL.

For the ease of analysis, we simplify the description of the GPU time model as a linear function of $q_i$, i.e., $T_i^{cp}(q_i) = c_i^2 q_i + c_i^1$. By substituting (26) into its expression, we obtain

$$\min_{\substack{H,\epsilon_q,\\ \mathbf{q},\mathbf{B}}} \quad \sum_{i=1}^{N} \frac{(A_1 H + A_2)^2}{MH(\epsilon - \epsilon_q)^2} \left(\frac{p_i^{cm} D_g}{\gamma_i} + Hp_i^{cp}(c_i^2 q_i + c_i^1)\right)$$
$$\text{s.t.} \quad (24b) - (24h). \tag{27}$$

The relaxed problem above is a mixed-integer non-linear programming. It is intractable due to the multiplicative form of the integer variables ($H$ and $\mathbf{q}$) and continuous variables ($\epsilon_q$ and $\mathbf{B}$) in both the objective function and constraints. In what follows, we develop an iterative algorithm with low complexity to seek feasible solutions.

## 4.3 Iterative algorithm for EE-FWQ

The proposed iterative algorithm divides the original problem (27) into two sub-problems: 1) Local update and quantization error optimization (for $H$ and $\epsilon_q$); 2) Joint weight quantization selection and bandwidth allocation (for $\mathbf{q}$ and $\mathbf{B}$), which can be solved in an iterative manner. For the two sub-problems, we are able to derive the closed-form solutions for local updates, bandwidth allocation and weight quantization levels. The details are presented in the following subsections.

### *4.3.1 Local update and quantization error optimization*

To obtain the optimal strategies for FWQ, we first relax $H$ as a continuous variable for theoretical analysis, which is later rounded back to the nearest integer. Given $\overline{\mathbf{B}}$ and $\overline{\mathbf{q}}$, problem (27) is written as follows

$$\min_{H,\epsilon_q} \quad \frac{(A_1 H + A_2)^2}{MH(\epsilon - \epsilon_q)^2}(E^{cm}(\overline{\mathbf{B}}) + HE^{cp}(\overline{\mathbf{q}})) \tag{28a}$$

$$\text{s.t.} \quad \frac{(A_1 H + A_2)^2}{MH(\epsilon - \epsilon_q)^2} \leq \frac{T_{\max}}{T_i^{cm}(\overline{B}_i) + HT_i^{cp}(\overline{q}_i)}, \forall i \in \mathcal{N}, \tag{28b}$$

$$\epsilon_q \geq \epsilon_q^{\min}, \tag{28c}$$

$$0 \leq \epsilon_q \leq \epsilon, H \geq 0, \tag{28d}$$

where $\epsilon_q^{\min} = \sum_{i=1}^{N} \frac{A_3 \pi_i^2 s}{2^{\overline{q}_i} - 1}$, $E^{cm}(\overline{\mathbf{B}}) = \sum_{i=1}^{N} E_i^{cm}(\overline{B}_i)$, and $E^{cp}(\overline{\mathbf{q}}) = \sum_{i=1}^{N} E_i^{cp}(\overline{q}_i)$.

**Theorem 2.** *The optimal $\epsilon_q^\star$ in problem (27) satisfies*

$$\epsilon_q^\star = \epsilon_q^{\min}, \tag{29}$$

*and the optimal $H^\star$ is given by*

$$\min_{H} \quad \Psi(H) \triangleq \frac{(A_1 H + A_2)^2 (E^{cm}(\overline{\mathbf{B}}) + HE^{cp}(\overline{\mathbf{q}}))}{MH(\epsilon - \epsilon_q^{\min})^2} \tag{30a}$$

$$\textit{s.t.} \quad H_{\min} \leq H \leq H_{\max}, \tag{30b}$$

*where $\rho(H_{\min}) = \rho(H_{\max}) = MN(\epsilon - \epsilon_q^{\min})^2 T_{\max}$ and $\rho(H)$ is defined in (49b).*

[3] These coefficients can be estimated by using a small sampling set of training experimental results.

*Proof.* Please refer to the detailed proof in Appendix C in the separate supplemental file. □

Noted that it can be verified that the objective function $\Psi(H)$ in (30) is convex. The optimal $H^\star$ can be obtained by setting the following first-order derivative to zero,

$$\frac{\mathrm{d}\Psi(H)}{\mathrm{d}H} = 2A_1^2 H E^{cp}(\overline{\mathbf{q}}) + A_1^2 E^{cm}(\overline{\mathbf{B}}) + 2A_1A_2E^{cp}(\overline{\mathbf{q}}) - \frac{A_2^2 E^{cm}(\overline{\mathbf{B}})}{H^2}. \tag{31}$$

It is a cubic equation of $H$ and can be solved analytically via Cardano formula [40]. Therefore, for the fixed values of $\overline{\mathbf{q}}$ and $\overline{\mathbf{B}}$, we have a unique real solution of $H$ in the closed form as follows

$$H = \sqrt[3]{\sqrt{\frac{\alpha^3\beta}{27} + \frac{\beta^2}{4}} - \frac{\alpha^3}{27} - \frac{\beta}{2}} + \sqrt[3]{-\sqrt{\frac{\alpha^3\beta}{27} + \frac{\beta^2}{4}} - \frac{\alpha^3}{27} - \frac{\beta}{2}} + \frac{\alpha}{3}, \tag{32}$$

with $\alpha = \frac{A_1E^{cm}(\overline{\mathbf{B}})+2A_2E^{cp}(\overline{\mathbf{q}})}{2A_1E^{cp}(\overline{\mathbf{q}})}$, and $\beta = -\frac{A_2^2E^{cm}(\overline{\mathbf{B}})}{2A_1^2E^{cp}(\overline{\mathbf{q}})}$.

