Energy-Efficient Radio Resource Allocation for Federated Edge Learning

Qunsong Zeng*, Yuqing Du*, Kaibin Huang*, and Kin K. Leung†
*Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong
† Department of Electrical and Electronic Engineering, Imperial College London, UK
Email: {qszeng, yqdu, huangkb}@eee.hku.hk, kin.leung@imperial.ac.uk

***Abstract*—Edge machine learning involves the development of learning algorithms at the network edge to leverage massive distributed data and computation resources. Among others, the framework of *federated edge learning* (FEEL) is particularly promising for its data-privacy preservation. FEEL coordinates global model training at a server and local model training at edge devices over wireless links. In this work, we explore the new direction of energy-efficient *radio resource management* (RRM) for FEEL. To reduce devices' energy consumption, we propose energy-efficient strategies for bandwidth allocation and scheduling. They adapt to devices' channel states and computation capacities so as to reduce their sum energy consumption while warranting learning performance. In contrast with the traditional rate-maximization designs, the derived optimal policies allocate more bandwidth to those scheduled devices with weaker channels or poorer computation capacities, which are the bottlenecks of synchronized model updates in FEEL. On the other hand, the scheduling priority function derived in closed form gives preferences to devices with better channels and computation capacities. Substantial energy reduction contributed by the proposed strategies is demonstrated in learning experiments.**

***Index Terms*—Energy efficient, radio resource allocation, federated edge learning.**

I. Introduction

Recent years have witnessed a phenomenal growth in mobile data, most of which are generated in real-time and distributed at edge devices (e.g., smartphones and sensors) [1]. Uploading these massive data to the cloud for training *artificial intelligence* (AI) models is impractical due to various issues including privacy, network congestion, and latency. To address these issues, the *federated edge learning* (FEEL) framework has been developed [2]–[4], which implements distributed machine learning at the network edge. In particular, a server updates a global model by aggregating local models (or stochastic gradients) transmitted by devices that are computed using local datasets. The updating of the global model using local models and the reverse are iterated till they converge. Besides preserving data privacy by avoiding data uploading, FEEL leverages distributed computation resources as well as allows rapid access to the real-time data generated by edge devices. One focus in the research area is communication-efficient FEEL where wireless techniques are designed to accelerate learning by reducing communication overhead and latency. However, the topic of energy-efficient communication for FEEL so far has not been explored. This is an important topic as training and transmission of large-scale models are energy consuming, while most edge devices especially sensors have limited battery lives. This topic is investigated in the current work where novel *radio-resource-management* (RRM) strategies for joint bandwidth allocation and user scheduling are proposed for minimizing the total device energy-consumption under a constraint on the learning speed.

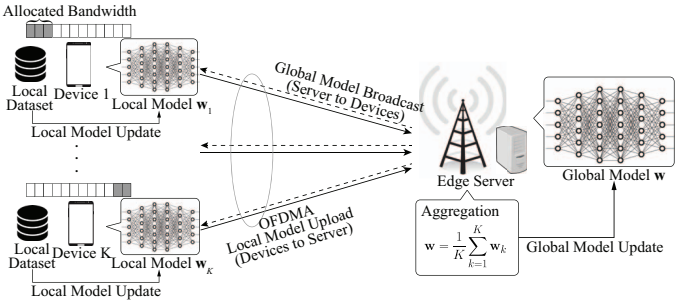

Figure 1. A framework for FEEL system.

The topic of communication-efficient FEEL has been extensively studied from different aspects. One branch of research focuses on edge-device selection so as to accelerate learning [5], [6]. In particular, a partial averaging scheme is proposed in [5], where only a portion of updates from fast-responding devices are used for global updating while those from stragglers are discarded. However, perfect device-update-uploading is assumed, which ignores the hostility of wireless channels, and at the same time overlooks the possibility of exploiting the sophisticated properties of wireless channels for improving the communication efficiency. By taking the properties into account, a joint device-selection and beamforming design is proposed for accelerating the federated edge learning [6]. Nevertheless, the device selection criterion is only based on the *channel-state information* (CSI) while ignoring the heterogeneous computation capacities of devices. On the other hand, to overcome the multi-access bottleneck, a *broadband analog aggregation* (BAA) multiple-access scheme is proposed in [2]. Specifically, by exploiting the waveform-superposition property of a multi-access channel, updates simultaneously transmitted by devices over broadband channels are analog aggregated "over-the-air" so as to reduce the multi-access latency. Given fixed communication cost per uploading, a control algorithm on uploading frequency is proposed in [4] by analyzing the convergence bound of distributed gradient

978-1-7281-7440-2/20/$31.00 ©2020 IEEE

descent to improve the learning performance. However, all these existing schemes are designed from the learning perspective while the energy-consumption issue of edge devices is out of scope, which is becoming increasingly important given the limited battery lives of devices. This motivates the current work on energy-efficient FEEL.

