A Joint Learning and Communications Framework for Federated Learning over Wireless Networks

Mingzhe Chen, Zhaohui Yang, *Member, IEEE*, Walid Saad, *Fellow, IEEE*, Changchuan Yin, *Senior Member, IEEE*, H. Vincent Poor, *Fellow, IEEE*, and Shuguang Cui, *Fellow, IEEE*

Abstract

In this paper, the problem of training federated learning (FL) algorithms over a realistic wireless network is studied. In particular, in the considered model, wireless users execute an FL algorithm while training their local FL models using their own data and transmitting the trained local FL models to a base station (BS) that will generate a global FL model and send it back to the users. Since all training parameters are transmitted over wireless links, the quality of the training will be affected by wireless factors such as packet errors and the availability of wireless resources. Meanwhile, due to the limited wireless bandwidth, the BS must select an appropriate subset of users to execute the FL algorithm so as to build a global FL model accurately. This joint learning, wireless resource allocation, and user selection problem is formulated as an optimization problem whose goal is to minimize an FL loss function that captures the performance of the FL algorithm. To address this problem, a closed-form expression for the expected convergence rate of the FL algorithm is first derived to quantify the impact of wireless factors on FL. Then, based on the expected convergence rate of the FL algorithm, the optimal transmit

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

Z. Yang is with the Centre for Telecommunications Research, Department of Informatics, King's College London, WC2B 4BG, UK, Email: yang.zhaohui@kcl.ac.uk.

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

C. Yin is with the Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing, 100876, China, Emails: ccyin@ieee.org.

H. V. Poor is with the Department of Electrical Engineering, Princeton University, Princeton, NJ, 08544, USA, Email: poor@princeton.edu.

S. Cui is with the Shenzhen Research Institute of Big Data and School of Science and Engineering, the Chinese University of Hong Kong, Shenzhen, 518172, China, Email: robert.cui@gmail.com

This work was supported in part by the U.S. National Science Foundation under Grants CNS-1836802 and CCF-0939370.

power for each user is derived, under a given user selection and uplink resource block (RB) allocation scheme. Finally, the user selection and uplink RB allocation is optimized so as to minimize the FL loss function. Simulation results show that the proposed joint federated learning and communication framework can reduce the FL loss function value by up to 10% and 16%, respectively, compared to: 1) An optimal user selection algorithm with random resource allocation and 2) a standard FL algorithm with random user selection and resource allocation.

Index Terms— Federated learning; wireless resource allocation; user selection.

I. Introduction

Standard machine learning approaches require centralizing the training data on one machine or in a data center [2]–[4]. However, due to privacy and limited communication resources for data transmission, it is impractical for all users engaged in learning to transmit all of their collected data to a data center or a cloud. This, in turn, motivates the development of distributed learning frameworks that allow devices to use individually collected data to train a learning model locally. One of the most promising of such distributed learning frameworks is the so-called federated learning (FL) algorithm developed in [5]. FL is a distributed machine learning algorithm that enables users to collaboratively learn a shared prediction model while keeping their collected data on their devices [6]–[10]. However, to train an FL algorithm in a distributed manner, the users must transmit the training parameters over wireless links which can introduce training errors, due to the limited wireless resources (e.g., bandwidth) and the inherent unreliability of wireless links.

A. Related Works

Recently, a number of existing works such as in [5], [11]–[21] have studied important problems related to the implementation of FL over wireless networks. The works in [5] and [11] provided a comprehensive survey on the design of FL algorithms and introduced various challenges, problems, and solutions for enhancing FL effectiveness. In [12], the authors developed two update methods to reduce the uplink communication costs for FL. The work in [13] presented a practical update method for a deep FL algorithm and conducted an extensive empirical evaluation for five different FL models using four datasets. An echo state network-based FL algorithm is developed

A preliminary version of this work [1] appears in the proceedings of IEEE GLOBECOM.

in [14] to analyze and predict the location and orientation for wireless virtual reality users. In [15], the authors proposed a novel FL algorithm that can minimize the communication cost. The authors in [16] studied the problem of joint power and resource allocation for ultra-reliable low latency communication in vehicular networks. The work in [17] developed a new approach to minimize the computing and transmission delay for FL algorithms. In [18], the authors used FL algorithms for traffic estimation so as to maximize the data rates of users. While interesting, these prior works [5] and [11]–[18] assumed that wireless networks can readily integrate FL algorithms. However, in practice, due to the unreliability of the wireless channels and to the wireless resource limitations (e.g., in terms of bandwidth and power), FL algorithms will encounter training errors due to the wireless links [19]. For example, symbol errors introduced by the unreliable nature of the wireless channel and by resource limitations can impact the quality and correctness of the FL updates among users. Such errors will, in turn, affect the performance of FL algorithms, as well as their convergence speed. Moreover, due to the wireless bandwidth limitations, the number of users that can perform FL is limited; a design issue that is ignored in [5] and [11]–[18]. Furthermore, due to limited energy consumption of each user's device and strict delay requirement of FL, not all wireless users can perform FL algorithms. Therefore, one must select the appropriate users to perform FL algorithms and optimize the performance of FL. In practice, to effectively deploy FL over real-world wireless networks, it is necessary to investigate how the wireless factors affect the performance of FL algorithms. Here, we note that, although some works such as [7] and [19]–[21] have studied communication aspects of FL, these works are limited in several ways. First, the works in [7] and [19] only provide a high-level exposition of the challenges of communication in FL. Meanwhile, the authors in [20] and [21] do not consider the effect of packet transmission errors on the performance of FL. Moreover, none of these prior works provided a comprehensive design and optimization of the joint wireless and FL performance.

B. Contributions

The main contribution of this paper is, thus, a novel framework for enabling the implementation of FL algorithms over wireless networks by jointly taking into account FL and wireless metrics and factors. To our best knowledge, *this is the first work that provides a fundamental connection between the performance of FL algorithms and the underlying wireless network.* Our

key contributions include:

- We propose a novel FL model in which cellular-connected wireless users transmit their locally trained FL models to a base station (BS) that generates the global FL model and transmits it back to the users. For the considered FL model, the bandwidth for uplink transmission is limited and, hence, the BS needs to select appropriate users to execute the FL algorithm so as to minimize the FL loss function. In addition, the impact of the wireless packet transmission errors on the parameter update process of the FL model is explicitly considered.
- In the developed joint communication and FL model, the BS must optimize its resource allocation and the users must optimize their transmit power allocation so as to decrease the packet error rates of each user thus improving the performance of federated learning. To this end, we formulate this joint resource allocation and user selection problem for FL as an optimization problem whose goal is to minimize the value of the FL loss function while meeting the delay and energy consumption requirements. Hence, our framework *jointly considers learning and wireless networking metrics*.
- To solve this problem, we first derive a closed-form expression for the expected convergence rate of the FL algorithm so as to build an explicit relationship between the packet error rates and the performance of the FL algorithm. Based on this relationship, the optimization problem can be simplified as an mixed-integer nonlinear programming problem. To solve this simplified problem, we first find the optimal transmit power under given user selection and resource block (RB) allocation. Then, we transform the original optimization problem into a bipartite matching problem that is solved using a Hungarian algorithm which finds the optimal, FL-aware user selection and RB allocation strategy.
- To further reduce the effect of the packet transmission errors on the performance and convergence speed of FL, we perform fundamental analysis on the expression of expected convergence rate of FL algorithms, which shows that, the transmit power, RB allocation, and user selection will significantly affect the convergence speed and performance of FL algorithms. Meanwhile, by appropriately setting the learning rate and selecting the number of users that perform FL algorithms, the effect of the transmission errors on FL algorithm can be reduced and the convergence of FL can be guaranteed.

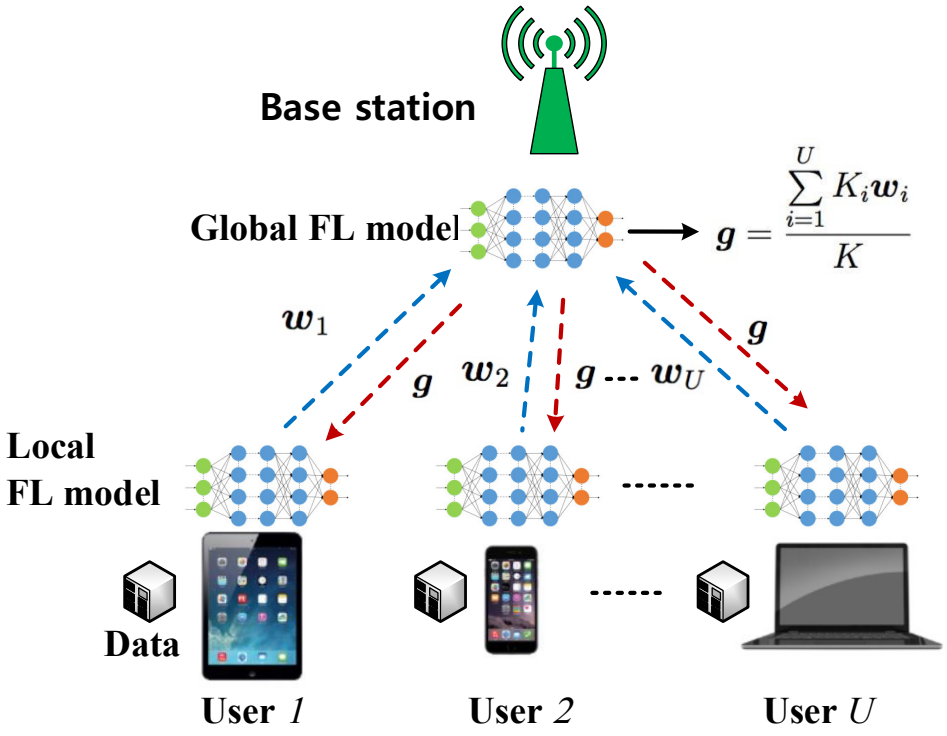

Fig. 1. The architecture of an FL algorithm that is being executed over a wireless network with multiple devices and a single base station.

Simulation results show that the transmit power, RB allocation, the number of users will jointly affect the performance of FL over wireless networks. In particular, the simulation result shows that the proposed FL algorithm that considers the wireless factors can achieve up to 10% and 16% reduction in the FL loss function compared, respectively, to an optimal user selection algorithm with random resource allocation and a standard FL algorithm (e.g., such as in [12]) FL with random user selection and resource allocation.

The rest of this paper is organized as follows. The system model and problem formulation are described in Section II. The expected convergence rate of FL algorithms is studied in Section III. The optimal resource allocation and user selection are determined in Section IV. Simulation results are analyzed in Section V. Conclusions are drawn in Section VI.

