Valuation-Aware Federated Learning: An Auction-Based Approach for User Selection

Pan-Yang Su*†, Pei-Huan Tsai*†, Yu-Kang Lin*†, Hung-Yu Wei†
*Equal Contribution †Department of Electrical Engineering, National Taiwan University, Taiwan

***Abstract*—Federated learning (FL) is a machine learning paradigm in which many users collaboratively train a model under the supervision of a central server (CS). Because the model performance is highly dependent on user quality, user selection becomes a critical issue in FL. In this paper, we develop a system model where the CS aims to select users with high computational power and valuation of the global model. In this regard, we propose an incentive mechanism to motivate users to reveal their computational power and model valuation. Then, we formulate a cost-minimization optimization problem of the CS and propose a polynomial-time dynamic programming algorithm to solve it. The proposed scheme effectively avoids the free-rider problem in which a user with little contribution can obtain the model by joining the FL process. Moreover, utilizing auction theory, our mechanism incentivizes users to report their computational power and model valuation truthfully. Finally, extensive theoretical analysis and numerical simulation validate the superiority of the proposed mechanism compared with two state-of-the-art user selection mechanisms.**

I. Introduction

With the proliferation of training data for mobile devices, federated learning (FL) emerges as a solution to the privacy consideration of local data and workload alleviation of the central server (CS) [1]. In FL, edge devices (ED) distributively perform machine learning (ML) training and transmit local parameters to the CS. The CS aggregates the parameters and distributes the aggregation result to EDs for subsequent training. This process is repeated until a global model which reaches a desired accuracy is obtained.

Since mobile devices carry out most of the training process, the selection of suitable participants is thus a critical issue. Typically, contract-based mechanisms are extensively adopted to encourage high-quality users to participate by giving corresponding rewards [2]–[5]. In [2], the authors discuss an incentive mechanism to compensate users for their privacy leakage based on contract theory. In [3], [4], a Stackelberg game is formulated along with the design of optimal contracts. In [3], the consideration is data quality and quantity of users, while privacy is the primary concern in [4]. Wu *et al.* jointly consider computation, communication, and privacy costs of data owners and devise a multidimensional contract [5]. Nonetheless, although contract theory plays a dominant role in user selection, it suffers two distinct disadvantages. First, by signing contracts, the CS approves the participation of all EDs, but the inclusion of some users may degrade the system performance. Second, the CS cannot know users' valuation of the global model because contract theory aims to give rewards to some workload, but model valuation is not a kind of workload contribution.

Some work has attempted to address the first issue [6]–[9]. One prominent aspect is the consideration of a bandwidth-constrained mobile network [6], [7]. In [6], the authors jointly consider resource allocation and user scheduling to minimize the maximum update delay of model training. In [7], the authors leverage the randomized auction to motivate EDs to join FL training and select appropriate participants simultaneously. Similar to [7], Kang *et al.* develop an incentive mechanism but use user reputation as a selection criterion [8]. Nishio and Yonetani further examine users' limited computational resources when devising a client selection mechanism [9].

Nevertheless, while the first issue has been investigated recently, as previously discussed, less attention is given to the inspection of users' valuation of the model and the resulting free-rider problem. The free-rider problem arises when some users participate in the FL training to acquire the global model without actually contributing their computational resources [10]. This situation may damage the overall system performance and lead to a waste of resources. In this regard, we propose an incentive mechanism based on a combination of English auction and second-price auction to avoid the free-rider problem. The English auction aims to incentivize users to report their computational power truthfully, and the second-price auction serves to motivate users to reveal their model valuation. Moreover, we formulate a cost-minimization optimization problem for the CS and devise a dynamic programming algorithm to solve it. We summarize the contributions of the paper as follows.

1) We propose a mechanism that takes user heterogeneity into account. In particular, we jointly consider users' computational contributions in the FL training process and their valuation of the global model.
2) We develop an auction-based incentive mechanism that motivates each ED to report its computational power and model valuation truthfully.
3) We present a cost-minimization optimization algorithm and a dynamic programming algorithm under the devised mechanism for the CS to select suitable EDs to train the model and effectively avoid the free-rider problem.

II. System Model

In this section, we describe the general FL system model (§II-A) and the computational model of EDs (§II-B). Then, we formulate the user selection problem of the CS (§II-C).

Corresponding Author: Hung-Yu Wei, email: hywei@ntu.edu.tw

A. Federated Learning Model

In ML optimization, the goal is to find a set of parameters w^* to minimize a loss function f given input datasets x:

$$w^* = \arg\min_{w} f(w, x). \tag{1}$$

In FL architecture, the global model parameters, denoted as $\mathbf{w}$, are a weighted summation of local parameters of all the EDs, as formulated below:

$$\mathbf{w} = \sum q_i w_i, \tag{2}$$

where q_i and w_i represent the weight and local parameters of ED ED_i, respectively.

