EAI Endorsed Transactions
on Energy Web

Research Article EAI.EU

Provision for Energy: A Resource Allocation Problem in Federated Learning for Edge Systems

Mingyue Liu[1,*], Leelavathi Rajamanickam[2] and Rajamohan Parthasarathy[3]

[1,2]SEGi University, Centre for Software Engineering, Faculty of Engineering, Built Environment Information Technology
[3]SEGi University, Centre for Software Engineering, Centre for Network Security and IoT, Faculty of Engineering, Built Environment Information Technology

Abstract

The article explores an energy-efficient method for allocating transmission and computation resources for federated learning (FL) on wireless communication networks. The model being considered involves each user training a local FL model using their limited local computing resources and the data they have collected. These local models are then transmitted to a base station, where they are aggregated and broadcast back to all users. The level of accuracy in learning, as well as computation and communication latency, are determined by the exchange of models between users and the base station. Throughout the FL process, energy consumption for both local computation and transmission must be taken into account. Given the limited energy resources of wireless users, the communication problem is formulated as an optimization problem with the goal of minimizing overall system energy consumption while meeting a latency requirement. To address this problem, we propose an iterative algorithm that takes into account factors such as bandwidth, power, and computational resources. Results from numerical simulations demonstrate that the proposed algorithm can reduce energy consumption compared to traditional FL methods up to 51% reduction.

Keywords: Energy Efficiency, Resource Allocation, Federated Learning

Received on 28 December 2023, accepted on 25 June 2024, published on 03 July 2024

doi: 10.4108/ew.6503

1. Introduction

There has been significant growth in mobile data in recent years, much of it generated in real-time and distributed to edge devices such as smartphones and sensors [1] [2]. Artificial intelligence (AI) technology is widely used to process this mobile data and support various services, such as computer vision and the internet of vehicles [3]. A common practice is to train AI models using elastic cloud computing, which allows operators to achieve optimal performance by accessing large-scale datasets. However, this process poses challenges due to privacy concerns [4], network congestion [5], and service latency [6]. Federated learning (FL) in the edge framework offers a solution to these issues. FL implements distributed machine learning at the network edge, where clients, or edge devices, train local models with their private data and only share parameters like model weights [7] [8] [9]. An FL server is used to aggregate these models into a global model and broadcast updates to each edge device. After several iterations, accuracy is achieved, and the training process is completed. FL avoids the need for data uploads and enables rapid access to real-time data, thus reducing pressure on communication resources and lowering service latency. It is a promising distributed learning algorithm that is likely to be applied in future internet of things systems [10, 11, 12, 13, 14, 15].

Wireless devices, such as those that communicate through cellular networks, benefit greatly from the use of distributed learning frameworks. These frameworks allow for the training of locally collected data using a shared learning model [16, 17, 18]. However, edge devices such

[*] Corresponding author. Email: mingyue2022@126.com

as smartphones have limited computation resources, and the spectrum resource is scarce. Only a few edge devices are able to upload their trained local models in each round. Additionally, the limited battery life of these devices is a growing concern, and the energy consumption caused by communications is increasing.

In addition, wireless devices can cooperate and execute a learning task by only uploading their local learning models to the base station (BS) instead of sharing the full training data [19]. One approach to improve this process is the use of a gradient quantization-based digital transmission scheme [20]. The coverage area of wireless devices is also taken into consideration to reduce the number of edge devices needed [21]. However, the limited wireless resources such as time or bandwidth, make it necessary for wireless devices to transmit their local training results over wireless links [22], which can affect the accuracy of the FL framework in edge systems [23]. Furthermore, the limited energy of wireless devices makes energy efficiency optimization crucial for the successful deployment of FL due to these resource constraints [24].

In order to address the challenges discussed above, we propose a framework that balances both energy consumption and learning accuracy. Specifically, we model the energy consumption of computation and communication in the FL framework for edge systems. The novelty of this work is to take both learning accuracy and resource constraints into consideration for FL. The main contribution of this paper is to construct an energy consumption minimization problem and provision of a solution to this problem. Our key contributions include:

- Using wireless communication networks, we investigate the performance of FL algorithms for a scenario in which each user locally computes its model under a given learning accuracy, while the BS broadcasts the aggregated model to all users. The convergence rate of FL is first determined for the considered algorithm.
- An optimization problem is formulated to minimize total energy consumption for both local computation and wireless transmission. A low- complexity iterative algorithm is proposed to solve this problem. This algorithm includes new closed-form solutions for time allocation, bandwidth allocation, power control, computation frequency, and learning accuracy.
- The FL completion time minimization problem is established as a feasible solution to the total energy minimization problem. Our theoretical analysis shows that the completion time is dependent on learning accuracy. To minimize FL completion time, we propose a bisection-based algorithm that is based on the theoretical result.

2. Related works

Mobile data is processed and supports various services through the use of artificial intelligence (AI) technology [3]. Elastic cloud computing is commonly used to train AI models, which can access large-scale datasets and achieve optimal performance [4,5,6]. However, this process presents challenges such as privacy concerns, network congestion, and service latency. To address these issues, federated learning (FL) in the edge framework implements distributed machine learning at the network edge [7,8,9]. In FL, clients or edge devices train local models with their private data and only share parameters like model weights.

