

# Joint Service Pricing and Cooperative Relay Communication for Federated Learning

Shaohan Feng<sup>1</sup>, Dusit Niyato<sup>1</sup>, Ping Wang<sup>2</sup>, Dong In Kim<sup>3</sup>, and Ying-Chang Liang<sup>4</sup>

<sup>1</sup> School of Computer Engineering, Nanyang Technological University, Singapore

<sup>2</sup> Department of Electrical Engineering and Computer Science, York University, Canada

<sup>3</sup> School of Information and Communication Engineering, Sungkyunkwan University, Korea

<sup>4</sup> CINC, University of Electronic Science and Technology of China, China

**Abstract**—For the sake of protecting data privacy and due to the rapid development of mobile devices, e.g., powerful central processing unit (CPU) and nascent neural processing unit (NPU), collaborative machine learning on mobile devices, e.g., federated learning, has been envisioned as a new AI approach with broad application prospects. However, the learning process of the existing federated learning platforms rely on the direct communication between the model owner, e.g., central cloud or edge server, and the mobile devices for transferring the model update. Such a direct communication may be energy inefficient or even unavailable in mobile environments. In this paper, we consider adopting the relay network to construct a cooperative communication platform for supporting model update transfer and trading. In the system, the mobile devices generate model updates based on their training data. The model updates are then forwarded to the model owner through the cooperative relay network. The model owner enjoys the learning service provided by the mobile devices. In return, the mobile devices charge the model owner certain prices. Due to the coupled interference of wireless transmission among the mobile devices that use the same relay node, the rational mobile devices have to choose their relay nodes as well as deciding on their transmission powers. Thus, we formulate a Stackelberg game model to investigate the interaction among the mobile devices and that between the mobile devices and the model owner. The Stackelberg equilibrium is investigated by capitalizing on the exterior point method. Moreover, we provide a series of insightful analytical and numerical results on the equilibrium of the Stackelberg game.

**Index Terms**—Federated learning, Security privacy, Energy efficient, Network optimization, Stackelberg game.

## I. INTRODUCTION

Due to the explosive growth of smart IoT devices at the edge of the Internet, massive data collection through embedded sensors on mobile devices, e.g., crowdsensing, for machine learning has found a number of applications and gained tremendous popularity rapidly [1]. However, the existing machine learning approaches rely on centralized storage of the training data. Consequently, they usually face a series of data security and privacy issues, e.g., data abuse and information leakage. A recent report from Ponemon Institute suggests an average cost of over \$200 for per record of data breach [2]. Such a high economic loss has hindered the adoption of data sharing among the different entities, and the machine learning that requiring centralized data storage is facing great challenges.

To overcome the limitations of traditional machine learning in the protection of data privacy, a novel paradigm has been

proposed. In [3], the federated learning system was introduced to address the issue of data privacy. Therein, the mobile devices perform computation of model training locally on their training data according to the model released by the model owner. Such design enables mobile device to collaboratively learn a shared prediction model while keeping all the training data on the device [4]. However, the independent and rational mobile devices need incentive to participate in federated learning. In practice, asking the mobile devices to work as sacrificial volunteers is not an economically viable and sustainable option. Moreover, in the federated learning paradigm, the direct communication between model owner and mobile devices is still required for transferring model updates, i.e., Fig. 1(a). In many scenarios, the direct communication may be unavailable because of limited transmission range and energy inefficiency because of high transmission power.

To address the incentive issues of the mobile devices, we adopt the service pricing scheme to motivate the mobile devices to participate in federated learning. Under the service pricing scheme, the machine learning is provided by the mobile devices as a service to the model owner. Then, the learning service, i.e., model update generation and trading as well as data collection, is performed in a decentralized manner. Additionally, to overcome the energy inefficiency in the model update transfer for the mobile devices, we resort to relay networking to ensure that the model updates are transferred in a cooperative manner. The mobile devices cooperatively form a relay network by providing relay service to each other and directly or indirectly connect to the access point of the model owner for model update transfer, e.g., Fig. 1(b). Note here that the mobile device that acts as the relay node will use the average operator to combine its own model update with its received model updates. Then, the file size of the model update is not affected, and hence providing relay service does not significantly affect the energy consumption of the mobile devices.

