2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)

A Truthful Procurement Auction for Incentivizing Heterogeneous Clients in Federated Learning

Ruiting Zhou*†, Jinlong Pang*, Zhibo Wang‡ , John C.S. Lui† and Zongpeng Li*
*School of Cyber Science and Engineering, Wuhan University, China
†Department of Computer Science and Engineering, The Chinese University of Hong Kong, China
‡Institute of Cyberspace Research, College of Computer Science and Technology, Zhejiang University, China
Email: {ruitingzhou, jinlongpang, zongpeng}@whu.edu.cn, zhibowang@zju.edu.cn, cslui@cse.cuhk.edu.hk

2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS) | 978-1-6654-4513-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICDCS51616.2021.00026

***Abstract*—Federated Learning (FL) is a new distributed machine learning (ML) approach which enables thousands of mobile devices to collaboratively train artificial intelligence (AI) models using local data without compromising user privacy. Although FL represents a promising computing paradigm, such training process can not be fully realized without an appropriate economic mechanism that incentivizes the participation of heterogeneous clients. This work targets social cost minimization, and studies the incentive mechanism design in FL through a procurement auction. Different from existing literature, we consider a practical scenario of FL where clients are selected and scheduled at different global iterations to guarantee the completion of the FL job, and capture the distinct feature of FL that the number of global iterations is determined by the local accuracy of all participants to balance between computation and communication. Our auction framework A_{FL} first decomposes the social cost minimization problem into a series of winner determination problems (WDPs) based on the number of global iterations. Then to solve each WDP, A_{FL} invokes a greedy algorithm to determine the winners, and a payment algorithm for computing remuneration to winners. Finally, A_{FL} returns the best solution among all WDPs. Theoretical analysis proves that A_{FL} is truthful, individual rational, computationally efficient, and achieves a near-optimal social cost. We further conduct large-scale simulation studies based on the real-world data. Simulation results show that A_{FL} can reduce the social cost by up to 75% compared with state-of-the-art algorithms.**

***Index Terms*—Federated Learning, Incentive Mechanism, Auction**

I. Introduction

The emergence of federated learning (FL) provides a new computing paradigm for artificial intelligence (AI) and its application. Traditional machine learning (ML) trains AI models centrally, which is privacy-intrusive, especially for mobile devices which contain owners' privacy-sensitive data [1], [2]. Compared to the centralized training process, FL is a decentralized training approach which distributes ML jobs to thousands of geo-distributed mobile devices (*a.k.a.* clients) [3], [4]. Mobile devices act as the computing nodes to collaboratively train a ML model using local data, without the risk of privacy disclosure. Major enterprises have launched FL projects. For example, Gboard, the Google Keyboard on Android, is adopting the FL process to make typing faster and easier. Mobile phones locally store users' typing preference every time when Gboard shows a suggested query. FL trains Gboard's query suggestion model using history on device to improve user experience in the next iteration [5].

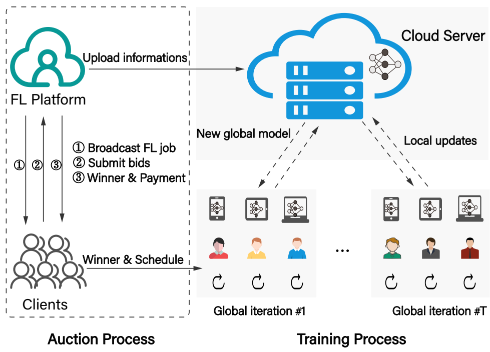

Fig. 1. An illustration of federated learning auction.

To fully realize the potential of FL in practice, two types of challenges need to be addressed: *technical* and *economic*. *First*, on the technical side, both computation and communication are the core challenges. The learning process in FL relies on frequent communication between the cloud server and mobile clients to update model, until the model converges [4]. To achieve a lower local accuracy[1], mobile clients spend more computation time to train their local models. Given a required global accuracy of the model, the number of communication rounds is proportional to the local accuracy achieved by all clients [7], [8]. Thus, how to balance between computation and communication time through the selection of clients with their local accuracy while guaranteeing fast convergence of the model becomes a vital problem. *Second*, on the economic side, incentive mechanism design is a necessary enabling technology for FL. The training process in FL is iterative, and needs thousands of clients to work collaboratively and continuously [4], [9]. However, it is not always practical to

The corresponding author is Zhibo Wang. This work is supported in part by the NSFC Grants (62072344 and U20A20177), Hubei Science Foundation (2020CFB195) and Compact Exponential Algorithm Project of Huawei (YBN2020035131). The work of John C.S. Lui is supported in part by the GRF 14200420.

[1]Here, local accuracy θ and global accuracy ε represent the relative gradient difference of loss function between two iterations [6], *i.e.*, $||\nabla F(w^{(t)})|| \leq \theta||\nabla F(w^{(t-1)})||$ and $||\nabla J(w^{(t)})|| \leq \varepsilon||\nabla J(w^{(t-1)})||$, where $F(w)$ and $J(w)$ are the loss function of local and global model, respectively.

2575-8411/21/$31.00 ©2021 IEEE
DOI 10.1109/ICDCS51616.2021.00026

assume that mobile clients are voluntary to fully participate in the complete training process, since mobile clients consume their own resources such as battery and GPU to calculate local model updates. Moreover, clients have their own schedule and may only participate in some particular time periods. Therefore, incentive mechanisms which pay rewards to compensate the cost of clients are the essential financial catalyst for making FL a reality.

To overcome aforementioned challenges, one needs to capture the distinct feature of FL while designing incentive mechanisms, *i.e.,* the relation between global accuracy and local accuracy among many mobile clients. Most existing research in FL focuses on the technical side, and investigates how to improve the training efficiency or reduce energy cost [6], [10]–[12]. There are only a few studies on the incentive mechanism design in FL. Most work in FL assume that the same set of clients can fully participate in the whole FL training process from beginning to end, and select energy-efficient clients to achieve fairness or utility maximization [13]–[16], which we will discuss in details in Sec. II. In this work, we propose a solution for incentivizing participation in FL as a procurement auction, A_{FL}. As shown in Fig. 1, the procurement auction consists of multiple sellers (mobile clients) and a single buyer (the cloud server). The goal of the auction design is social cost minimization while guaranteeing computational efficiency, truthfulness and individual rationality. Different from existing literature, we describe a richer and practical model of FL. We select and schedule clients at different global iterations to guarantee the completion of the FL job, and determine the number of global iterations (communication rounds) by the local accuracy of all participants to balance between computation and communication. We summarize our main contributions as follows.

First, we model and formulate the social cost minimization problem in FL as an integer linear program (ILP), and prove it is NP-hard. Different from existing literature that only determines winners (or selected mobile clients), we also need to decide how to schedule the participation of winners and the number of global iterations. To address the challenge introduced by the variation on the number of global iterations, our A_{FL} first calculates a range for the number of global iterations. Then for each fixed number of global iterations within the range, we formulate a winner determination problem (WDP). This way, A_{FL} decomposes the original optimization problem into a series of WDPs. To reduce computation, we exclude those bids which violate the communication round and computation time constraints, and further form a qualified bids set for each WDP. A_{FL} next invokes an algorithm A_{winner} to solve each WDP and finally announces the auction results, including the winning bids which generate the minimum social cost and the payment to winners.

Second, to determine and schedule winners for each WDP, we first reformulate each WDP to a new ILP by using compact exponential technique [17], which is a packing-type ILP with an exponential number of variables corresponding to valid schedules. This exponential-sized ILP and its dual are the foundation of algorithm design and analysis. We show that the new ILP can be solved by a greedy algorithm A_{winner}. A_{winner} iteratively selects a client with a schedule which can cover available global iterations at the lowest average cost, until there are enough participants in the winner set. Furthermore, a payment scheme based on the critical value rule [18], [19] is proposed as a subroutine of A_{winner} to ensure truthfulness and individual rationality.

Third, we conduct rigorous theoretical analysis to show that A_{FL} is truthful, individual rational and computationally efficient. Furthermore, we adopt the primal-dual theory to prove that A_{FL} achieves a good approximation ratio in social cost. We also evaluate the performance of A_{FL} through large-scale simulations based on real-world data. Numerical results demonstrate that A_{FL} always outperforms three benchmark algorithms. Moreover, A_{FL} produces a close-to-optimal social cost with a small ratio (< 1.3), and reduces the social cost by 10%, 40%, 75%, compared with Greedy [20], A_{online} [17] and FCFS [21], respectively.

In the rest of the paper, related work is given in Sec. II. The preliminary and system model of FL are introduced in Sec. III and Sec. IV. The procurement auction is presented and analyzed in Sec. V and Sec. VI. Sec. VII evaluates the performance and Sec. VIII concludes the paper.

