Energy Efficient Federated Learning Over Wireless Communication Networks

Zhaohui Yang, Mingzhe Chen, Walid Saad, *Fellow, IEEE*, Choong Seon Hong, *Senior Member, IEEE*, and Mohammad Shikh-Bahaei, *Senior Member, IEEE*

***Abstract*—In this paper, the problem of energy efficient transmission and computation resource allocation for federated learning (FL) over wireless communication networks is investigated. In the considered model, each user exploits limited local computational resources to train a local FL model with its collected data and, then, sends the trained FL model to a base station (BS) which aggregates the local FL model and broadcasts it back to all of the users. Since FL involves an exchange of a learning model between users and the BS, both computation and communication latencies are determined by the learning accuracy level. Meanwhile, due to the limited energy budget of the wireless users, both local computation energy and transmission energy must be considered during the FL process. This joint learning and communication problem is formulated as an optimization problem whose goal is to minimize the total energy consumption of the system under a latency constraint. To solve this problem, an iterative algorithm is proposed where, at every step, closed-form solutions for time allocation, bandwidth allocation, power control, computation frequency, and learning accuracy are derived. Since the iterative algorithm requires an initial feasible solution, we construct the completion time minimization problem and a bisection-based algorithm is proposed to obtain the optimal solution, which is a feasible solution to the original energy minimization problem. Numerical results show that the proposed algorithms can reduce up to 59.5% energy consumption compared to the conventional FL method.**

***Index Terms*—Federated learning, resource allocation, energy efficiency.**

I. Introduction

In future wireless systems, due to privacy constraints and limited communication resources for data transmission, it is impractical for all wireless devices to transmit all of their collected data to a data center that can use the collected data to implement centralized machine learning algorithms for data analysis and inference [2]–[5]. To this end, distributed learning frameworks are needed, to enable the wireless devices to collaboratively build a shared learning model with training their collected data locally [6]–[15]. One of the most promising distributed learning algorithms is the emerging federated learning (FL) framework that will be adopted in future Internet of Things (IoT) systems [16]–[24]. In FL, wireless devices can cooperatively execute a learning task by only uploading local learning model to the base station (BS) instead of sharing the entirety of their training data [25]. Using gradient sparsification, a digital transmission scheme based on gradient quantization was investigated in [26]. To implement FL over wireless networks, the wireless devices must transmit their local training results over wireless links [27], which can affect the performance of FL due to limited wireless resources (such as time and bandwidth). In addition, the limited energy of wireless devices is a key challenge for deploying FL. Indeed, because of these resource constraints, it is necessary to optimize the energy efficiency for FL implementation.

Some of the challenges of FL over wireless networks have been studied in [28]–[34]. To minimize latency, a broadband analog aggregation multi-access scheme was designed in [28] for FL by exploiting the waveform-superposition property of a multi-access channel. An FL training minimization problem was investigated in [29] for cell-free massive multiple-input multiple-output (MIMO) systems. For FL with redundant data, an energy-aware user scheduling policy was proposed in [30] to maximize the average number of scheduled users. To improve the statistical learning performance for on-device distributed training, the authors in [31] developed a novel sparse and low-rank modeling approach. The work in [32] introduced an energy-efficient strategy for bandwidth allocation under learning performance constraints. However, the works in [28]–[32] focused on the delay/energy for wireless transmission without considering the delay/energy tradeoff between learning and transmission. Recently, the works in [33] and [34] considered both local learning and wireless transmission energy. In [33], we investigated the FL loss function minimization problem with taking into account packet errors over wireless links. However, this prior work ignored the computation delay of local FL model. The authors in [34] considered the sum computation and transmission energy minimization problem for FL. However, the solution in [34] requires all users to upload their learning model synchronously. Meanwhile, the work in [34] did not provide any convergence analysis for FL.

The main contribution of this paper is a novel energy efficient computation and transmission resource allocation scheme for FL over wireless communication networks. Our key contributions include:

- We study the performance of FL algorithm over wireless communication networks for a scenario in which each user locally computes its FL model under a given learning accuracy and the BS broadcasts the aggregated FL model

This work was supported in part by the EPSRC through the Scalable Full Duplex Dense Wireless Networks (SENSE) grant EP/P003486/1, and by the U.S. National Science Foundation under Grant CNS-1814477. A preliminary version of this work was by IEEE ICML 2020 [1].

Z. Yang and M. Shikh-Bahaei are with the Centre for Telecommunications Research, Department of Engineering, King's College London, WC2R 2LS, UK, Emails: yang.zhaohui@kcl.ac.uk, m.sbahaei@kcl.ac.uk.

M. Chen is with the Electrical Engineering Department of Princeton University, NJ, 08544, USA, and also with the Chinese University of Hong Kong, Shenzhen, 518172, China, Email: mingzhec@princeton.edu.

W. Saad is with Wireless@VT, Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, 24060, USA, Email: walids@vt.edu.

C. Hong is with the Department of Computer Science and Engineering, Kyung Hee University, Yongin-si, Gyeonggi-do 17104, Rep. of Korea, Email: cshong@khu.ac.kr.

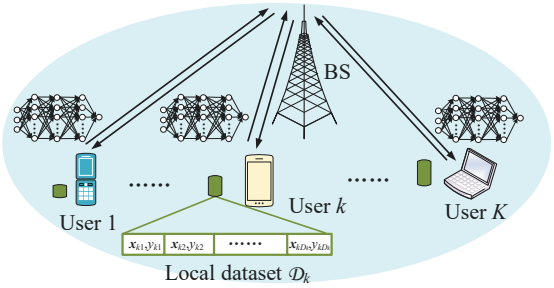

Fig. 1. Illustration of the considered model for FL over wireless communication networks.

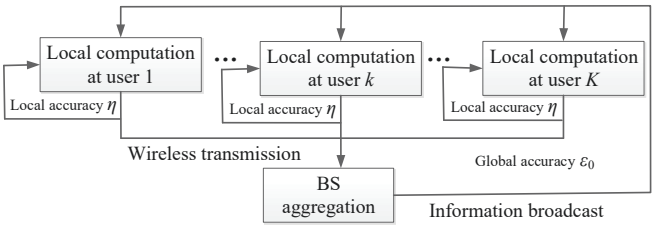

Fig. 2. The FL procedure between users and the BS.

to all users. For the considered FL algorithm, we first derive the convergence rate.

- We formulate a joint computation and transmission optimization problem aiming to minimize the total energy consumption for local computation and wireless transmission. To solve this problem, an iterative algorithm is proposed with low complexity. At each step of this algorithm, we derive new closed-form solutions for the time allocation, bandwidth allocation, power control, computation frequency, and learning accuracy.
- To obtain a feasible solution for the total energy minimization problem, we construct the FL completion time minimization problem. We theoretically show that the completion time is a convex function of the learning accuracy. Based on this theoretical finding, we propose a bisection-based algorithm to obtain the optimal solution for the FL completion time minimization.
- Simulation results show that the proposed scheme that jointly considers computation and transmission optimization can achieve up to 59.5% energy reduction compared to the conventional FL method.

The rest of this paper is organized as follows. The system model is described in Section II. Section III provides problem formulation and the resource allocation for total energy minimization. The algorithm to find a feasible solution of the original energy minimization problem is given in Section IV. Simulation results are analyzed in Section V. Conclusions are drawn in Section VI.

II. System Model

Consider a cellular network that consists of one BS serving K users, as shown in Fig. 1. Each user k has a local dataset $\mathcal{K}_k$ with D_k data samples. For each dataset $\mathcal{K}_k = \}\boldsymbol{x}_{kl}, y_{kl}|_{l=1}^{D_k}$, $\boldsymbol{x}_{kl} \ \forall \ \mathbb{R}^d$ is an input vector of user k and y_{kl} is its corresponding output[1]. The BS and all users cooperatively perform an FL algorithm over wireless networks for data analysis and inference. Hereinafter, the FL model that is trained by each user's dataset is called the *local FL model*, while the FL model that is generated by the BS using local FL model inputs from all users is called the *global FL model*.

A. Computation and Transmission Model

The FL procedure between the users and their serving BS is shown in Fig. 2. From this figure, the FL procedure contains three steps at each iteration: local computation at each user (using several local iterations), local FL model transmission for each user, and result aggregation and broadcast at the BS. The local computation step is essentially the phase during which each user calculates its local FL model by using its local data set and the received global FL model.

1) Local Computation: Let f_k be the computation capacity of user k, which is measured by the number of CPU cycles per second. The computation time at user k needed for data processing is:

$$\tau_k = \frac{I_k C_k D_k}{f_k}, \quad \mathcal{D}k \ \forall \ \mathcal{L}, \tag{1}$$

where C_k (cycles/sample) is the number of CPU cycles required for computing one sample data at user k and I_k is the number of local iterations at user k. According to Lemma 1 in [35], the energy consumption for computing a total number of $C_k D_k$ CPU cycles at user k is:

$$E_{k1}^{\mathrm{C}} = \kappa C_k D_k f_k^2, \tag{2}$$

where κ is the effective switched capacitance that depends on the chip architecture. To compute the local FL model, user k needs to compute $C_k D_k$ CPU cycles with I_k local iterations, which means that the total computation energy at user k is:

$$E_k^{\mathrm{C}} = I_k E_{k1}^{\mathrm{C}} = \kappa I_k C_k D_k f_k^2. \tag{3}$$

2) Wireless Transmission: After local computation, all users upload their local FL model to the BS via frequency domain multiple access (FDMA). The achievable rate of user k can be given by:

$$r_k = b_k \log_2 \Big)1 + \frac{g_k p_k}{N_0 b_k}\Big[, \quad \mathcal{D}k \ \forall \ \mathcal{L}, \tag{4}$$

where b_k is the bandwidth allocated to user k, p_k is the average transmit power of user k, g_k is the channel gain between user k and the BS, and N_0 is the power spectral density of the Gaussian noise. Note that the Shannon capacity (4) serves as an upper bound of the transmission rate. Due to limited bandwidth of the system, we have: $\sum_{k=1}^{K} b_k \geq B$, where B is the total bandwidth.

In this step, user k needs to upload the local FL model to the BS. Since the dimensions of the local FL model are fixed for all users, the data size that each user needs to upload is constant, and can be denoted by s. To upload data of size s within transmission time t_k, we must have: $t_k r_k \preceq s$. To transmit data of size s within a time duration t_k, the wireless transmit energy of user k will be: $E_k^{\mathrm{T}} = t_k p_k$.

[1]For simplicity, we consider an FL algorithm with a single output. In future work, our approach will be extended to the case with multiple outputs.

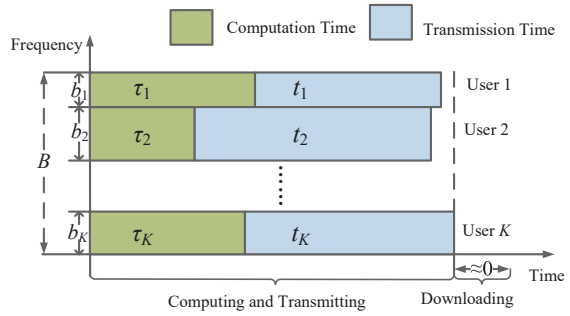

Fig. 3. An implementation for the FL algorithm via FDMA.

3) Information Broadcast: In this step, the BS aggregates the global FL model. The BS broadcasts the global FL model to all users in the downlink. Due to the high transmit power at the BS and the high bandwidth that can be used for data broadcasting, the downlink time is neglected compared to the uplink data transmission time. It can be observed that the local data $\mathcal{K}_k$ is not accessed by the BS, so as to protect the privacy of users, as is required by FL.