In this paper, we propose a novel framework of cooperative federated learning system. Our designed federated learning system involves two parties, i.e., the massive-scale mobile devices working as learning service providers, and the model owner handling the learning task dispatching (model releasing) and model updates collection. The mobile devices price their learning service by deciding on the price of one unit of their training data. In return, the model owner determines the size

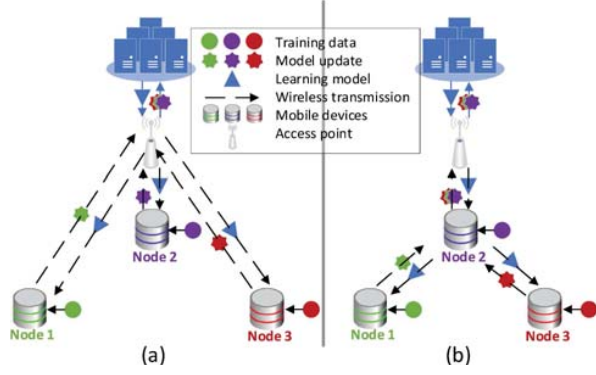


Figure 1: (a) Original communication scenario for federated learning; (b) Our designed communication scenario for federated learning.

of training data for each mobile device. As such, under the service pricing scheme, the mobile devices optimize the prices of their data to motivate the model owner to determine a larger size of training data and hence maximize their profits. However, under the cooperative relay network design, the larger size of training data implies the lower probability of enjoying the relay service. As a result, the learning service pricing and cooperative relaying should be considered jointly. By using the pricing-based data rent and a self-organized relay network design shown in Fig 1(b) for federated learning, the following key properties are guaranteed in our system:

- The model update throughput of the model owner scales well such that the massive model update volume from the mobile devices is handled smoothly.
- The rational and self-interested mobile devices noncooperatively decide on their own price of one unit of training data for individual profit optimization and cooperatively transfer their model updates.
- Significantly reduces the congestion in the communication for both the model owner and the mobile devices.

The rest of the paper is organized as follows. Section II describes the system model. Section III presents the formulation of a Stackelberg game. Section IV analyzes the equilibrium of the proposed Stackelberg game. Section V presents the numerical performance evaluation. Section VI concludes the paper.

## II. SYSTEM DESCRIPTION

We consider a cooperative federated learning system as shown in Fig. 1(b). Specifically, a model owner employs a set of mobile devices, e.g., mobile phones, to train a high-quality centralized model [5]. The set of mobile devices is denoted by  $\mathcal{N}^0$ , e.g., mobile devices 1, 2, and 3 in Fig. 1(b). Each mobile device  $i \in \mathcal{N}^0$  uses a part of its data and performs computation on its data locally to generate the model update for training the model of the model owner. The model owner negotiates with the mobile devices about the size of their training data, i.e.,  $s_i^d$ . In return, each mobile device  $i \in \mathcal{N}^0$  will receive

Table I: Notations

| Symbol                       | Description   |
|------------------------------|---|
| $i, N_D$                     | Node $i$ and the access point of the model owner.   |
| $\mathcal{N}^0$              | The set of mobile devices.  |
| $\mathcal{N},  \mathcal{N} $ | $\mathcal{N} = \mathcal{N}^0 \cup \{N_D\} = \{1, 2, \dots,  \mathcal{N}^0 , N_D\}$ is a sorted set and its cardinality.                             |
| $P_{ij}$                     | The transmission power used by mobile device $i$ to transfer the model update to mobile device $j$ .  |
| $T_i^s, T_i^a$               | The time used by mobile device $i$ for processing the model update and average operator, respectively.  |
| $h_{ij}, d_{ij}$             | The channel gain of the transmission between mobile devices $i$ and $j$ , and the distance between mobile devices $i$ and $j$ .                     |
| $q_i, s_i^d$                 | The price of one unit of training data for mobile device $i$ and the size of the data set used by mobile device $i$ for generated the model update. |
| $c_i^p, c_i^t$               | The costs that mobile device $i$ uses one unit of energy for processing one unit of data and transferring, respectively.                            |
| $c^a$                        | The fee that each mobile device $i \in \mathcal{N}$ charged by its rely node for the relay service.   |
| $I^d$                        | The size of the model update.   |
| $r_i$                        | The transmission rate of mobile device $i$ .  |

the revenue  $q_i s_i^d$  from the model owner, where  $q_i$  is the price for one unit of mobile device  $i$ 's training data. Intuitively, the learning accuracy of the model depends on the total size of all the mobile devices' training data. Specifically, the learning accuracy of the model becomes higher as the total size of all the mobile devices' training data increases. In this case, we incorporate the results in [6] to describe the relationship between the learning accuracy of the model and the total size of all the mobile devices' training data. As a result, the utility of the model owner is defined as follows:

$$U(s^d, \mathbf{q}) = f(s^d) - \sum_{i \in \mathcal{N}^0} q_i s_i^d, \quad (1)$$

where  $f(\cdot)$  is the function describing the relationship between the learning accuracy of the model and the total size of all the mobile devices' training data [6]. Note here that  $f(\cdot)$  is an increasing concave function implying that the learning accuracy of the model keeps increasing as the size of training data increases while the marginal increasing speed of the learning accuracy of the model decreases [6].

