Energy-Constrained D2D Assisted Federated Learning in Edge Computing

Yuchen Li
The Australian National University
Canberra, Australia

Weifa Liang
City University of Hong Kong
Hong Kong, P. R. China

Jing Li
The Hong Kong Polytech University
Hong Kong, P. R. China

Xiuzhen Cheng
Shandong University
Qingdao, China

Dongxiao Yu
Shandong University
Qingdao, China

Albert Y. Zomaya
The University of Sydney
Sydney, Australia

Song Guo
The Hong Kong Polytech University
Hong Kong, P. R. China

ABSTRACT

The surging of deep learning brings new vigor and vitality to shape the prospect of intelligent Internet of Things (IoT), and edge intelligence arises to provision real-time deep neural network (DNN) inference services for mobile users. To perform efficient and effective DNN model training in edge environments while preserving training data security and privacy of IoT devices, federated learning has been envisioned as an ideal learning paradigm for this purpose. In this paper we study energy-aware DNN model training in an edge environment. We first formulate a novel energy-aware, device-to-device (D2D) assisted federated learning problem with the aim to minimize the global loss of a training DNN model, subject to bandwidth capacity on an edge server and the energy capacity on each IoT device. We then devise an efficient heuristic algorithm for the problem. The crux of the proposed algorithm is to explore the energy usage of neighboring devices of each device for its local model uploading, by reducing the problem to a series of maximum weight matching problems in corresponding auxiliary graphs. We finally evaluate the performance of the proposed algorithm through experimental simulations. Experimental results show that the proposed algorithm is promising.

CCS CONCEPTS

• **Networks** → *Mobile networks.*

KEYWORDS

energy-aware federated learning, D2D-assisted edge learning.

ACM Reference Format:

Yuchen Li, Weifa Liang, Jing Li, Xiuzhen Cheng, Dongxiao Yu, Albert Y. Zomaya, and Song Guo. 2022. Energy-Constrained D2D Assisted Federated Learning in Edge Computing. In *Proceedings of the International Conference on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWiM '22), October 24–28, 2022, Montreal, QC, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3551659.3559062

MSWiM '22, October 24–28, 2022, Montreal, QC, Canada

ACM ISBN 978-1-4503-9479-6/22/10.
https://doi.org/10.1145/3551659.3559062

1 INTRODUCTION

Deep learning is a technique that utilizes DNNs to learn patterns of a group of data. Traditional training of DNNs needs users to upload their data to a centralized server, which has a high risk of privacy violation. Federated learning provides a solution to utilizing the data on training while protecting privacy by allowing each device uploads its local model to an edge server for aggregation [13]. However, devices with longer distances from the edge server need to use larger transmission power to upload their local models, thereby consuming much more energy than those with shorter distances. Also, training a large-scale DNN model needs a large volume of data transmissions between devices and the edge server, this introduces communication overhead [5]. To alleviate the communication overhead on uploading local models, in this paper we introduce an energy-aware, device-to-device (D2D) assisted uploading concept to address this issue. Devices are paired according to their distances and energy availability. For each pair of devices, the one with less energy budget sends its trained local model to the one with more energy budget, and the received device aggregates the received model with its local model and uploads the aggregated model.

Performing energy-aware federated learning in edge environments poses several challenges: First, by allocating more energy to computation, a device can train its local model on a larger volume portion of its collected data set, but this leaves the device less energy on its local model uploading or vice versa. How to strive for a non-trivial trade-off of energy allocations between its local model training and local model transmission/uploading is challenging. Second, due to the heterogeneity of computing power and volume of collected data, different devices have different energy budgets at each training round, the choice of pairing devices heavily impacts on not only the transmission energy consumption of devices but also the amount of data used for local model training, thereby affecting the accuracy and convergence of model learning, how to identify devices and form device pairs such that the training convergence can be guaranteed is challenging. Finally, the wireless bandwidth capacity of the edge server usually is bounded, which can support up to K devices rather than all devices to upload their local models to the server simultaneously at each round. As the contributions of different devices to the global model are different, the contributions of some devices are more important than those of others, and it is critical to identify the K important devices from all devices that are able to upload their (aggregated) local models to the edge server, in order to ensure the model training efficiency and accuracy. In the rest of this paper, we will tackle the challenges.

The novelty of this paper lies in that we consider energy-aware D2D assisted federated learning in edge computing, and utilize neighbor devices of each device for its local model uploading.