Studies have been conducted on the challenges of federated learning (FL) over wireless networks. For example, in [25], a broadband analog aggregation scheme was used to minimize communication latency in multi-access channels. In [26], the authors explored a minimization problem for FL in cell-free venues using multiple inputs and outputs (MIMO) systems, and in [27], an energy- aware user scheduling policy was proposed to maximize the number of scheduled users in FL with redundant data. In [28], a novel sparse and low-rank model was developed to improve statistical learning performance for on-device distributed training, and in [29], an energy-efficient bandwidth allocation scheme was proposed with constraints on learning performance. However, these works tend to focus on the trade-off between completion time and energy in wireless transmission and do not take into account the trade-off between learning and transmission. In recent works, such as [30], [31] and [32], the authors considered both local learning and communication energy but did not take into account computation delays on local FL models, and it was not feasible for all users to send their learning model synchronously.

3. System model

This paper presents a framework for a federated edge learning system that includes a set of edge devices and one edge server (BS) as shown in Figure 1, the applications in federated learning can be self-driving, users augmented or virtual reality headsets. Typically, AI-enabled edge applications can also be allocated in federated learning frameworks. The set of edge devices is denoted as K = 1, 2, · · ·, k, and each device has its own local data sample D_k. Additionally, we use xi and y_i to represent the input and output of data sample i, respectively. The edge server and users jointly execute the learning model, with the learning algorithm trained on both the user-local side (called the local model) and the server side (called the global model). The communication between the user and server is the main process that consumes cost and energy.

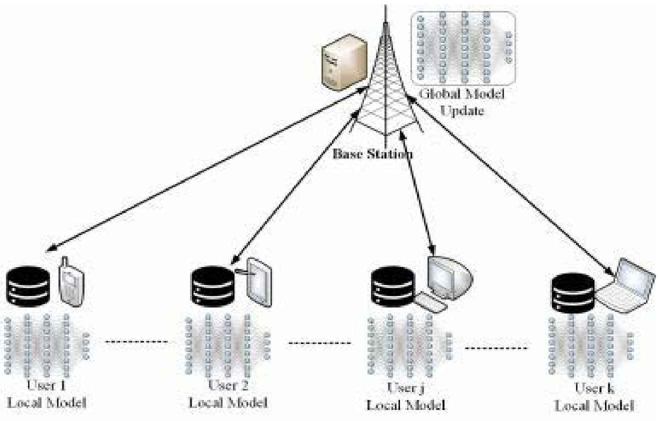

Figure 1. Federated learning framework

3.1 Computation and Communication Model

The process of FL is illustrated in Figure 1. It can be seen that FL consists of three parts: local computation on the user side, global computation on the server side, and global communication between users and the server. The process begins with the calculation of the model on the local side using its data and then obtaining the global model from the base station.

(1) Local Computation: The computation capacity of each device is denoted as f_k , which represents the CPU circle of the devices. So, the computation time can be calculated by the data size and capacity

$$\tau_k = \frac{\beta_k D_k C_k}{f_k} \quad (1)$$

while C_k is the number of CPU demand of data sample at user k and β is the fraction of data sample executed locally. The energy consumption is affected by the CPU frequency and CPU chip. So, the energy consumption caused by local computation for $\beta_k D_k C_k$ is denoted as

$$E_k^c = \alpha_k \beta_k D_k C_k f_k^2 \quad (2)$$

where α is the coefficient of the chip architecture at edge device k.

(2) Global Communication: After the process of local computation, the users upload their trained data to BS through the wireless access. The rate of user k is

$$r_k = b_k log_2 \left(1 + \frac{p_k h_k^2}{N_0}\right) \quad (3)$$

where p_k is the transmit power, N_0 is the Gaussian channel noise and h_k denotes the channel gain between the user k and the base station. We know that Shannon capacity gives an upper bound of the transmission rate. Because the transmitting data size is $\beta_k D_k$, so the rate can also be calculated as

$$r_k = \frac{\alpha_k D_k}{t_k} \quad (4)$$

It is clear that, the energy consumption of uploading data to BS is

$$E_k^t = \beta_k B p_k t_k = \frac{\beta_k B t_k N_0}{h_k^2}\left(2^{\frac{\beta_k D_k}{\beta_k B t_k}} - 1\right)\cdot \quad (5)$$

The overall consumption contains the cost in both process of computation E_k^c and transmission E_k^t:

$$E = \sum_{k=1}^{K} (E_k^c + E_k^t) \quad (6)$$

The completion time of FL algorithm is the main concern of the edge systems. So, the completion time of user k includes both the computation and communication time, denoted as

$$T_k = \tau_k + t_k = \frac{\beta_k D_k C_k}{f_k} + t_k \quad (7)$$

while the completion time of algorithm is the maximum completion time T_k.