In this work, we consider the problem of minimizing energy consumption of edge devices in the context of FEEL without compromising learning performance. To this end, two energy-efficient RRM strategies are proposed for joint bandwidth allocation and user scheduling. To the best known of authors' knowledge, this work represents the first attempt to consider the energy-efficient RRM for FEEL.

To design the first energy-efficient RRM strategy, we assume a given set of edge devices and focus on bandwidth allocation. The optimal policy for energy minimization is derived in closed-form. The solution suggests that each edge device should utilize all the allowed uploading time so as to minimize the energy consumption. Furthermore, it can be observed from the solution that under the constraint of synchronous updates, less bandwidth should be allocated to devices with more powerful computation capacities and better channel conditions. This is in contrast with the traditional rate-maximization design.

The second strategy extends the first to include user scheduling, namely selecting devices to participate in FEEL. We propose a practical algorithm for iterating between solving two sub-problems under the criterion of energy minimization: 1) user scheduling and 2) bandwidth allocation using the first strategy. For user scheduling, the optimal policy is derived in closed-form, indicating the selection priorities for devices. The solution suggests that a device with a poor computation capacity and a bad channel has a lower priority to be selected and vice versa. Furthermore, the proposed algorithm is further improved by accounting for the effects of radio-resource utilization, where devices with fast computation capacity are allowed to occupy additional bandwidths for model uploading.

II. System Model

Consider a FEEL system consisting of a single edge server and K edge devices, denoted by a set $\mathcal{K} = \{1, \cdots, K\}$. For the FEEL system in Fig. 1, each iteration between local-model uploading and global-model updating is called a *communication round*. It is assumed that the server has perfect knowledge of the model size as well as multiuser channel gains and local computation capacities, which can be obtained by feedback. Using this information, for each communication round, the server determines the energy-efficient strategy for user scheduling and allocating bandwidth. Because communication rounds are independent, it suffices to consider the problem for an arbitrary round without loss of generality.

A. Multiple-access Model

Consider *orthogonal frequency-division multiple access* (OFDMA) for local model uploading with the total bandwidth B. Define $\gamma_k \in [0, 1]$ as the bandwidth allocation ratio for device k, and the allocated bandwidth is $\gamma_k B$. Furthermore, let h_k denote the instant/ average channel gain, targeting small scale/ fast fading, respectively. Given synchronous updates [7], a time constraint is set for local model training and uploading in each communication round:

$$\text{(Time constraint)} \quad t_k^{\text{comp}} + t_k \leq T, \quad \forall k \in \mathcal{K}, \tag{1}$$

where t_k^{comp} and t_k denote the time for local model training and uploading of device k, respectively. T is the maximum total time. The fact that devices have heterogeneous computation capacities is reflected in the differences among the values of $\{t_k^{\text{comp}}\}$. It follows from (1) that $t_k \leq T_k, \ \forall k \in \mathcal{K}$, where $T_k = T - t_k^{\text{comp}}$ is referred to as the allowed uploading time.

B. Energy Consumption Model

For each round, the energy consumption of a typical device comprises two parts: one for transmission (model uploading) and the other for local model training.

1) Energy consumption for model uploading: Let p_k denote the transmission power (in Watt/Hz) of device k. The achievable rate (in bit/s), denoted by r_k, can be written as [8]

$$r_k = \gamma_k B \log\left(1 + \frac{p_k h_k^2}{N_0}\right), \tag{2}$$

where N_0 is the variance of the complex white Gaussian channel noise. Let L denote the data size (in bit), the data rate can then be calculated as

$$r_k = \frac{\beta_k L}{t_k}, \tag{3}$$

where the state indicator $\beta_k = 1$ if device k is selected for uploading, or 0 otherwise. By combining (2) and (3), the uploading energy consumption is

$$E_k^{\text{up}} = \gamma_k B p_k t_k = \frac{\gamma_k B t_k N_0}{h_k^2}\left(2^{\frac{\beta_k L}{\gamma_k B t_k}} - 1\right). \tag{4}$$

2) Energy consumption for local training: Consider the local training of a *neural network* model via the well-known *backpropagation* (BP) algorithm on *graphics processing unit* (GPU). According to experiments reported in [9], the energy consumption of GPU only depends on the complexity of the BP algorithm and the size of model parameters. Since all devices train the same model of size L using the BP algorithm, the energy consumption of devices for local training is identical and denoted as E^{comp}.