II. Related Work

Federated Learning. In FL, majority of researchers focus on learning algorithm design with verifiable convergence analysis, but ignore practical challenges and economic incentives. Several papers studied FL based on practical scenarios. Smith *et al.* [10] consider practical systems challenges in FL such as straggler effort and random drops, and propose an efficient optimization method to address these issues. Tran *et al.* [6] focus on the trade-off between communication and computation cost, and obtain the optimal number of communication rounds, accuracy-level and minimum energy cost. Nishio *et al.* [11] present a FL protocol, which selects as many clients as possible to maximize training efficiency under the stragglers' effort caused by heterogeneous resources. Considering devices' computation capacity and limited networking resources, Wang *et al.* [12] design a control algorithm to dynamically adapt aggregation frequency. To speed up the convergence of FL, Wang *et al.* [22] select clients (mobile devices) with non-IID data through deep reinforcement learning (DRL), rather than selecting randomly like FedAvg [4]. From the perspective of energy efficiency, Zhan *et al.* [23] design an experience-driven method based on DRL to control devices' CPU-cycle frequency in a synchronized setting. For the above work, their aim is to improve the performance or achieve cost minimization, but neglect the problem of how to incentivize the participation of clients.

Incentive Mechanisms. In mobile crowdsourcing or crowdsensing system, there has been a long study on incentive mechanisms, especially using contract theory [24], [25], auction [26], [27] and game theory [28], [29]. However, there are only a few studies on the incentive mechanism design for implementing FL. Kang *et al.* [15] aim to address unreliable updates, and propose a contract-theoretic method to motivate clients that have a high reputation and high-qualified data to do the update. Ding *et al.* [30] analyze the incentive mechanism design for FL by using contract theory when considering multi-dimensional privacy information. Pandey *et al.* [14] develop a Stackelberg game-based framework to incentive clients to participate in training to achieve utility maximization. Toyoda *et al.* [13] design an incentive-aware mechanism for a blockchain-enabled FL platform by using contest theory to guarantee fairness. Zeng *et al.* [16] consider multi-dimensional resources and present an incentive framework based on game theory to achieve utility maximization. Le *et al.* [31] develop an auction-based incentive mechanism to stimulate clients in the wireless networks scenario. Above studies are all based on an impractical assumption that the same set of clients can fully participate in the complete process from beginning to end. In addition, they fixed the number of global iterations at the beginning. Different from the above literature, we select cost-efficient clients and schedule them at different global iterations. In addition, to balance between computation and communication, we also determine the number of global iterations that is affected by winners' local accuracy.

III. Preliminary of Federated Learning

The learning process in FL [8] relies on the iterative interaction between the cloud server and clients. In each global iteration: i) each selected client trains its local model on its local dataset for a number of local iterations to achieve a desirable local accuracy; ii) then each client returns its local model update to the server; iii) the server aggregates all local model updates and sends back the global model update to clients. The above process terminates until the global model accuracy reaches a predefined threshold. The upper bound on the number of global iterations (T_g) can be expressed according to the definitions in [8], [10], as follows:

$$T_g = \frac{\mathcal{O}(\log(\frac{1}{\varepsilon}))}{1-\theta_{\max}}, \quad (1)$$

where $\varepsilon \in [0,1]$ is the predefined global accuracy and $\theta_{\max} \in [0,1]$ is the maximum local accuracy among all clients. For the ease of presentation, $\mathcal{O}(\log(\frac{1}{\varepsilon}))$ is normalized to 1 when we consider a fixed global accuracy ε. Let $T_l(\theta_i)$ denote the number of local iterations for client i to achieve its local accuracy θ_i, which can be defined as follows [8]:

$$T_l(\theta_i) = \eta \cdot \log(\frac{1}{\theta_i}), \quad (2)$$

where η is a positive constant.

IV. System Model

A. System Overview

As shown in Fig. 1, we consider a typical scenario of FL which involves a cloud server and a set of clients (*e.g.,* smartphone or personal computer). On a FL platform, the cloud server first broadcasts the information of a FL job to all clients and specifies the maximum number of global iterations T. To incentivize clients, a procurement auction is applied where the server acts as the auctioneer and each client submits a bid for job participation. After collecting all bids, the server determines and pays the selected winners, and then schedules them to collaboratively execute the FL job. Let $\mathcal{X}$ denote the integer set $\{1, 2, \ldots, X\}$.

B. Auction Model

Bid Information. Let I denote the number of available clients. The cloud server needs K (where $K \ll I$) clients in each global training iteration. In practice, a client may not be able to fully participate in the entire training process due to many factors, *e.g.,* battery level or personal schedule. Furthermore, a client values different periods and local accuracy differently. Therefore, we assume that each client i submits up to J bids, and client i's j-th bid (B_{ij}) is expressed as a tuple:

$$B_{ij} = \{b_{ij}, \theta_{ij}, [a_{ij}, d_{ij}], c_{ij}\}_{\forall j \in \mathcal{J}}, \quad (3)$$

where b_{ij} is the "claimed" cost that user i wants to charge for the service. θ_{ij} is the local accuracy. $[a_{ij}, d_{ij}]$ is the available time period within $\mathcal{T}$, which starts and ends at a_{ij}-th and d_{ij}-th global iteration. In period $[a_{ij}, d_{ij}]$, client i can only participate c_{ij} number of global iterations, which is limited by its battery level, and calculated based on θ_{ij}. Let v_{ij} be the "true" cost of client i's j-th bid. Note that even each client submits J bids, only one bid can be accepted. This is because each client can only participate in one time period due to its battery capacity.

Decision Variables. After receiving all bids from the clients, the cloud server needs to make the following decisions: i) $T_g \in \{1, 2, ..., T\}$, the number of global iterations; ii) $x_{ij} \in \{0,1\}$, whether or not to accept client i's j-th bid, and if so, iii) $y_i(t) \in \{0,1\}$, whether or not to schedule client i at t-th global iteration, and iv) p_i, payment to client i.

Auction Preliminary. We next introduce some definitions in auction design. The cloud server's utility is:

$$u_{server} = V(\varepsilon) - \sum_{i \in \mathcal{I}} p_i, \quad (4)$$

where $V(\varepsilon)$ indicates how the server values the FL job with global accuracy ε. Client i's utility is:

$$u_i = \begin{cases} p_i - v_{ij}, & \text{if } x_{ij} = 1 \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

In general, clients are selfish and tend to maximize their own utilities. They may even lie about their true cost to get a higher utility. We instead focus on the utilities of the entire FL system, and target *social welfare maximization*. Therefore, it is necessary to elicit truthful bids from clients.

Definition 1. *(Truthful in bidding price): An auction is truthful in bidding price if and only if each client's utility is maximized when it bids with its true cost,* i.e., *for all* $b_{ij} \neq v_{ij}$, $u_i(v_{ij}) \geq u_i(b_{ij})$.

Definition 2. *(Individual Rationality): An auction is individual rational if each client's utility is non-negative,* i.e., $u_i(b_{ij}) \geq 0$.

Definition 3. *(Social Welfare, Social cost): The social welfare of the FL system is the aggregate utility of the cloud sever and clients, and equals* $V(\varepsilon) - \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} v_{ij} x_{ij}$*. When* $V(\varepsilon)$ *is a fixed value, one can ignore it. Note that in the optimizing process, maximizing social welfare is equivalent to minimizing the social cost,* i.e., $\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} v_{ij} x_{ij}$.

TABLE I
LIST OF NOTATIONS

I	# of clients	$\mathcal{S}$	winner set
p_i	payment to client i	ε	global accuracy
J	# of submitted bids		
T	maximum number of global iterations		
T_g	# of global iterations		
B_{ij}	bid information of client i's j-th bid		
b_{ij}	asking price of client i's j-th bid		
v_{ij}	true cost of client i's j-th bid		
θ_{ij}	local accuracy of client i's j-th bid		
a_{ij}	starting global iteration of the available time period of client i's j-th bid		
d_{ij}	ending global iteration of the available time period of client i's j-th bid		
c_{ij}	# of participation rounds of client i's j-th bid		
x_{ij}	whether or not to accept client i's j-th bid		
$y_i(t)$	whether or not to schedule client i at t-th iteration		
t_i^{cmp}	computation time required for client i to perform one local iteration		
t_i^{com}	communication time required for client i in one global iteration		
$t_{\max}$	duration of a single global iteration		
$T_l(\theta_{ij})$	# of local iterations with local accuracy θ_{ij}		
$\mathcal{L}_i$	the set of feasible schedules of client i		
ρ_{il}	client i's bidding price with schedule l		
z_{il}	whether or not to accept client i's schedule l		
$H_{\hat{T}_g}$	a harmonic number, which is equal to $\sum_{t=1}^{\hat{T}_g} \frac{1}{t}$		
ω	an auxiliary variable defined in line 18 of Alg. 2, equals $\max_{t \in \hat{\mathcal{T}}_g} \omega_t$		