For each global iteration, the CS updates the global model's parameters as prescribed by the gradient descent method:

$$\mathbf{w^{t+1}} = \mathbf{w^t} - \alpha \sum q_i \nabla f_i(w_i, x_i), \tag{3}$$

where $\mathbf{w^t}$ is the parameters of the global model at global iteration t, α is the learning rate, and $\nabla f_i(w_i, x_i)$ is the gradient computed by ED_i's local model. In the following, we use the terms ED and user interchangeably.

B. Computational Model

In the FL training process, EDs contribute computational resources according to their respective capabilities. Typically, a higher computational power contribution leads to a more accurate local model. We denote an ED ED_i's local model accuracy as ϵ_i^l. Based on [3], given ED_i's computational power c_i^l, the local relative accuracy ϵ_i^l can be expressed as:

$$\epsilon_i^l = 1 - e^{-\eta c_i^l}, \tag{4}$$

where η is a positive constant. Note that we use the superscript l to differentiate it from the global model, where we use g as the superscript.

To reach a desired global model accuracy ϵ^g, the CS selects sufficient participants for training. The relationship between the global model accuracy and the upper bound on the required computational power c^g can be formulated as [11]:

$$c^g = \sum_{i=1}^{N} c_i^l = \frac{O(\log \frac{1}{1-\epsilon^g})}{\max \epsilon_i^l}. \tag{5}$$

The inverse dependence on $\max \epsilon_i^l$ indicates the existence of a global accuracy limit through solving local problems. There is at least an order of $\log \frac{1}{\epsilon^g}$ computational cost needed for training the global model.

C. User Selection System Model

The proposed system model consists of one CS and multiple EDs with different computational power and valuation of the global model. The CS aims to select suitable users to train an FL model, and EDs aim to receive monetary rewards by providing computational power in the training process. The monetary rewards serve to compensate for EDs' energy consumption in the FL training process. However, some EDs may benefit from obtaining the global model to perform some ML tasks, so they may request lower monetary rewards but still get positive utility. Hence, the CS can pay less to these EDs. In this regard, the CS wants to select EDs with high computational power and model valuation to minimize the FL training cost.

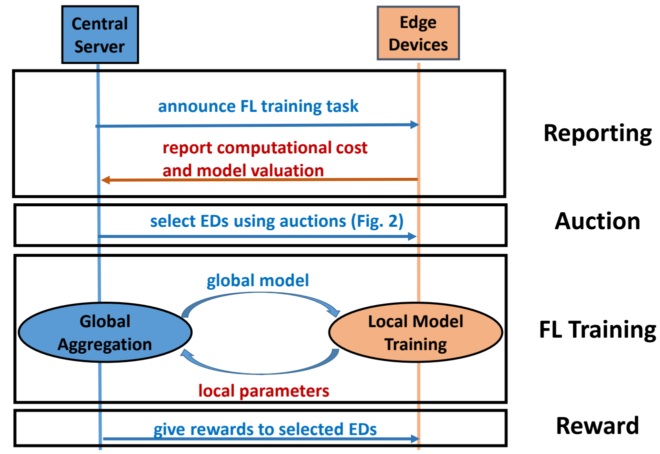


Fig. 1. Flow chart of the proposed mechanism.

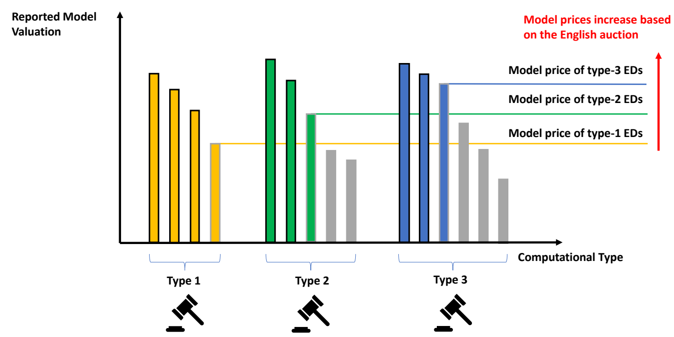


Fig. 2. Auction system of the proposed mechanism.

Nonetheless, EDs may untruthfully report their computational power and model valuation to increase personal utility, resulting in the degradation of system performance. To tackle this issue, we devise an auction-based mechanism to incentivize EDs to reveal their computational power and model valuation truthfully. Based on the EDs' information, we present a cost-minimization optimization problem of the CS and utilize a dynamic programming algorithm to solve it. In the following, we describe the system architecture in detail.

We assume that there are J levels of computational power, denoted by $\textbf{Type} = \{t_1, t_2, ..., t_J\}$, where $t_1 > t_2 > ... > t_J$. We refer to an ED with computational power t_j as a type-j ED. The number of type-j EDs is K_j, and the total number of EDs is $K = \sum_{j=1}^{J} K_j$. We denote each user ED as ED_i, whose computational power and valuation of the global model are denoted as $c_i \in \textbf{Type}$ and $v_i \in \mathbb{R}^+$, respectively. Note that for a type-j ED ED_i, its computational power c_i is t_j. An example with three EDs and two computational types is shown in Tbl. I, where $\textbf{Type} = \{t_1, t_2\} = \{200, 100\}$, $c_1 = 200 = t_1$, $c_2 = 100 = t_2$, and $c_3 = 100 = t_2$.