The mobile devices can cooperate with each other for transferring their model updates to the access point of the model owner. Let  $P_{ij}$  denote the transmission power used by mobile device  $i$  for transferring its model update to mobile device  $j$ , where we define the transmission power matrix  $\mathbf{P} = [p_{ij}]_{i,j \in \mathcal{N}}$ .  $I_{ij}$  is the element of the indicator matrix  $\mathbf{I}$ , i.e.,  $\mathbf{I} = [I_{ij}]_{i,j \in \mathcal{N}}$ , and defined as follows:

$$I_{ij} = \begin{cases} 1, & P_{ij} > 0 \\ 0, & \text{Otherwise} \end{cases}, \quad \forall i, j \in \mathcal{N}. \quad (2)$$

To provide the learning service, mobile device  $i$  has the cost of  $c_i^p s_i^d$  due to the energy consumption incurred from the computation. Moreover, each mobile device  $i \in \mathcal{N}^0$  has another cost incurred by the wireless transmission of its model update. Let  $c_i^t$  be the cost that mobile device  $i$  uses one unit of power for wireless transmission, the instantaneous cost for mobile

device  $i$  incurred by the transmission is  $c_i^t \sum_{j \in \mathcal{N}} P_{ij}$ . Under the assumption that the sizes of the model updates transferred by the mobile devices are the same, the energy consumption from the wireless transmission depends on the transmission rate of each mobile device, i.e.,  $r_i$ . We denote the size of the model update by  $I^d$ . The time for transferring the model update is accordingly  $\frac{I^d}{r_i}$ , and hence the energy consumption of mobile device  $i$  incurred by the wireless transmission is  $c_i^t \frac{I^d}{r_i} \sum_{j \in \mathcal{N}} P_{ij}$ . Furthermore, with the relay service among the mobile devices, each mobile device  $i \in \mathcal{N}^0$  will have a revenue of  $c^a \sum_{j \in \mathcal{N}^0} I_{ji}$  due to its provided relay service while a cost of  $c^a (1 - I_{iN_D})$  incurred by the use of the relay service.  $c^a (1 - I_{iN_D})$  comes from the fact that mobile device  $i$  uses the relay service and hence pay the fee  $c^a$  if it does not directly transfer the model update to the model owner, i.e.,  $I_{iN_D} = 0$ . Finally, with the revenue of providing the learning service to the model owner, i.e.,  $q_i s_i^d$ , the corresponding profit of mobile device  $i$  is

$$\begin{aligned} \Pi_i(\mathbf{P}, q_i, s_i^d) &= q_i s_i^d - c_i^t \frac{I^d}{r_i} \sum_{j \in \mathcal{N}} P_{ij} - c_i^p s_i^d \\ &+ c^a \sum_{j \in \mathcal{N}^0} I_{ji} - c^a (1 - I_{iN_D}) + v_i. \end{aligned} \quad (3)$$

To form the wireless relay networks among the mobile devices and hence achieve the energy efficient communication for the model update transfer, we have two sets of constraints. The first set of the constraints is for the routing. The second set of the constraints is for the model update arrival time at the relay point.

For the routing, we first have the constraint to ensure that every mobile device can connect to and transfer its model update to only one of other mobile devices or directly to the access point of the model owner. The constraint is expressed as follows:

$$\sum_{j \in \mathcal{N}} I_{ij} = 1 \quad \text{and} \quad I_{ii} = 0, \quad \forall i \in \mathcal{N}^0. \quad (4)$$

Secondly, we have the constraint that at least one mobile device connects to the access point of the model owner and acts as one of the last-hop nodes. Otherwise, none of the mobile devices can transfer the model update to the model owner. The corresponding constraint is expressed as follows:

$$\sum_{i \in \mathcal{N}^0} I_{iN_D} \geq 1. \quad (5)$$

We then have the constraint that the model update of each mobile device can finally and affirmatively arrive at the model owner. That is, each mobile device can transfer its model update to the model owner after a limited number of mobile devices as its relay nodes, i.e.,

$$\mathbf{I}^{|\mathcal{N}^0|-1} = \mathbf{I}_D, \quad (6)$$

where

$$\mathbf{I}_D^\top = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \\ 1 & \cdots & 1 \end{bmatrix} \begin{matrix} \leftarrow 1 \\ \vdots \\ \leftarrow |\mathcal{N}^0| \\ \leftarrow N_D \end{matrix}. \quad (7)$$