### *4.3.2 Joint weight quantization selection and bandwidth allocation*

Given the updated $H$, $\epsilon_q$, the optimal quantization levels $\mathbf{q}^\star$ and the bandwidth allocation $\mathbf{B}^\star$ can be obtained by solving the following problem,

$$\min_{\mathbf{q},\mathbf{B}} \quad K(H,\epsilon_q)\sum_{i=1}^{N}\frac{p_i^{cm}\alpha_i^1}{B_i} + Hp_i^{cp}\cdot(c_i^2 q_i + c_i^1) \tag{33a}$$

$$\text{s.t.} \quad (24b), (24c), (24f), (24g), \tag{33b}$$

$$\frac{\alpha_i^1}{B_i} + H(c_i^2 q_i + c_i^1) \leq \frac{T_{\max}}{K(H,\epsilon_q)}, \forall i \in \mathcal{N}. \tag{33c}$$

Based on the observation of problem (33), it is clear that problem (33) is a mixed-integer non-linear problem. Besides, the integer variable $q_i$ and a fractional form of continuous variable $B_i$ are linearly coupled in constraint (33c), which makes the optimization problem difficult to tackle. To address the above issues, we first introduce a new variable $\widetilde{q} = \log_2(q)$ and its finite set can be defined as $\widetilde{Q} = \{1,2,3,4,5\}$. We then relax $\widetilde{q}_i$ to be continuous and then round the solution. Since $\widetilde{q} = \log_2(q)$ is monotonously increasing function, we can transform an equivalent formulation as follows

$$\min_{\widetilde{\mathbf{q}},\mathbf{B}} \quad K(H,\epsilon_q)\sum_{i=1}^{N}\frac{p_i^{cm}\alpha_i^1}{B_i} + Hp_i^{cp}(c_i^2 2^{\widetilde{q}_i} + c_i^1) \tag{34a}$$

$$\text{s.t.} \quad (24f), \tag{34b}$$

$$\phi(\widetilde{q}_1,\cdots,\widetilde{q}_N) \triangleq \sum_{i=1}^{N}\frac{A_3\pi_i^2 s}{2^{2\widetilde{q}_i}-1} \leq \epsilon_q, \tag{34c}$$

$$c_3(2^{\widetilde{q}_i})U_i \leq C_i, \forall i \in \mathcal{N}, \tag{34d}$$

$$\frac{\alpha_i^1}{B_i} + H(c_i^2 2^{\widetilde{q}_i} + c_i^1) \leq \frac{T_{\max}}{K(H,\epsilon_q)}, \forall i \in \mathcal{N}, \tag{34e}$$

$$B_i > 0, q^{\min} \leq \widetilde{q}_i \leq q^{\max}, \forall i \in \mathcal{N}. \tag{34f}$$

For objective function in (34), $\frac{K(H,\epsilon_q)\alpha_i^1}{B_i}$ and $p_i^{cp}c_i^2 2^{\widetilde{q}_i}$ are convex functions in $B_i$ and $\widetilde{q}_i$, respectively. The affine combination of convex functions preserves convexity. Similarly, we can easily verify the convexity of the constraints.

**Algorithm 2** The proposed iterative algorithm for (34)

```
Input: Given $H$, $\epsilon_q$, two small constants, $\iota_1$, $\iota_2$, and a large positive number $\hat{\mu}$.
Output: Optimal $2^{\widetilde{q}_i^\star}$, $B_i^\star$
Initialization: $\mu_{LB}^1 = \omega_{LB} = 0$; $\mu_{UB}^1 = \hat{\mu}$; $\omega_{UB} = \hat{\omega}$;
Choose a feasible $\chi_0 \leftarrow (\mathbf{B}^{(0)}, \widetilde{\mathbf{q}}^{(0)})$
repeat
    Set $\mu^1 = (\mu_{UB}^1 + \mu_{LB}^1)/2$ and $\omega = (\omega_{UB} + \omega_{LB})/2$
    repeat
        Calculate $\widetilde{q}_i^{(z)}$ via (37)
        if $\phi(\widetilde{q}_1^{(z)}, \cdots, \widetilde{q}_N^{(z)}) > \epsilon_q$ then
            Set $\mu_{UB}^1 = \mu^1$
        else
            Set $\mu_{LB}^1 = \mu^1$
        end if
    until $\mu_{UB}^1 - \mu_{LB}^1 \leq \iota_1$
    repeat
        Set $\omega = (\omega_{UB} + \omega_{LB})/2$
        Calculate $B_i^{(z)}$ via (38)
        if $\sum B_i^{(z)} > B$ then
            Set $\omega_{UB} = \omega$
        else
            Set $\omega_{LB} = \omega$
        end if
    until $\omega_{UB} - \omega_{LB} \leq \iota_2$
    $\chi_z \leftarrow (\mathbf{B}^{(z)}, \widetilde{\mathbf{q}}^{(z)})$ and $z \leftarrow z + 1$
until $|\chi_{z+1} - \chi_z| \leq \iota_3$
```

Next, we propose an efficient iterative algorithm to reduce the computational complexity. The main idea of the proposed iterative algorithm as follows. In the $(z)$-th iteration, we first fix the bandwidth in the $(z-1)$-th iteration, denoted as $\mathbf{B}^{(z-1)}$, to solve problem (34) to obtain quantization strategy $\widetilde{\mathbf{q}}$; then, with the updated $\widetilde{\mathbf{q}}^{(z)}$, we can get the optimal $\mathbf{B}^{(z)}$. In the intermediate steps, we attempt to derive some analytical solutions to reduce the computation load.