C. Social Cost Minimization Problem

Problem Formulation. Under truthful bidding ($b_{ij} = v_{ij}$), the social cost minimization problem can be formulated into the following integer linear program (ILP):

$$\text{minimize} \quad \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} b_{ij} x_{ij} \tag{6}$$

subject to:

$$\sum_{i \in \mathcal{I}} y_i(t) \geq K, \quad \forall t \in \mathcal{T}_g, \tag{6a}$$

$$T_g \geq \frac{1}{1 - \theta_{ij} x_{ij}}, \quad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \tag{6b}$$

$$\sum_{t \in \mathcal{T}_g} y_i(t) = \sum_{j \in \mathcal{J}} c_{ij} x_{ij}, \quad \forall i \in \mathcal{I}, \tag{6c}$$

$$x_{ij} \cdot (T_l(\theta_{ij}) t_i^{cmp} + t_i^{com}) \leq t_{\max}, \quad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \tag{6d}$$

$$y_i(t) = 1 \ \text{ only if } x_{ij} = 1, \quad t \in [a_{ij}, d_{ij}], \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \tag{6e}$$

$$\sum_{j \in \mathcal{J}} x_{ij} \leq 1, \quad \forall i \in \mathcal{I}, \tag{6f}$$

$$x_{ij}, y_i(t) \in \{0, 1\}, \forall i \in \mathcal{I}, \quad \forall j \in \mathcal{J}, \forall t \in \mathcal{T}_g, \tag{6g}$$

$$T_g \in \{1, 2, ..., T\}. \tag{6h}$$

Constraint (6a) ensures that at least K clients are selected for each global iteration. Constraint (6b) calculates the number of global iterations, $\mathcal{T}_g$, according to the maximum local accuracy $\theta_{\max}$ among the winners. The number of participation rounds for each bid is implied by constraint (6c). In constraint (6d), $t_{\max}$ is the time limit for each global iteration. The time for client i's j-th bid to compute the local update in one global iteration consists of two parts: computation time $T_l(\theta_{ij}) t_i^{cmp}$ and communication time t_i^{com}. For simplicity, suppose that when a client registered at the FL platform, the cloud server can access its information. Therefore, t_i^{cmp} and t_i^{com} can be considered as constants. The relationship between $y_i(t)$ and x_{ij} is shown in constraint (6e). Constraint (6f) specifies that one can only accept at most one bid for each client.

Challenges. Note that even a simplified version of ILP (6) without constraints (6b), (6d) and (6h) is still NP-hard, which is equivalent to the set cover problem [32]. The challenge becomes more complicated when this problem involves a non-trivial variable T_g which relates to all winners. Moreover, two sets of binary variables determine clients' participation schedules, and eventually affect the total social cost.

V. AUCTION DESIGN

A. Overview of Auction Design

Algorithmic Idea. To solve the ILP (6), we present an auction framework, A_{FL}, to determine the winning bids and corresponding schedules to minimize the social cost with a bounded approximation ratio. The algorithmic idea of A_{FL} is shown in Fig. 2.

i. A_{FL} first computes the range of T_g according to clients' local accuracy. Then for each fixed $\hat{T}_g$ within the range, it formulates a winner determination problem (WDP). The input of each WDP is a qualified bids set which satisfies constraints (6b) and (6d). A_{FL} then decomposes ILP (6) into several WDPs. A_{FL} next calls A_{winner} to solve each WDP and finally outputs the winning bids which generate the minimum social cost.
ii. In Sec. V-B, we show how to determine the winners for each WDP. We first encode x_{ij} and $y_i(t)$ into one variable and reformulate each WDP into ILP (7). To solve it, we design an approximation algorithm A_{winner} based on a greedy strategy to select winning bids and schedule clients' participation.
iii. In Sec. V-C, we show how to charge winners for each WDP. We propose a payment algorithm $A_{payment}$ which is a subroutine of A_{winner} based on the critical value rule [18], [19].

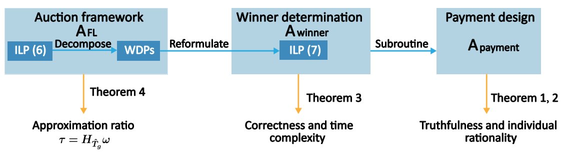

Fig. 2. Main idea of FL auction A_{FL}

Algorithm 1 FL Auction A_{FL}

Input: T, K, $B_{ij}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}$;
Output: T_g^*, $minicost$, $\mathcal{S}^*$;
1: Initialize $t_{ij} = T_l(\theta_{ij})t_i^{cmp} + t_i^{com}, \forall i, j$; $\mathcal{S}^* = \mathcal{P}^* = \mathcal{J}_{\hat{T}_g} = \varnothing$, $minicost = \infty$;
2: Find the minimum local accuracy $\theta_{\min}$ of all bids;
3: $T_0 = 1/(1 - \theta_{\min})$;
4: **for** $\hat{T}_g = T_0$ to T **do**
5: $\theta_{\max} = 1 - 1/\hat{T}_g$;
6: $\mathcal{J}_{\hat{T}_g} = \{(i,j)_{\forall i \in \mathcal{I}, \forall j \in \mathcal{J}} | \theta_{ij} \leq \theta_{\max} \& t_{ij} \leq t_{\max} \& a_{ij} + c_{ij} \leq \hat{T}_g\}$;
7: $(\mathcal{S}, \mathcal{P}, cost(\hat{T}_g))$=$A_{winner}(\mathcal{J}_{\hat{T}_g}, \hat{T}_g, K)$;
8: **if** $cost(\hat{T}_g) < minicost$ **then**
9: $T_g^* = \hat{T}_g$, $minicost = cost(\hat{T}_g)$, $\mathcal{S}^* = \mathcal{S}$, $\mathcal{P}^* = \mathcal{P}$;
10: **end if**
11: **end for**
12: **for all** $x_{ij} == 1, \forall x_{ij} \in \mathcal{S}^*$ **do**
13: Accept client i's j-th bid and schedule i according to $y_i(t) \in l_{ij}$; Pay $p_i \in \mathcal{P}^*$ to client i;
14: **end for**
15: return T_g^*, $minicost$, $\mathcal{S}^*$;

Auction Framework. Our FL auction A_{FL} is presented in Alg. 1. Let t_{ij} be the time for client i's j-th bid to compute and transmit the local update in one global iteration. Line 1 initializes all variables. Given the local accuracy of all bids, A_{FL} selects the minimum local accuracy to compute the initial value T_0 for T_g in lines 2-3. Then, A_{FL} enumerates the number of global iterations $\hat{T}_g$ from T_0 to T and computes the feasible maximum local accuracy $\theta_{\max}$ for different $\hat{T}_g$ according to Eq. (1) in lines 4-5. Next, $\theta_{\max}$, $t_{\max}$ and $\hat{T}_g$ are used to get a set of qualified bids $\mathcal{J}_{\hat{T}_g}$ for A_{winner} (line 6). In line 7, algorithm A_{winner} returns winners' set $\mathcal{S}$, the payment set $\mathcal{P}$ and the corresponding social cost $cost(\hat{T}_g)$. A_{FL} then compares the resulting costs at different $\hat{T}_g$, and records the best solution which achieves the minimum social cost (lines 8-10). Finally, A_{FL} announces the auction result in lines 12-15.

B. Winner Determination

To solve each WDP and determine the winners, we next present the algorithmic design of A_{winner}.

1) Problem Reformulation

For each fixed $\hat{T}_g$, there is a WDP with a qualified bids set $\mathcal{J}_{\hat{T}_g}$. Recall that each WDP is equivalent to a simplified version of ILP (6) without constraints (6b), (6d) and (6h). Note that two decision variables x_{ij} and $y_i(t)$ have a natural precedence correlation. To address this problem, we apply compact exponential technique [17] to reformulate the WDP to the following ILP (7) by encoding x_{ij} and $y_i(t)$ into a new decision variable z_{il}.

$$\text{minimize} \quad \sum_{i \in \mathcal{I}} \sum_{l \in \mathcal{L}_i} \rho_{il} z_{il} \tag{7}$$

subject to:

$$\sum_{i \in \mathcal{I}} \sum_{l: y_i(t) \in l} z_{il} \geq K, \quad \forall t \in \hat{\mathcal{T}}_g, \tag{7a}$$

$$\sum_{l \in \mathcal{L}_i} z_{il} \leq 1, \quad \forall i \in \mathcal{I}, \tag{7b}$$

$$z_{il} \in \{0, 1\}, \quad \forall l \in \mathcal{L}_i, \forall i \in \mathcal{I}. \tag{7c}$$

In the above ILP (7), $\mathcal{L}_i$ is the schedule set of client i. A feasible schedule l is a vector $l = \{\{x_{ij}\}_{\forall (i,j) \in \mathcal{J}_{\hat{T}_g}}, \{y_i(t)\}_{\forall i,t}\}$ which satisfies constraints (6c) and (6e). The value of ρ_{il} equals the corresponding b_{ij} based on l. z_{il} denotes whether or not to select client i's schedule l. Note that the number of feasible schedules z_{il} for client i is exponential, due to combinatorial property of variables x_{ij} and $y_i(t)$ (*i.e.,* the number of feasible schedules for client i is up to $\sum_j \binom{d_{ij} - a_{ij}}{c_{ij}}$). Constraint (7a) is equivalent to constraint (6a). And constraint (7b) ensures that one client can be selected according to at most one schedule. It is clear that a feasible solution to ILP (7) is also a feasible solution to the WDP, and vice versa, with the same objective value.