TABLE I
A SIMPLE EXAMPLE

ED ID	Computational power	Computational type
1	200	1
2	100	2
3	100	2

We denote the utility of ED_i as u_i. If ED_i participates in the FL training process, it will spend c_i computational cost in model training. On the other hand, it can get the global model with v_i utility, and the CS will provide r_i reward to compensate for the its energy consumption cost. r_i will be defined more detailedly later. Thus, we have $u_i = r_i + v_i - c_i$.

III. Mechanism Design

After defining some notations, which are summarized in Tbl. II, we present the flow of our proposed mechanism in Fig. 1, and the detail is described below.

1) **Reporting:** The CS announces the FL training mechanism and requests interested EDs to report their computational power and valuation of the model. Each interested ED ED_i reports to the CS with a tuple (d_i, b_i), where $d_i \in$ **Type** is the reported computational power, and $b_i \in \mathbb{R}^+$ is the reported valuation of the model. Then, EDs are classified into different groups such that each group consists of EDs with the same computational power.
2) **Auction:** The CS solves an optimization problem, which is described in §IV, to determine the number of winning EDs N_j in each group j. Then, the CS holds a second-price auction in each group to motivate EDs to reveal their true valuation.
   We denote the model price in group j as p_j. The model prices between different groups must follow: $p_j \leq p_{j+1}$ as prescribed by the English auction. In other words, the model price of the previous group j becomes the reserve price of the next group $j+1$, so type-$(j+1)$ EDs with bids lower than p_j is excluded from participating in the auction. As for other EDs, the second-price auction is conducted, and the model price is set as the highest bid among the losing EDs.
   Fig. 2 shows the auction procedure with $K_1 = 4$, $K_2 = 5$, and $K_3 = 6$. There are three computational types. The yellow line is p_1, the green line is p_2, and the blue is p_3. These values are the highest model valuations among the losing EDs in each group according to the second-price auction. We can observe that $p_3 \geq p_2 \geq p_1$ as demanded by the English auction. On the other hand, within each group, the gray bars are EDs whose model valuations are below the reserve prices, so they are forbidden from the second-price auction. The final result is $N_1 = 3$, $N_2 = 2$, and $N_3 = 2$, as indicated by the bordered bars.
3) **FL Training:** After deciding on the participating EDs, the FL training is conducted based on §II-A. The CS distributes the initial global model to the selected EDs. Then, the EDs perform local model training and send the updated parameters to the CS. The CS aggregates the parameters and distributes the resulting model to the EDs again. This process is repeated until the global model accuracy achieves the predetermined accuracy threshold.
4) **Reward:** After the FL training, the CS gives each ED ED_i who fulfills its task successfully a reward r_i and the final global model. The reward aims to compensate for the EDs' computational contribution while considering their model valuation. Specifically, for each type-j ED ED_i, we have $r_i = d_i - p_j$, where d_i is to compensate for its energy consumption, and p_j is the price for model valuation calculated from the second-price auction in group j. Thus, ED_i's utility u_i is as follows:

$$u_i = r_i + v_i - c_i = (d_i - p_j) + v_i - c_i. \tag{6}$$

TABLE II
Table of Notations

Notation	Definition
x	Input datasets in FL training.
w	Model parameters in FL training.
f	Loss function in FL training.
$\mathbf{w}$	Global model parameters in FL training.
α	Learning rate in FL training.
ED_i	Edge device whose ID is i.
q_i	ED_i's weight during global model aggregation.
$\nabla f_i(w_i, x_i)$	The gradient computed by ED_i's local model.
ϵ_i^l	Relative accuracy of ED_i's local model.
ϵ^g	Relative accuracy of the global model.
ϵ_{max}^l	The maximum relative accuracy of local models. $\epsilon_{max}^l = \max_{1\leq j \leq J} \epsilon_j^l$.
c^g	Required total computational power to achieve a certain accuracy of the global model.
t_j	Computational power of type-j EDs.
Type	The set of computational types (computational power) of EDs. **Type** $= \{t_j \mid j = 1, 2, ..., J\}$.
K_j	The number of type-j EDs.
K	The total number of EDs. $K = \sum_{j=1}^{J} K_j$.
N_j	The number of selected type-j EDs.
$\mathbf{N}$	The user selection result. $\mathbf{N} = \{N_j \mid j = 1, 2, ..., M\}$.
v_i	ED_i's valuation of the global model.
c_i	ED_i's computational power.
b_i	ED_i's reported valuation of the global model.
d_i	ED_i's reported computational power.
p_j	Payment of type-j winning players in the second-price auction.
r_i	ED_i's reward given by the CS. Note that $r_i = d_i - p_j$ if type-j ED_i contributes its power d_i in FL training, and $r_i = 0$, otherwise.
u_i	ED_i's utility. $u_i = r_i + v_i - c_i$.
B_j^i	The ith largest bid in group j. Note that $p_j = B_j^{N_j+1}$ based on the second-price auction.
T	Targeted accuracy threshold of the global model.
S	Required total computational power for the global model to reach the accuracy threshold.