II. System Model and Problem Formulation

Consider a cellular network in which one BS and a set $\mathcal{U}$ of U users cooperatively perform an FL algorithm for data analysis and inference. For example, the network can execute an

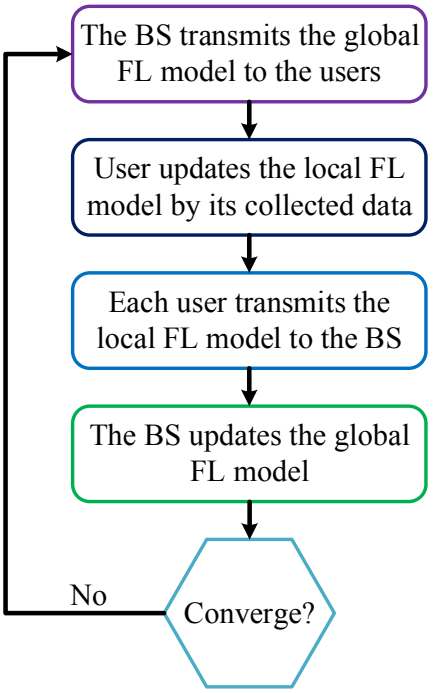

Fig. 2. The learning procedure of an FL algorithm.

FL algorithm to sense the wireless environment and generate a holistic radio environment mapping [22]. The use of FL for such applications is important because the data related to the wireless environment is distributed across the network [9] and the BS cannot collect all of this scattered data to implement a centralized learning algorithm. FL enables the BS and the users to collaboratively learn a shared learning model while keeping all of the training data at the device of each user. In an FL algorithm, each user will use its collected training data to train an FL model. For example, for radio environment mapping, each user will collect the data related to the wireless environment for training an FL model. Hereinafter, the FL model that is trained at the device of each user (using the data collected by the user itself) is called the *local FL model*. The BS is used to integrate the local FL models and generate a shared FL model. This shared FL model is used to improve the local FL model of each user so as to enable the users to collaboratively perform a learning task without training data transfer. Hereinafter, the FL model that is generated by the BS using the local FL models of its associated users is called the *global FL model*. As shown in Fig. 1, the *uplink* from the users to the BS is used to transmit the parameters related to the local FL model while the *downlink* is used to transmit the parameters

TABLE I
LIST OF NOTATIONS.

Notation	Description	Notation	Description
U	Number of users	$l_i^{\mathrm{U}}(\boldsymbol{r}_i, P_i)$	Uplink transmission delay
$\boldsymbol{X}_i$	Data collected by user i	$\boldsymbol{x}_{ik}$	FL input vector implemented by user i
y_{ik}	Output of $\boldsymbol{x}_{ik}$	$P_{\max}$	Maximum transmit power of each user
P_B	Transmit power of BS	$c_i^{\mathrm{U}}(\boldsymbol{r}_i, P_i)$	Uplink data rate of user i
P_i	Transmit power of user i	K_i	Number of samples collected by user i
R	Number of RBs	B^{D}	Total downlink bandwidth of each BS
$\boldsymbol{g}$	Global FL model	c_i^{D}	Downlink data rate of user i
$\mathcal{U}$	Set of users	l_i^{D}	Downlink transmission delay
$\boldsymbol{a}$	User selection vector	$Z(\boldsymbol{g})$	Data size of global FL model
λ	Learning rate	$q_i(\boldsymbol{r}_i, P_i)$	Packet error rate of user i
$\boldsymbol{R}$	RB allocation vector of all users	$Z(\boldsymbol{w}_i)$	Data size of local FL model
γ_{T}	Delay requirement	$f(\boldsymbol{g}(\boldsymbol{a}, \boldsymbol{R}), \boldsymbol{x}_{ik}, y_{ik})$	Loss function of FL
$\boldsymbol{w}_i$	Local FL model of user i	$e_i(\boldsymbol{r}_i, P_i)$	Energy consumption of user i
γ_{E}	Energy consumption requirement	$\boldsymbol{r}_i$	RB allocation vector of user i
K	Total number of training data samples	B^{U}	Bandwidth of each RB

related to the global FL model.

A. Machine Learning Model

In our model, each user i collects a marix $\boldsymbol{X}_i = [\boldsymbol{x}_{i1}, \ldots, \boldsymbol{x}_{iK_i}]$ of input data, where K_i is the number of the samples collected by each user i and each element $\boldsymbol{x}_{ik}$ is an input vector of the FL algorithm. The size of $\boldsymbol{x}_{ik}$ depends on the specific FL task. Our approach, however, is applicable to any generic FL algorithm and task. Let y_{ik} be the output of $\boldsymbol{x}_{ik}$. For simplicity, we consider an FL algorithm with a single output, however, our approach can be readily generalized to a case with multiple outputs [12]. The output data vector for training the FL algorithm of user i is $\boldsymbol{y}_i = [y_{i1}, \ldots, y_{iK_i}]$. We assume that the data collected by each user i is different from the other users, i.e., ($\boldsymbol{x}_i \neq \boldsymbol{x}_n, i \neq n, i, n \in \mathcal{U}$). We define a vector $\boldsymbol{w}_i$ to capture the parameters related to the local FL model that is trained by $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$. In particular, $\boldsymbol{w}_i$ determines the local FL model of each user i. For example, in a linear regression learning algorithm, $\boldsymbol{x}_{ik}^{\mathrm{T}}\boldsymbol{w}_i$ represents the predicted output and $\boldsymbol{w}_i$ is a weight vector that determines the performance of the linear regression learning algorithm. The training process of an FL algorithm is done in a way to solve

the following optimization problem:

$$\min_{\boldsymbol{w}_1,\ldots,\boldsymbol{w}_U} \frac{1}{K}\sum_{i=1}^{U}\sum_{k=1}^{K_i} f\left(\boldsymbol{w}_i, \boldsymbol{x}_{ik}, y_{ik}\right), \tag{1}$$

$$\text{s. t. } \boldsymbol{w}_1 = \boldsymbol{w}_2 = \ldots = \boldsymbol{w}_U = \boldsymbol{g}, \quad \forall i \in \mathcal{U}, \tag{1a}$$

where $K = \sum_{i=1}^{U} K_i$ is total size of training data of all users and $\boldsymbol{g}$ is the global FL model that is generated by the BS and $f\left(\boldsymbol{w}_i, \boldsymbol{x}_{ik}, y_{ik}\right)$ is a loss function. The loss function captures the performance of the FL algorithm. For different learning tasks, the FL performance captured by the loss function is different. For example, for a prediction learning task, the loss function captures the prediction accuracy of FL. In contrast, for a classification learning task, the loss function captures the classification accuracy. Meanwhile, for different FL algorithms, different loss functions can be defined [23]. For example, for a linear regression FL, the loss function is $f\left(\boldsymbol{w}_i, \boldsymbol{x}_{ik}, y_{ik}\right) = \frac{1}{2}\left(\boldsymbol{x}_{ik}^{\mathrm{T}}\boldsymbol{w}_i - y_{ik}\right)^2$. As the prediction errors (i.e., $\boldsymbol{x}_{ik}^{\mathrm{T}}\boldsymbol{w}_i - y_{ik}$) increase, the loss function $f\left(\boldsymbol{w}_i, \boldsymbol{x}_{ik}, y_{ik}\right)$ increases. Constraint (1a) is used to ensure that, once the FL algorithm converges, all of the users and the BS will share the same FL model for their learning task. This captures the fact that the purpose of an FL algorithm is to enable the users and the BS to learn an optimal global FL model without data transfer. To solve (1), the BS will transmit the parameters $\boldsymbol{g}$ of the global FL model to its users so that they train their local FL models. Then, the users will transmit their local FL models to the BS to update the global FL model. The detailed procedure of training an FL algorithm [24] to minimize the loss function in (1) is shown in Fig. 2. In FL, the update of each user i's local FL model $\boldsymbol{w}_i$ depends on the global model $\boldsymbol{g}$ while the update of the global model $\boldsymbol{g}$ depends on all of the users' local FL models. The update of the local FL model $\boldsymbol{w}_i$ depends on the learning algorithm. For example, one can use gradient descent, stochastic gradient descent, or randomized coordinate descent [12] to update the local FL model. The update of the global model $\boldsymbol{g}$ is given by [12]:

$$\boldsymbol{g} = \frac{\sum_{i=1}^{U} K_i \boldsymbol{w}_i}{K}. \tag{2}$$

During the training process, each user will first use its training data $\boldsymbol{X}_i$ and $\boldsymbol{y}_i$ to train the local FL model $\boldsymbol{w}_i$ and then, it will transmit $\boldsymbol{w}_i$ to the BS via wireless cellular links. Once the BS receives the local FL models from all participating users, it will update the global FL model

based on (2) and transmit the global FL model $\boldsymbol{g}$ to all users to optimize the local FL models. As time elapses, the BS and users can find their optimal FL models and use them to minimize the loss function in (1). Since all of the local FL models are transmitted over wireless cellular links, once they are received by the BS, they may contain erroneous symbols due to the unreliable nature of the wireless channel, which, in turn, will have a significant impact on the performance of FL. Meanwhile, the BS must update the global FL model once it receives all of the local FL models from its users and, hence, the wireless transmission delay will significantly affect the convergence of the FL algorithm. In consequence, to deploy FL over a wireless network, *one must jointly consider the wireless and learning performance and factors.*

B. Transmission Model

For uplink, we assume that an orthogonal frequency division multiple access (OFDMA) technique in which each user occupies one RB. The uplink rate of user i transmitting its local FL parameters to the BS is given by:

$$c_i^{\mathrm{U}}\left(\boldsymbol{r}_i, P_i\right) = \sum_{n=1}^{R} r_{i,n} B^{\mathrm{U}} \log_2 \left(1 + \frac{P_i h_i}{\sum\limits_{i' \in \mathcal{U}_n'} P_{i'} h_{i'} + B^{\mathrm{U}} N_0}\right), \tag{3}$$

where $\boldsymbol{r}_i = [r_{i,1}, \ldots, r_{i,R}]$ is an RB allocation vector with R being the total number of RBs, $r_{i,n} \in \{0, 1\}$ and $\sum\limits_{n=1}^{R} r_{i,n} = 1$; $r_{i,n} = 1$ indicates that RB n is allocated to user i, and $r_{i,n} = 0$, otherwise; $\mathcal{U}_n'$ represents the set of users that are located at the other service areas and transmit data over RB n; B^{U} is the bandwidth of each RB and P_i is the transmit power of user i; h_i is the channel gain between user i and the BS; N_0 is the noise power spectral density; $\sum\limits_{i' \in \mathcal{U}_n'} P_{i'} h_{i'}$ is the interference caused by the users that are located in other service areas (e.g., other BSs not participating in the FL algorithm) and use the same RB. Note that, although we ignore the optimization of resource allocation for the users located at the other service areas, we must consider the interference caused by the users in other service areas (if they are sharing RBs with the considered FL users), since this interference may significantly affect the packet error rates and the performance of FL.