Dual Problem. In order to analyze the performance of A_{winner}, we formulate the dual problem of ILP (7) by relaxing the integrality constraint (7c) into $0 \leq z_{il} \leq 1$, and introduce dual variables $g(t)$, q_i and λ_{il} to constraint (7a), (7b) and $z_{il} \leq 1$, respectively. Then, the dual problem of relaxed ILP (7) is:

$$\text{maximize} \quad \sum_{t \in \hat{\mathcal{T}}_g} K g(t) - \sum_{i \in \mathcal{I}} \sum_{l \in \mathcal{L}_i} \lambda_{il} - \sum_{i \in \mathcal{I}} q_i \tag{8}$$

$$\text{subject to:} \sum_{t: y_i(t) \in l} g(t) - \lambda_{il} - q_i \leq \rho_{il}, \quad \forall l \in \mathcal{L}_i, \forall i \in \mathcal{I}, \tag{8a}$$

$$g(t), \lambda_{il}, q_i \geq 0, \quad \forall t \in \hat{\mathcal{T}}_g, \forall l \in \mathcal{L}_i, \forall i \in \mathcal{I}. \tag{8b}$$

2) Winner Determination and Scheduling

Main Idea. To get a feasible solution of the exponential-sized ILP (7), we design an efficient algorithm A_{winner} which selects schedules iteratively based on a greedy strategy. We say the t-th global iteration is *available* if the number of selected clients in the t-th global iteration is less than K. A_{winner} starts with an empty set. In each iteration, A_{winner} selects a client with a schedule which can cover *available* global iterations at the lowest *average cost*. Then A_{winner} adds the selected client with its corresponding schedule to the winner set. This process terminates until there are enough participants in the winner set.

Average Cost. Let $\mathcal{S} = \{(i_1, l_1), (i_2, l_2), ...\}$ be a set where (i_1, l_1) is client i_1's l_1-th schedule. $\gamma_t^{\mathcal{S}} = \sum_{(i,l) \in \mathcal{S}: y_i(t) \in l} 1$ denotes the number of clients which are scheduled at the t-th global iteration in set $\mathcal{S}$. The utility of set $\mathcal{S}$ is its valid contribution, which is defined as $R(\mathcal{S}) = \sum_{t \in \hat{\mathcal{T}}_g} \min(\gamma_t^{\mathcal{S}}, K)$. The increased utility of adding client i's l-th schedule to $\mathcal{S}$ is:

$$\begin{aligned} R_{il}(\mathcal{S}) &= R(\mathcal{S} \bigcup (i,l)) - R(\mathcal{S}) \\ &= \sum_{t \in \hat{\mathcal{T}}_g} (\min(\gamma_t^{\mathcal{S} \cup (i,l)}, K) - \min(\gamma_t^{\mathcal{S}}, K)) \end{aligned} \tag{9}$$

The *average cost* of schedule l is $\frac{\rho_{il}}{R_{il}(\mathcal{S})}$. At the beginning, $\mathcal{S}$ is an empty set. In each iteration, the schedule with the minimum average cost is added to set $\mathcal{S}$, until there are enough participants. Although the number of feasible schedules for client i's j-th bid is up to $\binom{d_{ij}-a_{ij}}{c_{ij}}$, for each bid, we only need to consider one representative schedule which generates the maximum utility. Let l_{ij} denote the representative schedule for client i's j-th bid. l_{ij} consists of c_{ij} global iterations which have the smallest $\gamma_t^{\mathcal{S}}$ within the time period $[a_{ij}, d_{ij}]$.

Algorithm 2 Winner Determination Algorithm A_{winner}

Input: $\mathcal{J}_{\hat{T}_g}$, $\hat{T}_g$, K;
Output: $\mathcal{S}$, $\mathcal{P}$, $cost$;
1: Initialize $\mathcal{S} = \mathcal{P} = \mathcal{C} = \mathcal{G} = \varnothing$, $cost = 0$, $\gamma_t^{\mathcal{S}} = 0$, $\forall t$;
2: **while** $R(\mathcal{S}) < K\hat{T}_g$ **do**
3: Sort $\hat{\mathcal{T}}_g$ global iterations according to $\gamma_t^{\mathcal{S}}$ in nondecreasing order; Save the order as $\tilde{T}$;
4: **for** all bids in $\mathcal{J}_{\hat{T}_g}$ **do**
5: Select the top c_{ij} global iterations in $\tilde{T}$ and within time period $[a_{ij}, d_{ij}]$ to form the representative schedule l_{ij};
6: Compute $R_{il_{ij}}(\mathcal{S})$; Save/update schedule (i, l_{ij}) in sets $\mathcal{C}$ and $\mathcal{G}$;
7: **end for**
8: $(i^*, l^*) = \arg\min_{(i,l_{ij})\in\mathcal{C}} \dfrac{\rho_{il_{ij}}}{R_{il_{ij}}(\mathcal{S})}$;
9: $z_{i^*l^*} = 1; \phi(t, l^{i^*}) = \dfrac{\rho_{i^*l^*}}{R_{i^*l^*}(\mathcal{S})}, \forall t \in \mathcal{F}_{i^*l^*}$;
10: $p_{i^*} = A_{payment}(\mathcal{C}, (i^*, l^*), R_{il_{ij}}(\mathcal{S}))$;
11: $(i^\#, l^\#) = \arg\min_{(i,l_{ij})\in\mathcal{G}} \dfrac{\rho_{il_{ij}}}{R_{il_{ij}}(\mathcal{S})}$;
12: $\phi(t, l^{i^\#})' = \dfrac{\rho_{i^\# l^\#}}{R_{i^\# l^\#}(\mathcal{S})}, \forall t \in \mathcal{F}_{i^\# l^\#}$;
13: $\mathcal{C} = \mathcal{C}\backslash(\bigcup_l (i^*, l))$;
14: $\mathcal{S} = \mathcal{S}\bigcup(i^*, l^*)$; $\mathcal{G} = \mathcal{G}\backslash(i^*, l^*)$;
15: **end while**
16: $\psi_{\max}^t = \max_{t\in[a_{ij},d_{ij}]}\{\rho_{il_{ij}}\}, \forall t \in \hat{\mathcal{T}}_g$;
17: $\psi_{\min}^t = \min_l\{\{\phi(t,l)\}\bigcup\{\phi(t,l^{i^\#})'\}\}, \forall t \in \hat{\mathcal{T}}_g$;
18: $\omega_t = \psi_{\max}^t/\psi_{\min}^t, \forall t \in \hat{\mathcal{T}}_g$; $\omega = \max_{t\in\hat{\mathcal{T}}_g} \omega_t$;
19: $\eta_\phi(t) = \max_l\{\phi(t,l)\}$, $g(t) = \eta_\phi(t)/(H_{\hat{T}_g}\omega), \forall t \in \hat{\mathcal{T}}_g$;
20: **for** all $z_{il} == 1$ **do**
21: $\lambda_{il_{ij}} = \sum_{t:t\in\mathcal{F}_{il}} (\eta_\phi(t) - \phi(t,l))/(H_{\hat{T}_g}\omega)$;
22: Save p_i to $\mathcal{P}$; $cost = cost + \rho_{il_{ij}}$;
23: **end for**
24: Return $\mathcal{S}$, $\mathcal{P}$, $cost$;

Algorithm Details. The winner determination algorithm A_{winner} is shown in Alg. 2. Here $\mathcal{C}$ is a candidate set which records representative schedules for selection during each iteration. $\mathcal{G}$ is a grand set which records unselected representative schedules. Let $\mathcal{F}_{il}$ be the set that stores the current *available* global iterations in client i's l-th schedule. Line 1 initializes sets and variables. Note that the default value of all primal and dual variables is zero. In lines 2-15, the *while* loop selects schedules iteratively until all global iterations have K participants. Lines 3-7 compute the representative schedule l_{ij} for each bid. Line 8 selects the schedule (i^*, l^*) with the smallest average cost. Then, the corresponding variable $z_{i^*l^*}$ is updated to 1, and its average cost $\phi(t, l^{i^*})$ is recorded in line 9. Line 10 calls the subroutine $A_{payment}$ to calculate the payment for each new selected schedule (i^*, l^*). Lines 11-12 record auxiliary variables for updating dual variables. Lines 13-14 update sets $\mathcal{C}$, $\mathcal{S}$ and $\mathcal{G}$. To satisfy constraint (8b), set $\mathcal{C}$ removes all remaining schedules of client i^*. Then, the winner set $\mathcal{S}$ adds the new selected schedule (i^*, l^*), and set $\mathcal{G}$ excludes it. In order to bound the approximation ratio, A_{winner} updates dual variables. The value of dual variable $g(t)$ is calculated in lines 16-19, where $H_{\hat{T}_g} = \sum_{t=1}^{\hat{T}_g} \frac{1}{t}$ is a harmonic number. Finally, lines 20-23 compute value of dual variable $\lambda_{il_{ij}}$ for each selected schedule and save winners' information.