IV. User Selection Algorithm of the CS

In this section, we formulate the user selection optimization problem of the CS as follows. Note that we define the ith largest bid in group j as B_j^i, and $B_j^{N_j+1}$ is the payments of type-j EDs based on the second-price auction.

$$\min_{\mathbf{N}} \sum_{j=1}^{J} (t_j - B_j^{N_j+1}) N_j. \tag{7a}$$

$$\text{s.t. } N_j \in \{0, 1, ..., K_j - 1\}, 1 \leq j \leq J, \tag{7b}$$

$$B_j^{N_j+1} \geq B_{j-1}^{N_{j-1}+1}, 2 \leq j \leq J, \tag{7c}$$

$$\sum_{j=1}^{J} t_j N_j \geq S. \tag{7d}$$

The objective function represents the CS's cost function. The first constraint describes the range of variables $\mathbf{N} = \{N_1, N_2, ..., N_J\}$ for each type. The second constraint follows from the concept of English auction, where the price keeps increasing. Finally, the last constraint demands that the total computation power surpass a threshold S, which stems from the following three constraints from eq. (4) and eq. (5) with η_1 and η_2 being constants, and T being the accuracy threshold the aggregate model must surpass.

$$\epsilon_j^l = 1 - e^{-\eta_2 t_j}, 1 \le j \le J, \tag{8}$$

$$\epsilon_{max}^l = \max_{1 \le j \le J} \epsilon_j^l, \tag{9}$$

$$T \le 1 - e^{-\eta_1 (\sum_{i=1}^J t_j N_i) \epsilon_{max}^l}. \tag{10}$$

Thus, the relationship between S and T can be formulated as:

$$S = \frac{\ln(1-T)}{-\eta_1(1 - e^{-\eta_2 t_j})}. \tag{11}$$

In considering the optimality of the CS's cost, we identify an optimal substructure in the minimization problem. Let $mc(j, s, n_j)$ denote the minimum cost of the CS to reach a total computational cost s using n_j type-j EDs, some EDs with types lower than j, and no EDs with types higher than j. By traversing the different situations of using type-$(j-1)$ EDs, $mc(j, s, n_j)$ can be recursively calculated as:

$$mc(j, s, n_j) = \min_{0 \le n_{j-1} \le K_{j-1}} mc(j-1, s - n_j \cdot t_j, n_{j-1}) + g(j, n_j). \tag{12}$$

Note that we use $g(j, n_j)$ to denote $(t_j - B_j^{n_j+1})n_j$ defined in eq. (7a) to simply the notation. From the above recursive relationship, we thus propose the following dynamic programming algorithm to solve the optimization problem in polynomial time. The algorithm is described in Alg. 1.

Algorithm 1 CS optimization algorithm using dynamic programming

1: Initialization:
2: Let $mc[1...J, 1...S, 1...K_{max}]$ be a new table ($K_{max} = \max_j K_j$).
3: **for** $s = 0$ to S **do**
4:     **for** $n_1 = 0$ to K_1 **do**
5:         **if** $n_1 t_1 \ge s$ **then**
6:             $mc[1, s, n_1] \leftarrow g(1, n_1)$.
7: Use eq. (12) to compute mc.
8: Backtrack to obtain $N[1...J]$.
9: **return** $N[1...J]$.

In Alg. 1, we construct a three-dimensional table $mc[1...J, 1...S, 1...K_{max}]$ (line 2). $mc[j, s, n]$ is the minimum cost for the CS to achieve computational power s with exactly n type-j EDs selected, some EDs with types lower than j, and no EDs with types higher than j.

The algorithm first initializes mc by assigning $g(1, n_1)$ to $mc[1, s, n_1]$ once the total computational power reaches the threshold s (line 3 to line 6). Then, mc is recursively updated based on the principle of dynamic programming in eq. (12). Finally, we backtrack with the table mc to acquire the optimal solution, which is stored in $\mathbf{N}$ (line 8 to line 9).

We analyze the time complexity of Alg. 1 as follows. The construction of the 3D array in line 2 takes $O(SK_{max}J)$ time. Next, the initial condition setting from line 3 to line 6 requires $O(SK_{max})$ time. Then, we apply the dynamic programming method in line 7, which takes $O(SK_{max}^2 J)$ time. Finally, we perform backtracking in line 8, which takes $O(J)$ time to finish. Thus, the overall time complexity of Alg. 1 is $O(SK_{max}^2 J)$, which is polynomial-time.

V. Theoretical Analysis

In this section, we prove that our mechanism guarantees the following properties;

- **Individual Rationality:** If an ED ED_i reports truthfully, its utility is non-negative, i.e., $(d_i, b_i) = (c_i, v_i) \Rightarrow u_i \ge 0$.
- **Incentive Compatibility:** An ED ED_i achieves maximum utility when it reports truthfully, i.e., $(c_i, v_i) = \arg\max_{(d_i, b_i)} u_i$.