Regarding the constraint for the model update arrival time on the relay node, we have the constraint in (8). This constraint is to ensure that the model update of mobile device  $i$  will arrive at mobile device  $j$  before mobile device  $j$  finishes the computing of its model update if mobile device  $i$  wants to transfer its model update by choosing mobile device  $j$  as its relay node. The time used by mobile device  $i$  for providing the learning service can be divided into three periods. The first time period, denoted by  $T_i^s$ , is for performing the computation on the training data according to the model. The model update is generated at the end of this time period. Suppose that  $r_i^p$  is the processing rate of mobile device  $i$ ,  $T_i^s$  is accordingly defined as  $T_i^s = \frac{s_i^d}{r_i^p}$  [7]. The second time period is for mobile device  $i$  to provide relay service to other mobile devices. For each received model update from another mobile device, mobile device  $i$  needs to spend the time of  $T_i^a$  for combining the received model update with its own model update by using the average operator. The length of the second time period is therefore linearly related to the number of model updates received by mobile device  $i$ , i.e., the number of mobile devices that use mobile device  $i$ 's relay service  $\sum_{j \in \mathcal{N}} I_{ji}$ , and expressed as  $T_i^a \sum_{j \in \mathcal{N}} I_{ji}$ . The last time period is for mobile device  $i$  to

transfer its model update and expressed as  $\frac{I^d}{r_i}$ , where  $r_i$  is the transmission rate of mobile device  $i$  defined in (9). To ensure that mobile device  $i$  can successfully use the relay service from other mobile devices, the sum of three time periods at mobile device  $i$  must be shorter than the first time period of its relay node, i.e.,

$$T_i^s + T_i^a \sum_{j \in \mathcal{N}^0} I_{ji} + \frac{I^d}{r_i} \leq \sum_{j \in \mathcal{N}^0} I_{ij} T_j^s, \quad \forall i \in \mathcal{N}^0. \quad (8)$$

We assume that the mobile devices using the same relay node share the channel and hence generate mutual interference. Let  $h_{ij}$  denote the channel gain from mobile device  $i$  to mobile device  $j$ ,  $d_{ij}$  be the distance between mobile devices  $i$  and  $j$ , and  $\alpha \geq 2$  be the path loss coefficient of wireless communication. Then, we define a matrix  $\mathbf{H} = [H_{ij}]_{i,j \in \mathcal{N}}$ , the element of which, i.e.,  $H_{ij}$ , is  $\frac{h_{ij}}{(d_{ij})^\alpha}$ . Accordingly, the propagation model for data transmission of mobile device  $i$ , i.e., the transmission rate of mobile device  $i$ ,  $r_i$  is given by

$$\begin{aligned} r_i &= w_i \log_2 \left( 1 + \frac{\sum_{j \in \mathcal{N}} \frac{h_{ij}}{(d_{ij})^\alpha} P_{ij}}{(\mathbf{H}_\mathbf{I}(:,i))^\top (\mathbf{P}_\mathbf{I}(:,i)) - \sum_{j \in \mathcal{N}} \frac{h_{ij}}{(d_{ij})^\alpha} P_{ij} + \sigma^2}} \right) \\ &= w_i \log_2 \left( 1 + \frac{(\mathbf{H}(i,:))^\top (\mathbf{P}(i,:))}{(\mathbf{H}_\mathbf{I}(:,i))^\top (\mathbf{P}_\mathbf{I}(:,i)) - (\mathbf{H}(i,:))^\top (\mathbf{P}(i,:)) + \sigma^2} \right), \end{aligned} \quad (9)$$

where  $\sigma^2$  is the noise,  $w_i$  is the bandwidth of mobile device  $i$ ,  $\mathbf{P}_\mathbf{I} = \mathbf{P}\mathbf{I}^\top$ , and  $\mathbf{H}_\mathbf{I} = \mathbf{H}\mathbf{I}^\top$ . Note here that we use  $\mathbf{H}(i,:)$  and  $\mathbf{H}(:,i)$  to represent the  $i$ -th row vector and  $i$ -th column vector of  $\mathbf{H}$ , respectively, and the usage of this notation is defined similarly in the rest of this paper.

### III. STACKELBERG GAME FORMULATION

In this section, we model the interaction between the mobile devices and the model owner as a Stackelberg game. In the Stackelberg game, the model owner is the buyer as it uses the learning service provided by the mobile devices. Then, the mobile devices that are the service providers act as the sellers. The sellers typically make their decisions before the buyers. Following this case, the model owner inherently acts as the single follower in the lower level of the Stackelberg game while the mobile devices are the corresponding leaders. In the lower level of the game, i.e., the lower-level subgame, the model owner determines the size of training data for the mobile devices. In the upper level, i.e., the upper-level subgame, the mobile devices decide on the price for one unit of their training data. Moreover, since the mobile devices cooperatively send their model update to the model owner, each mobile device also needs to independently decide on its relay node as well as its transmission power. As a result, the Stackelberg game can be formally defined as follows:

- 1) *Lower-level subgame*: Given the fixed vector of the prices of one unit of training data  $\mathbf{q} = [q_1, \dots, q_{|\mathcal{N}^0|}]^\top$ , the lower-level subgame is defined by the a three-tuple  $\mathcal{G}_L = \{\mathbf{s}^d, \mathbb{S}^d, U\}$ , where
  - $\mathbf{s}^d = [s_1^d, \dots, s_{|\mathcal{N}^0|}^d]^\top$  is the vector of the sizes of training data;
  - $\mathbb{S}^d = \left\{ [s_1^d, \dots, s_{|\mathcal{N}^0|}^d]^\top \mid s_i^d \in [0, s_i^{d,u}], i \in \mathcal{N}^0 \right\} \subset \mathbb{R}^{|\mathcal{N}^0|}$  is the domain of definition for  $\mathbf{s}^d$  and an M-polyhedron, where  $s_i^{d,u}$  is the upper bound of  $s_i^d$ ;
  - $U = U(\mathbf{s}^d, \mathbf{q})$  is the utility function of the model owner defined in (1);
- 2) *Upper-level subgame*: After the model owner's demand of data size  $\mathbf{s}^d$  is determined in the lower-level subgame, the mobile devices form a upper-level subgame defined by a six-tuple  $\mathcal{G}_U = \{\mathcal{N}^0, \mathbf{P}, \mathbb{P}, \mathbf{q}, \mathbb{Q}, \mathbf{\Pi}\}$ , where
  - $\mathcal{N}^0$  is the set of the mobile devices;
  - $\mathbf{P} = [P_{ij}]_{i,j \in \mathcal{N}}$  is the matrix of the power for wireless transmission;
  - $\mathbb{P} = \left\{ [P_{ij}]_{i,j \in \mathcal{N}} \mid P_{ij} \in [0, P_{ij}^u], i, j \in \mathcal{N} \right\} \subset \mathbb{R}^{|\mathcal{N}| \times |\mathcal{N}|}$  is the domain of definition for  $\mathbf{P}$ , where  $P_{ij}^u$  is the upper bound of  $P_{ij}$ ;
  - $\mathbf{q} = [q_1, \dots, q_{|\mathcal{N}^0|}]^\top$  is the vector of prices for one unit of training data.
  - $\mathbb{Q} = \left\{ [q_1, \dots, q_{|\mathcal{N}^0|}]^\top \mid q_i \in [0, q_i^u], i \in \mathcal{N}^0 \right\} \subset \mathbb{R}^{|\mathcal{N}^0|}$  is the domain of definition for  $\mathbf{q}$  and an M-polyhedron, where  $q_i^u$  is the upper bound of  $q_i$ ;
  - $\mathbf{\Pi} = [\Pi_1, \dots, \Pi_{|\mathcal{N}^0|}]^\top$  is the vector of the profits for the mobile devices, where  $\Pi_i = \Pi_i(\mathbf{P}, q_i, s_i^d)$  is the profit of mobile device  $i$  defined in (3).

Based on the game formulation, we consider a Stackelberg equilibrium to be the solution for the model owner and the mobile devices.

### IV. EQUILIBRIUM ANALYSIS

By following the backward induction, we firstly use the first-order optimality condition to obtain the optimal solution to the lower-level subgame  $\mathcal{G}_L$ . The existence of the optimal solution to the lower-level subgame  $\mathcal{G}_L$  is proven to exist by showing the concavity of its utility function. This optimal solution is further proven to be unique by showing the negative definiteness of the Hessian matrix of the utility function of the lower-level subgame  $\mathcal{G}_L$ . Then, we substitute the NE of the lower-level subgame  $\mathcal{G}_L$  into the upper-level subgame  $\mathcal{G}_U$  and investigate the solution to the upper-level subgame  $\mathcal{G}_U$  by capitalizing on the exterior point method.

#### A. Solution to Lower-level Subgame

To find an optimal solution for the lower-level subgame  $\mathcal{G}_L$ , we need to take the first derivative of the utility function of the model owner given in (1) with respect to  $s_i^d$  as follows:

$$\begin{aligned} \frac{\partial}{\partial s_i^d} U(\mathbf{s}^d, \mathbf{q}) &= \frac{\partial}{\partial s_i^d} \left[ f(\mathbf{s}^d) - \sum_{i \in \mathcal{N}^0} q_i s_i^d \right] \\ &= -q_i + \frac{\partial}{\partial s_i^d} f(\mathbf{s}^d), \forall i \in \mathcal{N}^0, \end{aligned} \quad (10)$$