Example. We illustrate the process of A_{winner} by a simple example. Suppose $\hat{T}_g = 3$ and $K = 1$. For any client $i \in \mathcal{I}$, it only submits one bid with a form of $B_i(b_i, [a_i, d_i], c_i)$. There are three qualified bids in set $\mathcal{J}_3$: $B_1(\$2, [1, 2], 1)$, $B_2(\$6, [2, 3], 2)$, $B_3(\$5, [1, 3], 2)$. First, A_{winner} initializes $R(\mathcal{S}) = 0$ and $\gamma_t^{\mathcal{S}} = 0, \forall t \in [1, 2, 3]$.

- In the first iteration, the candidate set $\mathcal{C}$ includes three representative schedules: $l_1 = \{1\}, l_2 = \{2, 3\}, l_3 = \{1, 2\}$. Since $R(\mathcal{S}) < 3$, A_{winner} computes $\frac{\rho_1}{R_1(\mathcal{S})} = 2$, $\frac{\rho_2}{R_2(\mathcal{S})} = 3$, $\frac{\rho_3}{R_3(\mathcal{S})} = 2.5$. $(1, l_1)$ is selected since it has the minimum average cost. Its corresponding payment is calculated as $p_1 = R_1(\mathcal{S}) \cdot \frac{\rho_3}{R_3(\mathcal{S})} = 2.5$. Next, A_{winner} updates $\mathcal{S} = \{(1, l_1)\}$ and $R(\mathcal{S}) = 1$, and then removes l_1 from set $\mathcal{C}$.
- In the second iteration, the candidate set $\mathcal{C}$ contains two representative schedules: $l_2 = \{2, 3\}$, $l_3 = \{2, 3\}$. Because $R(\mathcal{S}) < 3$, A_{winner} computes $\frac{\rho_2}{R_2(\mathcal{S})} = 3$, $\frac{\rho_3}{R_3(\mathcal{S})} = 2.5$. $(3, l_3)$ is selected. The corresponding payment is $p_3 = R_3(\mathcal{S}) \cdot \frac{\rho_2}{R_2(\mathcal{S})} = 6$. So $\mathcal{S} = \{(1, l_1), (3, l_3)\}$ and $R(\mathcal{S}) = 3$. l_3 is removed from set $\mathcal{C}$. The while loop in A_{winner} terminates since $R(\mathcal{S}) = K\hat{T}_g = 3$ now.

C. Payment Design

Algorithm 3 Payment Algorithm $A_{payment}$

Input: $\mathcal{C}$, (i^*, l^*), $R_{il_{ij}}(\mathcal{S})$;
Output: p_{i^*};
1: $(i', l') = \arg\min_{(i,l_{ij})\in\mathcal{C}:(i,l_{ij})\neq(i^*,l^*)} \frac{\rho_{il_{ij}}}{R_{il_{ij}}(\mathcal{S})}$;
2: $p_{i^*} = R_{i^*l^*}(\mathcal{S}) \cdot \dfrac{\rho_{i'l'}}{R_{i'l'}(\mathcal{S})}$;
3: return p_{i^*};

We next discuss how to calculate the payment for winners. The basic idea is to calculate the payment based on the critical bid, *i.e.,* the schedule which has the second smallest average cost. (Please see Theorem 1 for details). $A_{payment}$ is shown in Alg. 3. Line 1 finds the critical bid (i', l') and line 2 calculates the payment for each new selected schedule (i^*, l^*) based on the critical value rule [18], [19].

VI. Theoretical Analysis

In this section, we analyze the property of A_{FL} in terms of truthfulness, individual rationality, correctness, time complexity and approximation ratio.

A. Truthfulness and Individual Rationality

Lemma 1. *A_{FL} is schedule-monotonic,* i.e., *$\forall i \in \mathcal{I}, \forall l, \tilde{l} \in \mathcal{L}_i$, if $\rho_{i\tilde{l}} < \rho_{il}$ and $R_{i\tilde{l}}(\mathcal{S}) = R_{il}(\mathcal{S})$, $z_{il} = 1$ implies $z_{i\tilde{l}} = 1$.*

Proof: Please see Appendix A. □

Lemma 2. *The payment to all selected schedules are critical,* i.e., *suppose that a selected schedule ($z_{i^*l^*} = 1$) has a bidding price $\tilde{\rho}_{i^*l^*}$ ($\neq \rho_{i^*l^*}$), then this schedule will win if $\tilde{\rho}_{i^*l^*} \leq p_{i^*}$, and will fail otherwise.*

Proof: Please see Appendix B. □

Theorem 1. *A_{FL} is a* truthful *auction.*

Proof: (*Truthfulness in bidding price b_{ij}*): The Myerson's theorem [18], [19] shows that a reverse auction is truthful in bidding price if the following conditions are satisfied: (i) the result of auction (z_{il}) is monotonically non-decreasing with the decrease of bidding price ρ_{il}; and (ii) the payment of each selected schedule is calculated based on the critical value. Combining Lemma 1 and 2, we finish this part of proof.

(*Truthfulness in local accuracy θ_{ij}*): We first prove that client i will not report a smaller local accuracy θ'_{ij} than its true local accuracy θ_{ij}. A smaller local accuracy leads to a longer computation time, which will risk failing to satisfy the time limitation of one single communication round, *i.e.*, constraint (6d). Even if client i submits a smaller local accuracy and it is selected by the FL platform, client i cannot achieve the local accuracy that it claimed. Therefore, the FL platform will refuse to pay when this happened.

If client i bids with a larger local accuracy, it would reduce the probability of being accepted by the FL platform. This is because the larger local accuracy may not satisfy the accuracy requirement of FL job, *i.e.*, constraint (6b). Thus, clients will not misreport the local accuracy of their bids.

(*Truthfulness in available time period $[a_{ij}, d_{ij}]$ and the number of participation rounds c_{ij}*): If client i reports a longer available time period and it is accepted by the FL platform, client i may not be able to participate in some rounds due to its actual schedule. Hence, the FL platform will refuse to pay when this happened. If client i claims a shorter available time period, the average cost of that bid may increase and it will further reduce the probability of acceptance. The reason is that the average cost is calculated based on the increased utility of adding one schedule, *i.e.*, $R_{il}(\mathcal{S})$, and the value of $R_{il}(\mathcal{S})$ may reduce because a shorter available time period will narrow down the range that the schedule l can select. In summary, there is no incentive to misreport the available time period.

Similarly, clients who submit a larger number of participation rounds would not get the payment from the FL platform, since they actually can not provide the service as they claimed. On the other hand, a smaller number of participation rounds submitted by clients results in a higher average cost, reducing the likelihood of acceptance. Therefore, clients will not misreport the number of participation rounds.

In conclusion, A_{FL} is a truthful auction. □

Theorem 2. *A_{FL} achieves individual rationality.*

Proof: The payment of selected schedule (i^*, l^*) is based on the critical value. It is clear that (i', l')'s average cost will be no less than (i^*, l^*)'s, *i.e.,* $\frac{\rho_{i^*l^*}}{R_{i^*l^*}(\mathcal{S})} \leq \frac{\rho_{i'l'}}{R_{i'l'}(\mathcal{S})}$. Then, we have $\rho_{i^*l^*} \leq R_{i^*l^*}(\mathcal{S}) \cdot \frac{\rho_{i'l'}}{R_{i'l'}(\mathcal{S})} = p_{i^*}$. Furthermore, Theorem 1 ensures truthfulness, *i.e.,* $v_{i^*l^*} = \rho_{i^*l^*}$. Therefore, each client's utility, $u_{i^*l^*} = p_{i^*} - v_{i^*l^*} \geq 0$, is always non-negative. □

B. Correctness and Time Complexity

Lemma 3. *A_{winner} produces a feasible solution to ILP (7) and LP (8).*

Proof: Please see Appendix C. □

Lemma 4. *The running time of A_{winner} is $O(I\hat{T}_g(\log(\hat{T}_g) + IJ))$.*

Proof: Please see Appendix D. □

Theorem 3. *A_{FL} produces a feasible solution to ILP (6) in polynomial time.*

Proof: We first prove the time complexity of A_{Fl}. Lines 1-2, and 12-14 of A_{FL} can be finished within $O(IJ)$ steps. Line 7 in Alg. 1 needs to search all bids, which takes $O(IJ)$ steps. According to Lemma 4, we know that the time complexity of A_{winner} is $O(I\hat{T}_g(\log(\hat{T}_g) + IJ))$. Therefore, the *for* loop in Alg. 1 which includes Alg. 2 can be done within $O(IT^2(\log(T)+IJ))$ steps. In summary, the time complexity of A_{FL} is $O(IT^2(\log(T) + IJ))$.