A. Individual Rationality

In this subsection, we prove that our mechanism satisfies individual rationality, meaning that a truthfully-reporting ED always gets non-negative utility when joining the FL training.

Theorem 1. *If an ED ED_i reports truthfully, its utility is non-negative, i.e., $(d_i, b_i) = (c_i, v_i) \Rightarrow u_i \ge 0$.*

Proof. If ED_i is selected, its utility can be formulated as $u_i = d_i - p_j + v_i - c_i$, where j is ED_i's computational type. Since the ED reports truthfully, we have $c_i = d_i$ and $u_i = v_i - p_j$. Also, $v_i = b_i > p_j$ due to the second-price auction. Thus, we arrive at $u_i = v_i - p_j > 0$. On the other hand, if ED_i is not selected, its utility remains zero. Hence, $u_i \ge 0$ is satisfied. □

B. Incentive Compatibility

In the following, we prove our mechanism guarantees incentive compatibility. First, we consider the situation when an ED ED_i over-reports its computational cost, i.e., $d_i > c_i$, in Prop. 1. We use u_i' to denote the utility of untruthful reporting.

Proposition 1. *If $d_i > c_i$, then $u_i' \le u_i$.*

Proof. If ED_i over-reports its computational cost, it cannot finish the training process, resulting in its inability to complete the training goal. Therefore, ED_i will not receive the global model or reward r_i. In addition, it will be forced to quit the FL system after being detected, and the power it contributes can be neglected, resulting in $u_i' = 0$. Since our mechanism satisfies individual rationality, we conclude that $u_i' \le 0 \le u_i$. □

Then, we consider the situation when an ED ED_i untruthfully reports its model valuation, but truthfully reports its computational cost, i.e., $b_i \ne v_i, d_i = c_i$, in Prop. 2 to Prop. 5.

Proposition 2. *If $b_i > v_i$, $v_i > p_j$, and $d_i = c_i$, then $u_i' \le u_i$.*

Proof. Since ED_i is already selected by reporting v_i, having a higher bid b_i does not change its utility. Thus, $u_i' = u_i$ is proved. □

Proposition 3. *If $b_i \leq v_i$, $v_i > p_j$, and $d_i = c_i$, then $u_i' \leq u_i$.*

Proof. If $b_i \leq p_j$, ED_i cannot be selected. Thus, $u_i' = 0$. On the other hand, if $b_i > p_j$, having a lower bid does not change its utility, resulting in $u_i' = u_i$. Thus, we conclude $u_i' \leq u_i$. □

Proposition 4. *If $b_i > v_i$, $v_i \leq p_j = B_j^{N_j+1}$, and $d_i = c_i$, then $u_i' \leq u_i$.*

Proof. If $b_i \geq B_j^{N_j}$, the payment in group j becomes $p_j' = B_j^{N_j}$, resulting in $b_i > p_j' > p_j > v_i$. Thus, $u_i' = v_i - p_j' < 0 = u_i$. As for $b_i < B_j^{N_j}$, the ED is not selected, so its utility remains zero, i.e., $u_i' = u_i = 0$. Thus, $u_i' \leq u_i$. □

Proposition 5. *If $b_i \leq v_i$, $v_i \leq p_j$, and $d_i = c_i$, then $u_i' \leq u_i$.*

Proof. Since $p_j \geq v_i > b_i$, the ED with bid b_i cannot be selected. Hence, $u_i' = u_i = 0$. □

Next, we consider the situation when an ED ED_i under-reports its computational cost, i.e., $d_i < c_i$, in Prop. 6 to Prop. 8. We denote the reported computational type as j', i.e., $d_i = t_{j'}$.

Proposition 6. *If $v_i > p_j$, and $d_i < c_j$, then $u_i' \leq u_i$.*

Proof. If ED_i is selected in group j', the change of utility is $u_i' - u_i = -(p_j - p_{j'})$. According to the English auction, $p_{j'} \geq p_{j+1} \geq p_j$, so $u_i' \leq u_i$. On the other hand, if ED_i is not selected by the CS, $u_i' = 0 \leq u_i$. Therefore, we prove $u_i' \leq u_i$. □

Proposition 7. *If $b_i > v_i$, $v_i \leq p_j$, and $d_i < c_j$, then $u_i' \leq u_i$.*

Proof. If ED_i is selected in group j', $b_i > p_{j'}'$, where $p_{j'}'$ is the model price in group j' after ED_i joins the group. Nonetheless, from the English auction, we have $p_{j'}' \geq p_{j'} > p_j$. The utility of ED_i becomes negative:

$$u_i' = v_i - p_{j'}' < v_i - p_j \leq 0 = u_i. \tag{13}$$

On the other hand, if ED_i is not selected by the CS, its utility remains $u_i' = 0 \leq u_i$. □

Proposition 8. *If $b_i \leq v_i$, $v_i \leq p_j$, and $d_i < c_j$, then $u_i' \leq u_i$.*

Proof. Since $p_{j'} > p_j \geq b_i \geq b_i'$, ED_i is not selected by the CS, resulting in the same utility $u_i' = 0 = u_i$. □

From the above propositions, we have the following theorem.