According to the above FL model, the energy consumption of each user includes both local computation energy E_k^{C} and wireless transmission energy E_k^{T}. Denote the number of global iterations by I_0, and the total energy consumption of all users that participate in FL will be:

$$E = I_0 \sum_{k=1}^{K} (E_k^{\mathrm{C}} + E_k^{\mathrm{T}}). \tag{5}$$

Hereinafter, the total time needed for completing the execution of the FL algorithm is called *completion time*. The completion time of each user includes the local computation time and transmission time, as shown in Fig. 3. Based on (1), the completion time T_k of user k will be:

$$T_k = I_0(\tau_k + t_k) = I_0 \left(\frac{I_k C_k D_k}{f_k} + t_k \right). \tag{6}$$

Let T be the maximum completion time for training the entire FL algorithm and we have:

$$T_k \geq T, \quad \forall k \in \mathcal{L}. \tag{7}$$

B. FL Model

We define a vector $\boldsymbol{w}$ to capture the parameters related to the global FL model. We introduce the loss function $f(\boldsymbol{w}, \boldsymbol{x}_{kl}, y_{kl})$, that captures the FL performance over input vector $\boldsymbol{x}_{kl}$ and output y_{kl}. For different learning tasks, the loss function will be different. For example, $f(\boldsymbol{w}, \boldsymbol{x}_{kl}, y_{kl}) = \frac{1}{2}(\boldsymbol{x}_{kl}^T \boldsymbol{w} - y_{kl})^2$ for linear regression and $f(\boldsymbol{w}, \boldsymbol{x}_{kl}, y_{kl}) = \log(1 + \exp(-y_{kl} \boldsymbol{x}_{kl}^T \boldsymbol{w}))$ for logistic regression. Since the dataset of user k is $\mathcal{K}_k$, the total loss function of user k will be:

$$F_k(\boldsymbol{w}, x_{k1}, y_{k1}, \ldots, x_{kD_k}, y_{kD_k}) = \frac{1}{D_k} \sum_{l=1}^{D_k} f(\boldsymbol{w}, \boldsymbol{x}_{kl}, y_{kl}). \tag{8}$$

Note that function $f(\boldsymbol{w}, \boldsymbol{x}_{kl}, y_{kl})$ is the loss function of user k with one data sample and function $F_k(\boldsymbol{w}, x_{k1}, y_{k1}, \ldots, x_{kD_k}, y_{kD_k})$ is the total loss function of user k with the whole local dataset. In the following, $F_k(\boldsymbol{w}, x_{k1}, y_{k1}, \ldots, x_{kD_k}, y_{kD_k})$ is denoted by $F_k(\boldsymbol{w})$ for simplicity of notation.

Algorithm 1 FL Algorithm

1: Initialize global model $\boldsymbol{w}^0$ and global iteration number $n = 0$.
2: **repeat**
3: Each user k computes $F_k(\boldsymbol{w}^{(n)})$ and sends it to the BS.
4: The BS computes $F(\boldsymbol{w}^{(n)}) = \frac{1}{K} \sum_{k=1}^{K} F_k(\boldsymbol{w}^{(n)})$, which is broadcast to all users.
5: **parallel for** user $k \in \mathcal{L} = \{1, \ldots, K\}$
6: Initialize the local iteration number $i = 0$ and set $\boldsymbol{h}_k^{(n),(0)} = \mathbf{0}$.
7: **repeat**
8: Update $\boldsymbol{h}_k^{(n),(i+1)} = \boldsymbol{h}_k^{(n),(i)} - \delta \nabla G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)})$ and $i = i + 1$.
9: **until** the accuracy η of local optimization problem (11) is obtained.
10: Denote $\boldsymbol{h}_k^{(n)} = \boldsymbol{h}_k^{(n),(i)}$ and each user sends $\boldsymbol{h}_k^{(n)}$ to the BS.
11: **end for**
12: The BS computes

$$\boldsymbol{w}^{(n+1)} = \boldsymbol{w}^{(n)} - \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{h}_k^{(n)}, \tag{10}$$

and broadcasts the value to all users.
13: Set $n = n + 1$.
14: **until** the accuracy ϵ_0 of problem (9) is obtained.

In order to deploy an FL algorithm, it is necessary to train the underlying model. Training is done in order to generate a unified FL model for all users without sharing any datasets. The FL training problem can be formulated as [2], [19], [25]:

$$\min_{\boldsymbol{w}} F(\boldsymbol{w}) \triangleq \sum_{k=1}^{K} \frac{D_k}{D} F_k(\boldsymbol{w}) = \frac{1}{D} \sum_{k=1}^{K} \sum_{l=1}^{D_k} f(\boldsymbol{w}, \boldsymbol{x}_{kl}, y_{kl}), \tag{9}$$

where $D = \sum_{k=1}^{K} D_k$ is the total data samples of all users.

To solve problem (9), we adopt the distributed approximate Newton (DANE) algorithm in [25], which is summarized in Algorithm 1. Note that the gradient descent (GD) is used at each user in Algorithm 1. One can use stochastic gradient descent (SGD) to decrease the computation complexity for cases in which a relatively low accuracy can be tolerated. However, if high accuracy was needed, gradient descent (GD) would be preferred [25]. Moreover, the number of global iterations is higher in SGD than in GD. Due to limited wireless resources, SGD may not be efficient since SGD requires more iterations for wireless transmissions compared to GD. The DANE algorithm is designed to solve a general local optimization problem, before averaging the solutions of all users. The DANE algorithm relies on the similarity of the Hessians of local objectives, representing their iterations as an average of inexact Newton steps. In Algorithm 1, we can see that, at every FL iteration, each user downloads the global FL model from the BS for local computation, while the BS periodically gathers the local FL model from all

users and sends the updated global FL model back to all users. According to Algorithm 1, since the local optimization problem is solved with several local iterations at each global iteration, the number of full gradient computations will be smaller than the number of local updates.

We define $\boldsymbol{w}^{(n)}$ as the global FL model at a given iteration n. In practice, each user solves the local optimization problem:

$$\min_{\boldsymbol{h}_k \in \mathbb{R}^d} \quad G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k) \triangleq F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k) \quad (\quad F_k(\boldsymbol{w}^{(n)}) \quad \xi \quad F(\boldsymbol{w}^{(n)}))^T \boldsymbol{h}_k. \tag{11}$$

In problem (11), ξ is a constant value. Any solution $\boldsymbol{h}_k$ of problem (11) represents the difference between the global FL model and local FL model for user k, i.e., $\boldsymbol{w}^{(n)} + \boldsymbol{h}_k$ is the local FL model of user k at iteration n. The local optimization problem in (11) depends only on the local data and the gradient of the global loss function. The local optimization problem is then solved, and the updates from individual users are averaged to form a new iteration. This approach allows any algorithm to be used to solve the local optimization problem. Note that the number of iterations needed to upload the local FL model (i.e., the number of global iterations) in Algorithm 1 is always smaller than in the conventional FL algorithms that do not consider a local optimization problem (11) [25]. To solve the local optimization problem in (11), we use the gradient method:

$$\boldsymbol{h}_k^{(n),(i+1)} = \boldsymbol{h}_k^{(n),(i)} \quad \delta \quad G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}), \tag{12}$$

where δ is the step size, $\boldsymbol{h}_k^{(n),(i)}$ is the value of $\boldsymbol{h}_k$ at the i-th local iteration with given vector $\boldsymbol{w}^{(n)}$, and $G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)})$ is the gradient of function $G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k)$ at point $\boldsymbol{h}_k = \boldsymbol{h}_k^{(n),(i)}$. For small step size δ, based on equation (12), we can obtain a set of solutions $\boldsymbol{h}_k^{(n),(0)}$, $\boldsymbol{h}_k^{(n),(1)}$, $\boldsymbol{h}_k^{(n),(2)}$, $\times\times\times$, $\boldsymbol{h}_k^{(n),(i)}$, which satisfy,

$$G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(0)}) \preceq \times\times\times \preceq G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}). \tag{13}$$

To provide the convergence condition for the gradient method, we introduce the definition of local accuracy, i.e., the solution $\boldsymbol{h}_k^{(n),(i)}$ of problem (11) with accuracy η means that:

$$G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}) \quad G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*}) \geq \eta(G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(0)}) \quad G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*})), \tag{14}$$

where $\boldsymbol{h}_k^{(n)*}$ is the optimal solution of problem (11). Each user is assumed to solve the local optimization problem (11) with a target accuracy η. Next, in Lemma 1, we derive a lower bound on the number of local iterations needed to achieve a local accuracy η in (14).

Lemma 1: Let $v = \frac{2}{(2-L\delta)\delta\gamma}$. If we set step $\delta < \frac{2}{L}$ and run the gradient method

$$i \preceq v \log_2(1/\eta) \tag{15}$$

iterations at each user, we can solve local optimization problem (11) with an accuracy η.

Proof: See Appendix A. □

The lower bounded derived in (15) reflects the growing trend for the number of local iterations with respect to accuracy η. In the following, we use this lower bound to *approximate the number of iterations* I_k needed for local computations by each user.

In Algorithm 1, the iterative method involves a number of global iterations (i.e., the value of n in Algorithm 1) to achieve a global accuracy ϵ_0 for the global FL model. In other words, the solution $\boldsymbol{w}^{(n)}$ of problem (9) with accuracy ϵ_0 is a point such that

$$F(\boldsymbol{w}^{(n)}) \quad F(\boldsymbol{w}^*) \geq \epsilon_0(F(\boldsymbol{w}^{(0)}) \quad F(\boldsymbol{w}^*)), \tag{16}$$

where $\boldsymbol{w}^*$ is the actual optimal solution of problem (9).

To analyze the convergence rate of Algorithm 1, we make the following two assumptions on the loss function.

- A1: Function $F_k(\boldsymbol{w})$ is L-Lipschitz, i.e., $^2F_k(\boldsymbol{w}) \succeq L\boldsymbol{I}$.
- A2: Function $F_k(\boldsymbol{w})$ is γ-strongly convex, i.e., $^2F_k(\boldsymbol{w}) \approx \gamma\boldsymbol{I}$.

The values of γ and L are determined by the loss function.

These assumptions can be easily satisfied by widely used FL loss functions such as linear or logistic loss functions [18].

Under assumptions A1 and A2, we provide the following theorem about convergence rate of Algorithm 1 (i.e., the number of iterations needed for Algorithm 1 to converge), where each user solves its local optimization problem with a given accuracy.

Theorem 1: If we run Algorithm 1 with $0 < \xi \geq \frac{\gamma}{L}$ for

$$n \preceq \frac{a}{1 \quad \eta} \tag{17}$$

iterations with $a = \frac{2L^2}{\gamma^2\xi} \ln \frac{1}{\epsilon_0}$, we have $F(\boldsymbol{w}^{(n)}) \quad F(\boldsymbol{w}^*) \geq \epsilon_0(F(\boldsymbol{w}^{(0)}) \quad F(\boldsymbol{w}^*))$.

Proof: See Appendix B. □

From Theorem 1, we observe that the number of global iterations n increases with the local accuracy η at the rate of $1/(1 \quad \eta)$. From Theorem 1, we can also see that the FL performance depends on parameters L, γ, ξ, ϵ_0 and η. Note that the prior work in [36, Eq. (9)] only studied the number of iterations needed for FL convergence under the special case in which $\eta = 0$. Theorem 1 provides a general convergence rate for FL with an arbitrary η. Since the FL algorithm involves the accuracy of local computation and the result aggregation, it is hard to calculate the exact number of iterations needed for convergence. In the following, we use $\frac{a}{1-\eta}$ to approximate the number I_0 of global iterations.

For parameter setup, one way is to choose a very small value of ξ , i.e., $\xi \infty 0$ satisfying $0 < \xi \geq \frac{\gamma}{L}$, for an arbitrary learning task and loss function. However, based on Theorem 1, the iteration time is pretty large for a very small value of ξ. As a result, we first choose a large value of ξ and decrease the value of ξ to $\xi/2$ when the loss does not decrease over time (i.e., the value of ξ is large). Note that in the simulations, the value of ξ always changes at least three times. Thm 1 is suitable for the situation when the value of ξ is fixed, i.e., after at most three iterations of Algorithm 1.

C. Extension to Nonconvex Loss Function

We replace convex assumption A2 with the following condition:

- B2: Function $F_k(\boldsymbol{w})$ is of γ-bounded nonconvexity (or γ-nonconvex), i.e., all the eigenvalues of $^2F_k(\boldsymbol{w})$ lie in $[\ \gamma, L]$, for some $\gamma\ \forall\ (0, L]$.

Due to the non-convexity of function $F_k(\boldsymbol{w})$, we respectively replace $F_k(\boldsymbol{w})$ and $F(\boldsymbol{w})$ with their regularized versions [37]

$$\tilde{F}_k^{(n)}(\boldsymbol{w}) = F_k(\boldsymbol{w}) + \gamma \nabla \boldsymbol{w} \quad \boldsymbol{w}^{(n)} \nabla^2, \tilde{F}^{(n)}(\boldsymbol{w}) = \sum_{k=1}^{K} \frac{D_k}{D} \tilde{F}_k^{(n)}(\boldsymbol{w}). \tag{18}$$

Based on assuption A2 and (18), both functions $F_k^{(n)}(\boldsymbol{w})$ and $F^{(n)}(\boldsymbol{w})$ are γ-strongly convex, i.e., $^2\tilde{F}_k^{(n)}(\boldsymbol{w}) \approx \gamma \boldsymbol{I}$ and $^2\tilde{F}^{(n)}(\boldsymbol{w}) \approx \gamma \boldsymbol{I}$. Moreover, it can be proved that both functions $F_k^{(n)}(\boldsymbol{w})$ and $F^{(n)}(\boldsymbol{w})$ are $(L + 2\gamma)$-Lipschitz. The convergence analysis in Section II-B can be direcltly applied.