Next, we discuss the correctness of A_{FL}. Constraint (6h) holds since we enumerate $\hat{T}_g$ in the *for* loop (lines 4-11). Before solving ILP (7), we pick those bids which satisfy constraint (6b) and (6d) at different fixed $\hat{T}_g$ and form a qualified set $\mathcal{J}_{\hat{T}_g}$ for ILP (7) (line 6). Therefore, constraint (6b) and (6d) hold. Finally, the correctness of A_{FL} can be guaranteed by combining Lemma 3.

In conclusion, A_{FL} produces a feasible solution to ILP (6) in polynomial time. □

C. Approximation Ratio

Here, we prove that the theoretical approximation ratio of A_{FL} is $H_{\hat{T}_g^*}\omega$. Furthermore, the value of $H_{\hat{T}_g^*}\omega$ is around 1.1, which will be verified in our simulations in Sec. VII.

Definition 4. *(Approximation Ratio): The approximation ratio of an algorithm A for a minimization problem is the upper bound ratio of the objective value of the solution found by A over the objective value returned by an optimal algorithm.*

Lemma 5. *Let P and D be the objective values of the primal problem (7) and the dual problem (8) returned by A_{winner}. $\tau D \geq P$ with $\tau = H_{\hat{T}_g}\omega$, where $H_{\hat{T}_g} = \sum_{t=1}^{\hat{T}_g} \frac{1}{t}$ and ω is defined in line 18 of A_{winner}. A_{winner} is a τ-approximation algorithm.*

Proof. Please see Appendix E. □

Theorem 4. *The approximation ratio of A_{FL} is τ^* where $\tau^* = H_{\hat{T}_g^*}\omega$.*

Proof: Suppose that the optimal number of global iterations is $\hat{T}_g^*$. Let C^* denote the optimal social cost. Let $C^{\#}$ be the social cost returned by A_{FL}. C denotes the cost of corresponding solution returned by A_{winner} under fixed $\hat{T}_g^*$. Then, we have $C^{\#} \leq C$. Hence, $C^{\#}/C^* \leq C/C^* \leq H_{\hat{T}_g^*}\omega = \tau^*$. Therefore, we can conclude that A_{FL} in Alg. 1 is a τ^*-approximation algorithm. □

VII. Performance Evaluation

A. Evaluation Setup

System Settings. For fair comparison, we follow the similar setting in [4], [6]. By default, there are 1000 (I) clients and each client submits 5 (J) bids [4]. Assume that the maximum number of global iterations equals 50 and each global iteration needs 20 clients to train collaboratively, *i.e.*, $T = 50$ and $K = 20$ [4], [22]. t_i^{cmp} and t_i^{com} are randomly picked within the range of $[5, 10]$ and $[10, 15]$, respectively [6]. The local accuracy θ_{ij} of all bids are uniformly distributed in $[0.3, 0.8]$ [10], [8]. We calculate the number of local iterations $T_l(\theta_{ij})$ according to a simplified equation: $T_l(\theta_{ij}) = \lceil 10(1 - \theta_{ij}) \rceil$ [4]. In our simulations, we do not consider the case that two available time periods overlap since they can be considered as one time period from the perspective of clients. Therefore, we select $2J$ non-repeated random numbers within the range $[1, T]$, and sort them in non-decreasing order to form J available time periods. The starting time (a_{ij}) and the ending time (d_{ij}) of each time period equal two adjacent random numbers in the order, respectively. The number of participation rounds (c_{ij}) is randomly generated within the range $[1,\ d_{ij} - a_{ij}]$. Finally, the claimed cost of bids are uniformly distributed in the range of $[10, 50]$. The default value of $t_{\max}$ is set to 60 [6].

Benchmark Algorithms. To evaluate the performance of A_{FL}, we compare it with three benchmark algorithms:

- *FCFS* [21]: first-come, first-served algorithm, which selects clients according to the non-decreasing order of each bid's start time, *i.e.,* a_{ij}.
- *Greedy* [20]: a greedy algorithm which iteratively selects bids with a lower average cost of one global iteration, which is calculated as b_{ij}/c_{ij}.
- A_{online} [17]: A_{online} first calculates the unit payment of each global iteration based on a payment function. Then it selects the client with larger utility and schedule the client according to the best schedule that maximizes its utility.

B. Evaluation Results

Performance ratio. The *performance ratio* of an algorithm A for a minimization problem = the objective value of the solution found by A / the objective value returned by an optimal algorithm. We first study the performance ratio of A_{winner}. To ensure there are enough bids, we assume that all bids can satisfy constraint (6b) and (6d). Fig. 3 depicts the trend of A_{winner}'s performance ratio under different number of global iterations ($\hat{T}_g$) and bids per client (J). We can observe that A_{winner} has a small ratio (< 1.3) and the ratio becomes smaller as $\hat{T}_g$ decreases and J increases. This result is coincident with the theoretical analysis in Lemma 5 that $\hat{T}_g$ determines $H_{\hat{T}_g}$. In addition, the increase of J will decrease the length of time period (*i.e.* $|d_{ij} - a_{ij}|$) since we select $2J$ non-repeated random numbers to form J time periods for each client. The value of $\psi_{\min}^t$ increases when the length of time period decreases. Therefore, parameter ω eventually decreases. Next, we also study the impact of the number of clients (I) and bids per client (J) on performance ratio of A_{FL}. Fig. 4 shows performance ratios of all algorithms under different I and J. We can observe that the performance ratio of A_{FL} is the smallest and not affected greatly by the change of I and J. One reason is that A_{FL} can find the best solution by enumerating the number of global iterations T_g from T_0 to T.

Social Cost. Fig. 5 and Fig. 6 further plot the social cost under different number of clients (I) and bids per client (J). In both Fig. 5 and Fig. 6, we can see that A_{FL} outperforms three benchmark algorithms. Furthermore, the social cost of A_{FL} in Fig. 5 will decrease slightly with the increase of I since there is higher probability to select bids with lower average cost. On the contrary, the cost of all algorithms increase when the value of J increases in Fig. 6. Since the length of time periods will decrease if the number of bids per clients (J) increases, the average cost gets higher when the claimed cost remains the same. As a result, the total cost of all algorithms become larger. Fig. 7 illustrates the social cost at different fixed $\hat{T}_g$ within the range $[T_0, T]$. From Fig. 7, we can find that A_{FL} still generates the lowest social cost. Moreover, we can see that all algorithms except FCFS achieve the smallest cost when $\hat{T}_g = 26$. This is because the computation cost occupies a large proportion of the total cost at the early stage and it drops with the increase of $\hat{T}_g$. When the number of global iterations $\hat{T}_g$ is close to 26, all algorithms except FCFS find the balance point between the computation and communication cost. After that, the total cost grows gradually with the increase of $\hat{T}_g$, since the communication cost dominates the total cost.

Running Time. In Fig. 8, we investigate the running time of A_{FL} and A_{online} under different number of clients, measured by *tic* and *toc* function in MATLAB. We evaluate the running time on our laptop with an Intel Core i7-4270HQ and 8-GB RAM memory. To minimize the error, we use the average result of five tests. We observe that the running time of A_{FL} is not affected greatly by the number of clients. Furthermore, A_{FL} can finish within 60 seconds even with a large input scale ($I = 9000$, $J = 10$), and runs fast than A_{online}.

Individual Rationality. Finally, Fig. 9 compares payment and claimed cost of all winners selected by A_{FL}. We can see that the payment for the winner is always larger than its corresponding claimed cost. Therefore, one can observe that the property of individual rationality can be satisfied and each winner's utility is non-negative.

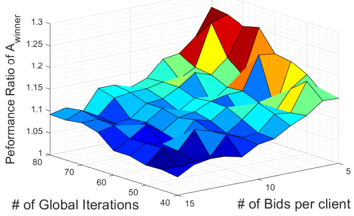

Fig. 3. Performance ratio of A_{winner}.