3.2 Federated learning model

In federated edge learning, we use θ to represent the training parameters related to the global model. We define D_k as the data sample for user k. We define the loss function $f_i(\theta) = (y_i - \theta^T x_i)^2$ which represents the difference between the input and output. We aim to minimize this loss function to find the optimal value of θ. The reason we adopt this expression is that in most federated learning applications, the training sample is large enough that we can assume that the difference between the input and output follows a normal distribution with a mean of 0. Therefore, for a specific sample x_i and given θ, the conditional probability of y_i can be expressed as:

$$p(y(i) \mid x(i); \theta) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y(i)-\theta^T x(i))^2}{2\sigma^2}\right)$$

(8)

We assume that all samples are independent and identically distributed. To find the optimal θ, we define the likelihood function. Our goal is to obtain as many observed outputs as possible, so we maximize the product of the likelihoods of each individual sample.

$$L(x, y) = \prod_{i=1}^{D_k} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y(i)-\theta^T x(i))^2}{2\sigma^2}\right) \quad (9)$$

We then convert the Eq. (9) into a log-likelihood function. The reason for this is that logarithms are strictly increasing functions, so maximizing the likelihood is equivalent to maximizing the log-likelihood:

$$\log L(x, y) = -\frac{D_k}{2}\log 2\pi - D_k \log \sigma - \frac{\sum_{i=1}^{D_k}((y(i)-\theta^T x(i))^2)}{2\sigma^2}$$

(10)

We remove items that are not related to $\theta: -\frac{D_k}{2}\log 2\pi$ and $D_k \log \sigma$, and convert the remaining item into a negative log-likelihood: $\frac{\sum_{i=1}^{D_k}((y(i)-\theta^T x(i))^2)}{2\sigma^2}$

By setting σ = 1 we obtain the original loss function. Essentially, maximizing the likelihood is equivalent to minimizing the loss function. Thus, we can obtain the optimal θ by minimizing the loss function.

The objective of federated learning is to find a set of parameters that minimize the value of F_k (θ).

$$F_k(\theta) = \frac{1}{D_k}\sum_k f_i(\theta) \quad (11)$$

Since FL is designed to share a model among users, data communication of various data samples is crucial in the FL problem. Therefore, the FL training problem can be formulated as

$$\underset{\theta}{min} F(\theta) = \sum\nolimits_{k} \tfrac{D_k}{D} F_k(\theta) = \tfrac{1}{D} \sum\nolimits_{k} \sum\nolimits_{D_k} f_i(\theta)$$

(12) where $D = \sum D_k$ is the total data size of all users in the system. The BS collects the partially trained models from all users and updates the global model based on the collection of trained models and then distributes the updated model to all users. The process involves several local iterations and global iterations. The number of efficient computations is smaller than local updates. From the system ' s perspective, it is crucial to schedule as much as possible under the limitations of wireless resources while providing good performance in terms of completion time and energy consumption.

4. Problem formulation

In this section, we formulate a problem of minimizing the energy consumption of all devices while taking into account the constraint of latency. The objective of our proposed model is to minimize the overall energy consumption of the K edge devices. The energy-oriented problem can be formulated as follows:

$$min_{(\beta_k, t_k)} \sum_{k=1}^{K} \frac{\beta_k B t_k N_0}{h_k^2} (2^{\frac{\beta_k D_k}{\beta_k B t_k}} - 1) \ (13)$$

$$\text{s.t. } \sum_{k=1}^{K} \beta_k = 1{,}0 \le \beta_k \le 1 \ (14)$$

$$t_k b_k log_2(1 + \frac{p_k h_k^2}{N_0}) \ge d_k, \forall k \in K \ (15)$$

$$0 \le t_k \le T_k, \forall k \in K \ (16)$$

$$0 \le f_k \le f_k^{max}, 0 \le p_k \le p_k^{max} \ \ \forall k \ (17)$$

$$t_k \ge 0, b_k \ge 0, \forall k \in K \ (18)$$

while f_k^{max} and p_k^{max} denote the maximum computation capacity and maximum transmit power of user k, respectively. This is the original problem P1, whose objective is to minimize the total energy consumption of all users, including the computation and communication phases. Constraint Eq. (14) is based on the definition of β_k, the resource allocated to all k users. Constraint Eq. (15) indicates that the communication time is sufficient to transmit the data sample to the BS. The completion time constraint is limited in Eq. (16), and constraint Eq. (17) describes the limitations of frequency and transmit power.

Lemma1 The problem P1 is a non-increasing function with t_k and β_k, $\forall k \in K$. The derivative of the P1 objective shows that it is a non-increasing function. Thus, the optimal solution is found by optimizing the transmission time of each device, which is independent of another parameter β_k.

Then, by applying the KKT conditions [33], the optimal solution of P1 can be obtained as follows.