The main contributions of this paper are given as follows. We formulate a novel energy-aware, D2D-assisted federated learning problem in an edge computing network with the aim to minimize the expected loss of a DNN model training, subject to the wireless bandwidth capacity on an edge server, and energy capacity and transmission range constraints on devices. We propose an efficient heuristic algorithm for the problem, by reducing the problem to a series of maximum weight matching problems in corresponding auxiliary graphs. In the end, we evaluate the performance of the proposed algorithm through experimental studies. Experimental results demonstrate that the proposed algorithm is promising.

The paper is organized as follows. Section 2 reviews related works. Section 3 introduces the system model and defines the problem formally. Section 4 proposes an efficient algorithm for the general problem. Section 5 evaluates the proposed algorithm empirically, and Section 6 concludes the paper.

2 RELATED WORK

Federated learning in edge computing environments has been extensively investigated recently, and most studies focused on the energy consumption of devices and edge servers, others dealt with the non-trivial trade-off between the communication cost, the accuracy of solutions, and the convergence speeds of different learning algorithms. For example, Wang *et. al* [11] focused on the trade-off between the communication cost and the convergence performance. Dinh *et al* [3] concentrated on the trade-off between the energy consumption and convergence, and proposed a new federated learning algorithm with the assumption of strongly convex and smooth loss functions. The crux of their algorithm is a new local surrogate function for each device to train its local model approximately up to a local accuracy level. Sun *et. al* [9] considered long-term energy-aware dynamic edge server scheduling problem, by developing an online scheduling algorithm with the aim to maximize the average number of edge server participation in training at each training iteration. Tu *et. al* [10] studied the federated learning in a fog computing environment where devices are allowed to offload their local data to other devices for processing and discard some of their data. They focused on the impact of data transfer between devices and data discarding on model training. They proposed a data transfer scheme to minimize both computing and communication costs while bounding the influence of data transfer and data discarding on training the model.

3 PRELIMINARIES

In this section, we first introduce the system model, notions and notations. We then define the problem precisely.

3.1 System model

We consider a set of devices $V = \{v_1, v_2, \ldots, v_{|V|}\}$ and an edge server s. Denote by $d_{i,j}$ the distance between device v_i and v_j with $0 \le i, j \le |V|$. Each device $v_i \in V$ has a data set $\mathcal{D}_i$. Denote by $\mathcal{D} = \cup_{i=1}^{|V|}\mathcal{D}_i$ the data set of all devices. Each device v_i has a set $\mathcal{P}$ of finite transmission power levels. Device v_i uses one of the transmission power level $p_i(t) \in \mathcal{P}$ for communication at round t where $1 \le t \le T$. We further assume that the federated learning process takes T training rounds. At each round t, each device $v_i \in V$ can perform its local training for τ epochs, assuming that the energy budget $\mathcal{E}_i(t)$ of v_i at round t is given [1]. Due to limited wireless bandwidth on edge server s, we assume that at most K devices can upload their local models to the server simultaneously at each round t, where $1 \le K \le |V|$.

3.2 Federated learning in edge computing

Let (x, y) be *one data point* in the data set $\mathcal{D}$, where x is the input feature and y is the label of the data point. We aim to train a DNN to minimize the error between the output of the neural network and label y under a given input x. Specifically, the error is defined by a loss function $l(w, x, y)$ depending on the application scenario of the neural network [13], and w is the *model parameter* of the DNN model, which is a vector. The loss function on data set $\mathcal{D}$ is

$$L(w \mid \mathcal{D}) = \frac{\sum_{(x,y)\in\mathcal{D}} l(w, x, y)}{|\mathcal{D}|}. \tag{1}$$

The training objective is to find an optimal model parameter w^* to minimize the value of $L(w \mid \mathcal{D})$.