C. Learning Speed Model

It is proved in [6], [10] that the convergence rate of distributed *stochastic gradient descent* (SGD) can be accelerated by involving more devices for global model updating. This is because the increment of devices can more significantly average out the effect of noise inherent in stochastic gradient, making the averaged gradient much closer to the ground-true one, and thereby accelerating learning. As such, we use the total number of scheduled devices as the measurement of

learning speed for tractability. By leveraging the indicators $\{\beta_k\}$, the learning speed can be expressed as

$$\text{(Learning speed)} \quad \sum_{k=1}^{K} \beta_k. \tag{5}$$

From the perspective of accelerating learning, it is desirable for the server to schedule as many devices as possible, which, however, is limited by finite radio resources.

III. Energy-Efficient Bandwidth Allocation

We consider the problem of bandwidth allocation for a given set of active devices which can all meet the time constraint in (1) ($\beta_k = 1, \forall k \in \mathcal{K}$). The allocation is performed by the server at the beginning of each round and fixed throughout the round. The constraint is relaxed later in Section V. The goal is to minimize the total energy consumption, i.e. $\sum_{k=1}^{K} \left(E^{\text{comp}} + E_k^{\text{up}}\right)$. Since the energy consumption for local model training, i.e. E^{comp}, is uniform and fixed, the problem focuses on minimizing uploading energy and thus is formulated as

$$(\mathbf{P1}) \quad \begin{aligned} \min_{\{\gamma_k, t_k\}} \ & \sum_{k=1}^{K} \frac{\gamma_k B t_k N_0}{h_k^2} \left(2^{\frac{\beta_k L}{\gamma_k B t_k}} - 1\right) \\ \text{s.t.} \ & \sum_{k=1}^{K} \gamma_k = 1, \quad 0 \le \gamma_k \le 1, \ k \in \mathcal{K}, \\ & 0 \le t_k \le T_k, \ k \in \mathcal{K}. \end{aligned}$$

By solving the above problem, the server can optimally determine bandwidth partitioning, as specified by $\{\gamma_k\}$, and the uploading time $\{t_k\}$ for devices. To begin with, one basic characteristic of Problem ($\mathbf{P1}$) is given as follows.

Lemma 1. The objective of Problem ($\mathbf{P1}$) is a non-increasing function in t_k and $\gamma_k, \forall k \in \mathcal{K}$.

The result follows from observing the derivative of the objective with the details omitted for brevity. It can be inferred from Lemma 1 that it is optimal to maximize the transmission time of each device, resulting in $t_k^\star = T_k, \forall k \in \mathcal{K}$, which is independent of the allocated bandwidth γ_k. Then, solving the *Karush-Kuhn-Tucker* (KKT) conditions, the optimal RRM policy of the convex Problem ($\mathbf{P1}$) is obtained as follows.

Theorem 1. (Optimal Bandwidth Allocation). The optimal policy for bandwidth allocation is

$$\gamma_k^\star = \frac{\beta_k L \ln 2}{B T_k \left[1 + \mathcal{W}\left(\frac{h_k^2 \nu^\star - B T_k N_0}{B T_k N_0 e}\right)\right]}, \ k \in \mathcal{K}, \tag{6}$$

$$t_k^\star = T_k, \ k \in \mathcal{K}, \tag{7}$$

where $\mathcal{W}(\cdot)$ is the Lambert W function, $T_k = T - t_k^{\text{comp}}$ is the restricted transmission time for device k, $\nu^\star$ is the solved value for the Lagrange multiplier and e is the Euler's number.

Proof: See Appendix A. □

Next, to gain more insight, a corollary is given as follows.

Corollary 1. $\gamma_k^\star$ is a non-increasing function with respect to T_k and h_k^2, respectively.