In the $(z)$-th iteration, we can decompose problem (34) into two convex subproblems as

$$\min_{\widetilde{\mathbf{q}}^{(z)}} \quad R\sum_{i=1}^{N} p_i^{cp}(c_i^2 2^{\widetilde{q}_i^{(z)}} + c_i^1) \tag{35a}$$

$$\text{s.t.} \quad (34c), (34d), (34f), \tag{35b}$$

$$\frac{\alpha_i^1}{HB_i^{(z-1)}} + (c_i^2 2^{\widetilde{q}_i^{(z)}} + c_i^1) \leq \frac{T_{\max}}{R}, \forall i \in \mathcal{N}, \tag{35c}$$

where $R = HK(H,\epsilon_q)$ and

$$\min_{\mathbf{B}^{(z)}} \quad K(H,\epsilon_q)\sum_{i=1}^{N}\frac{p_i^{cm}\alpha_i^1}{B_i^{(z)}} \tag{36a}$$

$$\text{s.t.} \quad (24f), (34f), \tag{36b}$$

$$\frac{K(H,\epsilon_q)\alpha_i^1}{B_i^{(z)}} \leq T_{\max} - RT_i^{cp}(2^{\widetilde{q}_i^{(z)}}), \forall i \in \mathcal{N}. \tag{36c}$$

In the above, the objective function in (35) is a monotonically increasing function w.r.t $\widetilde{\mathbf{q}}$, and the objective function in (36) monotonically decreasing function w.r.t $\mathbf{B}$. Hence, we have the unique solutions of $\widetilde{\mathbf{q}}, \mathbf{B}$ as follows

**Theorem 3.** *The optimal quantization levels $\widetilde{q}_i^\star$ and bandwidth allocation $B_i^\star$ for the i-th device are given by*

$$\widetilde{q}_i^{(z)\star} = \min\{\widetilde{q}_i^{\max}, \widetilde{q}_i(\mu^{1\star})\}, \tag{37}$$

*and*

$$B_i^{(z)\star} = \max\{B_{i,\min}^{(z)}(\widetilde{q}_i^{(z)\star}), B_i^{(z)}(\omega^\star)\}, \tag{38}$$

*where*

$$\widetilde{q}_i = \log_2\left(\log_2(\lambda_i + \sqrt{\lambda_i^2 + 4}) - 1\right), \tag{39}$$

$$\lambda_i = \frac{\ln(2)\mu^{1\star} A_3 \pi_i^2 s^2}{c_i^2 R(p_i^{cp} + \mu_i^2(\mu^{1\star}))}, \tag{40}$$

$$B_{i,\min}^{(z)}(\widetilde{q}_i^{(z)\star}) = \frac{K(H, \epsilon_q)}{T_{\max} - RT_i^{cp}(2^{\widetilde{q}_i^{(z)\star}})}, \tag{41}$$

$$B_i^{(z)}(\omega^\star) = \frac{\sqrt{p_i^{cm}\alpha_i^1}(A_1 H + A_2)}{\omega^\star\sqrt{MH}(\epsilon - \epsilon_q)}, \tag{42}$$

*$\mu^{1\star}$ and $\omega^\star$ are the optimal Lagrange multipliers to satisfy the quantization error constraint $\phi(\widetilde{q}_1^{(z)\star}, \cdots, \widetilde{q}_N^{(z)\star}) = \epsilon_q$ and bandwidth capacity constraint $\sum_{i=1}^N B_i^{(z)\star} = B_{\max}$, respectively.*

*Proof.* Please refer to the detailed proof in Appendix D in the separate supplemental file. □

Theorem 3 suggests that $\widetilde{q}_i^\star$ is determined by local computing capabilities. Specifically, small quantization levels can be allocated to devices with weaker computing capabilities for the benefit of sum computing energy reduction. Given the overall quantization error constraint, the devices with higher computing capabilities may use a higher quantization level to maintain the model accuracy. It also indicates that the optimal bandwidth allocation $\mathbf{B}$ depends not only on the channel conditions ($h_i$) but also on the quantization levels $\widetilde{q}_i^\star$. Concisely, assigned bandwidth increases with the poor channel condition to avoid the straggler issues. In addition, when the devices use a higher quantization level for local training (higher computing energy), the device should be assigned more bandwidth to reduce total energy consumption.

The algorithm that solves problem (27) is summarized in Alg 3, by iteratively solving problem (28) and problem (34). We first solve problem (28) to determine $(H, \epsilon_q)$ in the closed form. Then, with Alg. 2 we iteratively calculate (37) and (38) which keeps decreasing the objective function in (33) until we achieve the converged solutions $(\widetilde{\mathbf{q}}, \mathbf{B})$. In Alg. 3, since the optimal solution of problem (28) or (34) can be obtained in each loop, the objective value of the problem (27) keeps decreasing in the loop. Moreover, the objective value of problem (27) is lower bounded by zero. Thus, Alg. 3, will finally converges.

Next, we analyze the computational complexity of Alg. 3. To solve the EE-FWQ problem by using Alg. 3, two subproblems (28) and (34) need to be solved. For the subproblem (28), we can obtain a unique real solution of $H$ from (31) in closed form, which does not resort to any iterative solver. For the subproblem (34), it requires $O\left(\log_2\left(\left(\mu_{UB}^1 - \mu_{LB}^1\right)/\iota_1\right) + \log_2\left(\left(\omega_{UB} - \omega_{LB}\right)/\iota_2\right)\right)$ inner-loop iterations for the bisection method [41] to determine the optimal $\mu^1$ and $\omega$ and $\Phi_1$ outer-loop iterations (as shown in simulations, $\Phi_1$ is usually no more than 3). Hence, it requires $O\left(\Phi_1\left(\log_2\left(\left(\mu_{UB}^1 - \mu_{LB}^1\right)/\iota_1\right) + \log_2\left(\omega_{UB} - \omega_{LB}\right)/\iota_2\right)\right)$ iterations to converge in Alg. 2. The complexity is $\mathcal{O}(N\Phi_1 \log_2(1/\iota_1)\log_2(1/\iota_2))$ with accuracy $\iota_1$ and $\iota_2$. As a result, the total complexity of Alg. 3 is $\mathcal{O}(N\Phi_1\Phi_2 \log_2(1/\iota_1)\log_2(1/\iota_2))$ where $\Phi_2$ is the number of iterations required in Alg 3 (as shown in simulations, $\Phi_2$ is usually no more than 4). The complexity of Alg. 3 is low since $\mathcal{O}(NL\log_2(1/\iota_1)\log_2(1/\iota_2))$ grows linearly with the total number of participating devices.