Due to limited wireless bandwidth capacity B of edge server s in the defined system model, it is impossible to allow all devices to upload their models to the server at the same time. Instead, a subset of devices $V_{fed}^t \subset V$ is chosen to participate in model training at each round t. Specifically, each device $v_i \in V_{fed}^t$ uniformly samples a subset $\mathcal{S}_i(t)$ of its data set $\mathcal{D}_i$ for local model training under its energy budget $\mathcal{E}_i(t)$ for round t. Within each round t, recall that each device $v_i \in V_{fed}^t$ applies τ gradient descent steps to train its local model $w_i(t)$. Denote by $w_i^k(t)$ the local model parameter of device v_i after the kth local training epoch with $k = 1, 2, \ldots, \tau$. Then, $w_i^0(t) = w(t-1)$ is the global model parameter distributed by server s after finishing local model training at round $t-1$. During each epoch k of round t with $1 \le k \le \tau$, device v_i updates its local model as follows.

$$w_i^k(t) = w_i^{k-1}(t) - \eta \cdot \nabla L_i(w_i^{k-1}(t) \mid \mathcal{S}_i(t)), \tag{2}$$

while

$$\nabla L_i(w_i^{k-1}(t) \mid \mathcal{S}_i(t)) = \frac{\sum_{(x,y)\in\mathcal{S}_i(t)} \nabla l(w_i^{k-1}(t), x, y)}{|\mathcal{S}_i(t)|}, \tag{3}$$

where η is the learning rate, and $\nabla l(w_i^{k-1}(t), x, y)$ is the gradient of the loss function $l(w, x, y)$ with respect to $w_i^{k-1}(t)$ on each data point $(x, y) \in \mathcal{S}_i(t)$ at epoch $(k-1)$ within round t. Having τ-epoch local training at round t, some device $v_i \in V_{fed}^t$ uploads its trained (or aggregated) local model $w_i(t)$ $(= w_i^{\tau}(t))$ to server s. The uploaded local models from devices in V_{fed}^t are then aggregated at server s as follows [11].

$$w(t) = \frac{\sum_{v_i \in V^t_{fed}} |\mathcal{S}_i(t)| \cdot w_i^\tau(t)}{\sum_{v_i \in V^t_{fed}} |\mathcal{S}_i(t)|}. \quad (4)$$

Server s then distributes the aggregated global model $w(t)$ back to each device in V in the beginning of the next round $t + 1$.

3.3 Energy-budgeted D2D assisted uploading

We here allow devices with sufficient energy to serve as relay nodes to help those devices with less energy to upload their local models to the server, thereby reducing the energy consumption of those less-energy devices. Meanwhile, the relay devices can aggregate the local models of relayed neighbors locally prior to uploading the aggregated local models to the server. Specifically, each device $v_i \in V^t_{fed}$ has a destination $\phi_{v_i}(t)$ at round t, which is either another device or server s. To avoid a long training delay, each device can only serve as either relay or relayed node at each round exclusively. The transmission range $\theta_i(p_i(t))$ of device $v_i \in V$ at round t usually is determined by its transmission power level $p_i(t) \in \mathcal{P}$, a device v_j or server s can be the destination of device v_i only if it is within the transmission range of v_i, i.e., $d_{i,\phi_{v_i}(t)} \leq \theta_i(p_i(t))$.

Denote by $C_{v_i}(t)$ the set of nodes using v_i as their relays at training round t, we have $|C_{v_i}(t)| \leq 1$. Similarly, $C_s(t)$ is the set of nodes that can send their models to server s directly. Since energy is the main constraint on devices, we assume that devices communicate with server s by adopting the Orthogonal Frequency-Division Multiplexing Access (OFDMA) mode. Due to limited wireless bandwidth constraint on the edge server, we assume that at most K devices can send their local models to s directly with $1 \leq K \leq |V|$ at each round, that is, $|C_s(t)| \leq K$. Having local training on its data set $\mathcal{S}_i$ at round t, device $v_i \in V^t_{fed}$ then sends its local model $w_i^\tau(t)$ to its destination $\phi_{v_i}(t)$ or server s. If its destination $\phi_{v_i}(t)$ is not to server s, device v_j (= $\phi_{v_i}(t)$) will aggregate its local model with its received local model from v_i, and transmits the aggregated result to server s to reduce transmission energy consumption. For each device v_i whose destination is server s, denote by $w_i^g(t)$ the result of local aggregation of v_i. The aggregation at device v_i is

$$w_i^g(t) = |\mathcal{S}_i(t)| \cdot w_i^\tau(t) + \sum_{v_j \in C_{v_i}(t)} |\mathcal{S}_j(t)| \cdot w_j^\tau(t) \quad (5)$$

The global model at server s after round t is updated as follows.

$$w(t) = \frac{\sum_{v_i \in C_s(t)} w_i^g(t)}{\sum_{v_i \in V^t_{fed}} |\mathcal{S}_i(t)|}, \quad (6)$$

It can be seen that $w(t)$ in Eq. (6) is equivalent to it in Eq.(4).