III. Resource Allocation for Energy Minimization

In this section, we formulate the energy minimization problem for FL. Since it is challenging to obtain the globally optimal solution due to nonconvexity, an iterative algorithm with low complexity is proposed to solve the energy minimization problem.

A. Problem Formulation

Our goal is to minimize the total energy consumption of all users under a latency constraint. This energy efficient optimization problem can be posed as follows:

$$\min_{\boldsymbol{t}, \boldsymbol{b}, \boldsymbol{f}, \boldsymbol{p}, \eta} \quad E, \tag{19}$$

$$\text{s.t.} \quad \frac{a}{1 \quad \eta} \Big) \frac{A_k \log_2(1/\eta)}{f_k} + t_k \Big[\geq T, \quad \mathcal{D}k \ \forall\ \mathcal{L}, \tag{19a}$$

$$t_k b_k \log_2 \Big) 1 + \frac{g_k p_k}{N_0 b_k} \Big[\preceq s, \quad \mathcal{D}k \ \forall\ \mathcal{L}, \tag{19b}$$

$$\sum_{k=1}^{K} b_k \geq B, \tag{19c}$$

$$0 \geq f_k \geq f_k^{\max}, \quad \mathcal{D}k \ \forall\ \mathcal{L}, \tag{19d}$$

$$0 \geq p_k \geq p_k^{\max}, \quad \mathcal{D}k \ \forall\ \mathcal{L}, \tag{19e}$$

$$0 \geq \eta \geq 1, \tag{19f}$$

$$t_k \preceq 0, b_k \preceq 0, \quad \mathcal{D}k \ \forall\ \mathcal{L}, \tag{19g}$$

where $\boldsymbol{t} = [t_1, \times\times\times, t_K]^T$, $\boldsymbol{b} = [b_1, \times\times\times, b_K]^T$, $\boldsymbol{f} = [f_1, \times\times\times, f_K]^T$, $\boldsymbol{p} = [p_1, \times\times\times, p_K]^T$, $A_k = vC_kD_k$ is a constant, $f_k^{\max}$ and $p_k^{\max}$ are respectively the maximum local computation capacity and maximum value of the average transmit power of user k. Constraint (19a) is obtained by substituting $I_k = v \log_2(1/\eta)$ from (15) and $I_0 = \frac{a}{1-\eta}$ from (17) into (6). Constraints (19b) is derived according to (4) and $t_k r_k \preceq s$. Constraint (19a) indicates that the execution time of the local tasks and transmission time for all users should not exceed the maximum completion time for the whole FL algorithm. Since the total number of iterations for each user is the same, the constraint in (19a) captures a maximum time constraint for all users at each iteration if we divide the total number of iterations on both sides of the constraint in (19a). The data transmission constraint is given by (19b), while the bandwidth constraint is given by (19c). Constraints (19d) and (19e) respectively represent the maximum local computation capacity and average transmit power limits of all users. The local accuracy constraint is given by (19f).

B. Iterative Algorithm

The proposed iterative algorithm mainly contains two steps at each iteration. To optimize $(\boldsymbol{t}, \boldsymbol{b}, \boldsymbol{f}, \boldsymbol{p}, \eta)$ in problem (19), we first optimize $(\boldsymbol{t}, \eta)$ with fixed $(\boldsymbol{b}, \boldsymbol{f}, \boldsymbol{p})$, then $(\boldsymbol{b}, \boldsymbol{f}, \boldsymbol{p})$ is updated based on the obtained $(\boldsymbol{t}, \eta)$ in the previous step. The advantage of this iterative algorithm lies in that we can obtain the optimal solution of $(\boldsymbol{t}, \eta)$ or $(\boldsymbol{b}, \boldsymbol{f}, \boldsymbol{p})$ in each step.

In the first step, given $(\boldsymbol{b}, \boldsymbol{f}, \boldsymbol{p})$, problem (19) becomes:

$$\min_{\boldsymbol{t}, \eta} \quad \frac{a}{1 \quad \eta} \sum_{k=1}^{K} \kappa A_k \log_2(1/\eta) f_k^2 + t_k p_k \|, \tag{20}$$

$$\text{s.t.} \quad \frac{a}{1 \quad \eta} \Big) \frac{A_k \log_2(1/\eta)}{f_k} + t_k \Big[\geq T, \quad \mathcal{D}k \ \forall\ \mathcal{L}, \tag{20a}$$

$$t_k \preceq t_k^{\min}, \quad \mathcal{D}k \ \forall\ \mathcal{L}, \tag{20b}$$

$$0 \geq \eta \geq 1, \tag{20c}$$

where

$$t_k^{\min} = \frac{s}{b_k \log_2 \Big) 1 + \frac{g_k p_k}{N_0 b_k} \Big(}, \quad \mathcal{D}k \ \forall\ \mathcal{L}. \tag{21}$$

The optimal solution of (20) can be derived using the following theorem.

Theorem 2: The optimal solution $(\boldsymbol{t}^*, \eta^*)$ of problem (20) satisfies:

$$t_k^* = t_k^{\min}, \quad \mathcal{D}k \ \forall\ \mathcal{L}, \tag{22}$$

and η^* is the optimal solution to:

$$\min_{\eta} \quad \frac{\alpha_1 \log_2(1/\eta) + \alpha_2}{1 \quad \eta} \tag{23}$$

$$\text{s.t.} \quad \eta^{\min} \geq \eta \geq \eta^{\max}, \tag{23a}$$

where

$$\eta^{\min} = \max_{k \in \mathcal{K}} \eta_k^{\min}, \quad \eta^{\max} = \min_{k \in \mathcal{K}} \eta_k^{\max}, \tag{24}$$

$\beta_k(\eta_k^{\min}) = \beta_k(\eta_k^{\max}) = t_k^{\min}$, $t_k^{\min} \geq t_k^{\max}$, α_1, α_2 and $\beta_k(\eta)$ are defined in (C.2).

Proof: See Appendix C. □

Theorem 2 shows that it is optimal to transmit with the minimum time for each user. Based on this finding, problem (20) is equivalent to the problem (23) with only one variable. Obviously, the objective function (23) has a fractional form, which is generally hard to solve. By using the parametric approach in [38], we consider the following problem,

$$H(\zeta) = \min_{\eta^{\min} \leq \eta \leq \eta^{\max}} \alpha_1 \log_2(1/\eta) + \alpha_2 \quad \zeta(1 \quad \eta). \tag{25}$$

It has been proved [38] that solving (23) is equivalent to finding the root of the nonlinear function $H(\zeta)$. Since (25) with fixed ζ is convex, the optimal solution η^* can be obtained by setting the first-order derivative to zero, yielding the optimal solution: $\eta^* = \frac{\alpha_1}{(\ln 2)\zeta}$. Thus, problem (23) can be solved by using the Dinkelbach method in [38] (shown as Algorithm 2).

In the second step, given $(\boldsymbol{t}, \eta)$ calculated in the first step, problem (19) can be simplified as:

Algorithm 2 The Dinkelbach Method

1: Initialize $\zeta = \zeta^{(0)} > 0$, iteration number $n = 0$, and set the accuracy ϵ_3.
2: **repeat**
3: Calculate the optimal $\eta^* = \frac{\alpha_1}{(\ln 2)\zeta^{(n)}}$ of problem (25).
4: Update $\zeta^{(n+1)} = \frac{\alpha_1 \log_2(1/\eta^*)+\alpha_2}{1-\eta^*}$
5: Set $n = n + 1$.
6: **until** $\|H(\zeta^{(n+1)})\| / \|H(\zeta^{(n)})\| < \epsilon_3$.

$$\min_{\boldsymbol{b},\boldsymbol{f},\boldsymbol{p}} \frac{a}{1-\eta} \sum_{k=1}^{K} \kappa A_k \log_2(1/\eta) f_k^2 + t_k p_k \|, \tag{26}$$

$$\text{s.t.} \quad \frac{a}{1-\eta}\Big)\frac{A_k \log_2(1/\eta)}{f_k} + t_k\Big[\geq T, \quad \forall k \in \mathcal{L}, \tag{26a}$$

$$t_k b_k \log_2\Big)1 + \frac{g_k p_k}{N_0 b_k}\Big[\preceq s, \quad \forall k \in \mathcal{L}, \tag{26b}$$

$$\sum_{k=1}^{K} b_k \geq B, \tag{26c}$$

$$0 \geq p_k \geq p_k^{\max}, \quad \forall k \in \mathcal{L}, \tag{26d}$$

$$0 \geq f_k \geq f_k^{\max}, \quad \forall k \in \mathcal{L}. \tag{26e}$$

Since both objective function and constraints can be decoupled, problem (26) can be decoupled into two subproblems:

$$\min_{\boldsymbol{f}} \quad \frac{a\kappa \log_2(1/\eta)}{1-\eta} \sum_{k=1}^{K} A_k f_k^2, \tag{27}$$

$$\text{s.t.} \quad \frac{a}{1-\eta}\Big)\frac{A_k \log_2(1/\eta)}{f_k} + t_k\Big[\geq T, \quad \forall k \in \mathcal{L}, \tag{27a}$$

$$0 \geq f_k \geq f_k^{\max}, \quad \forall k \in \mathcal{L}, \tag{27b}$$

and

$$\min_{\boldsymbol{b},\boldsymbol{p}} \quad \frac{a}{1-\eta} \sum_{k=1}^{K} t_k p_k, \tag{28}$$

$$\text{s.t.} \quad t_k b_k \log_2\Big)1 + \frac{g_k p_k}{N_0 b_k}\Big[\preceq s, \quad \forall k \in \mathcal{L}, \tag{28a}$$

$$\sum_{k=1}^{K} b_k \geq B, \tag{28b}$$

$$0 \geq p_k \geq p_k^{\max}, \quad \forall k \in \mathcal{L}. \tag{28c}$$

According to (27), it is always efficient to utilize the minimum computation capacity f_k. To minimize (27), the optimal f_k^* can be obtained from (27a), which gives:

$$f_k^* = \frac{aA_k \log_2(1/\eta)}{T(1-\eta) - at_k}, \quad \forall k \in \mathcal{L}. \tag{29}$$

We solve problem (28) using the following theorem.

Theorem 3: The optimal solution $(\boldsymbol{b}^*, \boldsymbol{p}^*)$ of problem (28) satisfies:

$$b_k^* = \max\}b_k(\mu), b_k^{\min}|, \tag{30}$$

and

$$p_k^* = \frac{N_0 b_k^*}{g_k}\Big)2^{\frac{s}{t_k b_k^*}} \quad 1\Big(, \tag{31}$$

where

$$b_k^{\min} = \frac{(\ln 2)s}{t_k W}\Big) \frac{(\ln 2)N_0 s}{g_k p_k^{\max} t_k} \mathrm{e}^{-\frac{(\ln 2)N_0 s}{g_k p_k^{\max} t_k}}\Big[+ \frac{(\ln 2)N_0 s}{g_k p_k^{\max}}, \tag{32}$$

Algorithm 3 : Iterative Algorithm

1: Initialize a feasible solution $(\boldsymbol{t}^{(0)}, \boldsymbol{b}^{(0)}, \boldsymbol{f}^{(0)}, \boldsymbol{p}^{(0)}, \eta^{(0)})$ of problem (19) and set $l = 0$.
2: **repeat**
3: With given $(\boldsymbol{b}^{(l)}, \boldsymbol{f}^{(l)}, \boldsymbol{p}^{(l)})$, obtain the optimal $(\boldsymbol{t}^{(l+1)}, \eta^{(l+1)})$ of problem (20).
4: With given $(\boldsymbol{t}^{(l+1)}, \eta^{(l+1)})$, obtain the optimal ($\boldsymbol{b}^{(l)}, \boldsymbol{f}^{(l)}, \boldsymbol{p}^{(l)}$) of problem (26).
5: Set $l = l + 1$.
6: **until** objective value (19a) converges

$b_k(\mu)$ is the solution to

$$\frac{N_0}{g_k t_k}\Big)e^{\frac{(\ln 2)s}{t_k b_k(\mu)}} \quad 1 \quad \frac{(\ln 2)s}{t_k b_k(\mu)} e^{\frac{(\ln 2)s}{t_k b_k(\mu)}}\Big[+ \mu = 0, \tag{33}$$

and μ satisfies

$$\sum_{k=1}^{K} \max\}b_k(\mu), b_k^{\min}| = B. \tag{34}$$

Proof: See Appendix D. □

By iteratively solving problem (20) and problem (26), the algorithm that solves problem (19) is given in Algorithm 3. Since the optimal solution of problem (20) or (26) is obtained in each step, the objective value of problem (19) is nonincreasing in each step. Moreover, the objective value of problem (19) is lower bounded by zero. Thus, Algorithm 3 always converges to a local optimal solution.