**Algorithm 3** Joint design of flexible weight quantization and bandwidth allocation for EE-FWQ

```
Input: Initialize $H(0), \epsilon_q(0), q_i(0), B_i(0)$ of problem (27) and set $l = 0$.
Output: $H^\star$, $\epsilon_q^\star$, $\mathbf{q}^\star$, $\mathbf{B}^\star$
repeat
    With given $\mathbf{q}(l)$, $\mathbf{B}(l)$, compute $\epsilon_q(l+1)$ and $H(l+1)$ via (29) and (32), respectively
    With given $\epsilon_q(l+1)$ and $H(l+1)$, compute $q_i(l+1)$ and $b_i(l+1)$ by Alg. 2
until objective value (27) converges
Rounding $\hat{q}_i = \lfloor\widetilde{q}_i^\star\rceil$ and $\lfloor H^\star\rceil$ and obtain the quantization strategy $q_i^\star = 2^{\hat{q}_i}$ with the minimum objective value.
```

It should be noted that the FL server is in charge of solving the optimization in (24). It is practical because the FL protocol in [42] requires mobile devices to check in with the FL server first before the FL training begins. Hence, the FL server can collect the information ($c_1(q_i), c_2(q_i), C_i, p_i^{cm}$ and $h_i$) from mobile devices, determine the optimal strategies ($q_i$, $B_i$) of each device via Alg. 3, and inform the strategies to the participating devices. It only needs to be solved once if the network information remains unchanged. That is absolutely affordable for the FL server.

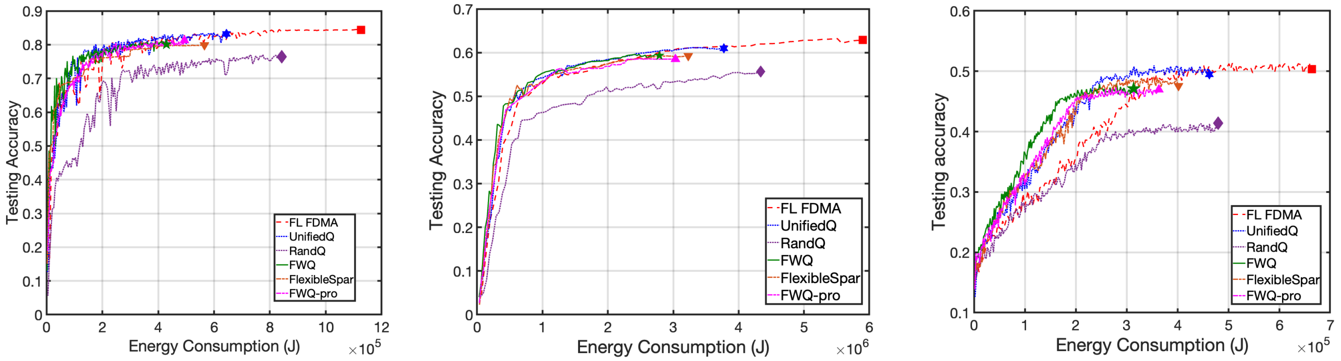
# 5 Performance Evaluation

## 5.1 Data and settings

1) *Learning Model and Dataset:* To test the model performance, we consider two types of learning tasks: image classification and next-character prediction. For the image classification task, we choose two commonly-used deep learning models: ResNet-34 [43], and MobileNet [44]. The well-known datasets, CIFAR-10 and CIFAR-100, are used to train FL models for image classification tasks. The CIFAR-10 dataset consists of 60000 32x32 color images in 10 classes with 5000 training images per class. The CIFAR-100 dataset has 100 classes and each class has 500 32x32 training images and 100 testing images. To generated heterogeneous data partition, we consider the label distribution of devices are different. Then, each device only has data samples of $J$ different labels. Without specific explanation, for the CIFAR-10 dataset, we consider the number of device, $N = 10$, and each device contains a total number of $30000/N$ training samples with only $J = 6$ classes. For the CIFAR-100, each device contains a total number of $20000/N$ training samples with only $J = 40$ classes. We use Shakespeare [45] dataset for the next character prediction task. This dataset is built on *The Complete Works of William Shakespeare* by separately extracting different roles' dialogues. We employ a two LSTM [46] layers, each with 256 nodes and a softmax layer (with dropout rate of 0.1). The heterogeneous dataset is the natural split of Shakespeare where each device corresponds to a role and the local dataset contains this role's sentences.

2) *Communication and Computing Models:* For the communication model, we assume the noise power is $N_0 = -174$ dBm. The transmitting power of each device is uniformly selected from $\{19, 20, 21, 22, 23\}$ dBm. Unless specified otherwise, we set the bandwidth $B_{\max} = 100$MHz and the channel gains $h_i$ are modeled as i.i.d. Rayleigh fading with average path loss set to $10^{-3}$.

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2022.3213766



(a) Testing accuracy vs energy cost. (b) Testing accuracy vs energy cost. (c) Testing accuracy vs energy cost.