3.4 Energy consumption of devices

Denote by C the size of the DNN model w. The channel gain $h_{i,\phi_{v_i}(t)}$ [8] between device v_i and device $\phi_{v_i}(t)$ is

$$h_{i,\phi_{v_i}(t)} = \frac{\alpha}{d^2_{i,\phi_{v_i}(t)}}, \quad (7)$$

where α is the channel gain at the reference distance of 1 meter. By uploading its local model $w_i^\tau(t)$ to device $\phi_{v_i}(t)$, the amount of transmission energy consumed by device v_i is

$$Tran_{v_i}(t) = \frac{C \cdot p_i(t)}{B \cdot \log_2(1 + \frac{p_i(t) \cdot h_{i,\phi_{v_i}(t)}}{\sigma^2})}, \quad (8)$$

where σ is the white Gaussian noise power, and B is the wireless bandwidth capacity.

Let ψ_i be the energy consumption of training on one data point per epoch at device v_i. The total computing energy consumption of v_i on local model training at round t is

$$Comp_{v_i}(t) = \psi_i \cdot |\mathcal{S}_i(t)| \cdot \tau. \quad (9)$$

The total energy consumption of device v_i at round t should be no greater than its energy budget $\mathcal{E}_i(t)$.

$$Tran_{v_i}(t) + Comp_{v_i}(t) \leq \mathcal{E}_i(t). \quad (10)$$

3.5 Problem definition

Given a set of devices V and an edge server s, they collaboratively perform federated learning to train a DNN model with a parameter vector (global model) w, each device $v_i \in V$ has a set of transmission power levels $\mathcal{P}$. Each device v_i can sample a subset $\mathcal{S}_i(t)$ of its data set $\mathcal{D}_i$ at each round t, and may upload its trained local model to server s directly or via another device, assuming that there are T rounds for the DNN model training, the *energy-aware D2D-assisted federated learning problem* in an edge computing network is to determine the devices to participate in training at each round t, the number of sampled data points, the transmission power level of each chosen device, and the destination device of each chosen device to upload its local model at each round t with $1 \leq t \leq T$, such that the expected loss $\mathbb{E}[L(w(T) \mid \mathcal{D})]$ of the global model over the data set $\mathcal{D}$ (= $\cup_{i=1}^{|V|} \mathcal{D}_i$) is minimized, subject to the maximum number K of devices that can communicate with the server simultaneously, $1 \leq K \leq |V|$, the wireless bandwidth capacity B, the energy capacity and the maximum transmission range of each device. The objective of the problem is to

$$\text{minimize} \qquad \mathbb{E}[L(w(T) \mid \mathcal{D})], \quad (11)$$

where $\mathbb{E}[L(w(T) \mid \mathcal{D})]$ is the expectation of $L(w \mid \mathcal{D})$.

4 ALGORITHM FOR THE ENERGY-AWARE D2D-ASSISTED FEDERATED LEARNING PROBLEM

4.1 Device choices at each round

Since data generated by different devices may not follow identical and individual distributions, the optimal model w_i^* for the local loss may not be optimal for the global model. As device v_i only samples data points in its data set, v_i can only train the local model w_i towards w_i^*. The global model w is strongly biased towards the chosen devices that participate in training. To mitigate biased model training caused by some but not all devices in V participating in training on the global model, we propose an efficient strategy for device choices. First, to ensure the quality of local training, it requires each chosen device to train on at least $\delta \cdot |\mathcal{D}_i|$ data points with $\delta \cdot |\mathcal{D}_i| \leq |\mathcal{S}_i(t)|$, where δ is a given threshold with $0 < \delta \leq 1$. Cho *et al.* showed that devices with higher local loss make the learning convergence faster [2], this implies that devices with higher local losses will have higher priorities to participate in training at that round. It thus is the key to choosing a subset V^t_{fed} of devices with high local losses to participate in training at each round t. To this end, each device v_i samples a small portion of its data set and makes use of the sampled data points to estimate the local loss $\hat{L}(w_i(t-1) \mid \mathcal{D}_i)$ at the beginning of the next round t. A device v_i that does not participate in training at round t has either a lower local loss than those of devices in V^t_{fed}, or a higher local loss than some of the devices in V^t_{fed} but may cause that the devices with higher local loss than it have inadequate energy to train.