Theorem 2 (Incentive Compatibility)**.** *The proposed auction mechanism is incentive-compatible. That is, an ED ED_i achieves maximum utility when it reports truthfully, i.e., $(c_i, v_i) = \arg\max_{(d_i, b_i)} u_i$.*

Proof. This is the direct result of Prop. 1 to Prop. 8. □

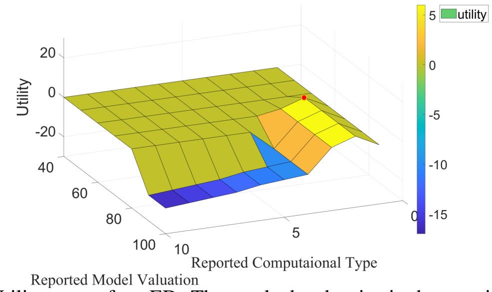
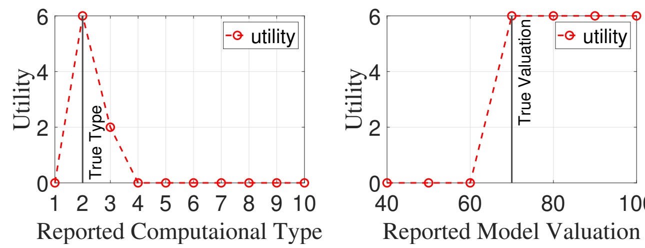


Fig. 3. Utility map of an ED. The marked red point is the true information of the ED.



(a) An ED's utility and the reported computational types.
(b) An ED's utility and the reported model valuation.

Fig. 4. Utility figures of an ED. The black line is the true information of the ED.

VI. Performance Evaluation

We first present the evaluation methodology in §VI-A. In §VI-B, we show that our proposed model incentivizes EDs to report truthfully. Then, in §VI-C, we compare the CS's cost using different mechanisms under distinct scenarios. Lastly, in §VI-D, we analyze the model accuracy and convergence rate of an actual FL image classification training.

A. Evaluation Methodology

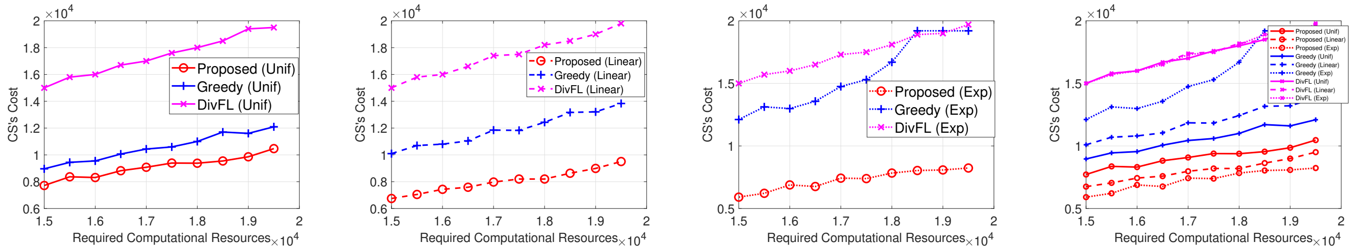
To show the superiority of the proposed model, we perform some numerical simulations. The experimental setting is described as follows. There are 5 types of EDs in the experiment. We consider three kinds of user distribution with different relative numbers of computational groups $K_1 : K_2 : K_3 : K_4 : K_5$.

1) *Uniform Distribution:* The relative number of EDs in each computational type is $1:1:1:1:1$.
2) *Linear Distribution:* The relative number of EDs in each computational type is $1:2:3:4:5$.
3) *Exponential Distribution:* The relative number of EDs in each computational type is $1:2:4:8:16$.

Also, we assume EDs' valuation is Gaussian distribution with the mean equal to 250 and the standard deviation equal to 90, i.e., $v_i \sim N(250, 3600)$. Finally, we set the computational resource difference between adjacent types as 100.

To evaluate the performance of our proposed model, we modify two mechanisms from [12] and [13] and compare the performance of the three models:

1) *Greedy Mechanism (Greedy) [12]:* This mechanism aims to take advantage of EDs' valuation to minimize the CS's cost. The CS selects sufficient EDs with computational power higher than a certain threshold with the second-price auction.

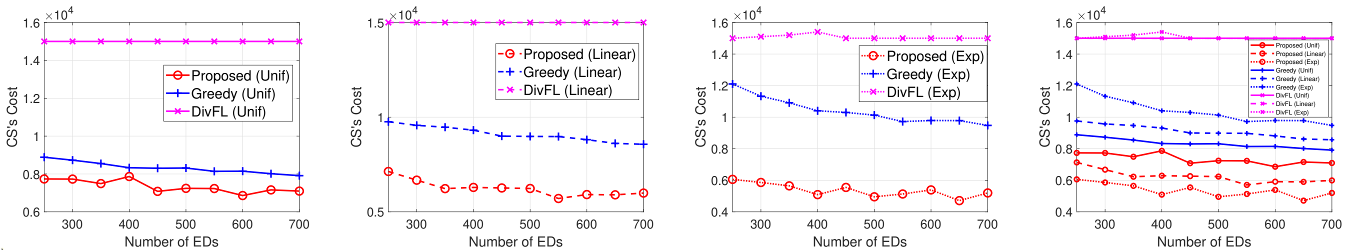


(a) Required computational resources under uniform distribution.