Fig. 2. Convergence analysis for diferent learning tasks. (a): ResNet-34 on CIFAR-10 with the estimated parameters $A_1 = 13.765, A_2 = 1.023, A_3 = 0.0435$. (b): MobileNet on CIFAR-100 with the estimated parameters $A_1 = 16.655, A_2 = 1.013, A_3 = 0.0795$. (c): LSTM on Shakespeare with the estimated parameters $A_1 = 6.34, A_2 = 2.003, A_3 = 0.039$.

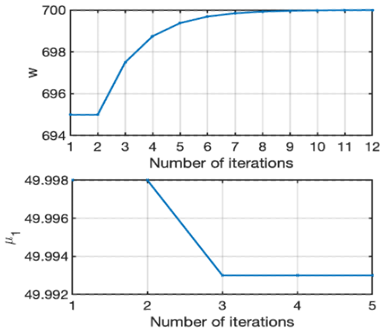


Fig. 3. Convergence of the inner loop of Alg. 2.

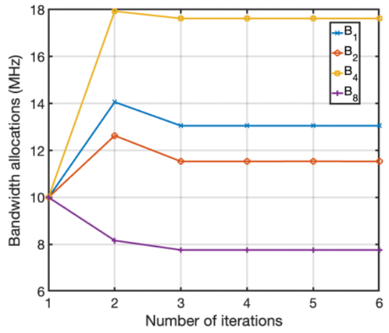


Fig. 4. Convergence of Alg. 2.

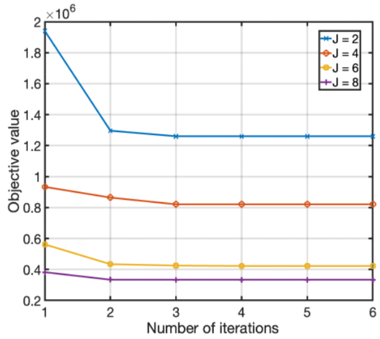


Fig. 5. Convergence of Alg. 3.

Furthermore, we assume that model parameter is quantized into 16 bits before transmission. For the GPU computing model, the scaling factors of quantization are measured by Nvidia profiling tools on Jetson Xavier NX. We use ResNet-34 model with CIFAR-10 multiple times and obtain the simulated function $c_1(q) = 7.12 \times 10^{-3}q + 0.274$ and $c_2(q) = 4.24 \times 10^{-4}q + 1.035$. The GPU core frequency $f_i^{core}, \forall i$ is uniformly selected from $\{1050, 1100, 1150, 1200\}$MHz and memory frequency $f_i^{mem}, \forall i$ is uniformly selected from $\{1450, 1500, 1550, 1600\}$MHz.

3) *Peer Schemes for Comparison:* We compare our proposed FWQ scheme with the following two different energy efficient FL schemes:

- *FL FDMA* [22]: All mobile devices train their local models with full precision, i.e., without quantization. Their scheme optimizes the computing and communication resources (i.e., CPU frequency and wireless bandwidth) to minimize the energy consumption in FL training. For a fair comparison, we change the CPU model in *FL FDMA* to GPU model and set $q = 32$.
- *FlexibleSpar* [24]: All mobile devices train their local models with the full precision and sparsity of their model updates before transmitting to the FL server. Their scheme optimizes the frequency of model updates and gradient sparsity ratio to minimize the energy consumption in FL training. Here, we set $q = 32$ in the GPU model.

Beside, we also consider two different quantization levels for our evaluation:

- *Unified Q*: All the devices are set to use the same quantization strategy regardless of resource budgets for different mobile devices.
- *Rand Q*: All mobile devices choose a quantization level randomly without considering the learning performance.
- *FWQ-pro*: We assign different weight quantization levels based on their GPU core and memory frequencies. Given the available combination of GPU core and memory frequencies, we divide devices into slow, medium, and fast groups. We set three different quantization levels, i.e., a small quantization level ($q = 8$), a medium quantization level ($q = 16$), a large quantization level ($q = 32$). Then, we assign the small quantization level to the slow group of devices. The rest can be done in the same manner.

The resource allocation strategies for *Unified Q*, *Rand Q*, and *FWQ-pro* are optimized by solving a simplified version of the problem (27).

## 5.2 Convergence analysis

First, we conduct convergence analysis. We implement the above learning models and choose a unified quantization strategy $q_1 = \cdots = q_N = 16$ in the *Unified Q* scheme. Fig. 2 shows the comparison of different FL schemes in terms of testing accuracy and corresponding energy consumption, when FL models are trained for a given epoch number[4]. We observe that the models

[4] We set different epoch numbers for different learning tasks: 200 epoch for CIFAR10, 300 epoches for CIFAR100, and 50 epoches for Shakespeare.

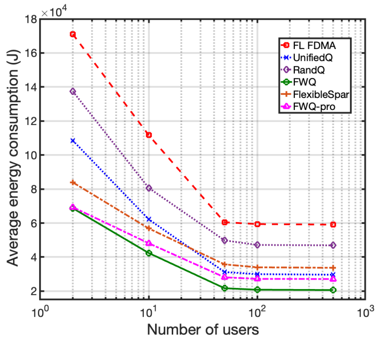


Fig. 6. Energy vs the numbers of devices.

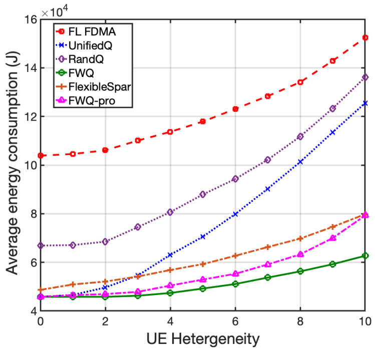


Fig. 7. Energy vs device heterogeneity.