4.2 Algorithm overview

The basic idea behind the proposed algorithm is to reduce the problem to a series of maximum weight matching problems in different auxiliary graphs. Specifically, we start by sorting devices in V in non-decreasing order of their estimated local loss. For the sake of convenience, let $v_1, v_2, \ldots, v_{|V|}$ be the sorted devices, where v_1 and $v_{|V|}$ have the lowest and highest local losses, respectively. We then construct an auxiliary graph $\mathcal{G}(t) = (U, E; w(\cdot,\cdot))$ that is similar to the auxiliary graph we did in the previous section at round t with $1 \le t \le T$, where the vertex set U consists of vertices u_i and u'_i for each device $v_i \in V$ and $2|V|-2K$ dummy vertices representing server s. The construction of the edge set E is as follows. For each device $v_i \in V$, to fulfill the number of sampled data points, v_i must spend at least the amount $\psi_i \cdot \delta \cdot |\mathcal{D}_i| \cdot \tau$ of energy on training. Denote by $\mathcal{E}'_i(t) = \mathcal{E}_i(t) - \psi_i \cdot \delta \cdot |\mathcal{D}_i| \cdot \tau$ the transmission energy budget of v_i at round t. From Eq. (8), to save energy of devices on transmissions, one device should select a minimum transmission power level in $\mathcal{P}$ such that its destination node is within the transmission range. Denote by $p^{min}_i(v_j)$ and $p^{min}_i(s)$ the minimum transmission power levels of v_i that v_j or s are within the transmission range of v_i when it adopts that power level accordingly. Denote by $tran(v_i, v_j)$ and $tran(v_i, s)$ the amounts of energy consumed by transmitting its local model to device v_j or service s respectively by adopting the minimum transmission power levels. If the required transmission energy constraint can be met (i.e., $tran(v_i, s) \le \mathcal{E}_i(t) - \mathcal{E}'_i(t)$), add an edge $e(u_i, u'_i)$ to E in $\mathcal{G}(t)$ with weight of $w(u_i, u'_i)$ ($= 2^i$), assuming that device v_i has the ith lowest estimated local loss, where edge $e(u_i, u'_i)$ with weight 2^i indicates the high priority of v_i as a potential uploading device at round t and upload its local model to s directly if it is chosen. For each pair of devices v_i and v_j with $i > j$, if $tran(v_i, v_j) \le \mathcal{E}_i(t) - \mathcal{E}'_i(t)$ and $tran(v_j, v_s) \le \mathcal{E}_i(t) - \mathcal{E}'_j(t)$, add an edge $e(u_i, u_j)$ to E with weight of $w(u_i, u_j)$ ($= 2^i + 2^j$). This means that v_i sends its trained local model to v_j, v_j then aggregates the received local model of v_i with its local model and uploads the aggregated model to server s; Otherwise, if $tran(v_j, v_i) \le \mathcal{E}_i(t) - \mathcal{E}'_j(t)$ and $tran(v_i, v_s) \le \mathcal{E}_i(t) - \mathcal{E}'_i(t)$), we add an edge $e(u_i, u_j)$ to E with weight of $w(u_i, u_j)$ ($= 2^i + 2^j$), the operations are similar, instead, v_i will upload the aggregated model to server s. For any pair of a dummy vertex u^ν_j and a device vertex u_i, add an edge $e(u^\nu_j, u_i)$ with weight of ∞ to E. Let $\mathcal{M}(t)$ be the maximum weight matching in $\mathcal{G}(t)$, and the K chosen devices driven from $\mathcal{M}(t)$ form a solution of the problem.