(b) Required computational resources under linear distribution.

(c) Required computational resources under exponential distribution.

(d) Required computational resources under three kinds of distributions.



(e) The number of EDs under uniform distribution.

(f) The number of EDs under linear distribution.

(g) The number of EDs under exponential distribution.

(h) The number of EDs under three kinds of distributions.

Fig. 5. The CS's cost with respect to different parameters and user distributions. (a)-(d): The CS's cost with respect to different required computational resources. (e)-(h): The CS's cost with respect to different numbers of EDs.

2) *Federated Averaging with Diverse Client Selection (DivFL) [13]:* The mechanism selects powerful devices to train a highly accurate global model. The CS sorts the EDs in descending order based on their computational power and then selects the EDs in order. The CS only pays the computational cost to EDs to compensate for their power consumption without considering valuation.

We evaluate the performance of the three mechanisms on the non-i.i.d. MNIST dataset using Federated Averaging (FedAvg) algorithm as the global aggregation method. We use Python3 with the TensorFlow library to simulate and conduct the experiments on Google Colab with Nvidia Tesla T4 GPU. In the experimental setting, a machine with more local iterations per global iteration represents a device with higher computational power.

B. Incentive Compatibility of Edge Devices

Fig. 3 and Fig. 4 demonstrate Thm. 2 that our mechanism guarantees incentive-compatibility of EDs. In the experimental setting, we consider ten computational types: $\textbf{Type} = \{t_1, t_2, ..., t_{10}\} = \{1100, 1000, ..., 200\}$. We consider a type-2 ED ED_i with computational power $c_i = 1000$ and model valuation $v_i = 70$. Fig. 3 presents a 3D utility map of the ED with different reported computational power d_i and model valuation b_i. Fig. 4(a) and Fig. 4(b) further demonstrate two views of the ED. Fig. 4(a) is the ED's utility with truthful model valuation and varying computational types, and Fig. 4(b) is the ED's utility with truthful computational type and varying model valuation.

Fig. 3 demonstrates that an ED acquires the highest utility when it reports its computational power and valuation truthfully. If the ED over-reports its computational power, it cannot fulfill the FL training task, resulting in a zero utility, as shown in Fig. 4(a) when the reported computation type is 1. Conversely, if the ED under-reports its computational capability, its utility decreases, as shown in Fig. 4(a) when the reported computation type exceeds 2. The reason is that our mechanism charges EDs with lower computational power a higher price on the model, so the ED is rewarded less when participating in the FL training, as shown in eq. (6).

As for truthfulness on model valuation, the second-price auction guarantees that revealing its true valuation is the dominant strategy, as shown in Fig. 4(b). Because the ED can participate in the FL training with truthful reporting, it will still be selected when over-reporting. Also, the model price is the same because the payment is determined by the highest losing bid as prescribed by the second-price auction. Therefore, the utility remains the same when over-reporting, as shown in Fig. 4(b) when the reported model valuation exceeds 70. Conversely, if the ED under-reports its model valuation, it may not be selected to join the FL training. Thus, its utility becomes zero, as shown in Fig. 4(a) when the reported model valuation is lower than 70. Moreover, when the ED reports truthfully, its utility is non-negative. This observation validates individual rationality as proved in Thm. 1.

C. Optimization Problem of the Central Server

The CS aims to minimize its cost when performing FL training, and we evaluate the expenditure with varying parameters in Fig. 5. Fig. 5(a) to Fig. 5(d) show that as the targeted global model accuracy threshold increases, the cost rises due to the requirement for more computational power. Besides, the proposed mechanism achieves the best performance since it directly solves a cost-minimization optimization problem, as shown in Alg. 1. As for *Greedy*, it obtains an inferior solution because the solution space is restricted by only choosing

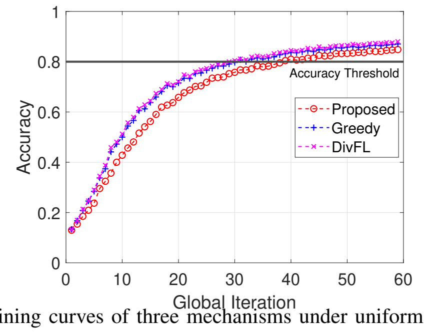


Fig. 6. Training curves of three mechanisms under uniform distribution.

users with computational power exceeding a certain threshold. Lastly, $DivFL$ solely concerns the computational power of devices, so it sacrifices costs to achieve the targeted accuracy and thus has the worst outcome.