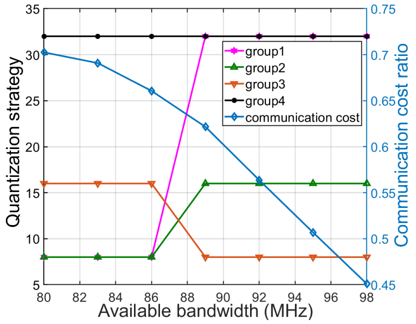


Fig. 8. Quantization vs bandwidth.

trained by FWQ, *Unified Q*, *Rand Q*, and *FWQ-pro* are inferior to the *FL FDMA* scheme, and the *Rand Q* has the worst performance. That is consistent with our convergence analysis that the discretization error induced by the quantization is unavoidable. This error is accumulated by all the participating mobile devices, which indicates some mobile devices take aggressive quantization levels (e.g., 8 bit) due to their resource limitation. As for our proposed FWQ scheme, since it considers this error in the quantization selection, the degradation is well controlled and relatively small. Compared with *FWQ-pro*, it demonstrates the effectiveness of the proposed optimization that can find the optimal strategies for different mobile devices. It shows when reaching FL convergence in the learning task of CIFAR10 with ResNet34, FWQ can reduce 62% energy consumption with 0.26% accuracy loss compared with *FL FDMA*, and reduce round 28% energy consumption with 0.16% accuracy loss compared with *FlexibleSpar*. For language task in Fig. 2(c), FWQ can reduce 52% energy with 0.18% accuracy loss compared with *FL FDMA*, and reduce 23% energy with 0.06% accuracy loss compared with *FlexibleSpar*. The FWQ scheme is superior to the other three schemes in terms of the trade-off between the overall energy efficiency for FL training and training accuracy, which is essential for battery-limited mobile devices.

Next, we show the convergence behavior of the proposed iterative algorithms, i.e., Algs. 2 and 3. The convergence results of Alg. 2 are shown in Fig. 3 and 4, and the convergence results of Alg. 3 is shown in Fig. 5. As observed from Figs. 3 and 4, the proposed iterative algorithm requires approximately seventeen inner iterations and 3 outer iterations. Hence, it takes total thirty to forty iterations to reach convergence, which indicates that Alg. 2 holds a desirable convergence rate. From Fig. 4, for different non-i.i.d levels (i.e., different values of $(A_1, A_2, A_3)$), they require approximately four iterations to reach convergence, which can be concluded that Alg. 3 is robust to the parameters $(A_1, A_2, A_3)$ in term of convergence rate.

### 5.3 Impacts of data heterogeneity

We evaluate the performance of FWQ with different data distributions in the context of skew class distribution. We set different $J$ values as $J \in \{2, 4, 6, 8\}$. Sample distributions become skewer as $J$ becomes small. The model is trained by ResNet34. As shown in Fig. 9, we find that training with small $J$ consumes more training energy compared with large $J$ values. In the case of $J = 2$, the proposed FWQ can efficiently reduce the energy consumption by 68% and 27%, compared with *FL FDMA* and *FlexibleSpar*, respectively. From the results in Fig. 9, the proposed FWQ achieves better trade-off between the energy consumption and model performance, compared to the peer schemes.

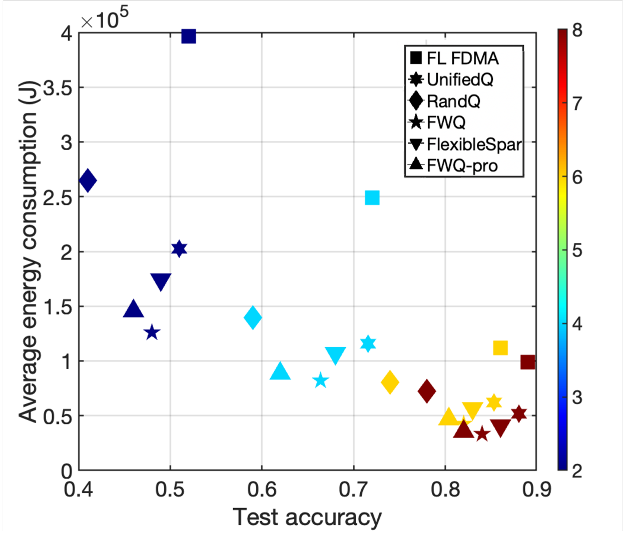


Fig. 9. Energy vs Test accuracy under fixed training iterations. Different colors represent different values of $J$.

### 5.4 Impact of the number of users

We now evaluate how the number of users affects the total energy consumption for FL training. Fig. 6 shows that the average energy consumption decreases as the number of mobile devices participating in FL increases. The average energy consumption per device does not experience too much change even after more devices participate in FL training under all the schemes. The reason is that bringing more devices to train the FL model helps speed up the model convergence and thus reduce energy consumption, which is consistent with the sub-linear speedup in Theorem 1. However, as $N$ continues increasing, the marginal reduction of the total number of training iterations becomes smaller and smaller. Besides, our proposed FWQ scheme outperforms *FL FDMA* and *FlexibleSpar*. For example, the proposed FWQ scheme saves the energy of *FL FDMA* by 56% and of *FlexibleSpar* by 35%, respectively. The reason is that the proposed FWQ leverages weight quantization to reduce the workload for on-device training and optimize the weight quantization levels for heterogeneous devices, while the computing workload is not optimized and fixed for all the devices in both *FL FDMA* and *FlexibleSpar* scheme.

TABLE 1
Computation overhead vs the numbers of devices.

| | N=10 | N=50 | N=100 | N=500 |
|---|---|---|---|---|
| Computation overhead (s) | 0.04 | 0.31 | 1.02 | 7.07 |

Moreover, the proposed FWQ scheme reduces the energy in the *Unified Q* strategy by 20%, the *Rand Q* strategy by 38.7%, and the *FWQ-pro* strategy by 13%, respectively, when the number of users is equal to ten. These results demonstrate the effectiveness of our proposed weight quantization scheme.