4.3 Algorithm

The proposed algorithm proceeds as follows. In the beginning of each round t, each device v_i samples a small subset of its data set $\mathcal{D}_i$ to estimate the local loss $\hat{L}(w(t-1) \mid \mathcal{D}_i)$. A weighted auxiliary graph $\mathcal{G}(t)$ then is constructed, and the estimated local loss and device ranking are used to assign weights to the edges in the graph. A maximum weight matching $\mathcal{M}(t)$ in $\mathcal{G}(t)$ then is found, and a feasible solution to the problem finally is derived from the maximum weight matching $\mathcal{M}(t)$. That is, if edge $e(u_i, u'_i) \in \mathcal{M}(t)$, device v_i uploads its local model to server s directly; otherwise, if edge $e(u_i, u_j) \in \mathcal{M}(t)$ with $i > j$ and $tran(v_j, s) \le \mathcal{E}_i(t) - \mathcal{E}'_i(t)$, device v_j is the destination of v_i, aggregates its local model with the local model of v_i and uploads the aggregated local model to s; Similarly, if $e(u_i, u_j) \in \mathcal{M}(t)$ with $i > j$ but $tran(v_j, s) > \mathcal{E}_i(t) - \mathcal{E}'_i(t)$, v_j sends its local model to v_i for aggregation and v_i uploads the aggregated model to server s. The rest devices that are not incident to any matching edge in $\mathcal{M}(t)$ will not participate in training at round t. As a result, the number of sampled data points $|\mathcal{S}_i(t)|$ of device v_i at round t is $\lfloor \frac{\mathcal{E}_i(t) - Tran_{v_i}(t)}{\psi_i} \rfloor$. The detailed algorithm for the energy-aware D2D assisted federated learning problem is given in `Algorithm` 1.

Algorithm 1 Algorithm for the energy-aware D2D-assisted federated learning problem

Input: A set of devices V, a server s, the data set $\mathcal{D}_i$ of each device $v_i \in V$, an edge budget $\mathcal{E}_i(t)$ of v_i at round t, a set of transmission power level $\mathcal{P}$, a given threshold δ and a DNN model w to be trained at t round with $1 \le t \le T$.

Output: The set V^t_{fed} of devices that participate in training, the offloading destination $\phi_{v_i}(t)$, the set of sampled data $\mathcal{S}_i(t)$, and the transmission power level $p_i(t)$ of each chosen device v_i at each round t.

for $t \leftarrow 1$ to T **do**
Each device $v_i \in V$ estimate the local loss $\hat{L}(w(t-1) \mid \mathcal{D}_i)$; Sort vertices in V in non-decreasing order of the estimated loss;
Initialize $G(t) = (U, E)$ where $U \leftarrow \emptyset$; $E \leftarrow \emptyset$;
for $i \leftarrow 1$ to $|V|$ **do**
$U \leftarrow U \cup \{u_i, u'_i\}$ and calculate $p^{min}_i(s)$, $tran(v_i, s)$;
if $tran(v_i, s) + \psi_i \cdot \delta \cdot |\mathcal{D}_i| \cdot \tau \le \mathcal{E}_i(t)$ **then**
$E \leftarrow E \cup \{e(u_i, u'_i)\}$; $w(u_i, u'_i) \leftarrow 2^i$;
for $i \leftarrow 1$ to $|V|$ **do**
for $j \leftarrow i+1$ to $|V|$ **do**
Calculate $p^{min}_j(s)$, $tran(v_i, v_j)$, and $tran(v_j, s)$;
if $(tran(v_i, v_j) + \psi_i \cdot \delta \cdot |\mathcal{D}_i| \cdot \tau < \mathcal{E}_i(t)$ and $tran(v_j, s) + \psi_j \cdot \delta \cdot |\mathcal{D}_j| \cdot \tau < \mathcal{E}_j(t))$ or $(tran(v_j, v_i) + \psi_j \cdot \delta \cdot |\mathcal{D}_j| \cdot \tau < \mathcal{E}_j(t)$ and $tran(v_i, s) + \psi_i \cdot \delta \cdot |\mathcal{D}_i| \cdot \tau < \mathcal{E}_i(t))$ **then**
$E \leftarrow E \cup \{e(u_i, u_j)\}$; $w(u_i, u_j) \leftarrow 2^i + 2^j$;
$U \leftarrow U \cup U_{dum}$, where U_{dum} consists of $2|V| - 2K$ vertices;
for each dummy vertex $u^\nu_j \in U_{dum}$ **do**
for each vertex $u_i \in U \backslash U_{dum}$ **do**
$E \leftarrow E \cup \{e(u_i, u^\nu_j)\}$; $w(u_i, u^\nu_j) \leftarrow \infty$;
Find a maximum weight matching $\mathcal{M}(t)$ in $G(t)$, by applying the algorithm in [4];
for $i \leftarrow 1$ to $|V|$ **do**
if $e(u_i, u'_i) \in \mathcal{M}(t)$ **then**
$\phi_{v_i}(t) \leftarrow s$; $V^t_{fed} \leftarrow V^t_{fed} \cup \{v_i\}$; $|\mathcal{S}_i(t)| \leftarrow \lfloor \frac{\mathcal{E}_i(t) - tran(v_i, s)}{\psi_i} \rfloor$;
if $\exists j > i, s.t.\ e(u_i, u_j) \in \mathcal{M}(t)$ **then**
$V^t_{fed} \leftarrow V^t_{fed} \cup \{v_i, v_j\}$;
if $tran(v_i, s) + \psi_i \cdot \delta \cdot |\mathcal{D}_i| \cdot \tau < \mathcal{E}_i(t)$ **then**
$\phi_{v_i}(t) \leftarrow s$; $\phi_{v_j}(t) \leftarrow v_i$;
$|\mathcal{S}_j(t)| \leftarrow \lfloor \frac{\mathcal{E}_j(t) - tran(v_j, v_i)}{\psi_j} \rfloor$;$|\mathcal{S}_i(t)| \leftarrow \lfloor \frac{\mathcal{E}_i(t) - tran(v_i, s)}{\psi_i} \rfloor$;
else
$\phi_{v_j}(t) \leftarrow s$; $\phi_{v_i}(t) \leftarrow v_j$;
$|\mathcal{S}_j(t)| \leftarrow \lfloor \frac{\mathcal{E}_j(t) - tran(v_j, s)}{\psi_j} \rfloor$;$|\mathcal{S}_i(t)| \leftarrow \lfloor \frac{\mathcal{E}_i(t) - tran(v_i, v_j)}{\psi_i} \rfloor$;
return V^t_{fed}, $\phi_{v_i}(t)$, $\mathcal{S}_i(t)$ for each device v_i at round t.