Similarly, the downlink data rate achieved by the BS when transmitting the parameters of

global FL model to each user i is given by:

$$c_i^{\mathrm{D}} = B^{\mathrm{D}} \log_2 \left(1 + \frac{P_B h_i}{\sum\limits_{j \in \mathcal{B}'} P_B h_{ij} + B^{\mathrm{D}} N_0} \right), \tag{4}$$

where B^{D} is the bandwidth that the BS used to broadcast the global FL model of each user i; P_B is the transmit power of the BS; $\mathcal{B}'$ is the set of other BSs that cause interference to the BS that performs the FL algorithm; h_{ij} is the channel gain between user i and BS j. Given the uplink data rate c_i^{U} in (3) and the downlink data rate c_i^{D} in (4), the transmission delays between user i and the BS over uplink and downlink are respectively specified as:

$$l_i^{\mathrm{U}}\left(\boldsymbol{r}_i, P_i\right) = \frac{Z\left(\boldsymbol{w}_i\right)}{c_i^{\mathrm{U}}\left(\boldsymbol{r}_i, P_i\right)}, \tag{5}$$

$$l_i^{\mathrm{D}} = \frac{Z\left(\boldsymbol{g}\right)}{c_i^{\mathrm{D}}}, \tag{6}$$

where function $Z\left(\boldsymbol{x}\right)$ is the data size of $\boldsymbol{x}$ which is defined as the number of bits that the users or the BS require to transmit vector $\boldsymbol{x}$ over wireless links. In particular, $Z\left(\boldsymbol{w}_i\right)$ represents the number of bits that each user i requires to transmit local FL model $\boldsymbol{w}_i$ to the BS while $Z\left(\boldsymbol{g}\right)$ is the number of bits that the BS requires to transmit the global FL model $\boldsymbol{g}$ to each user. Here, $Z\left(\boldsymbol{w}_i\right)$ and $Z\left(\boldsymbol{g}\right)$ are determined by the type of implemented FL algorithm. From (2), we see that the number of elements in the global FL model $\boldsymbol{g}$ is similar to that of each user i's local FL model $\boldsymbol{w}_i$. Hence, we assume $Z\left(\boldsymbol{w}_i\right) = Z\left(\boldsymbol{g}\right)$.

C. Packet Error Rates

For simplicity, we assume that each local FL model $\boldsymbol{w}_i$ will be transmitted as a single packet in the uplink. A cyclic redundancy check (CRC) mechanism is used to check the data errors in the received local FL models at the BS. In particular, $C\left(\boldsymbol{w}_i\right) = 0$ indicates that the local FL model received by the BS contains data errors; otherwise, we have $C\left(\boldsymbol{w}_i\right) = 1$. The packet error rate experienced by the transmission of each local FL model $\boldsymbol{w}_i$ to the BS is given by [25]:

$$q_i\left(\boldsymbol{r}_i, P_i\right) = \sum_{n=1}^{R} r_{i,n} q_{i,n}, \tag{7}$$

where $q_{i,n} = \left(1 - \exp\left(-\frac{m\left(\sum\limits_{i' \in \mathcal{U}_n'} P_{i'} h_{i'} + B^{\mathrm{U}} N_0\right)}{P_i h_i}\right)\right)$ is the packet error rate over RB n with m being a waterfall threshold [25].

In the considered system, whenever the received local FL model contains errors, the BS will not use it for the update of the global FL model. We also assume that the BS will not ask the corresponding users to resend their local FL models when the received local FL models contain data errors. Instead, the BS will directly use the remaining correct local FL models to update the global FL model. As a result, the global FL model in (2) can be given by:

$$\boldsymbol{g}\left(\boldsymbol{a},\boldsymbol{P},\boldsymbol{R}\right)=\frac{\sum\limits_{i=1}^{U}K_{i}a_{i}\boldsymbol{w}_{i}C\left(\boldsymbol{w}_{i}\right)}{\sum\limits_{i=1}^{U}K_{i}a_{i}C\left(\boldsymbol{w}_{i}\right)}, \tag{8}$$

where $\boldsymbol{a}=[a_1,\ldots,a_U]$ is the vector of the user selection index with $a_i=1$ indicating that user i performs the FL algorithm and $a_i=0$, otherwise, $\boldsymbol{R}=[\boldsymbol{r}_1,\cdots,\boldsymbol{r}_U]$, $\boldsymbol{P}=[P_1,\cdots,P_U]$, $\sum\limits_{i=1}^{U}K_{i}a_{i}C\left(\boldsymbol{w}_{i}\right)$ is the total number of training data samples, which depends on the user selection vector $\boldsymbol{a}$ and packet transmission $C\left(\boldsymbol{w}_{i}\right)$, $K_{i}\boldsymbol{w}_{i}C\left(\boldsymbol{w}_{i}\right)=0$ indicates that the local FL model of user i contains data errors and, hence, the BS will not use it to generate the global FL model, and $\boldsymbol{g}\left(\boldsymbol{a},\boldsymbol{P},\boldsymbol{R}\right)$ is the global FL model that explicitly incorporates the effect of wireless transmission. From (8), we see that the global FL model also depends on the resource allocation matrix $\boldsymbol{R}$, user selection vector $\boldsymbol{a}$, and transmit power vector $\boldsymbol{P}$.

D. Energy Consumption Model

In our network, the energy consumption of each user consists of the energy needed for two purposes: a) Transmission of the local FL model and b) Training of the local FL model. The energy consumption of each user i is given by [26]:

$$e_{i}\left(\boldsymbol{r}_{i},P_{i}\right)=\varsigma\omega_{i}\vartheta^{2}Z\left(\boldsymbol{X}_{i}\right)+P_{i}l_{i}^{\mathrm{U}}\left(\boldsymbol{r}_{i},P_{i}\right), \tag{9}$$

where ϑ is the frequency of the central processing unit (CPU) clock of each user i, ω_i is the number of CPU cycles required for computing per bit data of user i, and ς is the energy consumption coefficient depending on the chip of each user i's device [26]. In (9), $\varsigma\omega_{i}\vartheta^{2}Z\left(\boldsymbol{X}_{i}\right)$ is the energy consumption of user i training the local FL model at its own device and $P_{i}l_{i}^{\mathrm{U}}\left(\boldsymbol{r}_{i},P_{i}\right)$ represents the energy consumption of local FL model transmission from user i to the BS. Note that, since the BS can have continuous power supply, we do not consider the energy consumption of the BS in our optimization problem.

E. Problem Formulation

To jointly design the wireless network and the FL algorithm, we now formulate an optimization problem whose goal is to minimize the FL loss function, while factoring in the wireless network parameters. This minimization problem includes optimizing transmit power allocation as well as resource allocation for each user. The minimization problem is given by:

$$\min_{\boldsymbol{a},\boldsymbol{P},\boldsymbol{R}} \frac{1}{K} \sum_{i=1}^{U} \sum_{k=1}^{K_i} f\left(\boldsymbol{g}\left(\boldsymbol{a},\boldsymbol{P},\boldsymbol{R}\right),\boldsymbol{x}_{ik},y_{ik}\right) \tag{10}$$

$$\text{s. t. } a_i, r_{i,n} \in \{0,1\}, \quad \forall i \in \mathcal{U}, n = 1,\ldots,R, \tag{10a}$$

$$\sum_{n=1}^{R} r_{i,n} = a_i, \ \ \forall i \in \mathcal{U}, \tag{10b}$$

$$l_i^{\mathrm{U}}\left(\boldsymbol{r}_i, P_i\right) + l_i^{\mathrm{D}} \leq \gamma_{\mathrm{T}}, \ \ \forall i \in \mathcal{U}, \tag{10c}$$

$$e_i\left(\boldsymbol{r}_i, P_i\right) \leq \gamma_{\mathrm{E}}, \ \ \forall i \in \mathcal{U}, \tag{10d}$$

$$\sum_{i \in \mathcal{U}} r_{i,n} \leq 1, \ \ \forall n = 1,\ldots,R, \tag{10e}$$

$$0 \leq P_i \leq P_{\max}, \quad \forall i \in \mathcal{U}, \tag{10f}$$

where γ_{T} is the delay requirement for implementing the FL algorithm, γ_{E} is the energy consumption of the FL algorithm, and B is the total downlink bandwidth. (10a) and (10b) indicates that each user can occupy only one RB for uplink data transmission. (10c) is the delay needed to execute the FL algorithm. (10d) is the energy consumption requirement of performing an FL algorithm. (10e) indicates that each uplink RB can be allocated to at most one user. (10f) is a maximum transmit power constraint.

From (7) and (8), we see that the transmit power and resource allocation determine the packet error rate, thus affecting the update of the global FL model. In consequence, the loss function of the FL algorithm in (10) depends on the resource allocation and transmit power. Moreover, (10c) shows that, in order to perform an FL algorithm, the users must satisfy a specific delay requirement. In particular, in an FL algorithm, the BS must wait to receive the local model of each user before updating its global FL model. Hence, transmission delay plays a key role in the FL performance. In a practical FL algorithm, it is desirable that all users transmit their local FL models to the BS simultaneously. From (10d), we see that to perform the FL algorithm, a given user must have enough energy to transmit and update the local FL model throughout the

FL iterative process. If this given user does not have enough energy, the BS should choose this user to participate in the FL process. In consequence, in order to implement an FL algorithm in a real-world network, the wireless network must provide low energy consumption and latency, and highly reliable data transmission.

III. Analysis of the Performance of Federated Learning

To solve (10), we first need to analyze how the packet error rate affects the performance of the federated learning. To find the relationship between the packet error rates and the performance of the federated learning, we must first analyze the convergence rate of FL. However, since the update of the global FL model depends on the instantaneous signal-to-interference-plus-noise ratio (SINR), we can analyze only the expected convergence rate of FL. Here, we first analyze the expected convergence rate of FL. Then, we show how the packet error rate affects the performance of the FL in (10).

In the studied network, the users adopt a standard gradient descent method to update their local FL models as done in [12]. Therefore, during the training process, the update of user i's local FL model $\boldsymbol{w}_i$ at time t is given by:

$$\boldsymbol{w}_{i,t+1} = \boldsymbol{g}_t\left(\boldsymbol{a}, \boldsymbol{P}, \boldsymbol{R}\right) - \frac{\lambda}{K_i}\sum_{k=1}^{K_i} \nabla f\left(\boldsymbol{g}_t\left(\boldsymbol{a}, \boldsymbol{P}, \boldsymbol{R}\right), \boldsymbol{x}_{ik}, y_{ik}\right), \tag{11}$$

where λ is the learning rate and $\nabla f\left(\boldsymbol{g}_t\left(\boldsymbol{a}, \boldsymbol{P}, \boldsymbol{R}\right), \boldsymbol{x}_{ik}, y_{ik}\right)$ is the gradient of $f\left(\boldsymbol{g}_t\left(\boldsymbol{a}, \boldsymbol{P}, \boldsymbol{R}\right), \boldsymbol{x}_{ik}, y_{ik}\right)$ with respect to $\boldsymbol{g}_t\left(\boldsymbol{a}, \boldsymbol{P}, \boldsymbol{R}\right)$.