C. Complexity Analysis

To solve the energy minimization problem (19) by using Algorithm 3, the major complexity in each step lies in solving problem (20) and problem (26). To solve problem (20), the major complexity lies in obtaining the optimal η^* according to Theorem 2, which involves complexity $\{ (K \log 2(1/\epsilon_1))$ with accuracy ϵ_1 by using the Dinkelbach method. To solve problem (26), two subproblems (27) and (28) need to be optimized. For subproblem (27), the complexity is $\{ (K)$ according to (29). For subproblem (28), the complexity is $\{ (K \log_2(1/\epsilon_2) \log_2(1/\epsilon_3))$, where ϵ_2 and ϵ_3 are respectively the accuracy of solving (32) and (33). As a result, the total complexity of the proposed Algorithm 3 is $\{ (L_{\text{it}} SK)$, where L_{it} is the number of iterations for iteratively optimizing $(\boldsymbol{t}, \eta)$ and $(T, \boldsymbol{b}, \boldsymbol{f}, \boldsymbol{p})$, and $S = \log 2(1/\epsilon_1) + \log 2(1/\epsilon_2) \log 2(1/\epsilon_3)$.

The conventional successive convex approximation (SCA) method can be used to solve problem (19). The complexity of SCA method is $O(L_{\text{sca}} K^3)$, where L_{sca} is the total number of iterations for SCA method. Compared to SCA, the proposed Algorithm 3 grows linearly with the number of users K.

It should be noted that Algorithm 3 is done at the BS side before executing the FL scheme in Algorithm 1. To implement Algorithm 3, the BS needs to gather the information of g_k, $p_k^{\max}$, $f_k^{\max}$, C_k, D_k, and s, which can be uploaded by all users before the FL process. Due to small data size, the transmission delay of these information can be neglected. The BS broadcasts the obtained solution to all users. Since the BS

has high computation capacity, the latency of implementing Algorithm 3 at the BS will not affect the latency of the FL process.

IV. Algorithm to Find a Feasible Solution of Problem (19)

Since the iterative algorithm to solve problem (19) requires an initial feasible solution, we provide an efficient algorithm to find a feasible solution of problem (19) in this section. To obtain a feasible solution of (19), we construct the completion time minimization problem:

$$\min_{T,\boldsymbol{t},\boldsymbol{b},\boldsymbol{f},\boldsymbol{p},\eta} \quad T, \tag{35}$$

$$\text{s.t.} \quad (19a) \quad (19g). \tag{35a}$$

We define $(T^*, \boldsymbol{t}^*, \boldsymbol{b}^*, \boldsymbol{f}^*, \boldsymbol{p}^*, \eta^*)$ as the optimal solution of problem (35). Then, $(\boldsymbol{t}^*, \boldsymbol{b}^*, \boldsymbol{f}^*, \boldsymbol{p}^*, \eta^*)$ is a feasible solution of problem (19) if $T^* \geq T$, where T is the maximum delay in constraint (19a). Otherwise, problem (19) is infeasible.

Although the completion time minimization problem (35) is still nonconvex due to constraints (19a)-(19b), we show that the globally optimal solution can be obtained by using the bisection method.

A. Optimal Resource Allocation

Lemma 2: Problem (35) with $T < T^*$ does not have a feasible solution (i.e., it is infeasible), while problem (35) with $T > T^*$ always has a feasible solution (i.e., it is feasible).

Proof: See Appendix E. □

According to Lemma 2, we can use the bisection method to obtain the optimal solution of problem (35).

With a fixed T, we still need to check whether there exists a feasible solution satisfying constraints (19a)-(19g). From constraints (19a) and (19c), we can see that it is always efficient to utilize the maximum computation capacity, i.e., $f_k^* = f_k^{\max}, \forall k \in \mathcal{K}$. From (19b) and (19d), we can see that minimizing the completion time occurs when $p_k^* = p_k^{\max}, \forall k \in \mathcal{K}$. Substituting the maximum computation capacity and maximum transmission power into (35), the completion time minimization problem becomes:

$$\min_{T,\boldsymbol{t},\boldsymbol{b},\eta} \quad T \tag{36}$$

$$\text{s.t.} \quad t_k \geq \frac{(1-\eta)T}{a} + \frac{A_k \log_2 \eta}{f_k^{\max}}, \quad \forall k \in \mathcal{K}, \tag{36a}$$

$$\frac{s}{t_k} \geq b_k \log_2 \left(1 + \frac{g_k p_k^{\max}}{N_0 b_k}\right), \quad \forall k \in \mathcal{K}, \tag{36b}$$

$$\sum_{k=1}^{K} b_k \geq B, \tag{36c}$$

$$0 \geq \eta \geq 1, \tag{36d}$$

$$t_k \preceq 0, b_k \preceq 0, \quad \forall k \in \mathcal{K}. \tag{36e}$$

Next, we provide the sufficient and necessary condition for the feasibility of set (36a)-(36e).

Lemma 3: With a fixed T, set (36a)-(36e) is nonempty if an only if

$$B \preceq \min_{0 \leq \eta \leq 1} \sum_{k=1}^{K} u_k(v_k(\eta)), \tag{37}$$

where

Algorithm 4 Completion Time Minimization

1: Initialize $T_{\min}$, $T_{\max}$, and the tolerance ϵ_5.
2: **repeat**
3: Set $T = \frac{T_{\min}+T_{\max}}{2}$.
4: Check the feasibility condition (40).
5: If set (36a)-(36e) has a feasible solution, set $T_{\max} = T$. Otherwise, set $T = T_{\min}$.
6: **until** $(T_{\max} \quad T_{\min})/T_{\max} \geq \epsilon_5$.

$$u_k(\eta) = \frac{(\ln 2)\eta}{W\left(\frac{(\ln 2)N_0\eta}{g_k p_k^{\max}} e^{-\frac{(\ln 2)N_0\eta}{g_k p_k^{\max}}}\right) + \frac{(\ln 2)N_0\eta}{g_k p_k^{\max}}}, \tag{38}$$

and

$$v_k(\eta) = \frac{s}{\frac{(1-\eta)T}{a} + \frac{A_k \log_2 \eta}{f_k^{\max}}}. \tag{39}$$

Proof: See Appendix F. □

To effectively solve (37) in Lemma 3, we provide the following lemma.

Lemma 4: In (38), $u_k(v_k(\eta))$ is a convex function.

Proof: See Appendix G. □

Lemma 4 implies that the optimization problem in (37) is a convex problem, which can be effectively solved. By finding the optimal solution of (37), the sufficient and necessary condition for the feasibility of set (36a)-(36e) can be simplified using the following theorem.

Theorem 4: Set (36a)-(36e) is nonempty if and only if

$$B \preceq \sum_{k=1}^{K} u_k(v_k(\eta^*)), \tag{40}$$

where η^* is the unique solution to $\sum_{k=1}^{K} u_k'(v_k(\eta^*))v_k'(\eta^*) = 0$.

Theorem 4 directly follows from Lemmas 3 and 4. Due to the convexity of function $u_k(v_k(\eta))$, $\sum_{k=1}^{K} u_k'(v_k(\eta^*))v_k'(\eta^*)$ is an increasing function of η^*. As a result, the unique solution of η^* to $\sum_{k=1}^{K} u_k'(v_k(\eta^*))v_k'(\eta^*) = 0$ can be effectively solved via the bisection method. Based on Theorem 4, the algorithm for obtaining the minimum completion time is summarized in Algorithm 4. Theorem 4 shows that the optimal FL accuracy level η^* meets the first-order condition $\sum_{k=1}^{K} u_k'(v_k(\eta^*))v_k'(\eta^*) = 0$, i.e., the optimal η^* should not be too small or too large for FL. This is because, for small η, the local computation time (number of iterations) becomes high as shown in Lemma 1. For large η, the transmission time is long due to the fact that a large number of global iterations is required as shown in Theorem 1.

B. Complexity Analysis

The major complexity of Algorithm 4 at each iteration lies in checking the feasibility condition (40). To check the inequality in (40), the optimal η^* needs to be obtained by using the bisection method, which involves the complexity of $\{ (K \log_2(1/\epsilon_4))$ with accuracy ϵ_4. As a result, the total complexity of Algorithm 4 is $\{ (K \log_2(1/\epsilon_4) \log_2(1/\epsilon_5))$, where ϵ_5 is the accuracy of the bisection method used in the outer layer. The complexity of Algorithm 4 is low since $\{ (K \log_2(1/\epsilon_4) \log_2(1/\epsilon_5))$ grows linearly with the total number of users.

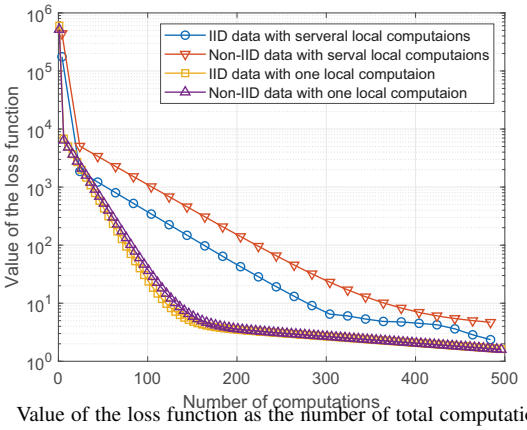

Fig. 4. Value of the loss function as the number of total computations varies for FL algorithms with IID and non-IID data.

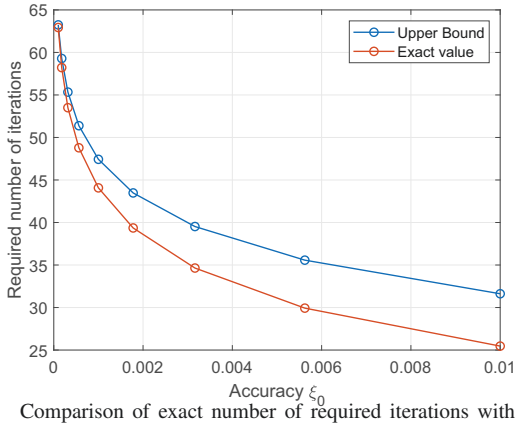

Fig. 5. Comparison of exact number of required iterations with the upper bound derived in Theorem 1.

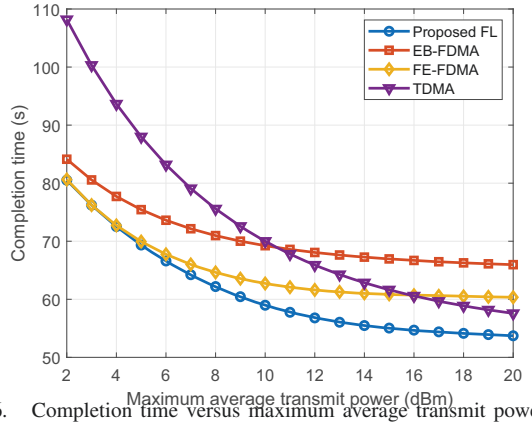

Fig. 6. Completion time versus maximum average transmit power of each user.

Similar to Algorithm 3 in Section III, Algorithm 4 is done at the BS side before executing the FL scheme in Algorithm 1, which will not affect the latency of the FL process.