where  $f(\mathbf{s}^d) = \sum_{i \in \mathcal{N}^0} f_i(s_i^d)$ . Without loss of generality and for the trackable analysis, we adopt Weibull model as suggested in [6], i.e.,  $f_i(s_i^d) = a_i - b_i \exp(-c_i s_i^d)$ . Let  $\frac{\partial}{\partial s_i^d} U(\mathbf{s}^d, \mathbf{q}) = 0, \forall i \in \mathcal{N}^0$ , we have  $s_i^{d*}$ , i.e., the best response of the model owner, as follows:

$$\begin{aligned} -q_i + \frac{\partial}{\partial s_i^d} f(\mathbf{s}^d) &= -q_i + \frac{\partial}{\partial s_i^d} [a_i - b_i \exp(-c_i s_i^d)] \\ &= -q_i + c_i b_i \exp(-c_i s_i^d) = 0 \Leftrightarrow s_i^{d*} = \frac{1}{c_i} \ln \frac{c_i b_i}{q_i} \end{aligned} \quad (11)$$

Due to the strict concavity of  $f_i(\cdot)$  and the linearity of  $-q_i s_i^d$  with respect to  $s_i^d$ ,  $U(\mathbf{s}^d, \mathbf{q})$  is therefore concave with respect to  $s_i^d$  and its concavity indicates the existence of the solution to the lower-level subgame  $\mathcal{G}_L$ . Moreover, with negative elements on the primary diagonal and zero elements on the off-diagonal, the Hessian matrix of  $U(\mathbf{s}^d, \mathbf{q})$  is negative definite, and hence the solution to the lower-level subgame  $\mathcal{G}_L$  is unique.

#### B. Solutions to Upper-level Subgame

After obtaining the optimal demand of the data size for the model owner, we investigate the upper-level subgame  $\mathcal{G}_U$  for the mobile devices. At the Nash Equilibrium (NE), no player can increase its profit by choosing a different strategy provided that the other players' strategy is unchanged [8]. We firstly substitute the optimal demand of the data size for the model owner given in (11) into the profit functions of the mobile devices given in (3) and have the new profit functions for the mobile devices as follows:

$$\begin{aligned} \Pi_i(\mathbf{P}, q_i) &= \Pi_i(\mathbf{P}, q_i, s_i^{d*}) = q_i \frac{1}{c_i} \ln \frac{c_i b_i}{q_i} - c_i^d \frac{I^d}{r_i} \sum_{j \in \mathcal{N}} P_{ij} \\ &\quad - c_i^p \frac{1}{c_i} \ln \frac{c_i b_i}{q_i} + c^a \sum_{j \in \mathcal{N}^0} I_{ji} - c^a (1 - I_{iN_D}) + v_i \end{aligned} \quad (12)$$

with the constraints (4)-(8).



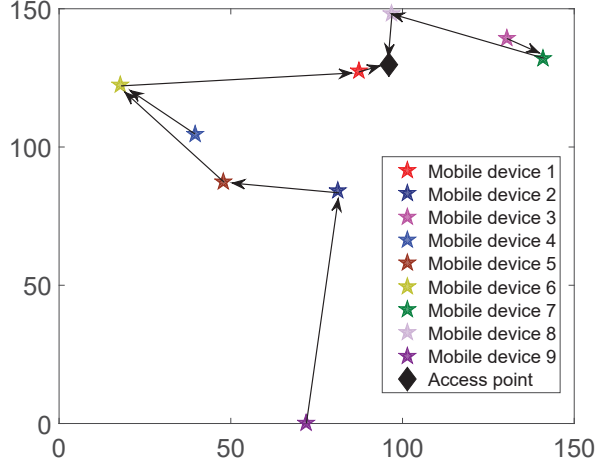


Figure 2: The wireless relay network for cooperative model update transfer.