Proof: See Appendix B. □

One observation can be made from Corollary 1 is that more bandwidths should be allocated to devices with weaker computation capacities, namely smaller T_k. The reason is that these devices are the bottlenecks in synchronized updates and sum energy minimization. To be specific, they require larger bandwidths so as to complete model uploading within the short allowed uploading time and also to reduce transmission power. Furthermore, it can be observed that more bandwidths should be allocated to devices with weaker channels. Overcoming the conditions requires boosting transmission power or more bandwidths. For energy minimization, the latter is preferred.

Remark 1. (Rate-centric vs. Learning-centric RRM). The conventional RRM strategies for sum-rate maximization, such as *water-filling*, allocate more resources to users with stronger channels. In contrast, the derived RRM policy for FEEL allocates more resources to users with weaker channels and/or poorer computation capacities.

IV. Energy-and-Learning Aware Scheduling

In the presence of devices with poor computation capacities or weak channels, scheduling only a subset of devices for model uploading can reduce sum energy consumption as well as meet the time constraint. By modifying Problem ($\mathbf{P1}$) to include the learning speed in the objective, the current problem can be formulated as

$$(\mathbf{P2}) \quad \begin{aligned} \min_{\{\gamma_k, t_k, \beta_k\}} \ & \sum_{k=1}^{K} \frac{\gamma_k B t_k N_0}{h_k^2} \left(2^{\frac{\beta_k L}{\gamma_k B t_k}} - 1\right) - \lambda \sum_{k=1}^{K} \beta_k \\ \text{s.t.} \ & \beta_k \in \{0, 1\}, \ k \in \mathcal{K}, \\ & \sum_{k=1}^{K} \gamma_k = 1, \quad 0 \le \gamma_k \le 1, \ k \in \mathcal{K}, \\ & 0 \le t_k \le T_k, \ k \in \mathcal{K}, \end{aligned}$$

where the trade-off factor $\lambda > 0$ is a pre-determined constant. Directly solving the above problem is difficult due to its non-convexity arising from the integer constraint. To solve this problem, we adopt the common method of *relaxation-and-rounding*. It firstly relaxes the integer constraint $\beta_k \in \{0, 1\}$ as $0 \le \beta_k \le 1$, and then the integer solution is determined by rounding after solving the relaxed problem. It is also noted that the continuous value of β_k can be viewed as the selection priority of device k. Mathematically, it is easy to prove that the relaxed problem is convex. A standard solution approach is to use a numerical method since the optimization variables are all coupled. In the remainder of the section, we propose a more insightful approach that iterates between solving two sub-problems: 1) the bandwidth-allocation in Problem ($\mathbf{P1}$); 2) user scheduling problem. To be specific, the first sub-problem is to allocate bandwidths given scheduled devices indicated by $\{\beta_k\}$, given in Theorem 1. The other sub-problem (user

scheduling) is to decide on the selection priorities of devices, i.e. $\{\beta_k\}$, given $\{\gamma_k, t_k\}$:

$$(\mathbf{P3}) \quad \min_{\{\beta_k\}} \sum_{k=1}^{K} \frac{\gamma_k B t_k N_0}{h_k^2} \left(2^{\frac{\beta_k L}{\gamma_k B t_k}} - 1\right) - \lambda \sum_{k=1}^{K} \beta_k$$
$$\text{s.t.} \quad 0 \leq \beta_k \leq 1, \ k \in \mathcal{K}.$$

Problem $(\mathbf{P3})$ is convex and has the following solution.

Theorem 2 (Edge-device Selection Priority)**.** The optimal selection priority for device k is given as

$$\beta_k^\star = \min\left\{\max\left\{\frac{\gamma_k B T_k}{L} \log\left(\frac{\lambda h_k^2}{N_0 L \ln 2}\right), 0\right\} 1\right\}, \ k \in \mathcal{K}. \tag{8}$$

Proof: See Appendix C. □

This theorem is consistent with the intuition that device k with a high computation capacity and a good channel should have a high priority to be selected, i.e. $\beta_k^\star$ is large.

Remark 2. (Effects of Parameters on Selection Priority). It can be observed from (8) that β_k, indicating the selection priority of device k, scales with the allowed transmission time, i.e. T_k, linearly and with the channel gain approximately as $\log(h_k)$. The former scaling is much faster than the latter. This shows that the allowed transmission time (or equivalently computation capacity) is dominant over the channel on determining the selection priority of the device.