Next, we show the conputation overhead of the proposed iterative algorithms in Table 1. It shows the computation overhead of Alg. 3 under varying number of devices. The computation overhead increases with the increase of the number of devices.

### 5.5 Impact of computing capacities

We evaluate the impact of device heterogeneity concerning computing capability. Here, we keep the number of mobile devices as ten and divide them into four groups. Fixing the minimum capacity as 1800MB, we set different capacities into 4 groups: $C$MB, $(C + 50L)$MB, $(C + 150L)$MB, and $(C + 200L)$MB, respectively. The values of $L$ range from 0 to 10. A larger value of $L$ means mobile devices have more diverse computing conditions, implying that the optimized quantization strategy has more diverse values. From Fig. 7, we observe that the total energy consumption grows as the value of $L$ increases. It indicates that device heterogeneity does impact the energy efficiency in FL training. Besides, it is observed that the gap between *FWQ-pro* and FWQ increases as the UE heterogeneity level grows. This indicates the effectiveness of the proposed FWQ under high UE heterogenity. Since the proposed FWQ scheme jointly optimizes the quantization levels and bandwidth allocation for heterogeneous devices, the FWQ scheme is superior to all other schemes in terms of high levels of computing heterogeneity across participating devices.

### 5.6 Impact of communication capacities

Figure 8 shows the impacts of the wireless conditions on the optimal quantization selection. We vary the total available bandwidth from 80 MHz to 98 MHz and divide the mobile devices into 4 different groups, denoted as $\{g_1, g_2, g_3, g_4\}$, where the channel gain $h(g_1) \le h(g_2) \le h(g_d 3) \le h(g_4)$. From Fig. 8, we observe that, as the overall bandwidth becomes small, the ratio of the communication energy consumption to the overall energy consumption grows, which means wireless communications have a larger impact on the total energy consumption than local computing. As a result, the mobile devices in group 1, with small channel gain, become the stragglers in FL training and could slow down the gradient update time for one iteration. To avoid the update delay for the next iteration and reduce the overall energy consumption, they have to take aggressive actions to compress their local models into the smallest number of bits. However this results in large discretization noise and degrades the performance, as stated in Theorem 1. To compensate for that, those who have better channel gain need to "work" more by using a higher precision model to perform local training. Similarly, when the available bandwidth increases, the computing contributes more to the overall energy consumption. Those mobile devices with smaller local computing capacities choose to compress their models more to save computing energy.

## 6 Conclusion

In this paper, we have studied the energy efficiency of FL training via joint design of wireless transmission and weight quantization. We have jointly exploited the flexible weight quantization selection and the bandwidth allocation to develop an energy efficient FL training algorithm over heterogeneous mobile devices, constrained by the training delay and learning performance. The weight quantization approach has been leveraged to deal with the mismatch between high model computing complexity and limited computing capacities of mobile devices. The convergence rate of FL with local quantization has been analyzed. Guided by the derived theoretical convergence bound, we have formulated the energy efficient FL training problem as a mixed-integer nonlinear programming. Since the optimization variables of the problem are strongly coupled, we have proposed an efficient iterative algorithm, where the closed-form solution of the bandwidth allocation and weight quantization levels are derived in each iteration. By comparing with different quantization levels through extensive simulations, we have demonstrated the effectiveness of our proposed scheme in handling device heterogeneity and reducing overall energy consumption in FL over heterogeneous mobile devices.

## References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics (AISTATS)*, Ft. Lauderdale, FL, April 2017.

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

[3] T. Yu, T. Li, Y. Sun, S. Nanda, V. Smith, V. Sekar, and S. Seshan, "Learning context-aware policies from multiple smart homes via federated multi-task learning," in *Proc. of the IEEE/ACM Fifth International Conference on Internet-of-Things Design and Implementation (IoTDI)*, Sydney, Australia, May 2020.

[4] T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi, "Federated learning of predictive models from federated electronic health records," *International journal of medical informatics*, vol. 112, pp. 59–67, 2018.

[5] D. Ye, R. Yu, M. Pan, and Z. Han, "Federated learning in vehicular edge computing: A selective model aggregation approach," *IEEE Access*, vol. 8, pp. 23 920–23 935, 2020.

[6] Y. Li, X. Dong, and W. Wang, "Additive powers-of-two quantization: An efficient non-uniform discretization for neural networks," in *International Conference on Learning Representations (ICLR)*, Virtual Conference, September 2020.

[7] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.

[8] S. Gupta, A. Agrawal, K. Gopalakrishnan, and P. Narayanan, "Deep learning with limited numerical precision," in *International Conference on Machine Learning (ICML)*, Lille, France, July 2015.

[9] D. Zhang, J. Yang, D. Ye, and G. Hua, "Lq-nets: Learned quantization for highly accurate and compact deep neural networks," in *Proceedings of the European conference on computer vision (ECCV)*, Munich, Germany, September 2018.

[10] F. Fu, Y. Hu, Y. He, J. Jiang, Y. Shao, C. Zhang, and B. Cui, "Don't waste your bits! squeeze activations and gradients for deep neural networks via tinyscript," in *International Conference on Machine Learning*. virtual: PMLR, July 2020, pp. 3304–3314.

[11] L. Hou, R. Zhang, and J. T. Kwok, "Analysis of quantized models," in *International Conference on Learning Representations*, Vancouver, Canada, May 2018.

[12] NSF, "Resilient & intelligent nextG systems (RINGS)," https://www.nsf.gov/pubs/2021/nsf21581/nsf21581.htm?WT.mc_id=USNSF_25&WT.mc_ev=click#toc, Last Accessed May 4, 2021.