## V. PERFORMANCE EVALUATION

In this section, we present numerical studies to evaluate the performance of the cooperative federated learning system. For the ease of illustration, we consider 9 mobile devices, i.e.,  $|\mathcal{N}^0| = 9$ , working as the learning entities. The bandwidth and channel gain are respectively  $h_{ij} = 1$  and  $w_i = 100$ ,  $\forall i, j \in \mathcal{N}^0$ , and the noise  $\sigma^2$  is 1. The size of the model update is assumed to be  $I^d = 0.001$ . The distances among the mobile devices as well as that between the mobile devices and the access point of the model owner, i.e.,  $d = [d_{ij}]_{i,j \in \mathcal{N}}$ , follows a uniform distribution under the plane of  $[0, 150] \times [0, 150]$ . The vector of the costs of consuming one unit of power for wireless communication is  $\mathbf{c}^t = [c_i^t]_{i \in \mathcal{N}^0} = [150]^\top$ . The vector of the costs of processing one unit of data is  $\mathbf{c}^p = [c_i^p]_{i \in \mathcal{N}^0} = [0.01]$ . We set the fee of relay service to be  $c^a = 0.01$ . The vector of processing rate is  $\mathbf{r}^p = [r_i^p]_{i \in \mathcal{N}^0} = [0.01]^\top$ . The vector of the time of using average operator for the mobile devices is  $\mathbf{T}^a = [T_i^a]_{i \in \mathcal{N}^0} = [0.01]$ . The coefficient of  $f_i(\cdot)$  defined in (1) is  $\mathbf{c} = [c_{ij}]_{i,j \in \mathcal{N}^0} = [13.28, 9.17, 14.31, 11.21, 9.12, 13.61, 13.27, 9.63, 14.32]$  and  $\mathbf{a} = [a_{ij}]_{i,j \in \mathcal{N}^0} = \mathbf{b} = [b_{ij}]_{i,j \in \mathcal{N}^0} = [9.78, 9.15, 11.35, 11.17, 12.7, 9.15, 12.38, 13.5, 10.59]$ .  $v_i$  is equal to 10 for all  $i \in \mathcal{N}$ .  $P_{ij}^u$  is 10 for all  $i, j \in \mathcal{N}$ .

### A. Numerical Result

For the convenience to observe the routing result for the cooperative model update transfer, We present it in Fig. 2 and Table II. As shown in Fig. 2, the mobile devices self-organize the wireless rely network for cooperative model update transfer. For example, mobile device 3 uses mobile device 7 as the relay node for transferring the model update. This can significantly reduce the energy consumption of wireless communication for mobile device 3. Moreover, both mobile devices 4 and 5 choose the same mobile device, i.e., mobile

Table II: The routing result for cooperative model update transfer.

| Mobile device No. | Routing   |
|-------------------|---|
| 1                 | $1 \rightarrow N_D$ .   |
| 2                 | $2 \rightarrow 5 \rightarrow 6 \rightarrow 1 \rightarrow N_D$ .               |
| 3                 | $3 \rightarrow 7 \rightarrow 8 \rightarrow N_D$ .                             |
| 4                 | $4 \rightarrow 6 \rightarrow 1 \rightarrow N_D$ .                             |
| 5                 | $5 \rightarrow 6 \rightarrow 1 \rightarrow N_D$ .                             |
| 6                 | $6 \rightarrow 1 \rightarrow N_D$ .   |
| 7                 | $7 \rightarrow 8 \rightarrow N_D$ .   |
| 8                 | $8 \rightarrow N_D$ .   |
| 9                 | $9 \rightarrow 2 \rightarrow 5 \rightarrow 6 \rightarrow 1 \rightarrow N_D$ . |

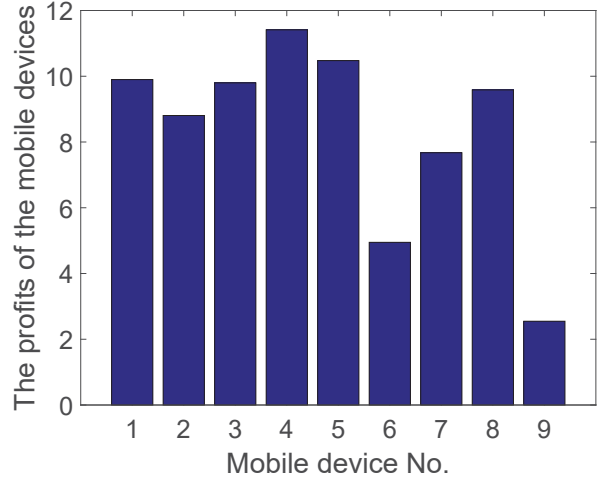


Figure 3: The profits of the mobile devices.

device 6, as their relay node. This will incur the mutual interference between mobile devices 4 and 5 and hence result in energy inefficient communication. However, compared with choosing other mobile devices as the relay node or directly transferring the model update to the central cloud, it is more energy efficient to choose mobile device 6 as the relay node for mobile devices 4 and 5.

We next evaluate the profits for the mobile devices. As shown in Fig. 3, we observe that the profit of mobile device 4 is the highest one among the mobile device. The reason is that the model update from mobile device 4 has a significant improvement on the model of the owner. This means that the model update from mobile device 4 contains much more valuable information than the other mobile devices. In contrast, although the model update from mobile device 6 is the combination of the model updates from the model updates of mobile devices 4 and 5, its profit is much less than that of mobile device 4. This is due to the data quality of mobile device 6 and the preference of the model owner.