The optimal solution of bandwidth allocation is

$$\beta_k^* = \frac{\beta_k D_k \ln 2}{B T_k [1 + W \frac{h_k^2 u_* - B T_k N_0}{B T_k N_0 e}]}, k \in K, \qquad (19)$$

$$t_k^* = T_k \qquad (20)$$

We use W to denote the Lambert W function [34]. The transmission time Tk is strictly constrained by the transmission time of the device k with the largest completion time, when u∗ and e are used as Lagrange multipliers.

Proof: From the Eq. (4.3.19), use T_k to replace t_k in P1, so the original problem P1 can be rewritten as:

$$\min_{(\beta_k)} \sum_{k=1}^{K} \frac{\beta_k B T_k N_0}{h_k^2} \left(2^{\frac{\beta_k D_k}{\beta_k B T_k}} - 1 \right), \qquad (21)$$

$$\text{s.t.} \qquad \text{Eq.}\,(14)(16)(17)(18), \qquad (22)$$

$$T_k b_k \log_2 \left(1 + \frac{p_k h_k^2}{N_0} \right) \ge d_k, \forall k \in K. \qquad (23)$$

As is mentioned before, it is a convex problem. Lagrange multiplier $\mu^* = [\mu_1, \mu_2, \dots, \mu_k^*]^T$ and $\gamma = [\gamma_1, \gamma_2, \dots, \gamma_k]^T$ are applied, and the multiplier u^*, so the KKT conditions can be written as

$$\beta \ge 0, \mu_k^* \ge 0, \mu_k^* \beta_k^* = 0,$$

$$\frac{B T_k N_0}{h_k^2} (2^{\frac{\beta_k D_k}{\beta_k B T_k}} - \frac{\beta_k D_k \ln 2}{\beta_k^* B T_k} 2^{\frac{\beta_k D_k}{\beta_k^* B T_k}} - 1) - \mu_k^* + u^* = 0.$$

To solve this problem, it can be obtained

$$\gamma_k^* = \frac{\beta_k D_k \ln 2}{B T_k \left[1 + W \frac{h_k^2 u^* - B T_k N_0}{B T_k N_0 e} \right]}, \qquad (24)$$

and the W(.) is the Lambert W function, the multiplier value u ∗ can be calculated by solving

$$\sum_{k=1}^{K} \frac{\beta_k D_k \ln 2}{B T_k [1 + W \frac{h_k^2 u^* - B T_k N_0}{B T_k N_0 e}]} = 1. \qquad (25)$$

Here, $x = \frac{h_k^2 u^* - B T_k N_0}{B T_k N_0 e}$ is used, then $T_k = \frac{h_k^2 u^*}{(x + \frac{1}{e}) B N_0 e}$, and substitute it into γ_k^*,

$$\gamma_k^* = \frac{\beta_k D_k \ln 2}{B T_k [1 + W \frac{h_k^2 u^* - B T_k N_0}{B T_k N_0 e}]} = \frac{N_0 e \beta_k D_k \ln 2}{h_k^2 u^*} \frac{x + \frac{1}{e}}{1 + W(x)}.$$

In order to solve it, this can be used

$$y = \frac{x + \frac{1}{e}}{1 + W(x)} = \frac{W e^{W(x)} + \frac{1}{e}}{1 + W(x)}. \qquad (26)$$

It is obvious that y is a non-decreasing variable with W(x), so γ_k^* is also non-decrease with T_k. From the equation of x, it can conclude that $h_k^2 = \frac{BN_0eT_k}{u^*}(1+\frac{1}{e})$. Then, replace h_k^2 in γ_k^*shown as

$$\gamma_k^* = \frac{\beta_kD_k\ln 2}{BT_k[1+W\frac{h_k^2u^*-BT_kN_0}{BT_kN_0e}]} = \frac{\beta_kD_k\ln 2}{BT_k}\frac{1}{1+W(x)}.$$

Here, if $z = \frac{1}{1+W(x)}$ is defined, it is easy to obtain z is non-increasing to W(x). Since W(x) is non-decreasing with x and h_k^2, so γ is also non-increasing with h_k^2.

It can be seen that more bandwidth resources should be allocated to devices with weak computation capacities. This is because the main factor affecting the minimization of energy consumption is the synchronous updating and execution of tasks. For simplicity, weak devices require larger bandwidth to complete the entire process of computation and communication. In addition, more bandwidth should be allocated to weak channels, as weak channels have a lower transmission rate, and therefore, require larger bandwidth.

5. Methodology

In this section, an efficient algorithm is proposed to solve problem P1, which was formulated in the previous section. The objective of solving problem P1 is to determine if these devices can finish their task within the completion time T. Therefore, it is equivalent to transforming P1 into the following problem P2:

$$\min_{(\beta_k,\gamma_k,t_k)} \sum_{k=1}^{K}\frac{\beta_kBt_kN_0}{h_k^2}(2^{\frac{\beta_kD_k}{\beta_kBt_k}}-1)-\lambda\sum_{k=1}^{K}\beta_k \tag{27}$$

$$\text{s.t.}\quad \beta_k \in \{0,1\}, k\in K, \tag{28}$$

$$t_kb_k\log_2\left(1+\frac{p_kh_k^2}{N_0}\right) \ge d_k, \forall k \in K, \tag{29}$$