V. Numerical Results

For our simulations, we deploy $K = 50$ users uniformly in a square area of size 500 m ≤ 500 m with the BS located at its center. The path loss model is $128.1 + 37.6 \log_{10} d$ (d is in km) and the standard deviation of shadow fading is 8 dB. In addition, the noise power spectral density is $N_0 = -174$ dBm/Hz. We use the real open blog feedback dataset in [39]. This dataset with a total number of 60,000 data samples originates from blog posts and the dimensional of each data sample is 281. The prediction task associated with the data is the prediction of the number of comments in the upcoming 24 hours. Parameter C_k is uniformly distributed in $[1, 3] \leq 10^4$ cycles/sample. The effective switched capacitance in local computation is $\kappa = 10^{-28}$ [35]. In Algorithm 1, we set $\xi = 1/10$, $\delta = 1/10$, and $\epsilon_0 = 10^{-3}$. Unless specified otherwise, we choose an equal maximum average transmit power $p_1^{\max} = \times\times\times = p_K^{\max} = p^{\max} = 10$ dB, an equal maximum computation capacity $f_1^{\max} = \times\times\times = f_K^{\max} = f^{\max} = 2$ GHz, a transmit data size $s = 28.1$ kbits, and a bandwidth $B = 20$ MHz. Each user has $D_k = 500$ data samples, which are randomly selected from the dataset with equal probability. All statistical results are averaged over 1000 independent runs.

A. Convergence Behavior

Fig. 4 shows the value of the loss function as the number of total computations varies with IID and non-IID data. For IID data case, all data samples are first shuffled and then partitioned into $60,000/500 = 120$ equal parts, and each device is assigned with one particular part. For non-IID data case [28], all data samples are first sorted by digit label and then divided into 240 shards of size 250, and each device is assigned with two shards. Note that the latter is a pathological non-IID partition way since most devices only obtain two kinds of digits. From this figure, it is observed that the FL algorithm with non-IID data shows similar convergence behavior with the FL algorithms with IID data. Besides, we find that the FL algorithm with multiple local updates requires more total computations than the FL algorithm with only one local update.

In Fig. 5, we show the comparison of exact number of required iterations with the upper bound derived in Theorem 1. According to this figure, it is found that the gap of the exact number of required iterations with the upper bound derived in Theorem 1 decreases as the accuracy ξ_0 decreases, which indicates that the upper bound is tight for small value of ξ_0.

B. Completion Time Minimization

We compare the proposed FL scheme with the FL FDMA scheme with equal bandwidth $b_1 = \times\times\times = b_K$ (labelled as 'EB-FDMA'), the FL FDMA scheme with fixed local accuracy $\eta = 1/2$ (labelled as 'FE-FDMA'), and the FL TDMA scheme in [34] (labelled as 'TDMA'). Fig. 6 shows how the completion time changes as the maximum average transmit power of each user varies. We can see that the completion time of all schemes decreases with the maximum average transmit power of each user. This is because a large maximum average transmit power can decrease the transmission time between users and the BS. We can clearly see that the proposed FL scheme achieves the best performance among all schemes. This is because the proposed approach jointly optimizes bandwidth and local accuracy η, while the bandwidth is fixed in EB-FDMA and η is not optimized in FE-FDMA. Compared to TDMA, the proposed approach can reduce the completion time by up to 27.3% due to the following two reasons. First, each user can directly transmit result data to the BS after local computation in FDMA, while the wireless transmission should be performed after the local computation for all users

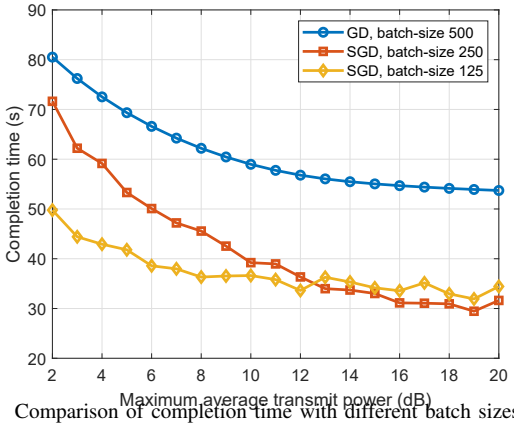

Fig. 7. Comparison of completion time with different batch sizes.

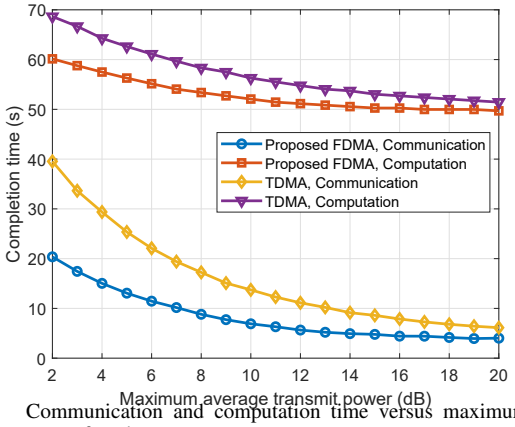

Fig. 8. Communication and computation time versus maximum average transmit power of each user.

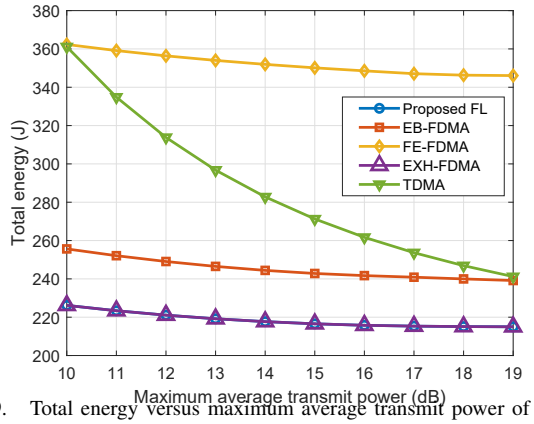

Fig. 9. Total energy versus maximum average transmit power of each user with $T = 100$ s.

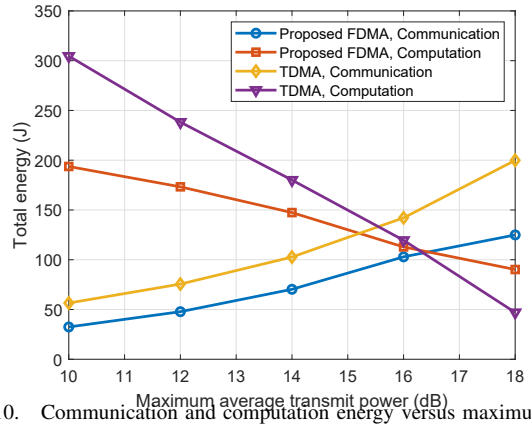

Fig. 10. Communication and computation energy versus maximum average transmit power of each user.

in TDMA, which needs a longer time compared to FDMA. Second, the noise power for users in FDMA is lower than in TDMA since each user is allocated to part of the bandwidth and each user occupies the whole bandwidth in TDMA, which indicates the transmission time in TDMA is longer than in FDMA.

Fig. 7 shows the completion time versus the maximum average transmit power with different batch sizes [40], [41]. The number of local iterations are keep fixed for these three schemes in Fig. 7. From this figure, it is found that the SGD method with smaller batch size (125 or 250) always outperforms the GD scheme, which indicates that it is efficient to use small batch size with SGD scheme.

Fig. 8 shows how the communication and computation time change as the maximum average transmit power of each user varies. It can be seen from this figure that both communication and computation time decrease with the maximum average transmit power of each user. We can also see that the computation time is always larger than communication time and the decreasing speed of communication time is faster than that of computation time.

C. Total Energy Minimization

Fig. 9 shows the total energy as function of the maximum average transmit power of each user. In this figure, the EXH-FDMA scheme is an exhaustive search method that can find a near optimal solution of problem (19), which refers to the proposed iterative algorithm with 1000 initial starting points. There are 1000 solutions obtained by using EXH-FDMA, and the solution with the best objective value is treated as the near optimal solution. From this figure, we can observe that the total energy decreases with the maximum average transmit power of each user. Fig. 9 also shows that the proposed FL scheme outperforms the EB-FDMA, FE-FDMA, and TDMA schemes. Moreover, the EXH-FDMA scheme achieves almost the same performance as the proposed FL scheme, which shows that the proposed approach achieves the optimum solution.

Fig. 10 shows the communication and computation energy as functions of the maximum average transmit power of each user. In this figure, it is found that the communication energy increases with the maximum average transmit power of each user, while the computation energy decreases with the maximum average transmit power of each user. For low maximum average transmit power of each user (less than 15 dB), FDMA outperforms TDMA in terms of both computation and communication energy consumption. For high maximum average transmit power of each user (higher than 17 dB), FDMA outperforms TDMA in terms of communication energy consumption, while TDMA outperforms FDMA in terms of computation energy consumption.

Fig. 11 shows the tradeoff between total energy consumption and completion time. In this figure, we compare the proposed scheme with the random selection (RS) scheme, where 25 users are randomly selected out from $K = 50$ users at each iteration. We can see that FDMA outperforms RS in terms of total energy consumption especially for low

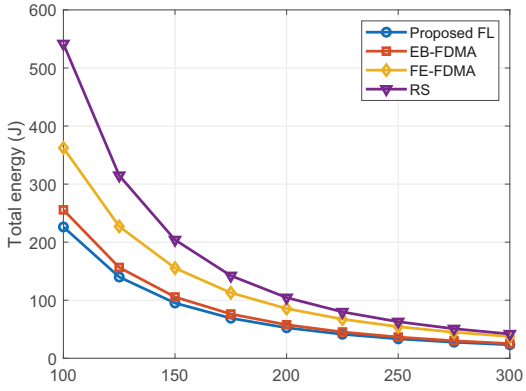

Fig. 11. Total energy versus completion time.

completion time. This is because, in FDMA all users can transmit data to the BS in each global iteration, while only half number of users can transmit data to the BS in each global iteration. As a result, the average transmission time for each user in FDMA is larger than that in RS, which can lead to lower transmission energy for the proposed algorithm. In particular, with given the same completion time, the proposed FL can reduce energy of up to 12.8%, 53.9%, and 59.5% compared to EB-FDMA, FE-FDMA, and RS, respectively.

VI. Conclusions

In this paper, we have investigated the problem of energy efficient computation and transmission resource allocation of FL over wireless communication networks. We have derived the time and energy consumption models for FL based on the convergence rate. With these models, we have formulated a joint learning and communication problem so as to minimize the total computation and transmission energy of the network. To solve this problem, we have proposed an iterative algorithm with low complexity, for which, at each iteration, we have derived closed-form solutions for computation and transmission resources. Numerical results have shown that the proposed scheme outperforms conventional schemes in terms of total energy consumption, especially for small maximum average transmit power.

Appendix A
Proof of Lemma 1

Based on (12), we have:

$$G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i+1)}) \overset{A1}{\geq} G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}) - \delta \left(G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}) \right)^2 + \frac{L\delta^2}{2} \left(G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}) \right)^2 = G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}) - \frac{(2 - L\delta)\delta}{2} \left(G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}) \right)^2. \quad \text{(A.1)}$$

Similar to (B.2) in the following Appendix B, we can prove that:

$$\nabla G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k) \preceq \gamma (G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k) - G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*})), \quad \text{(A.2)}$$

where $\boldsymbol{h}_k^{(n)*}$ is the optimal solution of problem (11). Based on (A.1) and (A.2), we have:

$$\begin{aligned} & G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i+1)}) - G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*}) \\ & \geq G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}) - G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*}) \\ & \quad - \frac{(2 - L\delta)\delta\gamma}{2} (G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}) - G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*})) \\ & \geq \left(1 - \frac{(2 - L\delta)\delta\gamma}{2}\right)^{i+1} (G_k(\boldsymbol{w}^{(n)}, \boldsymbol{0}) - G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*})) \\ & \geq \exp\left(-(i+1)\frac{(2 - L\delta)\delta\gamma}{2}\right) (G_k(\boldsymbol{w}^{(n)}, \boldsymbol{0}) - G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*})), \end{aligned} \quad \text{(A.3)}$$

where the last inequality follows from the fact that $1 - x \geq \exp(-x)$. To ensure that $G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n),(i)}) - G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*}) \geq \eta(G_k(\boldsymbol{w}^{(n)}, \boldsymbol{0}) - G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*}))$, we have (15).

Appendix B
Proof of Theorem 1

Before proving Theorem 1, the following lemma is provided.