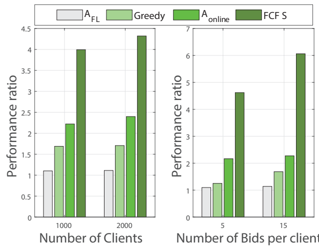

Fig. 4. Performance ratio of A_{FL}.

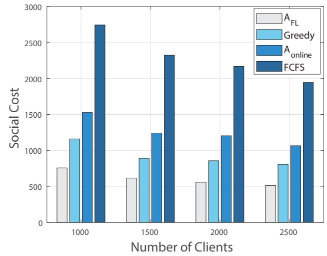

Fig. 5. Social cost under different number of clients.

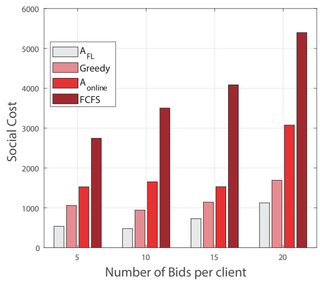

Fig. 6. Social cost under different number of bids per client.

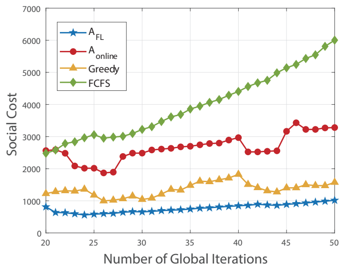

Fig. 7. Social cost at different fixed $\hat{T}_g$.

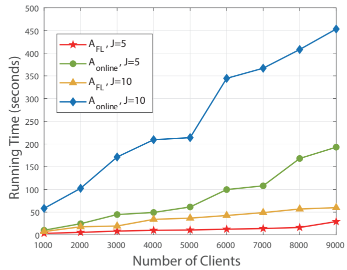

Fig. 8. Running time of A_{FL} and A_{online}.

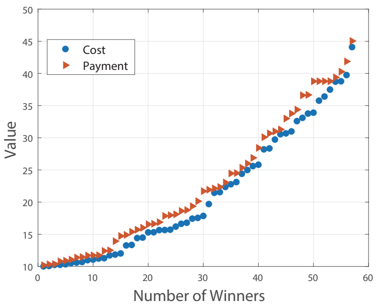

Fig. 9. Payment versus claimed cost of winning bid.

VIII. Conclusions

Federated learning (FL) is proving a remarkable privacy-preserving approach to train machine learning jobs without exchanging data samples. Besides technical challenges that are being studied in the literature, economic incentives of such distributed machine learning process is also critical for realizing practical applications. In this paper, we propose a reverse auction to incentivize the participation of heterogeneous clients. Different from previous research, we select and schedule winners (or mobile clients) to execute training job in different global iterations, with a goal of social cost minimization. In addition, the number of global iterations is determined by the global accuracy and local accuracy. Both theoretical analysis and large-scale simulations based on the real-world data verified that our proposed auction is truthful, individual rational, computationally efficient, and achieves near-optimal social cost.

In practice, one may not be able to obtain the actual local accuracy and there may be some variations in the training process due to hardware specifications. Furthermore, clients drop out with high probability since the network connection (4G or WiFi) can be unstable. As a future direction, it is interesting to further study a more practical scenario that combines these considerations.

References

[1] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proc. of ACM CCS*, 2015.

[2] P. Mohassel and Y. Zhang, "Secureml: A system for scalable privacy-preserving machine learning," in *Proc. of IEEE SP*, 2017.

[3] J. Konečnỳ, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.

[4] 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*, 2017, pp. 1273–1282.

[5] *Gboard, now available for Android*, https://blog.google/products/search/gboard-now-on-android/.

[6] N. H. Tran, W. Bao, A. Zomaya, N. M. NH, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *Proc. of IEEE INFOCOM*, 2019.

[7] M. Jaggi, V. Smith, M. Takác, J. Terhorst, S. Krishnan, T. Hofmann, and M. I. Jordan, "Communication-efficient distributed dual coordinate ascent," in *Proc. of NIPS*, 2014.

[8] C. Ma, J. Konečnỳ, 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.

[9] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *arXiv preprint arXiv:1812.06127*, 2018.

[10] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Proc. of NIPS*, 2017.

[11] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *Proc. of IEEE ICC*, 2019.

[12] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, 2019.

[13] K. Toyoda and A. N. Zhang, "Mechanism design for an incentive-aware blockchain-enabled federated learning platform," in *Proc. of IEEE Big Data*, 2019.

[14] S. R. Pandey, N. H. Tran, M. Bennis, Y. K. Tun, A. Manzoor, and C. S. Hong, "A crowdsourcing framework for on-device federated learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3241–3256, 2020.

[15] 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.

[16] R. Zeng, S. Zhang, J. Wang, and X. Chu, "Fmore: An incentive scheme of multi-dimensional auction for federated learning in mec," *arXiv preprint arXiv:2002.09699*, 2020.

[17] R. Zhou, Z. Li, C. Wu, and Z. Huang, "An efficient cloud market mechanism for computing jobs with soft deadlines," *IEEE/ACM Transactions on networking*, vol. 25, no. 2, pp. 793–805, 2016.

[18] R. B. Myerson, "Optimal auction design," *Mathematics of operations research*, vol. 6, no. 1, pp. 58–73, 1981.

[19] A. Archer and É. Tardos, "Truthful mechanisms for one-parameter agents," in *Proc. of IEEE FOCS*, 2001.

[20] 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.

[21] Z. Dong, N. Liu, and R. Rojas-Cessa, "Greedy scheduling of tasks with time constraints for energy-efficient cloud-computing data centers," *Journal of Cloud Computing*, vol. 4, no. 1, pp. 1–14, 2015.

[22] H. Wang, Z. Kaplan, D. Niu, and B. Li, "Optimizing federated learning on non-iid data with reinforcement learning," in *Proc. of IEEE INFOCOM*, 2020.

[23] Y. Zhan, P. Li, and S. Guo, "Experience-driven computational resource allocation of federated learning by deep reinforcement learning," in *Proc. of IPDPS*, 2020.

[24] 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*, 2020.

[25] M. Dai, Z. Su, Y. Wang, and Q. Xu, "Contract theory based incentive scheme for mobile crowd sensing networks," in *Proc. of IEEE MoWNeT*, 2018.

[26] X. Zhang, Z. Yang, Y. Liu, J. Li, and Z. Ming, "Toward efficient mechanisms for mobile crowdsensing," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 1760–1771, 2016.

[27] R. Zhou, Z. Li, and C. Wu, "A truthful online mechanism for location-aware tasks in mobile crowd sensing," *IEEE Transactions on Mobile Computing*, vol. 17, no. 8, pp. 1737–1749, 2017.

[28] L. Duan, T. Kubo, K. Sugiyama, J. Huang, T. Hasegawa, and J. Walrand, "Motivating smartphone collaboration in data acquisition and distributed computing," *IEEE Transactions on Mobile Computing*, vol. 13, no. 10, pp. 2320–2333, 2014.

[29] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," in *Proc. of ACM MOBICOM*, 2012.

[30] N. Ding, Z. Fang, and J. Huang, "Optimal contract design for efficient federated learning with multi-dimensional private information," *IEEE Journal on Selected Areas in Communications*, 2020.

[31] 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 *Proc. of IEEE WCNC*, 2020.

[32] J. Kleinberg and E. Tardos, *Algorithm design*. Pearson Education India, 2006.

[33] D. G. Luenberger, *Introduction to linear and nonlinear programming*. Addison-Wesley Reading, MA, 1973, vol. 28.

APPENDIX

A. Proof of Lemma 1

When client i's schedule l was selected, then it has the lowest average cost $\frac{\rho_{il}}{R_{il}(\mathcal{S})}$ in the current iteration. If client i changes its cost to a smaller one $\rho_{i\tilde{l}}$ $(< \rho_{il})$ and others remain the same, $\frac{\rho_{i\tilde{l}}}{R_{il}(\mathcal{S})} < \frac{\rho_{il}}{R_{il}(\mathcal{S})}$ implies that this schedule will still be selected in the current iteration by the greedy algorithm A_{winner}. Hence, Lemma 1 holds.

B. Proof of Lemma 2

According to $A_{paymemt}$, schedule (i', l') has the second smallest average cost in set $\mathcal{C}$ at the current iteration. Then, the payment $p_{i^*} = R_{i^*l^*}(\mathcal{S}) \cdot \frac{\rho_{i'l'}}{R_{i'l'}(\mathcal{S})}$ ensures that $\frac{\tilde{\rho}_{i^*l^*}}{R_{i^*l^*}(\mathcal{S})} \leq \frac{\rho_{i'l'}}{R_{i'l'}(\mathcal{S})}$ when $\tilde{\rho}_{i^*l^*} \leq p_{i^*}$ and $\frac{\tilde{\rho}_{i^*l^*}}{R_{i^*l^*}(\mathcal{S})} \geq \frac{\rho_{i'l'}}{R_{i'l'}(\mathcal{S})}$ when $\tilde{\rho}_{i^*l^*} \geq p_{i^*}$. Consequently, each winning schedule is paid with a critical value.