$$0 \le t_k \le T_k, \forall k \in K, \tag{30}$$

$$0 \le f_k \le f_k^{max}, 0 \le p_k \le p_k^{max}\forall k \in K, \tag{31}$$

$$t_k \ge 0, b_k \ge 0, \forall k \in K, \tag{32}$$

while λ is defined as a pre-trained parameter. This problem is difficult to solve due to the integer constraint. Therefore, relax the constraint $\beta_k \in 0{,}1$ to $0 \le \beta \le 1$, allowing the integer problem to be solved by the relaxed problem. β_k can be considered as the importance of various devices, which includes both bandwidth allocation and sequence scheduling. In detail, the bandwidth problem can be solved by P1, and the second part is to decide the importance of devices, defined as P3,

$$\min_{\beta_k} \sum_{k=1}^{K}\frac{\beta_kBt_kN_0}{h_k^2}(2^{\frac{\beta_kD_k}{\beta_kBt_k}}-1)-\lambda\sum_{k=1}^{K}\beta_k \tag{33}$$

$$\text{s.t.}\quad 0 \le \beta_k \le 1, k \in K, \tag{34}$$

$$\text{Eq.}(15)(16)(17)(18). \tag{35}$$

Based on the definition of $\beta = [\beta_1, \beta_2, \dots, \beta_k]$ and format of P3, another function is introduced:

$$G(\beta) = \sum_{k=1}^{K}\left[\frac{\gamma_kBt_kN_0}{h_k^2}\left(2^{\frac{\beta_kD_k}{\gamma_kBt_k}}-1\right) - \lambda\beta_k\right]. \tag{36}$$

And the partial derivative of $G(\beta)$ can be calculated as:

$$\frac{\partial(G(\beta))}{\partial(\beta_k)} = \frac{N_0D_k\ln 2}{h_k^2}\left(2^{\frac{\beta_kD_k}{\gamma_kBt_k}}-1\right) - \lambda. \tag{37}$$

If $\frac{\partial(G(\beta))}{\partial(\beta_k)} = 0$, the result can be obtained:

$$\beta_k' = \frac{\gamma_kBt_K}{D_k}\log\left(\frac{\lambda h_k^2}{N_0D_k\ln 2}\right). \tag{38}$$

Here discusses the result of β_k' under the limitations of the P3:

If $\beta_k' \le 0$, the minimizing value will be at $\beta_k = 0$;

If $0 < \beta_k' \le 0$, the minimizing value will be at $\beta_k' = \beta_k$;

if $\beta_k' \le 1$, the minimizing value will be at $\beta_k = 1$.

In conclusion, the optimizing value is

$$\beta_k' = \min\left\{\max\left\{\frac{\gamma_kBt_K}{D_k}\log\left(\frac{\lambda h_k^2}{N_0D_k\ln 2}\right), 0\right\}, 1\right\}. \tag{39}$$

From this result, it can be inferred that the importance of devices with strong computation capacity and bandwidth is higher than that of others.

Based on the result of β_k and Eq. (39), it can be concluded that the importance of device k is related to transmission time t_k and the condition of the channel h_k. The effect of t_k is larger than that of h_k based on Eq. (39).

Algorithm1 Allocation method

```
Algorithm1 Allocation for User
Input: The set of user devices, k;
The bandwidth resources, B;
The basic information of channel, $N_0$ and $h_k$;
Output:
Initialize parameter $\beta_k \in [0,1]$;
Calculate $\gamma_k$ using Eq. (4.3.19) with $\beta_k$;
Calculate $t_k$ using Eq. (4.3.20) with $\beta_k$;
Calculate $\beta_k$ using Eq. (4.3.39) with $\{\gamma_k, t_k\}$;
Calculate until convergence;
Compute $\beta_k$ to $\{0,1\}$;
   Compute $\{\gamma_k, t_k\}$ using Eq. (4.3.19)(4.3.20);
Return $\{\beta_k', \gamma_k^*, t_k^*\}$
```

The complexity of Algorithm1 is determined by the number of iterations of Eq. (5), (6), (7). The optimal solution of Eq. (11) is obtained by the bisection method, which has a complexity of $\mathcal{O}(K\log 2n)$, where n is the interval of β_k. In this algorithm, the bandwidth allocation is fixed during each time slot, which results in a certain energy consumption at each iteration.

6. Results and discussion

In this simulation settings, the device number is set as K=50 following the uniform distribution in the area of 500m*500m. In the edge system, the bandwidth is set as B= 2MHz. The transmit power is defined as 10 dB, and the computation frequency is 2 GHz. In the meantime, the power gain that presents noise is set as -30 dB. It is considered that the machine learning model is CNN, and the data set is applied by MNIST, and 500 data samples are in total.