It follows that the solution of $(\mathbf{P2})$ is provided in Algorithm 1 by iteratively solving $(\mathbf{P1})$ and $(\mathbf{P3})$ until convergence.

Algorithm 1 Joint Bandwidth Allocation and User Scheduling

Initialization: Randomly set indicators $\{\beta_k\} \in [0, 1]$.
Iteration:

- **(Energy-efficient Bandwidth Allocation)**: Given fixed $\{\beta_k\}$, compute $\{\gamma_k, t_k\}$ using (6) and (7);
- **(Energy-and-Learning Aware Scheduling)**: Given fixed $\{\gamma_k, t_k\}$, compute $\{\beta_k\}$ using (8);

Until Convergence.
Round indicators $\{\beta_k\}$ to $\{0, 1\}$.
Compute $\{\gamma_k, t_k\}$ using (6) and (7).
Output the optimal solution $\{\beta_k^\star, \gamma_k^\star, t_k^\star\}$.

V. Improvement by Opportunistic Spectrum Access

In the preceding section, bandwidth allocation is assumed fixed throughout the communication round. If it is allowed to be dynamic and devices have spectrum sensing capabilities, Algorithm 1 can be improved by allowing opportunistic spectrum access as follows. At the beginning of each round, scheduling and bandwidth allocation are performed using Algorithm 1. Then in the actual operation, the relatively slow computation speeds of some devices may result in unoccupied spectrums in particular time slots. The opportunities can be sensed and exploited by faster devices for more energy-efficient uploading. Let the round be divided into multiple time slots. The above enhancement of Algorithm 1 is presented in Algorithm 2. Last, it is worth mentioning that a globally optimal solution for centralized RRM is intractable and requires an exhaustive search.

Algorithm 2 Opportunistic Spectrum Access

Initialization: Apply Algorithm 1 to obtain $\{\beta_k^\star, \gamma_k^\star\}$. Denote τ as the time slot duration and let $t_{\mathsf{count}} = 0$. For the subset of devices $\mathcal{S} = \{k \in \mathcal{K} \mid \beta_k^\star = 1\}$:
While $t_{\mathsf{count}} < T$:

- Denote $\mathcal{S}_l$ the set of devices that have not completed local computation at time t_{count}. For $k \in \mathcal{S}_l$, no bandwidth will be occupied by them;
- $\forall k \in \mathcal{S}/\mathcal{S}_l$, besides the allocated bandwidth $\gamma_k^\star B$, each device could sense additional $\frac{\sum_{k \in \mathcal{S}_l} \gamma_k^\star B}{|\mathcal{S}/\mathcal{S}_l|}$ bandwidths;
- $t_{\mathsf{count}} = t_{\mathsf{count}} + \tau$;

VI. Simulation Results

The simulation settings are as follows unless specified otherwise. There are $K = 50$ devices with local model training time, $\{t_k^{\text{comp}}\}$, following the uniform distribution in the range of $(0, 10]$ ms. Consider an OFDMA system where the bandwidth $B = 1$ MHz. The channel gains $\{h_k\}$ are modeled as independent Rayleigh fading with average path loss set as 10^{-4}. The noise variance is $N_0 = 10^{-8}$ W/Hz. The classifier model size is set as $L = 10^4$ bits and the task aims at classifying handwritten digits using the MNIST dataset. Each device is randomly assigned 20 samples. The model is a 6-layer *convolutional neural network* (CNN).

1) Energy-efficient bandwidth allocation: Consider the scenario that all devices are scheduled for uploading, the performance of the proposed RRM policy and its improved version are benchmarked against the *uniform bandwidth allocation* policy, which allocates equal bandwidth to devices. Particularly, the curves of total energy consumption by devices versus the communication round time T are shown in Fig. 2. Several observations can be made. First, the total energy consumption reduces as T grows for both cases. This agrees with Lemma 1 that the energy consumption is smaller if the allowed transmission time is larger. Second, it can be found the proposed RRM policies outperform the baseline, showing their effectiveness. Last, the opportunistic RRM policy contributes negligible energy consumption reduction as compared to the proposed strategy in (6) under current settings. Thus, we only consider Algorithm 1 in the following. It is noted that the gain can be more significant when the computation capacities among devices become increasingly heterogenous as more radio resources are efficiently utilized.