Lemma 5: Under the assumptions A1 and A2, the following conditions hold:

$$\begin{aligned} & \frac{1}{L}\nabla F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}^{(n)}) - \nabla F_k(\boldsymbol{w}^{(n)}) \\ & \geq (\nabla F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}^{(n)}) - \nabla F_k(\boldsymbol{w}^{(n)}))^T \boldsymbol{h}^{(n)} \\ & \geq \frac{1}{\gamma}\nabla F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}^{(n)}) - \nabla F_k(\boldsymbol{w}^{(n)}), \end{aligned} \quad \text{(B.1)}$$

and

$$\nabla F(\boldsymbol{w}) - \nabla F(\boldsymbol{w}^*) \preceq \gamma (F(\boldsymbol{w}) - F(\boldsymbol{w}^*)). \quad \text{(B.2)}$$

Proof: According to the Lagrange median theorem, there always exists a $\boldsymbol{w}$ such that

$$\nabla F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}^{(n)}) - \nabla F_k(\boldsymbol{w}^{(n)}) = \nabla^2 F_k(\boldsymbol{w}) \boldsymbol{h}^{(n)}. \quad \text{(B.3)}$$

Combining assumptions A1, A2, and (B.3) yields (B.1).

For the optimal solution $\boldsymbol{w}^*$ of $F(\boldsymbol{w}^*)$, we always have $\nabla F(\boldsymbol{w}^*) = \boldsymbol{0}$. Combining (9) and (B.1), we also have $\gamma \boldsymbol{I} \succeq \nabla^2 F(\boldsymbol{w})$, which indicates that

$$\nabla F(\boldsymbol{w}) - \nabla F(\boldsymbol{w}^*) \preceq \gamma \nabla \boldsymbol{w} - \boldsymbol{w}^* \nabla, \quad \text{(B.4)}$$

and

$$F(\boldsymbol{w}^*) \preceq F(\boldsymbol{w}) + \nabla F(\boldsymbol{w})^T (\boldsymbol{w}^* - \boldsymbol{w}) + \frac{\gamma}{2} \nabla \boldsymbol{w}^* - \boldsymbol{w} \nabla^2. \quad \text{(B.5)}$$

As a result, we have:

$$\begin{aligned} \nabla F(\boldsymbol{w}) \nabla^2 &= \nabla F(\boldsymbol{w}) - \nabla F(\boldsymbol{w}^*) \nabla^2 \\ &\overset{\text{(B.4)}}{\preceq} \gamma \nabla F(\boldsymbol{w}) - \nabla F(\boldsymbol{w}^*) \nabla \nabla \boldsymbol{w} - \boldsymbol{w}^* \nabla \\ &\preceq \gamma (\nabla F(\boldsymbol{w}) - \nabla F(\boldsymbol{w}^*))^T (\boldsymbol{w} - \boldsymbol{w}^*) \\ &= \gamma \nabla F(\boldsymbol{w})^T (\boldsymbol{w} - \boldsymbol{w}^*) \\ &\overset{\text{(B.5)}}{\preceq} \gamma (F(\boldsymbol{w}) - F(\boldsymbol{w}^*)), \end{aligned} \quad \text{(B.6)}$$

which proves (B.2). □

For the optimal solution of problem (11), the first-order derivative condition always holds, i.e.,

$$\nabla F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k^{(n)*}) - \nabla F_k(\boldsymbol{w}^{(n)}) + \xi \nabla F(\boldsymbol{w}^{(n)}) = \boldsymbol{0}. \quad \text{(B.7)}$$

We are new ready to prove Theorem 1. With the above inequalities and equalities, we have:

$$
\begin{aligned}
&F(\boldsymbol{w}^{(n+1)}) \\
&\overset{(10),A1}{\geq} F(\boldsymbol{w}^{(n)}) + \frac{1}{K}\sum_{k=1}^{K} F(\boldsymbol{w}^{(n)})^T \boldsymbol{h}_k^{(n)} + \frac{L}{2K^2}\left(\sum_{k=1}^{K} \boldsymbol{h}_k^{(n)}\right)^2 \\
&\overset{(11)}{=} F(\boldsymbol{w}^{(n)}) + \frac{1}{K\xi}\sum_{k=1}^{K}\Big[G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)}) + F_k(\boldsymbol{w}^{(n)})\boldsymbol{h}_k^{(n)} \\
&\quad F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k^{(n)})\Big) + \frac{L}{2K^2}\left(\sum_{k=1}^{K} \boldsymbol{h}_k^{(n)}\right)^2 \\
&\overset{A2}{\geq} F(\boldsymbol{w}^{(n)}) + \frac{1}{K\xi}\sum_{k=1}^{K}\Big[G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)}) \quad F_k(\boldsymbol{w}^{(n)}) \\
&\quad \frac{\gamma}{2}\nabla \boldsymbol{h}_k^{(n)}\nabla^2\Big) + \frac{L}{2K^2}\left(\sum_{k=1}^{K} \boldsymbol{h}_k^{(n)}\right)^2. \qquad \text{(B.8)}
\end{aligned}
$$

According to the triangle inequality and mean inequality, we have

$$
\left(\frac{1}{K}\sum_{k=1}^{K} \boldsymbol{h}_k^{(n)}\right)^2 \geq \Big)\frac{1}{K}\sum_{k=1}^{K}\left(\boldsymbol{h}_k^{(n)}\right)^2 \geq \frac{1}{K}\sum_{k=1}^{K}\left(\boldsymbol{h}_k^{(n)}\right)^2. \qquad \text{(B.9)}
$$

Combining (B.8) and (B.9) yields:

$$
\begin{aligned}
&F(\boldsymbol{w}^{(n+1)}) \\
&\geq F(\boldsymbol{w}^{(n)}) + \frac{1}{K\xi}\sum_{k=1}^{K}\Big[G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)}) \quad F_k(\boldsymbol{w}^{(n)}) \\
&\quad \frac{\gamma \quad L\xi}{2}\nabla \boldsymbol{h}_k^{(n)}\nabla^2\Big) \\
&\overset{(11)}{=} F(\boldsymbol{w}^{(n)}) + \frac{1}{K\xi}\sum_{k=1}^{K}\Big[G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)}) \quad G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*}) \\
&\quad (G_k(\boldsymbol{w}^{(n)}, \mathbf{0}) \quad G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*})) \quad \frac{\gamma \quad L\xi}{2}\nabla \boldsymbol{h}_k^{(n)}\nabla^2\Big) \\
&\overset{(14)}{\geq} F(\boldsymbol{w}^{(n)}) \quad \frac{1}{K\xi}\sum_{k=1}^{K}\Big]\frac{\gamma \quad L\xi}{2}\nabla \boldsymbol{h}_k^{(n)}\nabla^2 \\
&\quad + (1 \quad \eta)(G_k(\boldsymbol{w}^{(n)}, \mathbf{0}) \quad G_k(\boldsymbol{w}^{(n)}, \boldsymbol{h}_k^{(n)*}))\Big) \\
&\overset{(11)}{=} F(\boldsymbol{w}^{(n)}) \quad \frac{1}{K\xi}\sum_{k=1}^{K}\Big]\frac{\gamma \quad L\xi}{2}\nabla \boldsymbol{h}_k^{(n)}\nabla^2 \\
&\quad + (1 \quad \eta)(F_k(\boldsymbol{w}^{(n)}) \quad F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k^{(n)*}) \\
&\quad + (\quad F_k(\boldsymbol{w}^{(n)}) \quad \xi \quad F(\boldsymbol{w}^{(n)}))^T \boldsymbol{h}_k^{(n)*}\Big) \\
&\overset{(B.7)}{=} F(\boldsymbol{w}^{(n)}) \quad \frac{1}{K\xi}\sum_{k=1}^{K}\Big]\frac{\gamma \quad L\xi}{2}\nabla \boldsymbol{h}_k^{(n)}\nabla^2 \\
&\quad + (1 \quad \eta)(F_k(\boldsymbol{w}^{(n)}) \quad F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k^{(n)*}) \\
&\quad + \quad F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k^{(n)*})^T \boldsymbol{h}_k^{(n)*}\Big). \qquad \text{(B.10)}
\end{aligned}
$$

Based on assumption A2, we can obtain:

$$
\begin{aligned}
&F_k(\boldsymbol{w}^{(n)}) \preceq F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k^{(n)*}) \\
&\quad + \quad F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k^{(n)*})^T(\quad \boldsymbol{h}_k^{(n)*}) + \frac{\gamma}{2}\nabla \boldsymbol{h}_k^{(n)*}\nabla^2. \qquad \text{(B.11)}
\end{aligned}
$$

Applying (B.11) to (B.10), we can obtain:

$$
\begin{aligned}
&F(\boldsymbol{w}^{(n+1)}) \geq F(\boldsymbol{w}^{(n)}) \quad \frac{1}{K\xi}\sum_{k=1}^{K}\Big]\frac{\gamma \quad L\xi}{2}\nabla \boldsymbol{h}_k^{(n)}\nabla^2 \\
&\quad + \frac{(1 \quad \eta)\gamma}{2}\nabla \boldsymbol{h}_k^{(n)*}\nabla^2\Big). \qquad \text{(B.12)}
\end{aligned}
$$

From (B.1), the following relationship is obtained:

$$
\nabla \boldsymbol{h}_k^{(n)*}\nabla^2 \preceq \frac{1}{L^2}\nabla \quad F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k^{(n)*})) \qquad F_k(\boldsymbol{w}^{(n)})\nabla^2. \qquad \text{(B.13)}
$$

For the constant parameter ξ, we choose

$$
\gamma \quad L\xi > 0. \qquad \text{(B.14)}
$$

According to (B.14) and (B.12), we can obtain:

$$
\begin{aligned}
&F(\boldsymbol{w}^{(n+1)}) \geq F(\boldsymbol{w}^{(n)}) \quad \frac{(1 \quad \eta)\gamma}{2K\xi}\sum_{k=1}^{K}\nabla \boldsymbol{h}_k^{(n)*}\nabla^2 \\
&\overset{(B.13)}{\geq} F(\boldsymbol{w}^{(n)}) \quad \frac{(1 \quad \eta)\gamma}{2KL^2\xi}\sum_{k=1}^{K}\Big] \\
&\quad \nabla \quad F_k(\boldsymbol{w}^{(n)} + \boldsymbol{h}_k^{(n)*}) \qquad F_k(\boldsymbol{w}^{(n)})\nabla^2\Big) \\
&\overset{(B.7)}{=} F(\boldsymbol{w}^{(n)}) \quad \frac{(1 \quad \eta)\gamma\xi}{2L^2}\nabla \quad F(\boldsymbol{w}^{(n)})\nabla^2 \\
&\overset{(B.2)}{\geq} F(\boldsymbol{w}^{(n)}) \quad \frac{(1 \quad \eta)\gamma^2\xi}{2L^2}(F(\boldsymbol{w}^{(n)}) \quad F(\boldsymbol{w}^*)). \qquad \text{(B.15)}
\end{aligned}
$$

Based on (B.15), we get:

$$
\begin{aligned}
&F(\boldsymbol{w}^{(n+1)}) \quad F(\boldsymbol{w}^*) \\
&\geq \Big)1 \quad \frac{(1 \quad \eta)\gamma^2\xi}{2L^2}\Big[\ (F(\boldsymbol{w}^{(n)}) \quad F(\boldsymbol{w}^*)) \\
&\geq \Big)1 \quad \frac{(1 \quad \eta)\gamma^2\xi}{2L^2}\Big[^{n+1} \quad (F(\boldsymbol{w}^{(0)}) \quad F(\boldsymbol{w}^*)) \\
&\geq \exp\Big) \quad (n+1)\frac{(1 \quad \eta)\gamma^2\xi}{2L^2}\Big[\ (F(\boldsymbol{w}^{(0)}) \quad F(\boldsymbol{w}^*)), \qquad \text{(B.16)}
\end{aligned}
$$

where the last inequality follows from the fact that $1 \quad x \geq \exp(\quad x)$. To ensure that $F(\boldsymbol{w}^{(n+1)}) \quad F(\boldsymbol{w}^*) \geq \epsilon_0(F(\boldsymbol{w}^{(0)}) \quad F(\boldsymbol{w}^*))$, we have (17).