Theorem 1. Given a set of devices V and a server s executing federated learning, there is an efficient algorithm, `Algorithm 1`, for the energy-aware D2D-assisted federated learning problem, which takes $O(T \cdot |V|^4)$.

Proof The proof is omitted due to space limitation.

5 PERFORMANCE EVALUATION

We consider an edge network that consists of 100 devices randomly deployed in a circular area with a 500 meter radius and an edge server co-located with an access point at the center of the area [7], where the server can communicate with up to 20% of the devices directly. The devices and the server collaboratively conduct a federated learning model training with 50 rounds, and each training round consists of $\tau = 1$ epoch. The set of transmission power levels of each device is $\{0.2, 0.3, 0.4, 0.5\}$ $Watt$[7], and the corresponding transmission distances of these transmission power levels are set as $\{282, 346, 400, 447\}$ $meters$, respectively [12]. The energy budget on each device at each round is randomly drawn in [0.02, 0.04] $Joules$[1]. The data set we adopt in this experiment is the MNIST [11]. The DNN model we here use is Le-net5, which has a size C of 1,960 $Kbits$ [6]. The value in each figure is the mean of the results out of 50 network instances of the same size. The running time of an algorithm is obtained based on a machine with a 3.6 GHz Intel i7 single-core CPU and 16 GB RAM. Unless otherwise specified, these parameters will be adopted in the default setting.

To evaluate the performance of the proposed algorithms, we first compare our algorithm against traditional federated learning where we randomly select K devices to participate in the training at each round [2], which is referred to as `Baseline`. We then devise a heuristic algorithm `Heur`, which chooses the device with the highest local loss from those non-chosen devices iteratively until no more devices can be added at each round.

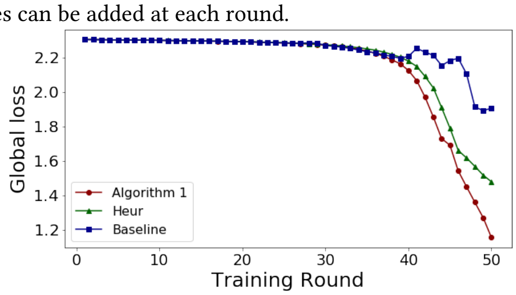
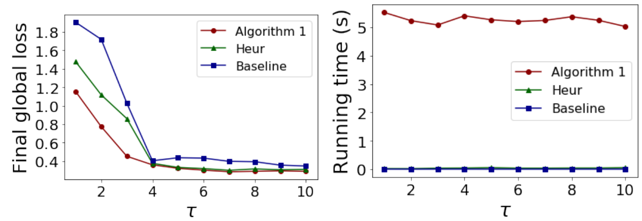


Figure 1: Convergence of different algorithms for the energy-aware D2D-assisted federated learning problem with $T = 50$.