2) Energy-and-learning aware scheduling: Consider the scenario that the communication time is short and the edge server needs to select the edge-devices for uploading. The performance of the proposed Algorithm 1 for joint bandwidth allocation and user scheduling is benchmarked against the previous case that all edge-devices are selected. Particularly,

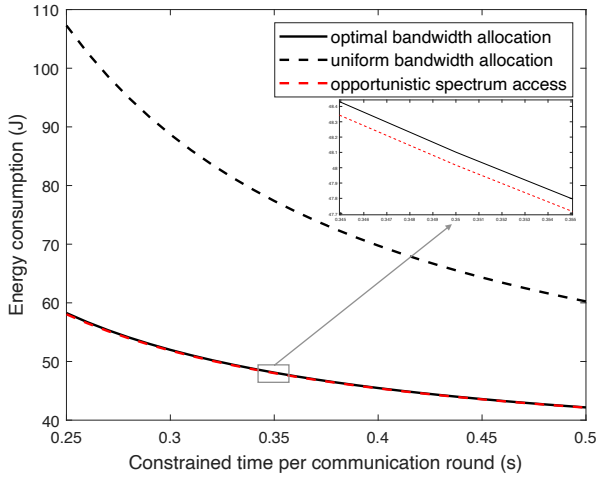

Figure 2. Sum device energy consumption vs. constrained time per communication round in a FEEL system.

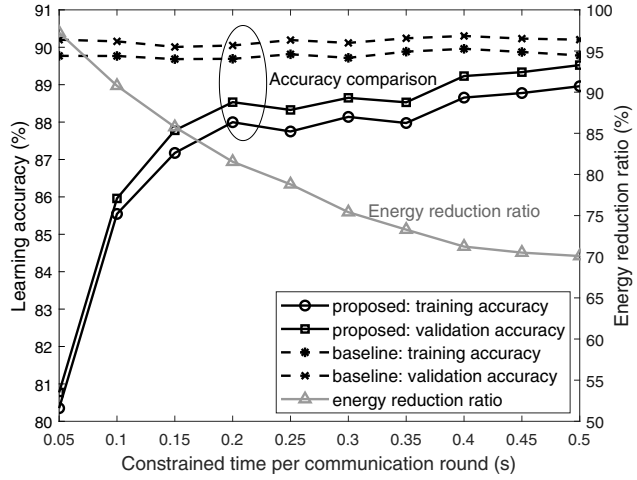

Figure 3. The learning accuracy vs. constrained communication-round time for the proposed scheme and the baseline are illustrated by the black solid and dashed lines, respectively. By defining the *energy reduction ratio* as $r = \frac{E_{\text{baseline}} - E_{\text{proposed}}}{E_{\text{baseline}}} \times 100\%$ with E_{proposed} and E_{baseline} denoting the sum energy consumptions of the proposed scheme and the baseline, respectively, the relationship between the energy reduction ratio and constrained communication-round time is shown by the grey line.

the relationship between the average learning accuracy of the federated learning algorithm and the constrained time T is illustrated in Fig. 3 given the fixed communication round 10. Several observations can be made. First, the performance of the baseline is independent of T. The reason is that the learning performance only depends on the number of scheduled edge-devices for uploading (i.e. $\sum_{k=1}^{K} \beta_k$), and this number is fixed for the baseline (i.e. $K = 50$). Second, the average learning accuracy of the proposed algorithm is an increasing function of T, whose performance approaches the baseline for the large T. The reason is that as the allowed transmission time increases, more devices will be scheduled for model uploading, giving rise to the performance improvement. Furthermore, it can be observed that the energy reduction ratio r, defined in the caption of Fig. 3, is a decreasing function of T, which approximately ranges from 70% to 98%. This is because that as T increases, the scheduled devices in the proposed scheme increases, and thereby the resulting sum energy consumption is larger. This reduces its difference to the sum energy consumption of the baseline of scheduling all devices.

VII. Concluding Remarks

In this paper, we have proposed energy-efficient RRM (bandwidth allocation and scheduling) for federated edge learning. By adapting to both channel states and computation capacities, the strategies effectively reduce sum device energy consumption while providing a guarantee on learning speed. This work makes the first attempt to explore the direction of energy-efficient RRM for federated edge learning. In the future, this work can be generalized into RRM for the asynchronous model-update scenario. Apart from the channel states and computation capacities, the sparsity of the updates can be also considered while allocating radio resources. Moreover, the effects of energy consumption model for local computing can be further taken into consideration to include the feature of local batch-size adaptation.