Appendix C
Proof of Theorem 2

According to (20), transmitting with minimal time is always energy efficient, i.e., the optimal time allocation is $t_k^* = t_k^{\min}$. Substituting $t_k^* = t_k^{\min}$ into problem (20) yields:

$$
\begin{aligned}
\min_{\eta} \quad & \frac{\alpha_1 \log_2(1/\eta) + \alpha_2}{1 \quad \eta} && \text{(C.1)} \\
\text{s.t.} \quad & t_k^{\min} \geq \beta_k(\eta), \quad \mathcal{D}k \ \forall \ \mathcal{L}, && \text{(C.1a)} \\
& 0 \geq \eta \geq 1, && \text{(C.1b)}
\end{aligned}
$$

where

$$
\begin{aligned}
&\alpha_1 = a \quad \kappa A_k f_k^2, \alpha_2 = a \quad t_k^{\min} p_k, \\
&\beta_k(\eta) = \frac{(1 \quad \eta)T}{a} + \frac{A_k \log_2 \eta}{f_k^{\max}}. \qquad \text{(C.2)}
\end{aligned}
$$

From (C.2), it can be verified that $\beta_k(\eta)$ is a concave function. Due to the concavity of $\beta_k(\eta)$, constraints (C.1a) can be equivalently transformed to:

$$\eta_k^{\min} \geq \eta \geq \eta_k^{\max}, \tag{C.3}$$

where $\beta_k(\eta_k^{\min}) = \beta_k(\eta_k^{\max}) = t_k^{\min}$ and $t_k^{\min} \geq t_k^{\max}$. Since $\beta_k(1) = 0$, $\lim_{\eta \to 0+} \beta_k(\eta) = \quad \in$ and $t_k^{\min} > 0$, we always have $0 < t_k^{\min} \geq t_k^{\max} < 1$. With the help of (C.3), problem (C.1) can be simplified as (23).

APPENDIX D
PROOF OF THEOREM 3

To minimize $\sum_{k=1}^{K} t_k p_k$, transmit power p_k needs to be minimized. To minimize p_k from (28a), we have:

$$p_k^* = \frac{N_0 b_k}{g_k} \Big) 2^{\frac{s}{t_k b_k}} \quad 1 \Big(. \tag{D.1}$$

The first order and second order derivatives of p_k^* can be respectively given by

$$\frac{\partial p_k^*}{\partial b_k} = \frac{N_0}{g_k} \Big) e^{\frac{(\ln 2)s}{t_k b_k}} \quad 1 \quad \frac{(\ln 2)s}{t_k b_k} e^{\frac{(\ln 2)s}{t_k b_k}} \Big[, \tag{D.2}$$

and

$$\frac{\partial^2 p_k^*}{\partial b_k^2} = \frac{N_0 (\ln 2)^2 s^2}{g_k t_k^2 b_k^3} e^{\frac{(\ln 2)s}{t_k b_k}} \preceq 0. \tag{D.3}$$

From (D.3), we can see that $\frac{\partial p_k^*}{\partial b_k}$ is an increasing function of b_k. Since $\lim_{b_k \to 0+} \frac{\partial p_k^*}{\partial b_k} = 0$, we have $\frac{\partial p_k^*}{\partial b_k} < 0$ for $0 < b_k < \in$, i.e., p_k^* in (D.1) is a decreasing function of b_k. Thus, maximum transmit power constraint $p_k^* \geq p_k^{\max}$ is equivalent to:

$$b_k \preceq b_k^{\min} \triangleq \quad \frac{(\ln 2)s}{t_k W \Big) \frac{(\ln 2)N_0 s}{g_k p_k^{\max} t_k} \mathrm{e}^{-\frac{(\ln 2)N_0 s}{g_k p_k^{\max} t_k}} \Big[+ \frac{(\ln 2)N_0 s}{g_k p_k^{\max}}}, \tag{D.4}$$

where $b_k^{\min}$ is defined in (32).

Substituting (D.1) into problem (28), we can obtain:

$$\min_{\boldsymbol{b}} \quad \sum_{k=1}^{K} \frac{N_0 t_k b_k}{g_k} \Big) 2^{\frac{s}{b_k t_k}} \quad 1 \Big(, \tag{D.5}$$

$$\text{s.t.} \quad \sum_{k=1}^{K} b_k \geq B, \tag{D.5a}$$

$$b_k \preceq b_k^{\min}, \quad \forall k \in \mathcal{L}, \tag{D.5b}$$

According to (D.3), problem (D.5) is a convex function, which can be effectively solved by using the Karush-Kuhn-Tucker conditions. The Lagrange function of (D.5) is:

$$\mathcal{O}(\boldsymbol{b}, \mu) = \sum_{k=1}^{K} \frac{N_0 t_k b_k}{g_k} \Big) 2^{\frac{s}{b_k t_k}} \quad 1 \Big(+ \mu \Big) \sum_{k=1}^{K} b_k \quad B, \tag{D.6}$$

where μ is the Lagrange multiplier associated with constraint (D.5a). The first order derivative of $\mathcal{O}(\boldsymbol{b}, \mu)$ with respect to b_k is:

$$\frac{\partial \mathcal{O}(\boldsymbol{b}, \mu)}{\partial b_k} = \frac{N_0}{g_k t_k} \Big) e^{\frac{(\ln 2)s}{t_k b_k}} \quad 1 \quad \frac{(\ln 2)s}{t_k b_k} e^{\frac{(\ln 2)s}{t_k b_k}} \Big[+ \mu. \tag{D.7}$$

We define $b_k(\mu)$ as the unique solution to $\frac{\partial \mathcal{L}(\boldsymbol{b}, \mu)}{\partial b_k} = 0$. Given constraint (D.5b), the optimal b_k^* can be founded from (30). Since the objective function (D.5) is a decreasing function of b_k, constrain (D.5a) always holds with equality for the optimal solution, which shows that the optimal Lagrange multiplier is obtained by solving (34).

APPENDIX E
PROOF OF LEMMA 2

Assume that problem (35) with $T = \bar{T} < T^*$ is feasible, and the feasible solution is $(\bar{T}, \bar{\boldsymbol{t}}, \bar{\boldsymbol{b}}, \bar{\boldsymbol{f}}, \bar{\boldsymbol{p}}, \bar{\eta})$. Then, the solution $(\bar{T}, \bar{\boldsymbol{t}}, \bar{\boldsymbol{b}}, \bar{\boldsymbol{f}}, \bar{\boldsymbol{p}}, \bar{\eta})$ is feasible with lower value of the objective function than solution $(T^*, \boldsymbol{t}^*, \boldsymbol{b}^*, \boldsymbol{f}^*, \boldsymbol{p}^*, \eta^*)$, which contradicts the fact that $(T^*, \boldsymbol{t}^*, \boldsymbol{b}^*, \boldsymbol{f}^*, \boldsymbol{p}^*, \eta^*)$ is the optimal solution.

For problem (35) with $T = \bar{T} > T^*$, we can always construct a feasible solution $(\bar{T}, \boldsymbol{t}^*, \boldsymbol{b}^*, \boldsymbol{f}^*, \boldsymbol{p}^*, \eta^*)$ to problem (35) by checking all constraints.

APPENDIX F
PROOF OF LEMMA 3

To prove this, we first define function

$$y = x \ln \Big) 1 + \frac{1}{x} \Big[, \quad x > 0. \tag{F.1}$$

Then, we have

$$y' = \ln \Big) 1 + \frac{1}{x} \Big[\quad \frac{1}{x+1}, y'' = \quad \frac{1}{x(x+1)^2} < 0. \tag{F.2}$$

According to (F.2), y' is a decreasing function. Since $\lim_{t_i \to +\infty} y' = 0$, we can obtain that $y' > 0$ for all $0 < x < +\in$. As a result, y is an increasing function, i.e., the right hand side of (36b) is an increasing function of bandwidth b_k.

To ensure that the maximum bandwidth constraint (36c) can be satisfied, the left hand side of (36b) should be as small as possible, i.e., t_k should be as long as possible. Based on (36a), the optimal time allocation should be:

$$t_k^* = \frac{(1 \quad \eta)T}{a} + \frac{A_k \log_2 \eta}{f_k^{\max}}, \quad \forall k \in \mathcal{L}. \tag{F.3}$$

Substituting (F.3) into (36b), we can construct the following problem:

$$\min_{\boldsymbol{b}, \eta} \quad \sum_{k=1}^{K} b_k \tag{F.4}$$

$$\text{s.t.} \quad v_k(\eta) \geq b_k \log_2 \Big) 1 + \frac{g_k p_k^{\max}}{N_0 b_k} \Big[, \quad \forall k \in \mathcal{L}, \tag{F.4a}$$

$$0 \geq \eta \geq 1, \tag{F.4b}$$

$$b_k \preceq 0, \quad \forall k \in \mathcal{L}, \tag{F.4c}$$

where $v_k(\eta)$ is defined in (39). We can observe that set (36a)-(36e) is nonempty if an only if the optimal objective value of (F.4) is less than B. Since the right hand side of (36b) is an increasing function, (36b) should hold with equality for the optimal solution of problem (F.4). Setting (36b) with equality, problem (F.4) reduces to (37).

APPENDIX G
PROOF OF LEMMA 4

We first prove that $v_k(\eta)$ is a convex function. To show this, we define:

$$\phi(\eta) = \frac{s}{\eta}, \quad 0 \geq \eta \geq 1, \tag{G.1}$$

and

$$\varphi_k(\eta) = \frac{(1 \quad \eta)T}{a} + \frac{A_k \log_2 \eta}{f_k^{\max}}, \quad 0 \geq \eta \geq 1. \tag{G.2}$$

According to (39), we have: $v_k(\eta) = \phi(\varphi_k(\eta))$. Then, the second-order derivative of $v_k(\eta)$ is:

$$v_k''(\eta) = \phi''(\varphi_k(\eta))(\varphi_k'(\eta))^2 + \phi'(\varphi_k(\eta))\varphi_k''(\eta). \tag{G.3}$$

According to (G.1) and (G.2), we have:

$$\phi'(\eta) = \frac{s}{\eta^2} \geq 0, \quad \phi''(\eta) = \frac{2s}{\eta^3} \preceq 0, \tag{G.4}$$

$$\varphi_k''(\eta) = \frac{A_k}{(\ln 2) f_k^{\max} \eta^2} \geq 0. \tag{G.5}$$

Combining (G.3)-(G.5), we can find that $v_k''(\eta) \preceq 0$, i.e., $v_k(\eta)$ is a convex function.

Then, we can show that $u_k(\eta)$ is an increasing and convex function. According to Appendix B, $u_k(\eta)$ is the inverse function of the right hand side of (36b). If we further define function:

$$z_k(\eta) = \eta \log_2 \Big) 1 + \frac{g_k p_k^{\max}}{N_0 \eta} \Big[, \quad \eta \preceq 0, \tag{G.6}$$

$u_k(\eta)$ is the inverse function of $z_k(\eta)$, which gives $u_k(z_k(\eta)) = \eta$.

According to (F.1) and(F.2) in Appendix B, function $z_k(\eta)$ is an increasing and concave function, i.e., $z_k'(\eta) \preceq 0$ and $z_k''(\eta) \geq 0$. Since $z_k(\eta)$ is an increasing function, its inverse function $u_k(\eta)$ is also an increasing function.

Based on the definition of concave function, for any $\eta_1 \preceq 0$, $\eta_2 \preceq 0$ and $0 \geq \theta \geq 1$, we have:

$$z_k(\theta\eta_1 + (1 \quad \theta)\eta_2) \preceq \theta z_k(\eta_1) + (1 \quad \theta) z_k(\eta_2). \tag{G.7}$$

Applying the increasing function $u_k(\eta)$ on both sides of (G.7) yields:

$$\theta\eta_1 + (1 \quad \theta)\eta_2 \preceq u_k(\theta z_k(\eta_1) + (1 \quad \theta) z_k(\eta_2)). \tag{G.8}$$

Denote $\bar{\eta}_1 = z_k(\eta_1)$ and $\bar{\eta}_2 = z_k(\eta_2)$, i.e., we have $\eta_1 = u_k(\bar{\eta}_1)$ and $\eta_2 = u_k(\bar{\eta}_2)$. Thus, (G.8) can be rewritten as:

$$\theta u_k(\bar{\eta}_1) + (1 \quad \theta) u_k(\bar{\eta}_1) \preceq u_k(\theta\bar{\eta}_1 + (1 \quad \theta)\bar{\eta}_2), \tag{G.9}$$

which indicates that $u_k(\eta)$ is a convex function. As a result, we have proven that $u_k(\eta)$ is an increasing and convex function, which shows:

$$u_k'(\eta) \preceq 0, \quad u_k''(\eta) \preceq 0. \tag{G.10}$$

To show the convexity of $u_k(v_k(\eta))$, we have:

$$u_k''(v_k(\eta)) = u_k''(v_k(\eta))(v_k'(\eta))^2 + u_k'(v_k(\eta)) v_k''(\eta) \preceq 0, \tag{G.11}$$

according to $v_k''(\eta) \preceq 0$ and (G.10). As a result, $u_k(v_k(\eta))$ is a convex function.