Fig. 5(e) to Fig. 5(h) demonstrates that the cost decreases with the growth of the number of EDs K. Compared with $Greedy$, the proposed mechanism declines more drastically since the CS can find less costly participants with more available EDs. Also, $DivFL$ tends to select more powerful EDs, so the participating EDs are all from the highest computational group. Thus, the cost of the CS remains unchanged regardless of K.

Now, we compare the performance under different user distributions. First, Fig. 5(d) and Fig. 5(h) indicate that the proposed mechanism achieves the minimum cost under the exponential distribution. Since the number of EDs with high computational power is low, the model prices are low. Thus, the price constraints imposed on the remaining types are loose, so the solution space is enlarged. As for the other two distributions, the CS's cost is higher under the uniform distribution because it causes the prices in high computational groups to increase, resulting in the difficulty for the CS to select feasible EDs in lower types. However, in $Greedy$, the CS's cost is the highest under exponential distribution because the number of EDs exceeding a certain computational power threshold is the fewest. Hence, the uniform distribution achieves the best performance since the number of powerful EDs is more, which enables the CS to explore better solutions. Lastly, in $DivFL$, there are no significant differences between the three distributions in terms of the CS's cost. Since the CS aims to select powerful EDs, the selected EDs are mostly from the highest computational group no matter the distributions. Thus, the participating EDs are similar, and the CS's cost barely changes.

TABLE III

AN EXAMPLE OF SELECTED EDS UNDER UNIFORM DISTRIBUTION

Computational type	1	2	3	4	5
Total number	50	50	50	50	50
Proposed	4	3	9	2	3
Greedy	0	19	0	0	0
DivFL	16	0	0	1	0

D. Accuracy Curve of Federated Learning

Fig. 6 shows the training accuracy and convergence rate of the three mechanisms based on Tbl. III. All of them reach the accuracy threshold of 0.8 in similar numbers of iterations. Because $DivFL$ selects EDs with higher computational power, and $Greedy$ selects EDs with computational power beyond a certain threshold, the participating EDs are more powerful. Thus, both of them achieve a more accurate global model. However, this high accuracy is obtained through an increase in expenditure, as demonstrated in Fig. 5. Therefore, there is a trade-off between accuracy and cost.

VII. CONCLUSION

In this paper, we propose a user selection mechanism for the CS to perform FL training. The proposed scheme jointly considers EDs' computational power and valuation of the global model. With the devised rewarding rule based on auction theory, the mechanism incentivizes users to reveal their types truthfully and prevents the free-rider problem from happening. Then, leveraging a dynamic programming algorithm, we solve a cost-minimization optimization problem of the CS. Furthermore, the theoretical analysis demonstrates that our mechanism ensures individual rationality and incentive compatibility. Finally, extensive simulations show the effectiveness of the proposed method.

REFERENCES

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[2] P. Sun, H. Che, Z. Wang, Y. Wang, T. Wang, L. Wu, and H. Shao, "Pain-fl: Personalized privacy-preserving incentive for federated learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3805–3820, 2021.

[3] W. Y. B. Lim, Z. Xiong, C. Miao, D. Niyato, Q. Yang, C. Leung, and H. V. Poor, "Hierarchical incentive mechanism design for federated machine learning in mobile networks," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9575–9588, 2020.

[4] T. Liu, B. Di, P. An, and L. Song, "Privacy-preserving incentive mechanism design for federated cloud-edge learning," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 3, pp. 2588–2600, 2021.

[5] M. Wu, D. Ye, J. Ding, Y. Guo, R. Yu, and M. Pan, "Incentivizing differentially private federated learning: A multidimensional contract approach," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 639–10 651, 2021.

[6] J. Shen, N. Cheng, Z. Yin, and W. Xu, "Joint resource allocation and user scheduling scheme for federated learning," in *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, 2021, pp. 1–5.

[7] T. H. T. Le, N. H. Tran, Y. K. Tun, Z. Han, and C. S. Hong, "Auction based incentive design for efficient federated learning in cellular wireless networks," in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, 2020, pp. 1–6.

[8] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 700–10 714, 2019.

[9] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–7.

[10] Y. Fraboni, R. Vidal, and M. Lorenzi, "Free-rider attacks on model aggregation in federated learning," 2020. [Online]. Available: https://arxiv.org/abs/2006.11901

[11] C. Ma, J. Konečný, M. Jaggi, V. Smith, M. I. Jordan, P. Richtárik, and M. Takáč, "Distributed optimization with arbitrary local solvers," *Optimization Methods and Software*, vol. 32, no. 4, pp. 813–848, 2017.

[12] J. Xu, J. Xiang, and D. Yang, "Incentive mechanisms for time window dependent tasks in mobile crowdsensing," *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 6353–6364, 2015.

[13] R. Balakrishnan, T. Li, T. Zhou, N. Himayat, V. Smith, and J. Bilmes, "Diverse client selection for federated learning: Submodularity and convergence analysis," in *ICML 2021 International Workshop on Federated Learning for User Privacy and Data Confidentiality*, 2021.