We then investigate the sizes of the training data for the model owner. As shown in Fig. 4, the size of the training data from mobile device 1 is the largest among the mobile devices. This is due to the shortest distance between the access point and mobile device 1. Moreover, due to the longest data

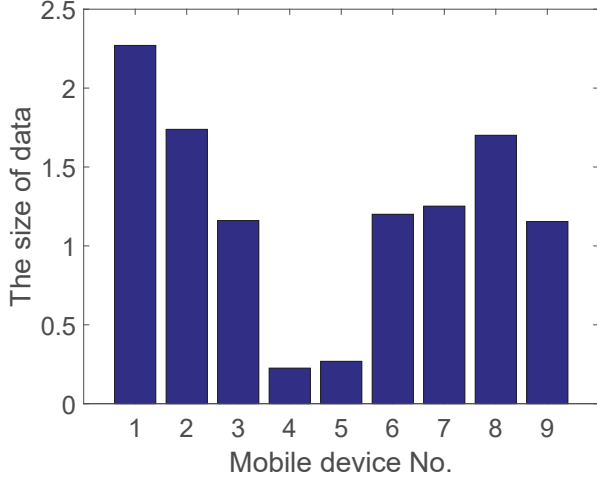


Figure 4: The size of training data for each mobile device.

processing time, mobile device 1 can make more profit by acting as the relay node for other mobile devices. Comparing with mobile device 1, the size of data of mobile device 4 is much less. The reason is that mobile device 4 needs to use the other mobile devices as its relay node, and hence the update of mobile device 4 needs to be transferred before its relay node finishes the computation. Thus, to guarantee this, mobile device 4 can only perform the computation on its data with a very small size.

## VI. CONCLUSION

In this paper, we have presented the Stackelberg game model to analyze the transmission strategy and training data pricing strategy of the self-organized mobile device as well as the learning service subscription of the model owner in the cooperative federated learning system. We have focused on the interactions among the mobile devices and considered the impact of the interference cost on the mobile devices' profits. Moreover, we have investigated the impact of the size of the training data on both the model owner's utility and the mobile devices's profits. Specifically, we have established a model describing the impact of the mobile devices' transmission strategies on their transmission rates and relay node. The model also describes the impact of the model owner's learning service subscription strategy on the model owner's utility. We have studied the optimal strategy of the model owner by using best response and the equilibrium strategies of the mobile devices by using the exterior point method. Our future work will extend to the study in the long-run among the model owner and the mobile devices.

## ACKNOWLEDGEMENT

This work was supported in part by A\*STAR-NTU-SUTD Joint Research Grant Call on Artificial Intelligence for the Future of Manufacturing RGANS1906, WASP/NTU M4082187

(4080), Singapore MOE Tier 1 under Grant 2017-T1-002-007 RG122/17, MOE Tier 2 under Grant MOE2014-T2-2-015 ARC4/15, Singapore NRF2015-NRF-ISF001-2277, and Singapore EMA Energy Resilience under Grant NRF2017EWT-EP003-041.

## REFERENCES

- [1] S. Feng, W. Wang, D. Niyato, D. I. Kim and P. Wang, "Competitive data trading in wireless-powered internet of things (iot) crowdsensing systems with blockchain," *arXiv preprint arXiv:1808.10217*, 2018.
- [2] D. W. Opderbeck, "Cybersecurity, data breaches, and the economic loss doctrine in the payment card industry," *Md. L. Rev.*, vol. 75, pp. 935, 2015.
- [3] google, "Federated learning: Collaborative machine learning without centralized training data," <https://ai.googleblog.com/2017/04/federated-learning-collaborative.html>, Accessed April 6, 2017.
- [4] TensorFlow, "Machine learning and mobile: Deploying models on the edge," <https://blog.algorthmia.com/machine-learning-and-mobile-deploying-models-on-the-edge/>, Accessed June 21, 2018.
- [5] J. Konecný, H. B. McMahan, D. Ramage and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.
- [6] B. Gu, F. Hu and H. Liu, "Modelling classification performance for large data sets," in *International Conference on Web-Age Information Management*. Springer, 2001, pp. 317–328.
- [7] A. Gharaibeh, T. Reza, E. Santos-Neto, L. B. Costa, S. Sallinen and M. Ripeanu, "Efficient large-scale graph processing on hybrid cpu and gpu systems," *arXiv preprint arXiv:1312.3018*, 2013.
- [8] M. J. Osborne et al., *An introduction to game theory*, vol. 3, Oxford university press New York, 2004.
- [9] X. Yang, "An exterior point method for computing points that satisfy second-order necessary conditions for a c1, 1 optimization problem," *Journal of Mathematical Analysis and Applications*, vol. 187, no. 1, pp. 118–133, 1994.