We assume that $F\left(\boldsymbol{g}\right) = \frac{1}{K}\sum_{i=1}^{U}\sum_{k=1}^{K_i} f\left(\boldsymbol{g}, \boldsymbol{x}_{ik}, y_{ik}\right)$ and $F_i\left(\boldsymbol{g}\right) = \sum_{k=1}^{K_i} f\left(\boldsymbol{g}, \boldsymbol{x}_{ik}, y_{ik}\right)$ where $\boldsymbol{g}$ is short for $\boldsymbol{g}\left(\boldsymbol{a}, \boldsymbol{P}, \boldsymbol{R}\right)$. Based on (11), the update of global FL model $\boldsymbol{g}$ at time t can be given by:

$$\boldsymbol{g}_{t+1} = \boldsymbol{g}_t - \lambda\left(\nabla F(\boldsymbol{g}_t) - \boldsymbol{o}\right), \tag{12}$$

where $\boldsymbol{o} = \nabla F(\boldsymbol{g}_t) - \frac{\sum_{i=1}^{U} K_i a_i \boldsymbol{w}_i C(\boldsymbol{w}_i)}{\sum_{i=1}^{U} K_i a_i C(\boldsymbol{w}_i)}$ with

$$C\left(\boldsymbol{w}_i\right) = \begin{cases} 1, & \text{with probability } 1 - q_i\left(\boldsymbol{r}_i, P_i\right), \\ 0, & \text{with probability } q_i\left(\boldsymbol{r}_i, P_i\right). \end{cases} \tag{13}$$

We also assume that the FL algorithm converges to an optimal global FL model $\boldsymbol{g}^*$ after the learning steps. To derive the expected convergence rate of FL, we first make the following assumptions:

- First, we assume that the gradient $\nabla F(\boldsymbol{g})$ of $F(\boldsymbol{g})$ is uniformly Lipschitz continuous with respect to $\boldsymbol{g}$ [27]. Hence, we have:

$$\|\nabla F(\boldsymbol{g}_{t+1}) - \nabla F(\boldsymbol{g}_t)\| \leq L\|\boldsymbol{g}_{t+1} - \boldsymbol{g}_t\|, \tag{14}$$

  where L is a positive constant and $\|\boldsymbol{g}_{t+1} - \boldsymbol{g}_t\|$ is the norm of $\boldsymbol{g}_{t+1} - \boldsymbol{g}_t$.
- Second, we assume that $F(\boldsymbol{g})$ is strongly convex with positive parameter μ, such that:

$$F(\boldsymbol{g}_{t+1}) \geq F(\boldsymbol{g}_t) + (\boldsymbol{g}_{t+1} - \boldsymbol{g}_t)^T \nabla F(\boldsymbol{g}_t) + \frac{\mu}{2}\|\boldsymbol{g}_{t+1} - \boldsymbol{g}_t\|^2. \tag{15}$$

- We also assumed that $F(\boldsymbol{g})$ is twice-continuously differentiable. Based on (14) and (15), we have:

$$\mu \boldsymbol{I} \preceq \nabla^2 F(\boldsymbol{g}) \preceq L\boldsymbol{I}. \tag{16}$$

- We also assume that $\|\nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{ik}, y_{ik}\right)\|^2 \leq \zeta_1 + \zeta_2 \nabla \|F\left(\boldsymbol{g}_t\right)\|^2$ with $\zeta_1 \geq 0$ and $\zeta_2 \geq 1$.

These assumptions can be easily satisfied by the general FL loss functions such as linear or logistic loss functions. The expected convergence rate of the FL algorithms can now be obtained by the following theorem.

Theorem 1. Given the transmit power vector $\boldsymbol{P}$, RB allocation matrix $\boldsymbol{R}$, user selection vector $\boldsymbol{a}$, optimal global FL model $\boldsymbol{g}^*$, and the learning rate $\lambda = \frac{1}{L}$, the upper bound of $\mathbb{E}[F(\boldsymbol{g}_{t+1}) - F(\boldsymbol{g}^*)]$ can be given by:

$$\mathbb{E}[F(\boldsymbol{g}_{t+1}) - F(\boldsymbol{g}^*)] \leq \underbrace{\frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i q_i\left(\boldsymbol{r}_i, P_i\right)\frac{1 - A^t}{1 - A}}_{\text{Impact of wireless factors on FL convergence}} + A^t \mathbb{E}(F(\boldsymbol{g}_0) - F(\boldsymbol{g}^*)), \tag{17}$$

where $A = 1 - \frac{2\mu}{L} + \frac{\mu\zeta_2}{LK}\sum_{i=1}^{U} K_i q_i\left(\boldsymbol{r}_i, P_i\right)$.

Proof. See Appendix A. □

From Theorem 1, we see that, when the learning rate λ is a constant ($\lambda = \frac{1}{L}$), the FL algorithm that considers the effect of the packet error rates will finally converge as t increases. However, a gap, $\frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i\left(1 - a_i + a_i q_i\left(\boldsymbol{r}_i, P_i\right)\right)$, exists between $\mathbb{E}[F(\boldsymbol{g}_t)]$ and $\mathbb{E}[F(\boldsymbol{g}^*)]$. This gap is caused by the packet errors and user selection policy. As the packet error rate decreases, the gap between $\mathbb{E}[F(\boldsymbol{g}_t)]$ and $\mathbb{E}[F(\boldsymbol{g}^*)]$ decreases. Meanwhile, as the number of users that implement the FL algorithm increases, the gap also decreases. Moreover, as the packet error rate decreases,

the value of A also decreases, which indicates that the convergence speed of the FL algorithm improves. Hence, it is necessary to optimize resource allocation, user selection, and transmit power for the implementation of any FL algorithm over a realistic wireless network.

According to Theorem 1, the following result is derived to guarantee the convergence of the FL algorithm.

Proposition 1. Given the learning rate $\lambda = \frac{1}{L}$, to guarantee convergence and reduce the effect of packet errors on the FL algorithm, ζ_2 must satisfy:

$$1 < \zeta_2 < 2. \tag{18}$$

Proof. From Theorem 1, we see that when $A < 1$, $A^t = 0$. Hence, $\mathbb{E}[F(\boldsymbol{g}_{t+1}) - F(\boldsymbol{g}^*)] = \sum_{i=1}^{U} K_i q_i(\boldsymbol{r}_i, P_i) \frac{1}{1-A}$ and the FL algorithm converges. In consequence, to guarantee the convergence, we only need to make $A_{\max} = 1 - \frac{2\mu}{L} + \frac{\mu\zeta_2}{LK} \sum_{i=1}^{U} K_i < 1$. Since $\sum_{i=1}^{U} K_i = K$. we have $A_{\max} = 1 - \frac{2\mu}{L} + \frac{\mu\zeta_2}{L}$. From (16), we see that $\mu < L$ and, hence, $\frac{\mu}{L} < 1$. To make $A_{\max} < 1$, we only need to ensure that $\frac{\mu\zeta_2}{L} - \frac{2\mu}{L} < 0$. Therefore, we have $\zeta_2 < 2$. To enable $\|\nabla f(\boldsymbol{g}_t, \boldsymbol{x}_{ik}, y_{ik})\|^2 \leq \zeta_1 + \zeta_2 \nabla \|F(\boldsymbol{g}_t)\|^2$, we have $\zeta_2 > 1$. This completes the proof. □

From Proposition 1, we see that the convergence of the FL algorithm depends on the parameters related to the approximation of $\nabla \|F(\boldsymbol{g}_t)\|^2$. Using Proposition 1, we can determine the convergence of the FL algorithm based on the approximation of $\nabla \|F(\boldsymbol{g}_t)\|^2$. From Theorem 1 and Proposition 1, we can also see that the number of training data samples will not affect the convergence of the FL algorithm but it affects the value that the FL algorithm converges to.

Based on Theorem 1, next, we can also derive the convergence rate of an FL algorithm when there are no packet errors.

Lemma 1. Given the optimal global FL model $\boldsymbol{g}^*$ and the learning rate $\lambda = \frac{1}{L}$, the upper bound of $\mathbb{E}[F(\boldsymbol{g}_{t+1}) - F(\boldsymbol{g}^*)]$ for an FL algorithm without considering packet errors and user selection is given by:

$$\mathbb{E}[F(\boldsymbol{g}_{t+1}) - F(\boldsymbol{g}^*)] \leq \left(1 - \frac{2\mu}{L}\right)^t \mathbb{E}(F(\boldsymbol{g}_0) - F(\boldsymbol{g}^*)). \tag{19}$$

Proof. Since the FL algorithms do not consider the packet error rates and user selection, we have $q_i(\boldsymbol{r}_i, P_i) = 0$, $a_i = 1$, $A = 1 - \frac{2\mu}{L}$. Hence, $\frac{\zeta_1}{2LK} \sum_{i=1}^{U} K_i (1 - a_i + a_i q_i(\boldsymbol{r}_i, P_i)) \frac{1-A^t}{1-A} = 0$.

Then (19) can be derived based on (17). □

From Lemma 1, we can observe that, if we do not consider the packet transmission errors, the FL algorithm will converge to the optimal global FL model without any gaps. This result also corresponds to the result in the existing works (e.g., [27]). In the following section, we show how one can leverage the result in Theorem 1 to solve the proposed problem (10).