REFERENCES

[1] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, "Delay minimization for federated learning over wireless communication networks," *Proc. Int. Conf. Machine Learning Workshop*, 2020.

[2] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "When edge meets learning: Adaptive control for resource-constrained distributed machine learning," in *IEEE Conf. Computer Commun.*, Honolulu, HI, USA, Apr. 2018, pp. 63–71.

[3] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Commun. Surveys & Tut.*, vol. 21, no. 4, pp. 3072–3108, June 2019.

[4] Y. Liu, S. Bi, Z. Shi, and L. Hanzo, "When machine learning meets big data: A wireless communication perspective," *arXiv preprint arXiv:1901.08329*, 2019.

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

[6] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Network*, to appear, 2019.

[7] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *Proceedings of the IEEE*, vol. 107, no. 11, pp. 2204–2239, Nov. 2019.

[8] M. Chen, O. Semiari, W. Saad, X. Liu, and C. Yin, "Federated echo state learning for minimizing breaks in presence in wireless virtual reality networks," *IEEE Trans. Wireless Commun.*, to appear, 2019.

[9] H. Tembine, *Distributed strategic learning for wireless engineers*. CRC Press, 2018.

[10] S. Samarakoon, M. Bennis, W. Saad, and M. Debbah, "Distributed federated learning for ultra-reliable low-latency vehicular communications," *arXiv preprint arXiv:1807.08127*, 2018.

[11] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, "Machine learning in the air," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2184–2199, Oct. 2019.

[12] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Commun. Surveys Tut.*, pp. 1–1, 2019.

[13] M. S. H. Abad, E. Ozfatura, D. Gündüz, and O. Ercetin, "Hierarchical federated learning across heterogeneous cellular networks," *arXiv preprint arXiv:1909.02362*, 2019.

[14] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *Proc. IEEE Int. Conf. Commun.* Shanghai, China: IEEE, May 2019, pp. 1–7.

[15] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for on-device federated learning," *arXiv preprint arXiv:1910.06378*, 2019.

[16] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-efficient learning of deep networks from decentralized data," *arXiv preprint arXiv:1602.05629*, 2016.

[17] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Advances Neural Information Process. Syst.* Curran Associates, Inc., 2017, pp. 4424–4434.

[18] H. H. Yang, Z. Liu, T. Q. S. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Trans. Commun.*, to appear, 2019.

[19] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, June 2019.

[20] E. Jeong, S. Oh, J. Park, H. Kim, M. Bennis, and S. Kim, "Multi-hop federated private data augmentation with sample compression," *arXiv preprint arXiv:1907.06426*, 2019.

[21] J.-H. Ahn, O. Simeone, and J. Kang, "Wireless federated distillation for distributed edge learning with heterogeneous data," *arXiv preprint arXiv:1907.02745*, 2019.

[22] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1046–1061, May 2017.

[23] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile unmanned aerial vehicles (uavs) for energy-efficient internet of things communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7574–7589, Nov. 2017.

[24] Z. Yang, C. Pan, M. Shikh-Bahaei, W. Xu, M. Chen, M. Elkashlan, and A. Nallanathan, "Joint altitude, beamwidth, location, and bandwidth optimization for UAV-enabled communications," *IEEE Commun. Lett.*, vol. 22, no. 8, pp. 1716–1719, 2018.

[25] J. Konečnỳ, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.

[26] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," in *Proc. IEEE Int. Symp. Information Theory*. Paris, France: IEEE, Jan. 2019, pp. 1432–1436.

[27] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Towards an intelligent edge: Wireless communication meets machine learning," *arXiv preprint arXiv:1809.00343*, 2018.

[28] G. Zhu, Y. Wang, and K. Huang, "Low-latency broadband analog aggregation for federated edge learning," *arXiv preprint arXiv:1812.11494*, 2018.

[29] 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," *arXiv preprint arXiv:1909.12567*, 2019.

[30] Y. Sun, S. Zhou, and D. Gündüz, "Energy-aware analog aggregation for federated learning with redundant data," *arXiv preprint arXiv:1911.00188*, 2019.

[31] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *arXiv preprint arXiv:1812.11750*, 2018.

[32] Q. Zeng, Y. Du, K. K. Leung, and K. Huang, "Energy-efficient radio resource allocation for federated edge learning," *arXiv preprint arXiv:1907.06040*, 2019.
[33] 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.
[34] N. H. Tran, W. Bao, A. Zomaya, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *Proc. IEEE Conf. Computer Commun.*, Paris, France, June 2019, pp. 1387–1395.
[35] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.
[36] O. Shamir, N. Srebro, and T. Zhang, "Communication-efficient distributed optimization using an approximate newton-type method," in *Proc. Int. Conf. Machine Learning*, Beijing, China, June 2014, pp. 1000–1008.
[37] Z. Allen-Zhu, "Natasha 2: Faster non-convex optimization than SGD," in *Advances in neural information processing systems*, 2018, pp. 2675–2686.
[38] W. Dinkelbach, "On nonlinear fractional programming," *Management Science*, vol. 13, no. 7, pp. 492–498, 1967.
[39] K. Buza, "Feedback prediction for blogs," in *Data analysis, machine learning and knowledge discovery*. Springer, 2014, pp. 145–152.
[40] C. J. Shallue, J. Lee, J. Antognini, J. Sohl-Dickstein, R. Frostig, and G. E. Dahl, "Measuring the effects of data parallelism on neural network training," *arXiv preprint arXiv:1811.03600*, 2018.
[41] T. Lin, S. U. Stich, K. K. Patel, and M. Jaggi, "Don't use large mini-batches, use local SGD," *arXiv preprint arXiv:1808.07217*, 2018.

Zhaohui Yang (Member, IEEE) received the B.S. degree in information science and engineering from Chien-Shiung Wu Honors College, Southeast University, Nanjing, China, in June 2014, and the Ph.D. degree in communication and information system with the National Mobile Communications Research Laboratory, Southeast University, Nanjing, China, in May 2018. He is currently a Postdoctoral Research Associate with the Center for Telecommunications Research, Department of Informatics, King's College London, U.K., where he has been involved in the EPSRC SENSE Project. He has guest edited a 2020 IEEE Feature topic of IEEE Communications Magazine on Communication Technologies for Efficient Edge Learning. He serves as the Co-Chair for IEEE International Conference on Communications (ICC), 1st Workshop on Edge Machine Learning for 5G Mobile Networks and Beyond, June 2020, IEEE International Symposium on Personal, Indoor and Mobile Radio Communication (PIMRC) Workshop on Rate-Splitting and Robust Interference Management for Beyond 5G, August 2020. His research interests include federated learning, reconfigurable intelligent surface, UAV, MEC, energy harvesting, and NOMA. He was a TPC member of IEEE ICC during 2015-2020 and Globecom during 2017-2020. He was an exemplary reviewer for IEEE TRANSACTIONS ON COMMUNICATIONS in 2019.

Mingzhe Chen (Member, IEEE) received the Ph.D. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2019. From 2016 to 2019, he was a Visiting Researcher at the Department of Electrical and Computer Engineering, Virginia Tech. He is currently a Post-Doctoral Researcher at the Electrical Engineering Department, Princeton University and at the Chinese University of Hong Kong, Shenzhen, China. His research interests include federated learning, reinforcement learning, virtual reality, unmanned aerial vehicles, and wireless networks. He was a recipient of the IEEE International Conference on Communications (ICC) 2020 Best Paper Award and an exemplary reviewer for IEEE Transactions on Wireless Communications in 2018 and IEEE Transactions on Communications in 2018 and 2019. He has served as a co-chair for workshops on edge learning and wireless communications in several conferences including IEEE ICC, IEEE Global Communications Conference, and IEEE Wireless Communications and Networking Conference. He currently serves as an associate editor for Frontiers in Communications and Networks, Specialty Section on Data Science for Communications and a guest editor for IEEE Journal on Selected Areas in Communications (JSAC) Special Issue on Distributed Learning over Wireless Edge Networks.

Walid Saad (Fellow, IEEE) received his Ph.D degree from the University of Oslo in 2010. He is currently a Professor at the Department of Electrical and Computer Engineering at Virginia Tech, where he leads the Network sciEnce, Wireless, and Security (NEWS) laboratory. His research interests include wireless networks, machine learning, game theory, security, unmanned aerial vehicles, cyber-physical systems, and network science. Dr. Saad is a Fellow of the IEEE and an IEEE Distinguished Lecturer. He is also the recipient of the NSF CAREER award in 2013, the AFOSR summer faculty fellowship in 2014, and the Young Investigator Award from the Office of Naval Research (ONR) in 2015. He was the author/co-author of nine conference best paper awards at WiOpt in 2009, ICIMP in 2010, IEEE WCNC in 2012, IEEE PIMRC in 2015, IEEE SmartGridComm in 2015, EuCNC in 2017, IEEE GLOBECOM in 2018, IFIP NTMS in 2019, and IEEE ICC in 2020. He is the recipient of the 2015 Fred W. Ellersick Prize from the IEEE Communications Society, of the 2017 IEEE ComSoc Best Young Professional in Academia award, of the 2018 IEEE ComSoc Radio Communications Committee Early Achievement Award, and of the 2019 IEEE ComSoc Communication Theory Technical Committee. He was also a co-author of the 2019 IEEE Communications Society Young Author Best Paper. From 2015-2017, Dr. Saad was named the Stephen O. Lane Junior Faculty Fellow at Virginia Tech and, in 2017, he was named College of Engineering Faculty Fellow. He received the Dean's award for Research Excellence from Virginia Tech in 2019. He currently serves as an editor for the IEEE Transactions on Wireless Communications, IEEE Transactions on Mobile Computing, and IEEE Transactions on Cognitive Communications and Networking. He is an Editor-at-Large for the IEEE Transactions on Communications.

Choong Seon Hong (Senior Member, IEEE) received the B.S. and M.S. degrees in electronic engineering from Kyung Hee University, Seoul, South Korea, in 1983 and 1985, respectively, and the Ph.D. degree from Keio University, Minato, Japan, in 1997. In 1988, he joined Korea Telecom, where he worked on broadband networks as a Member of Technical Staff. In September 1993, he joined Keio University. He worked for the Telecommunications Network Laboratory, Korea Telecom, as a Senior Member of Technical Staff and the Director of the Net- working Research Team, until August 1999. Since September 1999, he has been a Professor with the Department of Computer Science and Engineering, Kyung Hee University. His research interests include future Internet, ad hoc networks, network management, and network security. He is a member of ACM, IEICE, IPSJ, KIISE, KICS, KIPS, and OSIA. He has served as the General Chair, a TPC Chair/Member, or an Organizing Committee Member of international conferences, such as NOMS, IM, APNOMS, E2EMON, CCNC, ADSN, ICPP, DIM, WISA, BcN, TINA, SAINT, and ICOIN. He was an Associate Editor of the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, the International Journal of Network Management, and the Journal of Communications and Networks and an Associate Technical Editor of IEEE Communications Magazine.

Mohammad Shikh-Bahaei (Senior Member, IEEE) received the B.Sc. degree from the University of Tehran, Tehran, Iran, in 1992, the M.Sc. degree from the Sharif University of Technology, Tehran, Iran, in 1994, and the Ph.D. degree from King's College London, U.K., in 2000. He has worked for two start-up companies, and for National Semiconductor Corp., Santa Clara ,CA, USA (now part of Texas Instruments Incorporated), on the design of third-generation (3G) mobile handsets, for which he has been awarded three U.S. patents as inventor and co-inventor, respectively. In 2002, he joined King's College London as Lecturer, and is now a full Professor in Telecommunications at theCenter for Telecommunications Research, Department of Engineering, King's College London. He has since authored or co-authored numerous journal and conference articles. He has been engaged in research in the area of wireless communications and signal processing for 25 years both in academic and industrial organizations. His research interests include the fields of Learning based resource allocation for multimedia applications over heterogeneous communication networks, Full Duplex and UAV communications, and Secure communication over wireless networks. Prof. Shikh-Bahaei was the Founder and Chair of the Wireless Advanced (formerly SPWC) Annual International Conference from 2003 to 2012. He was the recipient of the overall King's College London Excellence in Supervisory Award in 2014.