Appendix

A. Proof of Theorem 1

As aforementioned, one can have that $t_k^\star = T_k, \forall k$. Next, we prove the optimal bandwidth allocation strategy. Substituting $t_k = T_k$ into $(\mathbf{P1})$, it follows that the original Problem $(\mathbf{P1})$ can be rewritten as

$$\begin{aligned} \min_{\gamma_k} \ & \sum_{k=1}^{K} \frac{\gamma_k B T_k N_0}{h_k^2} \left(2^{\frac{\beta_k L}{\gamma_k B T_k}} - 1 \right) \\ \text{s.t.} \ & 0 \le \gamma_k \le 1, \ k \in \mathcal{K}, \ \sum_{k=1}^{K} \gamma_k = 1. \end{aligned} \tag{9}$$

Since the above problem is a convex problem, by introducing Lagrange multipliers $\boldsymbol{\mu}^\star = [\mu_1^\star, \mu_2^\star, \cdots, \mu_K^\star]^T \in \mathbb{R}^K$ for the inequality constraints $\boldsymbol{\gamma} \succeq 0$ with $\boldsymbol{\gamma} = [\gamma_1, \gamma_2, \cdots \gamma_K]^T$, and a multiplier $\nu^\star \in \mathbb{R}$ for the equality constraint $\mathbf{1}^T \boldsymbol{\gamma} = 1$, $\forall k \in \mathcal{K}$, the KKT conditions can be written as follows

$$\begin{aligned} & \boldsymbol{\gamma}^\star \succeq 0, \quad \mathbf{1}^T \boldsymbol{\gamma}^\star = 1, \quad \boldsymbol{\mu}^\star \succeq 0, \quad \mu_k^\star \gamma_k^\star = 0, \\ & \frac{B T_k N_0}{h_k^2} \left(2^{\frac{\beta_k L}{\gamma_k^\star B T_k}} - \frac{\beta_k L \ln 2}{\gamma_k^\star B T_k} 2^{\frac{\beta_k L}{\gamma_k^\star B T_k}} - 1 \right) - \mu_k^\star + \nu^\star = 0. \end{aligned} \tag{10}$$

By solving the above equations, one can have

$$\gamma_k^\star = \frac{\beta_k L \ln 2}{B T_k \left[1 + \mathcal{W}\left(\frac{h_k^2 \nu^\star - B T_k N_0}{B T_k N_0 e} \right) \right]}, \tag{11}$$

where $\mathcal{W}(\cdot)$ is the Lambert W function, and the Lagrange multiplier value $\nu^\star$ is calculated by solving

$$\sum_{k=1}^{K} \frac{\beta_k L \ln 2}{B T_k \left[1 + \mathcal{W}\left(\frac{h_k^2 \nu^\star - B T_k N_0}{B T_k N_0 e} \right) \right]} = 1. \tag{12}$$

B. Proof of Corollary 1

First, we prove that $\gamma_k^\star$ is non-increasing with respect to T_k. Denote $x = \frac{h_k^2\nu^\star - BT_kN_0}{BT_kN_0e}$, then it follows that $T_k = \frac{h_k^2\nu^\star}{\left(x+\frac{1}{e}\right)BN_0e}$. Substituting it to the expression for $\gamma_k^\star$, one can have

$$\gamma_k^\star = \frac{\beta_k L\ln 2}{BT_k\left[1+\mathcal{W}\left(\frac{h_k^2\nu^\star - BT_kN_0}{BT_kN_0e}\right)\right]} = \frac{N_0e\beta_kL\ln 2}{h_k^2\nu^\star}\frac{x+\frac{1}{e}}{1+\mathcal{W}(x)}. \tag{13}$$

Further, we denote

$$y = \frac{x+\frac{1}{e}}{1+\mathcal{W}(x)} = \frac{\mathcal{W}e^{\mathcal{W}(x)}+\frac{1}{e}}{1+\mathcal{W}(x)}. \tag{14}$$

It is easy to prove that y is non-decreasing with respect to $\mathcal{W}(x)$. Since $\mathcal{W}(x)$ is non-decreasing with respect to x and $x(T_k)$ is non-increasing with respect to T_k, it follows that $\gamma_k^\star$ is non-increasing with respect to T_k.