IV. Optimization of Prediction Errors for Federated Learning Algorithm

In this section, our goal is to minimize the FL loss function when considering the underlying wireless network constraints. To solve the problem in (10), we must first simplify it. From Theorem 1, we can see that, to minimize the loss function in (10), we need to only minimize the gap, $\frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i\left(1-a_i+a_i q_i\left(\boldsymbol{r}_i,P_i\right)\right)\frac{1-A^t}{1-A}$. When $A\geq 1$, the FL algorithm will not converge. In consequence, here, we only consider the minimization of the FL loss function when $A<1$. Hence, as t is large enough, which captures the asymptotic convergence behavior of FL, we have $A^t=0$. The gap can be rewritten as follows:

$$\frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i\left(1-a_i+a_i q_i\left(\boldsymbol{r}_i,P_i\right)\right)\frac{1-A^t}{1-A}=\frac{\frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i\left(1-a_i+a_i q_i\left(\boldsymbol{r}_i,P_i\right)\right)}{\frac{2\mu}{L}-\frac{\mu\zeta_2}{LK}\sum_{i=1}^{U} K_i\left(1-a_i+a_i q_i\left(\boldsymbol{r}_i,P_i\right)\right)}. \tag{20}$$

From (20), we can observe that minimizing $\frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i\left(1-a_i+a_i q_i\left(\boldsymbol{r}_i,P_i\right)\right)\frac{1-A^t}{1-A}$ only requires minimizing $\sum_{i=1}^{U} K_i\left(1-a_i+a_i q_i\left(\boldsymbol{r}_i,P_i\right)\right)$. Meanwhile, since $a_i=\sum_{n=1}^{R} r_{i,n}$ and $q_i\left(\boldsymbol{r}_i,P_i\right)=\sum_{n=1}^{R} r_{i,n}q_{i,n}$, when $a_i=1$, $q_i\left(\boldsymbol{r}_i,P_i\right)\leq 0$ and when $a_i=0$, $q_i\left(\boldsymbol{r}_i,P_i\right)=0$. In consequence, we have $a_i q_i\left(\boldsymbol{r}_i,P_i\right)=q_i\left(\boldsymbol{r}_i,P_i\right)$. The problem in (10) can be simplified as follows:

$$\min_{\boldsymbol{P},\boldsymbol{R}} \sum_{i=1}^{U} K_i\left(1-\sum_{n=1}^{R} r_{i,n}+q_i\left(\boldsymbol{r}_i,P_i\right)\right) \tag{21}$$

$$\text{s. t. (10c) – (10f).}$$

$$r_{i,n}\in\{0,1\},\quad \forall i\in\mathcal{U}, n=1,\ldots,R, \tag{21a}$$

$$\sum_{n=1}^{R} r_{i,n}\leq 1,\ \ \forall i\in\mathcal{U}. \tag{21b}$$

Next, we first find the optimal transmit power for each user given the uplink RB allocation matrix $\boldsymbol{R}$. Then, we find the uplink RB allocation to minimize the FL loss function.

A. Optimal Transmit Power

The optimal transmit power of each user i can be determined by the following lemma.

Proposition 2. Given the uplink RB allocation vector $\boldsymbol{r}_i$ of each user i, the optimal transmit power of each user i, P_i^* is given by:

$$P_i^* = \min\left\{P_{\max}, P_{i,\gamma_{\mathrm{E}}}\right\}, \tag{22}$$

where $P_{i,\gamma_{\mathrm{E}}}$ satisfies the equality $\varsigma\omega_i\vartheta^2 Z\left(\boldsymbol{X}_i\right) + \frac{P_{i,\gamma_{\mathrm{E}}} Z(\boldsymbol{w}_i)}{c_i^{\mathrm{U}}\left(\boldsymbol{r}_i, P_{i,\gamma_{\mathrm{E}}}\right)} = \gamma_{\mathrm{E}}$.

Proof. See Appendix B. □

From Proposition 2, we see that the optimal transmit power depends on the size of the collected data $Z\left(\boldsymbol{X}_i\right)$, the size of the local FL model $Z\left(\boldsymbol{w}_i\right)$, and the interference in each RB. In particular, as the size of the collected data and local FL model increases, each user must spend more energy for training FL model and, hence, the energy that can be used for data transmission decreases. In consequence, the value of the FL loss function increases.

B. Optimal Uplink Resource Block Allocation

Based on Proposition 2 and (7), the optimization problem in (21) can be simplified as follows:

$$\min_{\boldsymbol{R}} \sum_{i=1}^{U} K_i \left(1 - \sum_{n=1}^{R} r_{i,n} + \sum_{n=1}^{R} r_{i,n} q_{i,n}\right) \tag{23}$$

$$\text{s. t. (10a), (10b), and (10e),}$$

$$l_i^{\mathrm{U}}\left(\boldsymbol{r}_i, P_i^*\right) + l_i^{\mathrm{D}} \leq \gamma_{\mathrm{T}}, \quad \forall i \in \mathcal{U}, \tag{23a}$$

$$e_i\left(\boldsymbol{r}_i, P_i^*\right) \leq \gamma_{\mathrm{E}}, \quad \forall i \in \mathcal{U}. \tag{23b}$$

Obviously, the objective function (23) is a mixed-integer linear programming problem, which can be solved by using bipartite matching algorithm [28]. Compared to traditional convex optimization algorithms, using bipartite matching to solve problem (23) does not require computing the gradients of each variable nor dynamically adjusting the step size for convergence.

To use a bipartite matching algorithm for solving problem (23), we first transform the optimization problem into a bipartite matching problem. We construct a bipartite graph $\mathcal{A} = (\mathcal{U} \times \mathcal{R}, \mathcal{E})$ where $\mathcal{R}$ is the set of RBs that can be allocated to each user, each vertex in $\mathcal{U}$ represents a user

and each vertex in $\mathcal{R}$ represents an RB, and $\mathcal{E}$ is the set of edges that connect to the vertices from each set $\mathcal{U}$ and $\mathcal{R}$. Let $\vartheta_{in} \in \mathcal{E}$ be the edge connecting vertex i in $\mathcal{U}$ and vertex n in $\mathcal{R}$ with $\vartheta_{in} \in \{0, 1\}$, where $\vartheta_{in} = 1$ indicates that RB n is allocated to user i, otherwise, we have $\vartheta_{in} = 0$. Let matching $\mathcal{T}$ be a subset of edges in $\mathcal{E}$, in which no two edges share a common vertex in $\mathcal{R}$, such that each RB n can only be allocated to one user (constraint (10e) is satisfied). Nevertheless, in $\mathcal{T}$, all of the edges associated with a vertex $i \in \mathcal{U}$ will not share a common vertex $n \in \mathcal{R}$, such that each user i can occupy only one RB (constraint (10b) is satisfied). The weight of edge ϑ_{in} is given by:

$$\psi_{in} = \begin{cases} K_i \left(q_{i,n} - 1\right), l_i^{\mathrm{U}}\left(r_{i,n}, P_i^*\right) + l_i^{\mathrm{D}} \leq \gamma_{\mathrm{T}} \text{ and } e_i\left(r_{i,n}, P_i^*\right) \leq \gamma_{\mathrm{E}}, \\ +\infty, \qquad \text{otherwise.} \end{cases} \tag{24}$$

From (24), we can see that when RB n is allocated to user i, if the delay and energy requirements cannot be satisfied, we will have $\psi_{in} = +\infty$, which indicates that RB n will not be allocated to user i. The goal of this formulated bipartite matching problem is to find an optimal matching set $\mathcal{T}^*$ that can minimize the weights of the edges in $\mathcal{T}^*$. A standard Hungarian algorithm [29] can be used to find the optimal matching set $\mathcal{T}^*$. When the optimal matching set is found, the optimal RB allocation is determined.

C. Implementation and Complexity

Next, we first analyze the implementation of the Hungarian algorithm. To implement the Hungarian algorithm for finding the optimal matching set $\mathcal{T}^*$, the BS must first calculate the packet error rate $q_{i,n}$, total delay $l_i^{\mathrm{U}}\left(r_{i,n}, P_i^*\right) + l_i^{\mathrm{D}}$, and the energy consumption $e_i\left(r_{i,n}, P_i^*\right)$ of each user transmitting the local FL model over each RB n. To calculate the packet error rate $q_{i,n}$ and total delay $l_i^{\mathrm{U}}\left(r_{i,n}, P_i^*\right) + l_i^{\mathrm{D}}$, the BS must know the SINR over each RB and the data size of FL model. The BS can use channel estimation methods to learn the SINR over each RB. The data size of the FL model depends on the learning task. To implement an FL mechanism, the BS must first send the FL model information and the learning task information to the users. In consequence, the BS will learn the data size of FL model before the execution of the FL algorithm. To calculate the energy consumption $e_i\left(r_{i,n}, P_i^*\right)$ of each user, the BS must learn each user's device information such as CPU. This device information can be learned by the BS when the users initially connect to the BS. Given the packer error rate $q_{i,n}$, total delay $l_i^{\mathrm{U}}\left(r_{i,n}, P_i^*\right) + l_i^{\mathrm{D}}$,

and the energy consumption $e_i\left(r_{i,n}, P_i^*\right)$ of each user, the BS can compute ψ_{in} according to (24). Given $\psi_{in}, i \in \mathcal{U}, n \in \mathcal{R}$, the Hungarian algorithm can be used to find the optimal matching set $\mathcal{T}^*$. Since (23) is a mixed-integer linear programming problem, it admits an optimal matching set $\mathcal{T}^*$ and the Hungarian algorithm will finally find the optimal matching set $\mathcal{T}^*$.

With regards to the complexity of the Hungarian algorithm, it must first use UR iterations to calculate the packer error rate, total delay, and energy consumption of each user over each RB. After that, the Hungarian algorithm will update the values of ψ_{in} so as to find the optimal matching set $\mathcal{T}^*$. The worst complexity of the hungarian algorithm to find the optimal matching set $\mathcal{T}^*$ is $\mathcal{O}\left(U^2R\right)$ [30]. In contrast, the best complexity is $\mathcal{O}\left(UR\right)$. In consequence, the major complexity lies in calculating the weight of each edge and updating the edges in the matching set $\mathcal{T}$. However, in the Hungarian algorithm, we need to only perform simple operations such as $K_i\left(q_{i,n}-1\right)$ without calculation for the gradients of each valuables nor adjusting the step sizes as done in the optimization algorithms. Meanwhile, Algorithm 1 is implemented by the BS in a centralized manner and the BS will have sufficient computational resources to implement it.

V. Simulation Results and Analysis

For our simulations, we consider a circular network area having a radius $r = 500$ m with one BS at its center servicing $U = 20$ uniformly distributed users. The other parameters used in simulations are listed in Table I. The data used to train the FL algorithm is generated randomly from $[0, 1]$. The input x and the output y follow the function $y = -2x + 1 + n \times 0.4$ where n follows a Gaussian distribution $\mathcal{N}\left(0, 1\right)$. The FL algorithm is used to model the relationship between x and y (i.e., FL is used as a linear regression). For comparison purposes, we use two baselines: a) an FL algorithm that optimizes user selection with random resource allocation and b) an FL algorithm that randomly determines user selection and resource allocation. *Baseline a)* is actually an FL algorithm without consideration of wireless factors. *Baseline b)* is a conventional FL in [12] without consideration of wireless factors nor optimizing FL performance.