[13] 3GPP, "Technical Specification Group Services and System Aspects; Release 15 Description; Summary of Rel-15 Work Items ," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 21.915, 09 2019, version 2.0.0. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3389

[14] N. H. Tran, W. Bao, A. Zomaya, N. M. NH, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *IEEE Conference on Computer Communications (INFOCOM)*, Paris, France, April 2019.

[15] T. T. Vu, D. T. Ngo, N. H. Tran, H. Q. Ngo, M. N. Dao, and R. H. Middleton, "Cell-free massive mimo for wireless federated learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 6377 – 6392, 2020.

[16] 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," *arXiv preprint arXiv:1909.07972*, 2019.

[17] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 2022–2035, 2020.

[18] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *IEEE International Conference on Communications (ICC)*, Shanghai, China, May 2019.

[19] D. Shi, L. Li, R. Chen, P. Prakash, M. Pan, and Y. Fang, "Towards energy efficient federated learning over 5G+ mobile devices," *arXiv preprint arXiv:2101.04866*, 2021.

[20] W. Shi, S. Zhou, Z. Niu, M. Jiang, and L. Geng, "Joint device scheduling and resource allocation for latency constrained wireless federated learning," *IEEE Transactions on Wireless Communications*, 2020.

[21] Q. Zeng, Y. Du, K. Huang, and K. K. Leung, "Energy-efficient resource management for federated edge learning with cpu-gpu heterogeneous computing," *IEEE Transactions on Wireless Communications*, vol. 20, no. 12, 2021.

[22] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, "Energy efficient federated learning over wireless communication networks," *IEEE Transactions on Wireless Communications (Early Access)*, 2020.

[23] X. Mo and J. Xu, "Energy-efficient federated edge learning with joint communication and computation design," *Journal of Communications and Information Networks*, vol. 6, no. 2, pp. 110–124, 2021.

[24] L. Li, D. Shi, R. Hou, H. Li, M. Pan, and Z. Han, "To talk or to work: Flexible communication compression for energy efficient federated learning over heterogeneous mobile edge devices," in *Proc. IEEE Conference on Computer Communications (INFOCOM)*, Virtual Conference, May 2021.

[25] C. De Sa, M. Leszczynski, J. Zhang, A. Marzoev, C. R. Aberger, K. Olukotun, and C. Ré, "High-accuracy low-precision training," *arXiv preprint arXiv:1803.03383*, 2018.

[26] M. Courbariaux, I. Hubara, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1," *arXiv preprint arXiv:1602.02830*, 2016.

[27] Z. Li and C. M. De Sa, "Dimension-free bounds for low-precision training," in *Advances in Neural Information Processing Systems (NIPS)*, Vancouver, BC, Canada, December 2019.

[28] H. Li, S. De, Z. Xu, C. Studer, H. Samet, and T. Goldstein, "Training quantized nets: A deeper understanding," in *Advances in Neural Information Processing Systems*, 2017, pp. 5811–5821.

[29] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-iid data," in *International Conference on Learning Representations (ICLR)*, New Orleans, Louisiana, May 2019.

[30] H. Yu, S. Yang, and S. Zhu, "Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Honolulu, USA, January 2019.

[31] D. Zhou, P. Xu, and Q. Gu, "Stochastic nested variance reduction for nonconvex optimization," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 4130–4192, 2020.

[32] S. Ghadimi and G. Lan, "Stochastic first-and zeroth-order methods for nonconvex stochastic programming," *SIAM Journal on Optimization*, vol. 23, no. 4, pp. 2341–2368, 2013.

[33] P. Jiang and G. Agrawal, "A linear speedup analysis of distributed deep learning with sparse and quantized communication," in *Advances in Neural Information Processing Systems (NIPS'18)*, Montreal, Canada, December 2018, pp. 2525–2536.

[34] Y. Drori and O. Shamir, "The complexity of finding stationary points with stochastic gradient descent," in *International Conference on Machine Learning*. PMLR, 2020, pp. 2658–2667.

[35] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2021–2031.

[36] X. Mei, X. Chu, H. Liu, Y.-W. Leung, and Z. Li, "Energy efficient real-time task scheduling on cpu-gpu hybrid clusters," in *IEEE Conference on Computer Communications (INFOCOM)*, Atlanta, GA, USA, May 2017.

[37] T.-J. Yang, Y.-H. Chen, J. Emer, and V. Sze, "A method to estimate the energy consumption of deep neural networks," in *2017 51st asilomar conference on signals, systems, and computers*. Pacific Grove, CA: IEEE, November 2017.

[38] S. K. Saha, P. Deshpande, P. P. Inamdar, R. K. Sheshadri, and D. Koutsonikolas, "Power-throughput tradeoffs of 802.11 n/ac in smartphones," in *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 2015, pp. 100–108.

[39] E. Torti, A. Fontanella, M. Musci, N. Blago, D. Pau, F. Leporati, and M. Piastra, "Embedded real-time fall detection with deep learning on wearable devices," in *2018 21st euromicro conference on digital system design (DSD)*. Prague, Czech Republic: IEEE, August 2018, pp. 405–412.

[40] K.-H. Schlote, "Bl van der waerden, moderne algebra, (1930–1931)," in *Landmark Writings in Western Mathematics 1640-1940*. Elsevier, 2005, pp. 901–916.

[41] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[42] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečnỳ, S. Mazzocchi, H. B. McMahan *et al.*, "Towards federated learning at scale: System design," *arXiv preprint arXiv:1902.01046*, 2019.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learningfor image recognition," *ComputerScience*, 2015.

[44] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[45] W. Shakespeare, "The complete works of william shakespeare," ttps: //www.gutenberg.org/ebooks/100, Last Accessed August 4, 2022.

[46] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-aware neural language models, 2015," *arXiv preprint arXiv:1508.06615*, 2015.