C. Proof of Lemma 3

We first prove that A_{winner} in Alg. 2 returns a feasible solution to ILP (7). If there exists enough clients, Alg. 2 has at least one feasible solution and it can terminate either before or when set $\mathcal{C} = \varnothing$. When Alg. 2 terminates, the ending condition of the *while* loop can guarantee that constraint (7a) is satisfied. Then, constraint (7b) is not violated since line 13 in Alg. 2 removes all remaining schedules corresponding to bids of client i^* from set $\mathcal{C}$. Constraint (7c) holds because the default value of $z_{il_{ij}}$ is zero and it is updated to 1 only when client i's l_{ij}-th schedule is selected. In conclusion, A_{winner} generates a feasible solution to ILP (7).

We next prove that A_{winner} in Alg. 2 returns a feasible solution to LP (8) in two cases.

Case 1: If client i's l-th schedule is not selected by Alg. 2. we first sort all global iterations in client i's l-th schedule in non-decreasing according of $\gamma_t^{\mathcal{S}}$, and denote it as $\tilde{t} = \{t_1, t_2, ..., t_{|c_{ij}|}\}$. Let t_k be the k-th global iteration in $\tilde{t}$. If t_k is *available* (*i.e.* $\gamma_{t_k}^{\mathcal{S}} \leq K$), schedule l has at least k *available* global iterations. Thus, the average cost for client i's l-th schedule to cover *available* global iterations is no larger than $\dfrac{\rho_{il}}{k}$. Note that $\psi_{\max}^t$ is the maximum bidding pricing in t-th global iteration, and $\psi_{\min}^t$ is the minimum average cost. Therefore, the cost of t_k-th global iteration $\eta_\phi(t_k)$ is no larger than $\dfrac{\rho_{il}}{k}\dfrac{\psi_{\max}^{t_k}}{\psi_{\min}^{t_k}}$, when t_k-th global iteration has K participants. Then

$$\begin{aligned}
&\sum_{t:y_i(t)\in l} g(t) - \lambda_{il} - q_i = \sum_{t:y_i(t)\in l} g(t) \\
&= \frac{1}{H_{\hat{T}_g}\omega} \sum_{t:y_i(t)\in l} \eta_\phi(t) \leq \frac{\rho_{il}}{H_{\hat{T}_g}\omega} \sum_{k=1}^{|c_{ij}|} \frac{1}{k} \frac{\psi_{\max}^{t_k}}{\psi_{\min}^{t_k}} \\
&= \frac{\rho_{il}}{H_{\hat{T}_g}\omega} H_{|c_{ij}|}\omega \leq \frac{\rho_{il}}{H_{\hat{T}_g}\omega} H_{\hat{T}_g}\omega = \rho_{il}.
\end{aligned}$$

Therefore, constraint (8a) can be satisfied when client i's l-th schedule is not selected.

Case 2: If client i's l-th schedule is selected by Alg. 2. Then, we have

$$\sum_{t:y_i(t)\in l} g(t) - \lambda_{il} - q_i = \sum_{t:y_i(t)\in l} g(t) - \lambda_{il}$$
$$= \frac{1}{H_{\hat{T}_g}\omega}(\sum_{t:t\in l\setminus\mathcal{F}_{il}} \eta_\phi(t) + \sum_{t\in\mathcal{F}_{il}} \phi(t,l))$$
$$= \frac{1}{H_{\hat{T}_g}\omega}(\sum_{t:t\in l\setminus\mathcal{F}_{il}} \eta_\phi(t) + \rho_{il})$$
$$\leq \frac{\rho_{il}}{H_{\hat{T}_g}}(\sum_{k=1}^{|l\setminus\mathcal{F}_{il}|} \frac{1}{k+|\mathcal{F}_{il}|} + \frac{1}{\omega})$$
$$\leq \rho_{il}(\frac{H_{|c_{ij}|} - H_{|\mathcal{F}_{il}|} + 1}{H_{\hat{T}_g}}) \leq \rho_{il}.$$

Recall that $\mathcal{F}_{il}$ represents a set involves those global iterations within schedule l that still *available*. And $l\setminus\mathcal{F}_{il}$ denotes the set of *non-available* global iterations in schedule l. Similarly, we order all global iterations in set $l\setminus\mathcal{F}_{il}$ as $\{t_1, t_2, ..., t_{|l\setminus\mathcal{F}_{il}|}\}$. If t_k-th global iteration is just *non-available* after client i's l-th schedule is selected, its cost $\eta_\phi(t)$ should be at most $\frac{\rho_{il}}{k+|\mathcal{F}_{il}|}\frac{\psi^{t_k}_{\max}}{\psi^{t_k}_{\min}}$. Hence, constraint (8a) can also be satisfied.

In summary, A_{winner} generates a feasible solution to ILP (7) and LP (8).

D. Proof of Lemma 4

Line 1 of A_{winner} in Alg. 2 first defines and initializes variables in $O(\hat{T}_g)$ steps. The *while* loop (lines 2-15) runs at most I iterations since the number of clients can be selected is at most I. Sorting all global iterations within $\hat{\mathcal{T}}_g$ needs at least $O(\hat{T}_g \log(\hat{T}_g))$ steps. The inner *for* loop in lines 4-7 selects and updates the representative schedule for each bid, which can be calculated in $O(IJ\hat{T}_g)$ steps. To find a schedule (i^*, l^*) or $(i^\#, l^\#)$, we need to search all schedules within the corresponding set, which takes $O(IJ)$ steps. Then, computing the payment in Alg. 3, updating three related sets $\mathcal{C}, \mathcal{G}$ and $\mathcal{S}$, and recording its cost can be done within $O(IJ\hat{T}_g)$ steps. Consequently, the running time of the *while* loop in Alg. 2 is $O(I\hat{T}_g(\log(\hat{T}_g) + IJ))$. Lines 16-19 calculate dual variable $g(t)$, which take $O(IJ\hat{T}_g)$ steps. The second *for* loop takes $O(IJ)$ steps to calculate dual variable λ_{il}, save corresponding payment p_i, and record the total cost. In conclusion, the time complexity of A_{winner} is $O(I\hat{T}_g(\log(\hat{T}_g) + IJ))$.

E. Proof of Lemma 5

The objective value of dual problem (8) is

$$D = \frac{1}{H_{\hat{T}_g}\omega}\sum_{t\in\hat{\mathcal{T}}_g} K\eta_\phi(t) - \frac{1}{H_{\hat{T}_g}\omega}\sum_{(i,l)}\sum_{t:t\in\mathcal{F}_{il}}(\eta_\phi(t) - \phi(t,l))$$
$$= \frac{1}{H_{\hat{T}_g}\omega}(\sum_{t\in\hat{\mathcal{T}}_g} K\eta_\phi(t) - \sum_{t:t\in\mathcal{F}_{il}}\sum_{(i,l)}\eta_\phi(t))$$
$$+ \frac{1}{H_{\hat{T}_g}\omega}\sum_{(i,l)}\sum_{t:t\in\mathcal{F}_{il}}\phi(t,l)$$
$$\geq \frac{1}{H_{\hat{T}_g}\omega}\sum_{(i,l)}\sum_{t\in\hat{\mathcal{T}}_g}\phi(t,l).$$

For each global iteration in set $\mathcal{F}_{il}$, the number of selected clients is no larger than K. Hence, the first term of the second equality is larger than 0. Meanwhile, $\phi(t,l)$ is assigned a value only when t-th global iteration belongs to set $\mathcal{F}_{il}$. Therefore, it is rational to extend the range of t in the term $1/(H_{\hat{T}_g}\omega)\sum_{(i,l)}\sum_{t:t\in\mathcal{F}_{il}}\phi(t,l)$ from $t : t\in\mathcal{F}_{il}$ to $t\in\hat{\mathcal{T}}_g$.

Then, the objective value of primal problem (7) is

$$P = \sum_{(i,l)\in\mathcal{S}} \rho_{il} = \sum_{(i,l)}\sum_{t\in\hat{\mathcal{T}}_g}\phi(t,l).$$

The above equation holds since when client i's l-th schedule is selected by A_{winner}, ρ_{il} is evenly distributed to variables $\phi(t,l)$, *i.e.*, all global iterations in $\mathcal{F}_{il}$.

Obviously, $H_{\hat{T}_g}\omega \cdot D \geq P$. Let P^* denote the optimal objective value of ILP (7). We have $P^* \geq D$ according to LP duality [33]. Consequently, $P/P^* \leq P/D \leq H_{\hat{T}_g}\omega = \tau$. Therefore, the approximation ratio of A_{winner} is τ.