Fig. 3 shows an example of using FL for linear regression. In this figure, the red crosses are the data samples. In the optimal FL, the optimal RB allocation, user association, and transmit power powers are derived using a heuristic search method. From Fig. 3, we see that the proposed FL algorithm can fit the data samples more accurately than baselines a) and b). This is due to the fact that the proposed FL algorithm jointly considers the learning and wireless factors and,

TABLE II
SYSTEM PARAMETERS

Parameter	Value	Parameter	Value
α	2	N_0	-174 dBm/Hz
P_B	1 W	B^{D}	20 MHz
M	64	B^{U}	150 kHz
σ_i	1	$P_{\max}$	0.01 W
f	10^9	K_i	[12,10,8,4,2]
ς	10^{-27}	γ_{T}	100 ms
ω_i	40	γ_{E}	0.02 J

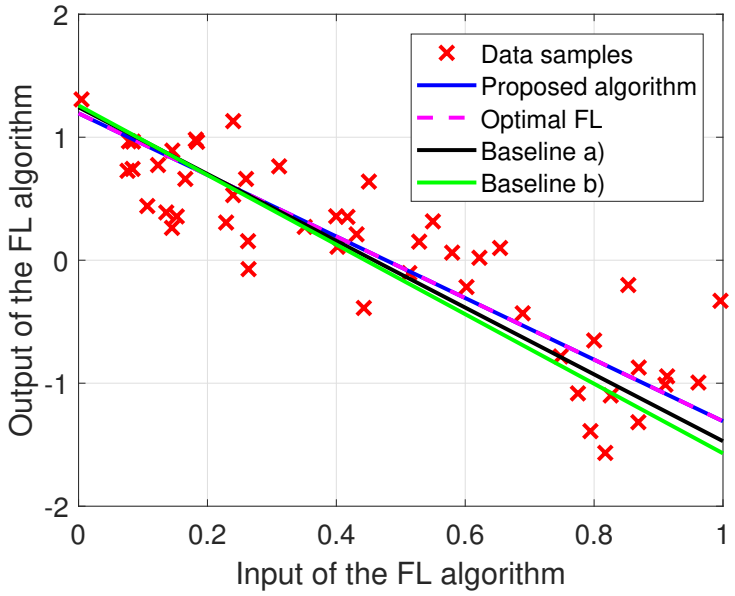

Fig. 3. An example of implementing FL for linear regression.

hence, it can optimize user selection and resource allocation to reduce the effect of wireless transmission errors on training FL algorithm and improve the performance of the FL algorithm. Fig. 3 also shows that the proposed algorithm can reach the same performance as the optimal FL, which verifies that the proposed algorithm can find an optimal solution using the Hungarian algorithm.

Fig. 4 shows how the value of the FL loss function changes as the total number of users varies. In this figure, an appropriate subset of users is selected to perform the FL algorithm. From Fig. 4, we can observe that, as the number of users increases, the value of the loss function decreases.

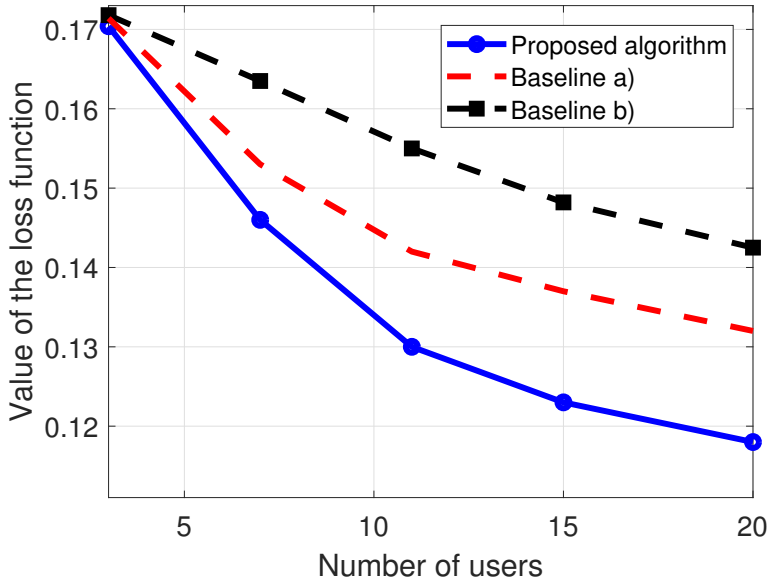

Fig. 4. Value of the loss function as the number of users varies.

Moreover, as the number of users increases, the effect of packet errors on the global FL model decreases. This is due to the fact that an increase in the number of users leads to more data available for the FL algorithm training and, hence, improving the accuracy of approximation of the gradient of the loss function. Fig. 4 also shows that the proposed algorithm reduces the loss function by, respectively, up to 10% and 16% compared to baselines a) and b). The 10% reduction of the loss function stems from the fact that the proposed algorithm optimizes the resource allocation. The 16% reduction stems from the fact that the proposed algorithm joint considers learning and wireless effects and, hence, it can optimize the user selection and resource allocation to reduce the FL loss function. Fig. 4 also shows that when the number of users is less than 12, the value of the loss function decreases quickly. In contrast, as the number of users continues to increase, the value of the FL loss function decreases slowly. This is because, for a higher number of users, the BS will have enough data samples to accurately approximate the gradient of the loss function.

In Fig. 5, we show how the transmission errors affect the convergence of the global FL model. From Fig. 5, we see that, as the number of iterations increases, the global FL model of all considered learning algorithms decreases first and, then remains unchanged. Here, the global FL model remains unchanged which shows that the global FL model converges. From Fig. 5, we

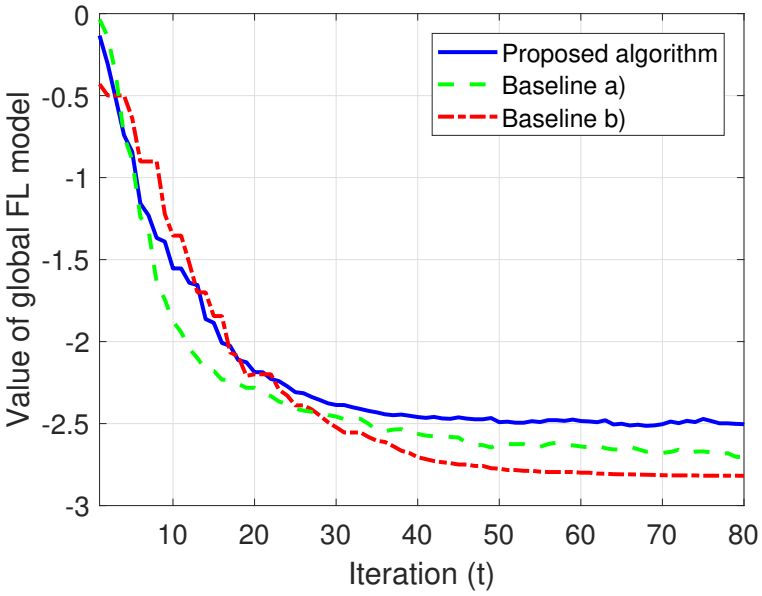

Fig. 5. Value of the loss function as the number of iteration varies.

can also see that the decrease speed in the value of the global FL model is different during each iteration. This is due to the fact that the local FL models that are received by the BS may contain data errors and the BS may not be able to use them for the update of the global FL model. In consequence, at each iteration, the number of local FL models that can be used for the update of the global FL model will be different. Fig. 5 also shows that a gap exists between the proposed algorithm and baselines a) and b). This gap is caused by the packet errors. Meanwhile, Fig. 5 clearly shows that the proposed algorithm can converge faster than both baselines a) and b). This is because the proposed algorithm can optimize the user selection and resource allocation to improve the convergence speed.

Fig. 6 shows how the value of the FL loss function changes as the number of data samples of each user varies. From this figure, we observe that, as the number of data samples of each user increases, the values of the FL loss function of all of considered FL algorithms decrease. This is due to the fact that, as the number of data samples increases, all of the considered learning algorithms can use more data samples for training. Fig. 6 also demonstrates that, when the number of data samples is less than 30, the value of the loss function decreases quickly. However, as the number of data samples continues to increase, the value of the loss function remains unchanged. This is due to the fact that as the number of data samples is over 30, the

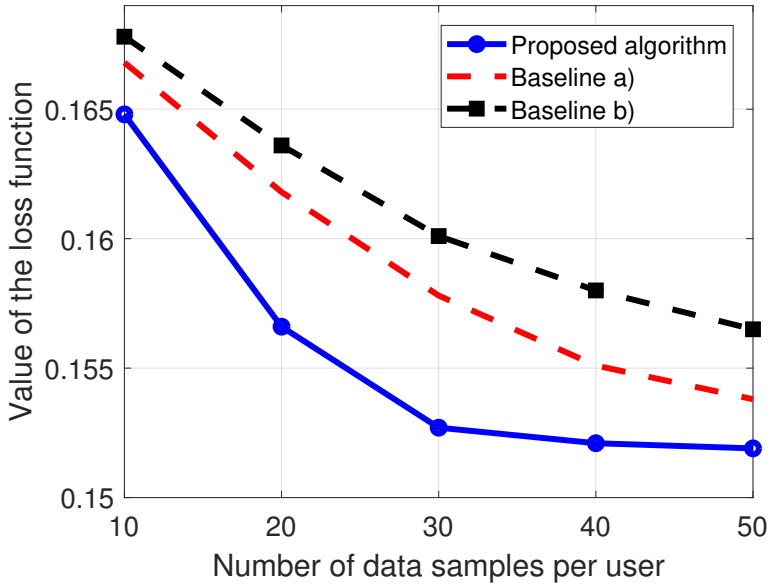

Fig. 6. Value of the loss function as the number of data samples per user varies.

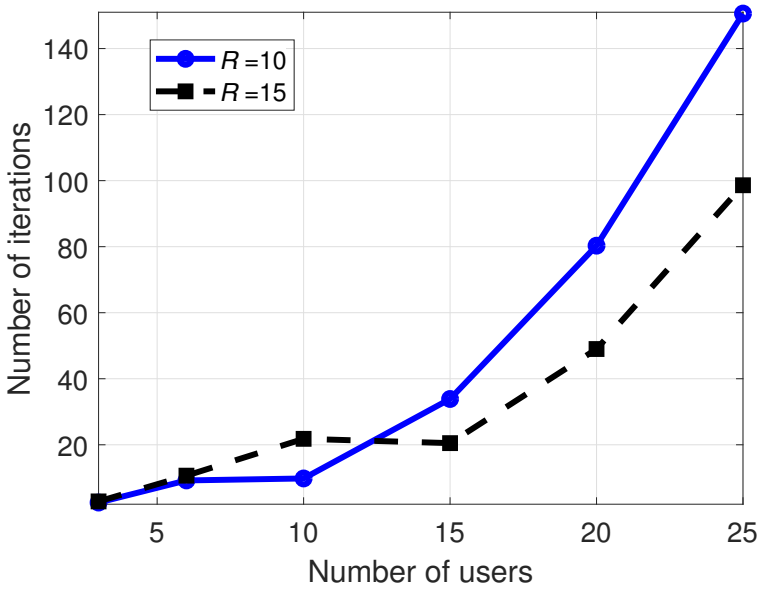

Fig. 7. Number of iterations as the number of users varies.