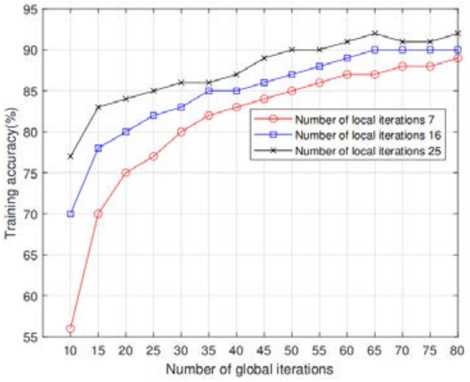

Figure 2. The training accuracy versus the number of global iterations with different local iteration

Figure 2 shows the results of training accuracy under different iterations. It can be seen that the accuracy of the running model is affected by the number of local and global iterations. According to this figure, it can be observed that with a fixed local iteration, the accuracy increases as the number of global iterations get larger. In the meantime, the overall accuracy of more significant local iterations is larger than smaller iterations. This shows that more local computation can be traded for communication among the edge nodes.

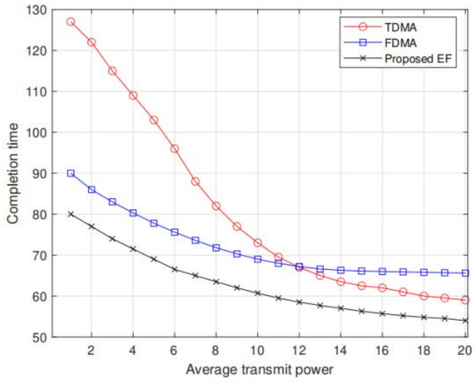

Figure 3. Completion time comparison with various transmit power

Most applications in the edge systems desire for lower completion time, so it can compare the proposed EF methods with FDMA and TDMA methods (Tran et al., 2019). Figure 3 shows the variation of completion time with the increase of average transmit power. It can be observed that the completion time decreases with the increase of transmit power, which is because the increase of transmit power can reduce the communication time of the end-user and BS. The proposed EF method demonstrates a 19.8% and 13.2% reduction in energy consumption compared to other methods. The method takes into account both energy and latency factors of tasks in the edge system. FDMA performs better than the TDMA method because the TDMA method divides time slots among different users, the task of the user may not be ready when it's their turn to use the time slot. Thus, the completion time of the TDMA method is worst in this scenario.

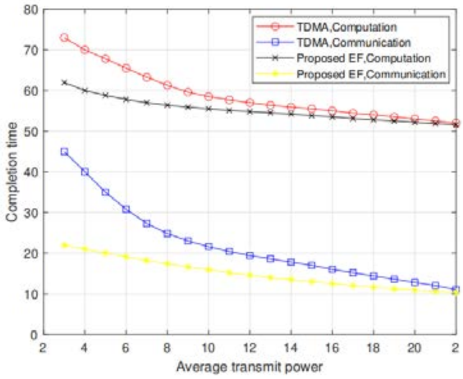

Figure 4. Computation and communication completion time comparison with various transmit power

In addition, it is necessary to find out the relationship between computation and communication in both methods. The changes in the maximum average transmit power of each user are illustrated in Figure 4, which shows how these variations affect the amount of time needed for communication and processing. It is clear from looking at the chart that both the amount of time spent communicating and the amount of time spent computing decreases as the maximum average transmit power of each user grows. It is also possible to notice that the amount of time required for computing is invariably greater than the amount of time required for communication, and that the rate at which the amount of time required for communication is decreasing is higher than that of the computation time.

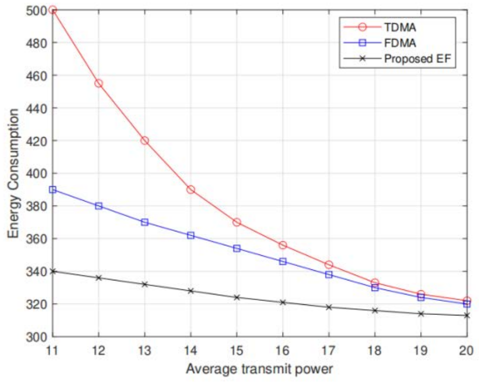

Figure 5. Energy consumption comparison with various transmit power within T=150

Energy consumption is an important concern for edge users. Figure 5 illustrates how energy consumption changes as transmit power increases over a period of T=150s. It can be seen from the figure that the proposed EF method outperforms TDMA and FDMA. Additionally, energy consumption decreases as transmit power increases, as a result of the reduced completion time as shown in Figure 3. TDMA method allocates specific time slots to users and the users respond periodically, which leads to wasted energy.

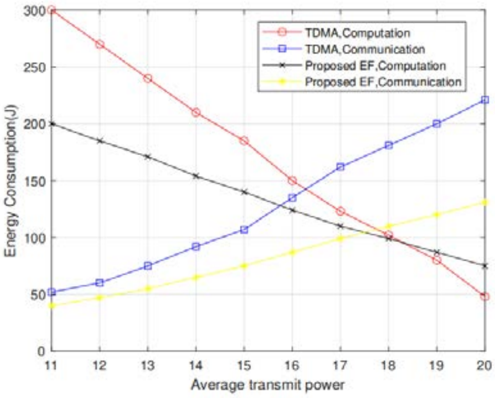

Figure 6. Computation and communication consumption comparison with various transmit power

Figure 6 shows the relationship between computation and communication energy consumption and transmit power. With the increase of transmit power, communication consumption increases because communication costs are affected by transmit power and task size. While the transmit power is smaller than 16dB, the proposed EF outperforms the TDMA method in both computation and communication energy consumption. When the transmit power is larger than 18dB, TDMA outperforms in computation energy consumption.