We first investigate the performance of different algorithms for the energy-aware D2D-assisted federated learning problem. From Fig. 1, `Algorithm 1` has the minimum final global loss among the three algorithms„ and the reason is that `Algorithm 1` selects the devices with the highest estimated local losses.

We then study the impact of the number τ of training epochs per round. To better illustrate the impact of τ, we scale up the energy budget on each device v_i to $\tau \cdot \mathcal{E}_i(t)$ at each round accordingly. Fig. 2 (a) plots the final global losses of different algorithms. It can be seen that `Algorithm 1` has the best performance, as it has a lower final global loss when τ is no greater than 4. This means that the federated learning can train the model better with fewer training epochs when applying `Algorithm 1`. Also, when $\tau = 10$, the final global loss of `Heur` is 10.7% larger than that of `Algorithm 1`.



(a) Final global loss **(b) Average running time**

Figure 2: Performance of different algorithms by varying the number of training epochs τ per round.

6 CONCLUSION

In this paper, we studied the energy-aware D2D-assisted federated learning problem in an edge computing environment, by utilizing neighbor devices of a certain number of devices to help their local model uploading. We developed an efficient heuristic algorithm for the problem. We finally evaluated the performance of the proposed algorithm by experimental simulations. Experimental results show that the proposed algorithm is promising.

REFERENCES

[1] Mingzhe Chen, Zhaohui Yang, Walid Saad, Changchuan Yin, H Vincent Poor, and Shuguang Cui. 2020. A joint learning and communications framework for federated learning over wireless networks. *IEEE Transactions on Wireless Communications* 20, 1 (2020), 269–283.

[2] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. 2020. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. *arXiv preprint arXiv:2010.01243* (2020).

[3] Canh T Dinh, Nguyen H Tran, Minh NH Nguyen, Choong Seon Hong, Wei Bao, Albert Y Zomaya, and Vincent Gramoli. 2020. Federated learning over wireless networks: Convergence analysis and resource allocation. *IEEE/ACM Transactions on Networking* 29, 1 (2020), 398–409.

[4] Jack Edmonds. 1965. Maximum matching and a polyhedron with 0, 1-vertices. *Journal of research of the National Bureau of Standards B* 69 (1965), 55–56.

[5] Sohei Itahara, Takayuki Nishio, Yusuke Koda, Masahiro Morikura, and Koji Yamamoto. 2020. Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-iid private data. *arXiv preprint arXiv:2008.06180* (2020).

[6] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998).

[7] Jing Li, Weifa Liang, Yuchen Li, Zichuan Xu, Xiaohua Jia, and Song Guo. 2021. Throughput Maximization of Delay-Aware DNN Inference in Edge Computing by Exploring DNN Model Partitioning and Inference Parallelism. *IEEE Transactions on Mobile Computing* (2021).

[8] Zhaohua Lv, Jianjun Hao, and Yijun Guo. 2020. Energy minimization for MEC-enabled cellular-connected UAV: Trajectory optimization and resource scheduling. In *Proc. of INFOCOM'20*. IEEE, 478–483.

[9] Yuxuan Sun, Sheng Zhou, and Deniz Gündüz. 2020. Energy-aware analog aggregation for federated learning with redundant data. In *Proc. of ICC'20*. IEEE.

[10] Yuwei Tu, Yichen Ruan, Satyavrat Wagle, Christopher G Brinton, and Carlee Joe-Wong. 2020. Network-aware optimization of distributed learning for fog computing. In *Proc. of INFOCOM'20*. IEEE, 2509–2518.

[11] Shiqiang Wang, Tiffany Tuor, Theodoros Salonidis, Kin K Leung, Christian Makaya, Ting He, and Kevin Chan. 2019. Adaptive federated learning in resource constrained edge computing systems. *IEEE Journal on Selected Areas in Communications* 37, 6 (2019), 1205–1221.

[12] Zhe Wang, Lingjie Duan, and Rui Zhang. 2019. Adaptive deployment for UAV-aided communication networks. *IEEE transactions on wireless communications* 18, 9 (2019), 4531–4543.

[13] Jie Zhang, Song Guo, Zhihao Qu, Deze Zeng, Yufeng Zhan, Qifeng Liu, and Rajendra A Akerkar. 2021. Adaptive federated learning on Non-IID data with resource constraint. *IEEE Trans. Comput.* (2021).