BS has enough data samples to approximate the gradient of the loss function.

In Fig. 7, we show the number of iterations that the Hungarian algorithm needs to find the optimal RB allocation as a function of the number of users. From this figure, we can see that, as the number of users increases, the number of iterations needed to find the optimal RB allocation

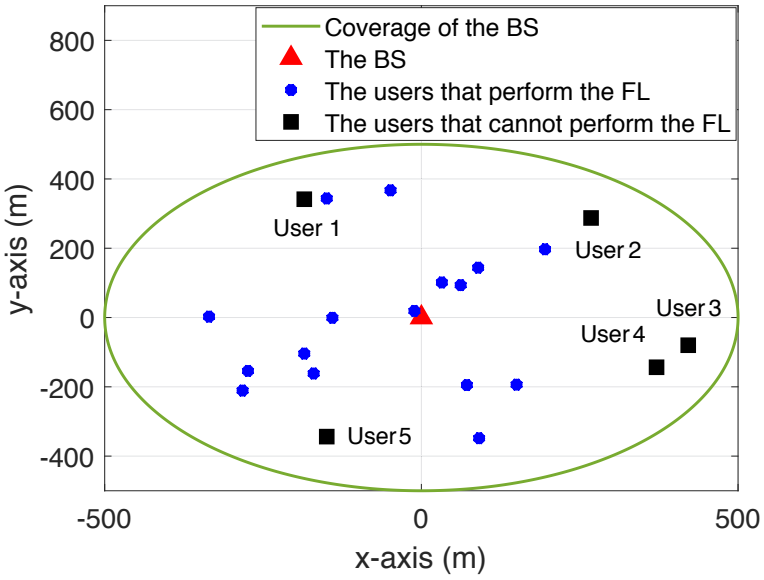

Fig. 8. An example of the users that perform an FL algorithm over a wireless network.

increases. This is because, as the number of users increases, the size of the edge weight matrix in (24) increases and, hence, the Hungarian algorithm needs to use more iterations to find the optimal RB allocation. Fig. 7 also shows when the number of users is smaller than the number of RBs, the number of iterations needed to find the optimal RB allocation increases slowly. However, as the number of users continues to increase, the number of iterations significantly increases. Fig. 7 also shows that, when the number of users is larger than 10, the number of iterations needed to find the optimal RB allocation for a network with 10 RBs is larger than that of a network with 15 RBs. This is due to the fact that as the number of users is larger than 10, the gap between the number of users and the number of RBs for a network with 10 RBs is larger than that for a network with 15 RBs.

Fig. 8 shows an example of the users that participate in the FL algorithm over a wireless network with 20 users. In this figure, the blue points indicate the users that are selected to perform the FL algorithm while the black points indicate users that are not selected for the implementation of the FL algorithm. In particular, due to the energy consumption and delay requirements, users 2, 3, and 4 are not selected to perform the FL algorithm. Users 1 and 5 were also not selected for the implementation of the FL algorithm due to the limited number of RBs.

VI. Conclusion

In this paper, we have developed a novel framework that enables the implementation of FL algorithms over wireless networks. We have formulated an optimization problem that jointly considers user selection and resource allocation for the minimization of the value of FL loss function. To solve this problem, we have derived the closed-form expression of the expected convergence rate of the FL algorithm that considers the wireless factors. Based on the derived expected convergence rate, the optimal transmit power is determined given the user selection and uplink RB allocation. Then, the Hungarian algorithm is used to find the optimal user selection and RB allocation so as to minimize the FL loss function. Simulation results have shown that the joint federated learning and communication framework yields significant improvements in the performance compared to the existing implementation of the FL algorithm that does not account for the wireless factors.

Appendix

A. *Proof of Theorem* 1

To prove Theorem 1, we first rewrite $F(\boldsymbol{g}_{t+1})$ using the second-order Taylor expansion, which can be expressed by:

$$\begin{aligned} F(\boldsymbol{g}_{t+1}) &= F(\boldsymbol{g}_t) + (\boldsymbol{g}_{t+1} - \boldsymbol{g}_t)^T \nabla F(\boldsymbol{g}_t) + \frac{1}{2}(\boldsymbol{g}_{t+1} - \boldsymbol{g}_t)^T \nabla^2 F(\boldsymbol{g})(\boldsymbol{g}_{t+1} - \boldsymbol{g}_t), \\ &\leq F(\boldsymbol{g}_t) + (\boldsymbol{g}_{t+1} - \boldsymbol{g}_t)^T \nabla F(\boldsymbol{g}_t) + \frac{L}{2}\|\boldsymbol{g}_{t+1} - \boldsymbol{g}_t\|^2, \end{aligned} \tag{25}$$

where the inequality stems from the assumption in (16). Given the learning rate $\lambda = \frac{1}{L}$, based on (12), the expected optimization function $\mathbb{E}[F(\boldsymbol{g}_{t+1})]$ can be expressed as:

$$\begin{aligned} \mathbb{E}[F(\boldsymbol{g}_{t+1})] \leq& \mathbb{E}\left(F(\boldsymbol{g}_t) - \lambda(\nabla F(\boldsymbol{g}_t) - \boldsymbol{o})^T \nabla F(\boldsymbol{g}_t) + \frac{L\lambda^2}{2}\|\nabla F(\boldsymbol{g}_t) - \boldsymbol{o}\|^2 \right), \\ \overset{(a)}{=}& \mathbb{E}\left(F\left(\boldsymbol{g}_t\right)\right) - \frac{1}{2L}\|\nabla F(\boldsymbol{g}_t)\|^2 + \frac{1}{2L}\mathbb{E}\left(\|\boldsymbol{o}\|^2\right), \end{aligned} \tag{26}$$

where (a) stems from the fact that $\frac{L\lambda^2}{2}\|\nabla F(\boldsymbol{g}_t)-\boldsymbol{o}\|^2=\frac{1}{2L}\|\nabla F(\boldsymbol{g}_t)\|^2-\frac{1}{L}\boldsymbol{o}^T\nabla F(\boldsymbol{g}_t)+\frac{1}{2L}\|\boldsymbol{o}\|^2$. Next, we derive $\mathbb{E}\left(\|\boldsymbol{o}\|^2\right)$, which can be given as follows:

$$\begin{aligned}
\mathbb{E}\left(\|\boldsymbol{o}\|^2\right) &= \mathbb{E}\left(\left\|\nabla F(\boldsymbol{g}_t)-\frac{\sum_{i=1}^{U}\sum_{k=1}^{K_i}a_i\nabla f\left(\boldsymbol{g},\boldsymbol{x}_{ik},y_{ik}\right)C\left(\boldsymbol{w}_i\right)}{\sum_{i=1}^{U}K_ia_iC\left(\boldsymbol{w}_i\right)}\right\|^2\right),\\
&= \mathbb{E}\left(\left\|\frac{\sum_{i=1}^{U}\sum_{k=1}^{K_i}\nabla f\left(\boldsymbol{g},\boldsymbol{x}_{ik},y_{ik}\right)}{K}-\frac{\sum_{i=1}^{U}\sum_{k=1}^{K_i}a_i\nabla f\left(\boldsymbol{g},\boldsymbol{x}_{ik},y_{ik}\right)C\left(\boldsymbol{w}_i\right)}{\sum_{i=1}^{U}K_ia_iC\left(\boldsymbol{w}_i\right)}\right\|^2\right),\\
&\leq \mathbb{E}\left(\left\|\frac{\sum_{i=1}^{U}\sum_{k=1}^{K_i}\nabla f\left(\boldsymbol{g},\boldsymbol{x}_{ik},y_{ik}\right)\left(1-a_iC\left(\boldsymbol{w}_i\right)\right)}{K}\right\|^2\right),
\end{aligned}\tag{27}$$

Since $\|\sum_{i=1}^{U}\sum_{k=1}^{K_i}\nabla f\left(\boldsymbol{g}_t,\boldsymbol{x}_{ik},y_{ik}\right)\|^2\leq K\sum_{i=1}^{U}\sum_{k=1}^{K_i}\|\nabla f\left(\boldsymbol{g}_t,\boldsymbol{x}_{ik},y_{ik}\|^2\right)$ and $\|\nabla f\left(\boldsymbol{g}_t,\boldsymbol{x}_{ik},y_{ik}\right)\|^2\leq\zeta_1+\zeta_2\nabla\|F\left(\boldsymbol{g}_t\right)\|^2$, (27) can be simplified as follows:

$$\begin{aligned}
\mathbb{E}\left(\|\boldsymbol{o}\|^2\right) &\leq \mathbb{E}\left(\frac{1}{K}\sum_{i=1}^{U}\sum_{k=1}^{K_i}\|\nabla f\left(\boldsymbol{g},\boldsymbol{x}_{ik},y_{ik}\right)\|^2\left(1-a_iC\left(\boldsymbol{w}_i\right)\right)\right),\\
&\leq \mathbb{E}\left(\left(\zeta_1+\zeta_2\nabla\|F\left(\boldsymbol{g}_t\right)\|^2\right)\left(1-a_iC\left(\boldsymbol{w}_i\right)\right)\right),\\
&= \frac{1}{K}\sum_{i=1}^{U}K_i\left(\zeta_1+\zeta_2\nabla\|F\left(\boldsymbol{g}_t\right)\|^2\right)\left(1-a_i+a_iq_i\left(\boldsymbol{r}_i,P_i\right)\right).
\end{aligned}\tag{28}$$

Therefore, we have:

$$\begin{aligned}
\mathbb{E}[F(\boldsymbol{g}_{t+1})] &\leq \mathbb{E}(F(\boldsymbol{g}_t))-\frac{1}{L}\|\nabla F(\boldsymbol{g}_t)\|^2+\frac{1}{2LK}\sum_{i=1}^{U}K_i\left(\zeta_1+\zeta_2\nabla\|F\left(\boldsymbol{g}_t\right)\|^2\right)\left(1-a_i+a_iq_i\left(\boldsymbol{r}_i,P_i\right)\right),\\
&= \mathbb{E}(F(\boldsymbol{g}_t))-\frac{1}{L}\left(1-\frac{\zeta_2}{2K}\sum_{i=1}^{U}K_i\left(1-a_i+a_iq_i\left(\boldsymbol{r}_i,P_i\right)\right)\right)\|\nabla F(\boldsymbol{g}_t)\|^2\\
&+\frac{\zeta_1}{2LK}\sum_{i=1}^{U}K_i\left(1-a_i+a_iq_i\left(\boldsymbol{r}_i,P_i\right)\right).
\end{aligned}\tag{29}$$