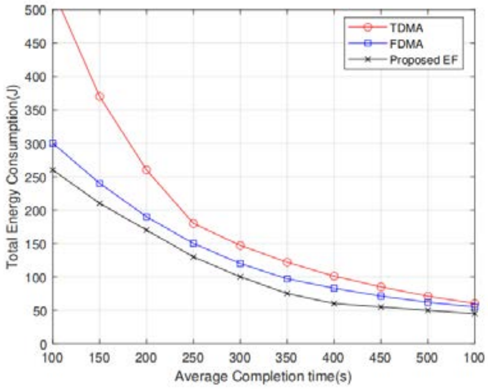

Figure 7. Energy consumption comparison with various completion time

The relationship between completion time and energy consumption is a crucial aspect of the edge system, as shown in Figure 7. It can be observed that the performance of FDMA is better in terms of low completion time since it schedules all data to the BS in a global iteration while only a fraction of users send data to the BS at one time. On the other hand, the performance of TDMA is similar to other methods when the completion time is larger. The proposed EF method can significantly reduce energy consumption, up to 51% and 27% compared to TDMA and FDMA respectively.

7. Conclusion

We have studied the problem of energy-efficient computation and resource allocation for FL over wireless networks. Based on the convergence rate, we developed time and energy consumption models for FL and formulated a joint learning and communication problem with the objective of minimizing the total computation and transmission energy of the network. We have derived closed- form solutions for computation and transmission resources at each iteration and proposed an iterative algorithm with low complexity to solve this problem. The proposed scheme is efficient in terms of energy consumption and outperforms conventional schemes TDMA and FDMA with 51% and 27% reduction, especially when the maximum average transmit power is low.

Acknowledgement

The authors wish to express gratitude towards SEGi University (SEGiIRF/2020- 8/FoEBEIT-37/103) for supporting the research.

References

[1] P. K. R. Maddikunta, Q.-V. Pham, D. C. Nguyen, T. Huynh-The, O. Aouedi, G. Yenduri, S. Bhattacharya, T. R. Gadekallu, Incentive tech- niques for the internet of things: a survey, Journal of Network and Com- puter Applications (2022) 103464.

[2] Y. Sun, M. Peng, Y. Zhou, Y. Huang, S. Mao, Application of machine learning in wireless networks: Key techniques and open issues, IEEE Com- munications Surveys & Tutorials 21 (4) (2019) 3072–3108.

[3] X. Wang, Y. Han, V. C. Leung, D. Niyato, X. Yan, X. Chen, Convergence of Edge Computing and Deep Learning: A Comprehensive Survey, IEEE Commun. Surv. Tutor. 22 (2) (2020) 869–904.

[4] T. Alam, R. Gupta, Federated learning and its role in the privacy preser- vation of iot devices, Future Internet 14 (9) (2022) 246.

[5] Y. Wang, Y. Xu, Q. Shi, T.-H. Chang, Robust federated learning in wireless channels with transmission outage and quantization errors, in: 2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), IEEE, 2021, pp. 586–590.

[6] Z. Xiong, R. Yu, D. Niyato, Blockchain-based federated learning for industrial metaverses: Incentive scheme with optimal aoi, in: 2022 IEEE International Conference on Blockchain (Blockchain), IEEE, 2022, pp. 71–78.

[7] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, K. Chan, When edge meets learning: Adaptive control for resource-constrained dis-tributed machine learning, in: IEEE INFOCOM 2018-IEEE Conference on Computer Communications, IEEE, 2018, pp. 63–71.

[8] M. M. Amiri, D. G¨und¨uz, Machine learning at the wireless edge: Dis- tributed stochastic gradient descent over-the-air, IEEE Transactions on Signal Processing 68 (2020) 2155–2169.

[9] Q. Duan, S. Hu, R. Deng, Z. Lu, Combined federated and split learning in edge computing for ubiquitous intelligence in internet of things: State-of- the-art and future directions, Sensors 22 (16) (2022) 5983.

[10] P. Manoharan, R. Walia, C. Iwendi, T. A. Ahanger, S. Suganthi, M. Kam- ruzzaman, S. Bourouis, W. Alhakami, M. Hamdi, Svm-based generative ad- verserial networks for federated learning and edge computing attack model and outpoising, Expert Systems (2022) e13072.

[11] A. M. Albaseer, M. Abdallah, A. Al-Fuqaha, A. Erbad, Fine-grained data selection for improved energy efficiency of federated edge learning, IEEE Transactions on Network Science and Engineering.