Next, we prove that $\gamma_k^\star$ is non-increasing with respect to h_k^2. From $x = \frac{h_k^2\nu^\star - BT_kN_0}{BT_kN_0e}$, one can have $h_k^2 = \frac{BN_0eT_k}{\nu^\star}\left(x+\frac{1}{e}\right)$. Substituting it into the expression for $\gamma_k^\star$, it follows that

$$\gamma_k^\star = \frac{\beta_k L\ln 2}{BT_k\left[1+\mathcal{W}\left(\frac{h_k^2\nu^\star - BT_kN_0}{BT_kN_0e}\right)\right]} = \frac{\beta_kL\ln 2}{BT_k}\frac{1}{1+\mathcal{W}(x)}. \tag{15}$$

Further, we let

$$z = \frac{1}{1+\mathcal{W}(x)}. \tag{16}$$

It is obvious that z is non-increasing with respect to $\mathcal{W}(x)$. Since $\mathcal{W}(x)$ is non-decreasing with respect to x and $x(h_k^2)$ is non-decreasing with respect to h_k^2, we can conclude that $\gamma_k^\star$ is non-increasing with respect to h_k^2. This completes the whole proof.

C. Proof of Theorem 2

Denote $\boldsymbol{\beta} = [\beta_1, \beta_2, \cdots, \beta_K]^T \in \mathbb{R}^K$ and define the function as follows:

$$J(\boldsymbol{\beta}) = \sum_{k=1}^{K}\left[\frac{\gamma_kBT_kN_0}{h_k^2}\left(2^{\frac{\beta_kL}{\gamma_kBT_k}}-1\right) - \lambda\beta_k\right], \tag{17}$$

then it follows that

$$\frac{\partial J(\boldsymbol{\beta})}{\partial\beta_k} = \frac{N_0L\ln 2}{h_k^2}2^{\frac{\beta_kL}{\gamma_kBT_k}} - \lambda, \tag{18}$$

Let $\frac{\partial J(\boldsymbol{\beta})}{\partial\beta_k} = 0$ and one can obtain the following result:

$$\hat{\beta}_k = \frac{\gamma_kBT_k}{L}\log\left(\frac{\lambda h_k^2}{N_0L\ln 2}\right). \tag{19}$$

When considering the constraint, it can be divided into three cases with respect to $\hat{\beta}_k$:

1) if $\hat{\beta}_k \le 0$, then the minimum will be obtained at $\beta_k = 0$;
2) if $0 < \hat{\beta}_k < 1$, then the minimum will be obtained at $\beta_k = \hat{\beta}_k$;
3) if $\hat{\beta}_k \ge 1$, then the minimum will be obtained at $\beta_k = 1$.

In summary, the optimal point is

$$\beta_k^\star = \min\left\{\max\left\{\frac{\gamma_kBT_k}{L}\log\left(\frac{\lambda h_k^2}{N_0L\ln 2}\right), 0\right\}, 1\right\}. \tag{20}$$

This completes the whole proof.

REFERENCES

[1] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Towards an intelligent edge: Wireless communication meets machine learning," *to appear in IEEE Commun. Mag.*, 2019.

[2] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. on Wireless Comm. DOI: 10.1109/TWC.2019.2946245,*, 2019.

[3] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," 2019. [Online]. Available: http://arxiv.org/abs/1901.00844.

[4] 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 Comm., INFOCOM*, pp. 63–71, Honolulu, HI, USA, Apr 16–19 2018.

[5] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *IEEE Intl. Conf. on Comm. (ICC)*, 2019.

[6] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," 2019. [Online]. Available: http://arxiv.org/abs/1812.11750.

[7] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proc. of the 20th Intel. Conf. Artificial Intell. and Statistics*, vol. 54, pp. 1273–1282, Fort Lauderdale, FL, USA, Apr 20–22 2017.

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

[9] X. Mei, Q. Wang, and X. Chu, "A survey and measurement study of GPU DVFS on energy conservation," *Digital Comm. and Networks*, vol. 3, no. 2, pp. 89–100, 2017.

[10] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, "signSGD: Compressed optimisation for non-convex problems," in *Proc. of the 35th Intl. Conf. Mach. Learning (ICML)*, vol. 80, pp. 560–569, Stockholmsmässan, Stockholm Sweden, Jul 10–15 2018.