Subtract $\mathbb{E}[F(\boldsymbol{g}^*)]$ in both sides of (29), we have:

$$\begin{aligned}\mathbb{E}[F(\boldsymbol{g}_{t+1}) - F(\boldsymbol{g}^*)] \leq & \mathbb{E}(F(\boldsymbol{g}_t) - F(\boldsymbol{g}^*)) + \frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i\left(1 - a_i + a_i q_i\left(\boldsymbol{r}_i, P_i\right)\right) \\ & -\frac{1}{L}\left(1 - \frac{\zeta_2}{2K}\sum_{i=1}^{U} K_i\left(1 - a_i + a_i q_i\left(\boldsymbol{r}_i, P_i\right)\right)\right)\|\nabla F(\boldsymbol{g}_t)\|^2. \end{aligned} \tag{30}$$

By minimizing both sides of (15) with respect to $\boldsymbol{g}_{t+1}$, we have:

$$\min_{\boldsymbol{g}_{t+1}} F(\boldsymbol{g}_{t+1}) \geq \min_{\boldsymbol{g}_{t+1}} \left[F(\boldsymbol{g}_t) + (\boldsymbol{g}_{t+1} - \boldsymbol{g}_t)^T \nabla F(\boldsymbol{g}_t) + \frac{\mu}{2}\|\boldsymbol{g}_{t+1} - \boldsymbol{g}_t\|^2\right]. \tag{31}$$

The minimization of the left-hand side is achieved by $\boldsymbol{g}_{t+1} = \boldsymbol{g}^*$, while the minimization of the right-hand side is achieved by $\boldsymbol{g}_{t+1} = \boldsymbol{g}^t - \frac{1}{\mu}\nabla F(\boldsymbol{g}_t)$. Minimizing (31) yields:

$$F(\boldsymbol{g}^*) \geq F(\boldsymbol{g}_t) - \frac{1}{2\mu}\|\nabla F(\boldsymbol{g}_t)\|^2. \tag{32}$$

Hence, we have

$$\|\nabla F(\boldsymbol{g}_t)\|^2 \geq 2\mu(F(\boldsymbol{g}_t) - F(\boldsymbol{g}^*)). \tag{33}$$

Substituting (33) into (30), we have:

$$\begin{aligned}\mathbb{E}[F(\boldsymbol{g}_{t+1}) - F(\boldsymbol{g}^*)] \leq & \frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i\left(1 - a_i + a_i q_i\left(\boldsymbol{r}_i, P_i\right)\right) \\ & + \left(1 - \frac{2\mu}{L} + \frac{\mu\zeta_2}{LK}\sum_{i=1}^{U} K_i\left(1 - a_i + a_i q_i\left(\boldsymbol{r}_i, P_i\right)\right)\right)\mathbb{E}(F(\boldsymbol{g}_t) - F(\boldsymbol{g}^*)). \end{aligned} \tag{34}$$

Let $A = 1 - \frac{2\mu}{L} + \frac{\mu\zeta_2}{LK}\sum_{i=1}^{U} K_i\left(1 - a_i + a_i q_i\left(\boldsymbol{r}_i, P_i\right)\right)$. Applying (34) recursively, we have:

$$\begin{aligned}\mathbb{E}[F(\boldsymbol{g}_{t+1}) - F(\boldsymbol{g}^*)] \leq & \frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i\left(1 - a_i + a_i q_i\left(\boldsymbol{r}_i, P_i\right)\right)\sum_{k=1}^{t-1} A^k + A^t\mathbb{E}(F(\boldsymbol{g}_0) - F(\boldsymbol{g}^*)), \\ = & \frac{\zeta_1}{2LK}\sum_{i=1}^{U} K_i\left(1 - a_i + a_i q_i\left(\boldsymbol{r}_i, P_i\right)\right)\frac{1 - A^t}{1 - A} + A^t\mathbb{E}(F(\boldsymbol{g}_0) - F(\boldsymbol{g}^*)). \end{aligned} \tag{35}$$

This completes the proof.

B. Proof of Proposition 2

To prove Proposition 2, we first prove that $e_i\left(\boldsymbol{r}_i, P_i\right)$ is an increasing function of P_i. Based on (3) and (9), we have:

$$e_i\left(\boldsymbol{r}_i, P_i\right) = \varsigma\omega_i\vartheta^2 Z\left(\boldsymbol{X}_i\right) + \frac{P_i}{\sum\limits_{n=1}^{R} r_{i,n}B^{\mathrm{U}}\log_2\left(1+\kappa_{i,n}P_i\right)}, \tag{36}$$

where $\kappa_{i,n} = \frac{h_i}{\sum\limits_{i'\in\mathcal{U}'_n} P_{i'}h_{i'}+B^{\mathrm{U}}N_0}$. The first derivative of $e_i\left(\boldsymbol{r}_i, P_i\right)$ with respect to P_i is given by:

$$\frac{\partial e_i\left(\boldsymbol{r}_i, P_i\right)}{\partial P_i} = \frac{\left(\ln 2\right)\sum\limits_{n=1}^{R}\frac{r_{i,n}}{1+\kappa_{i,n}P_i}\left(\left(1+\kappa_{i,n}P_i\right)\ln(1+\kappa_{i,n}P_i)-\kappa_{i,n}P_i\right)}{\left(\sum\limits_{n=1}^{R} r_{i,n}B^{\mathrm{U}}\ln\left(1+\kappa_{i,n}P_i\right)\right)^2}. \tag{37}$$

Since $\frac{\partial e_i(\boldsymbol{r}_i,P_i)}{\partial P_i}$ is always positive, $e_i\left(\boldsymbol{r}_i, P_i\right)$ is a monotonically increasing function. Contradiction is used to prove Proposition 2. We assume that P_i' ($P_i' \neq P_i^*$) is the optimal transmit power of user i. In (10d), $e_i\left(\boldsymbol{r}_i^*, P_{i,\gamma_{\mathrm{E}}}\right)$ is a monotonically increasing function of P_i. Hence, as $P_i' > P_i^*$, $e_i\left(\boldsymbol{r}_i^*, P_i'\right) > \gamma_{\mathrm{E}}$, which does not meet the constraint (10f). From (7), we see that, the packer error rates decrease as the transmit power increases. Thus, as $P_i' < P_i^*$, we have $q_i\left(\boldsymbol{r}_i, P_i^*\right) \leqslant q_i\left(\boldsymbol{r}_i, P_i'\right)$. In consequence, as $P_i' < P_i^*$, P_i' cannot minimize the function in (21). Hence, we have $P_i' = P_i^*$. This completes the proof.

REFERENCES

[1] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "Performance optimization of federated learning over wireless networks," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, December 2019.

[2] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Communications Surveys & Tutorials*, to appear, 2019.

[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 Communications Surveys & Tutorials*, to appear, 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*, Jan. 2019.

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

[6] V. Smith, C. K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Proc. of Neural Information Processing Systems*, Long Beach, CA, USA, Dec. 2017.

[7] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Network*, to appear, 2019.

[8] Y. Zhao, J. Zhao, L. Jiang, R. Tan, and D. Niyato, "Mobile edge computing, blockchain and reputation-based crowdsourcing IoT federated learning: A secure, decentralized and privacy-preserving system," *arXiv preprint arXiv:1906.10893*, June 2019.

[9] 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.

[10] E. Jeong, S. Oh, J. Park, H. Kim, M. Bennis, and S. L. Kim, "Multi-hop federated private data augmentation with sample compression," in *Proc. of International Joint Conference on Artificial Intelligence Workshop on Federated Machine Learning for User Privacy and Data Confidentiality*, Macao, China, Aug. 2019.

[11] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *arXiv preprint arXiv:1908.07873*, Aug. 2019.

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

[13] 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*, Feb. 2017.

[14] 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," *arXiv preprint arXiv:1812.01202*, Dec. 2018.

[15] J. Konečnỳ, B. McMahan, and D. Ramage, "Federated optimization: Distributed optimization beyond the datacenter," *arXiv preprint arXiv:1511.03575*, Nov. 2015.

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

[17] S. Ha, J. Zhang, O. Simeone, and J. Kang, "Coded federated computing in wireless networks with straggling devices and imperfect CSI," *arXiv preprint arXiv:1901.05239*, Jan. 2019.

[18] O. Habachi, M. A. Adjif, and J. P. Cances, "Fast uplink grant for NOMA: A federated learning based approach," *arXiv preprint arXiv:1904.07975*, Mar. 2019.

[19] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *arXiv preprint arXiv:1812.02858*, Dec. 2018.

[20] Q. Zeng, Y. Du, K. K. Leung, and K. Huang, "Energy-efficient radio resource allocation for federated edge learning," *arXiv preprint arXiv:1907.06040*, July 2019.

[21] 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 Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, June 2019.

[22] S. Bi, J. Lyu, Z. Ding, and R. Zhang, "Engineering radio maps for wireless resource management," *IEEE Wireless Communications*, to appear, 2019.

[23] C. Hennig and M. Kutlukaya, "Some thoughts about the design of loss functions," *REVSTAT–Statistical Journal*, vol. 5, no. 1, pp. 19–39, March 2007.

[24] J. Konecny, H. B. McMahan, F. X. Yu, P. Richtarik, A. Theertha Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," in *Proc. of NIPS Workshop on Private Multi-Party Machine Learning*, Barcelona, Spain, Dec. 2016.

[25] Y. Xi, A. Burr, J. Wei, and D. Grace, "A general upper bound to evaluate packet error rate over quasi-static fading channels," *IEEE Transactions on Wireless Communications*, vol. 10, no. 5, pp. 1373–1377, May 2011.

[26] Y. Pan, C. Pan, Z. Yang, and M. Chen, "Resource allocation for D2D communications underlaying a NOMA-based cellular network," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 130–133, Feb 2018.

[27] M. P. Friedlander and M. Schmidt, "Hybrid deterministic-stochastic methods for data fitting," *SIAM Journal on Scientific Computing*, vol. 34, no. 3, pp. A1380–A1405, May 2012.

[28] M. Mahdian and Q. Yan, "Online bipartite matching with random arrivals: An approach based on strongly factor-revealing LPs," in *Proc. of the ACM symposium on Theory of computing*, San Jose, California, USA, June 2011.

[29] R. Jonker and T. Volgenant, "Improving the hungarian assignment algorithm," *Operations Research Letters*, vol. 5, no. 4, pp. 171–175, 1986.

[30] N. Landman K. Moore and J. Khim, "Hungarian maximum matching algorithm," https://brilliant.org/wiki/hungarian-matching/.