[12] B. Luo, X. Li, S. Wang, J. Huang, L. Tassiulas, Cost-effective federated learning design, in: IEEE INFOCOM 2021-IEEE Conference on Computer Communications, IEEE, 2021, pp. 1– 10.

[13] J. Xu, H. Wang, Client selection and bandwidth allocation in wireless fed- erated learning networks: A long-term perspective, IEEE Transactions on Wireless Communications 20 (2) (2020) 1188– 1200.

[14] Y. Liu, Y. Zhu, J. James, Resource-constrained federated learning with het- erogeneous data: Formulation and analysis, IEEE Transactions on Network Science and Engineering.

[15] M. Kim, W. Saad, M. Mozaffari, M. Debbah, On the tradeoff between energy, precision, and accuracy in federated quantized neural networks, arXiv preprint arXiv:2111.07911.

[16] H. G. Abreha, M. Hayajneh, M. A. Serhani, Federated learning in edge computing: a systematic survey, Sensors 22 (2) (2022) 450.

[17] M. Chen, O. Semiari, W. Saad, X. Liu, C. Yin, Federated echo state learn- ing for minimizing breaks in presence in wireless virtual reality networks, IEEE Transactions on Wireless Communications 19 (1) (2019) 177– 191.

[18] Q. Wang, Y. Xiao, H. Zhu, Z. Sun, Y. Li, X. Ge, Towards energy-efficient federated edge intelligence for iot networks, in: 2021 IEEE 41st Interna- tional Conference on Distributed Computing Systems Workshops (ICDCSW), IEEE, 2021, pp. 55–62.

[19] Y. Li, F. Li, L. Chen, L. Zhu, P. Zhou, Y. Wang, Power of redundancy: Surplus client scheduling for federated learning against user uncertainties, IEEE Transactions on Mobile Computing.

[20] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, A. T. Suresh, Scaffold: Stochastic controlled averaging for federated learning, in: Inter- national Conference on Machine Learning, PMLR, 2020, pp. 5132–5143.

[21] S. L. Paulswamy, A. A. Roobert, K. Hariharan, A novel coverage improved deployment strategy for wireless sensor network, Wireless Personal Com- munications 124 (1) (2022) 867–891.

[22] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, K. Huang, Toward an intelligent edge: Wireless communication meets machine learning, IEEE communica- tions magazine 58 (1) (2020) 19–25.

[23] C. Thapa, P. C. M. Arachchige, S. Camtepe, L. Sun, Splitfed: When feder- ated learning meets split learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 8485–8493.

[24] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, K. Chan, Adaptive federated learning in resource constrained edge computing sys- tems, IEEE Journal on Selected Areas in Communications 37 (6) (2019) 1205–1221.

[25] T. T. Vu, D. T. Ngo, H. Q. Ngo, M. N. Dao, N. H. Tran, R. H. Middleton, Joint resource allocation to minimize execution time of federated learning in cell-free massive mimo, IEEE Internet of Things Journal 9 (21) (2022) 21736–21750.

[26] M. Salehi, E. Hossain, Federated learning in unreliable and resource- constrained cellular wireless networks, IEEE Transactions on Communi- cations 69 (8) (2021) 5136–5151.

[27] Y. Sun, S. Zhou, D. G¨und¨uz, Energy-aware analog aggregation for feder- ated learning with redundant data, in: ICC 2020-2020 IEEE International Conference on Communications (ICC), IEEE, 2020, pp. 1– 7.

[28] K. Yang, T. Jiang, Y. Shi, Z. Ding, Federated learning via over-the-air com- putation, IEEE Transactions on Wireless Communications 19 (3) (2020) 2022–2035.

[29] W. Shi, S. Zhou, Z. Niu, M. Jiang, L. Geng, Joint device scheduling and re- source allocation for latency constrained wireless federated learning, IEEE Transactions on Wireless Communications 20 (1) (2020) 453–467.

[30] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, S. Cui, A joint learning and communications framework for federated learning over wireless networks, IEEE Transactions on Wireless Communications 20 (1) (2020) 269–283.

[31] R. Hamdi, M. Chen, A. B. Said, M. Qaraqe, H. V. Poor, Federated learning over energy harvesting wireless

networks, IEEE Internet of Things Journal 9 (1) (2021) 92–103.

[32] A. A. Abdellatif, N. Mhaisen, A. Mohamed, A. Erbad, M. Guizani, Z. Dawy, W. Nasreddine, Communication-efficient hierarchical federated learning for iot heterogeneous systems with imbalanced data, Future Generation Computer Systems 128 (2022) 406–419.

[33] H.-C. Wu, The karush–kuhn–tucker optimality conditions in an optimiza- tion problem with interval-valued objective function, European Journal of Operational Research 176 (1) (2007) 46–59.

[34] E. W. Weisstein, Lambert w-function, https://mathworld. wolfram. com/.

[35] Z. Yang, M. Chen, W. Saad, C. S. Hong, M. Shikh-Bahaei, Energy efficient federated learning over wireless communication networks, IEEE Transac- tions on Wireless Communications 20 (3) (2020) 1935– 1949.