SIoT Selection, Clustering, and Routing for Federated Learning with Privacy-Preservation

Min-Siou Chung†, Chih-Hang Wang$^\sharp$, De-Nian Yang$^\sharp$, Guang-Siang Lee$^\sharp$, Wen-Tsuen Chen†, and Jang-Ping Sheu†
†Dept. of Computer Science, National Tsing Hua University, Taiwan
$^\sharp$Institute of Information Science, Academia Sinica, Taiwan
E-mail: s108062603@m108.nthu.edu.tw, {superwch7805, dnyang, gslee9822802}@iis.sinica.edu.tw,
and {wtchen, sheujp}@cs.nthu.edu.tw

ICC 2022 - IEEE International Conference on Communications | 978-1-5386-8347-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICC45855.2022.9838641

Abstract—**With the advances in Social Internet of Things (SIoT) and Federated learning (FL), smart devices are now able to cooperatively and locally perform learning tasks to protect sensitive data by Differential Privacy (DP). On the other hand, Hierarchical FL (HFL) clusters SIoTs into multiple local training groups to reduce communication overheads by local aggregation. In this paper, we explore SIoT Training Group Construction (STGC) for HFL to minimize the total SIoT computation, communication and hiring costs, and the privacy cost for exploiting DP. We prove that STGC is NP-hard and inapproximable within any factor unless P = NP. Then, we design an algorithm with the ideas of *Coverage Efficiency Indicator*, *Data Balance-aware Dual Adjustment*, and *Privacy-Aware Rerouting* to choose and cluster SIoTs and to determine the aggregator for local training and SIoT routing in each cluster. Simulation results manifest that the proposed algorithm outperforms state-of-the-arts regarding the total cost, model accuracy, and convergence time.**

I. Introduction

With the advances in Artificial Intelligence (AI), the notion of Social Internet of Things (SIoT) has emerged to create and maintain the collaborative social relations among smart IoT devices [1].[1] First, SIoTs possessed by the same owner can share *ownership object relation*. Next, they can build *co-location object relation* and *co-work object relation* if they are located in adjacent areas and designed to manage similar events, respectively. For example, SIoTs with ownership or co-work object relations in hospitals can cooperatively perform learning tasks (i.e., federated learning [2], [3]) for smart health monitoring and patient tracking of COVID-19 [4], [5].

Federated Learning (FL) [2], [3] is a novel machine learning paradigm which allows SIoTs to locally train learning models, and the SIoTs only exchange their model weights with the central server, executing Federated Averaging (FedAvg) for model aggregation, to prevent accessing others' private data. In order to reduce communication overheads, Hierarchical FL (HFL) [6]–[8] was proposed to cluster SIoTs into multiple local training groups, where each group includes an aggregator (AG) to execute a number of local aggregations before uploading the aggregated model to the central server for global aggregation.[2] Liu *et al*. [6] designed a client-edge-cloud learning framework with low communication costs. Mhaisen *et al*. [7] chose users for each training group according to the distribution distance of data labels. Wang *et al*. [8] derived the optimal cluster for minimizing resource consumption in HFL. However, the above works ignored the SIoT selection, clustering, and routing for HFL, to support each SIoT with a different privacy demand and to collect data with different labels and quality.

Greater data coverage (i.e., multifarious training data that includes a number of labels [9]) brings out better training results (e.g., high-accuracy event identifications) [2]. To increase the variety of data, a potential way is to hire users' private SIoTs via human social networks [10], [11], and the employer will pay the reward for participants [10], [11]. For example, Google calls for Gboard users to participate in training language models for next word prediction. Jamaican exploits the social relations between public (provided by governments) and private SIoTs to cooperatively monitor criminal events.[3] However, privacy leakage will occur if the hired SIoTs are malicious to recover private data from their relayed model weights by Generative Adversarial Networks (GANs) [12]. To preserve privacy, *Differential Privacy* (DP) adds Gaussian noise to the model weights according to the required privacy,[4] which can be set based on SIoT social relations (e.g., trust between neighbor SIoTs) [16], to prevent attackers from recovering private data [13]–[15]. Nevertheless, Bagdasarya *et al*. [15] indicated that the accuracy degrades under DP-based model training, and it requires additional training rounds to restore the required accuracy [12], [17]. Abadi *et al*. [13] evaluated the privacy loss for DP, and Wei *et al*. [14] analyzed the FL convergence with DP. However, the above works did not explore the SIoT selection and clustering with different data coverage, and they also ignored the heterogeneous privacy demands of SIoTs according to their social relations.

In this paper, we explore joint SIoT selection, clustering, and routing in HFL with privacy preservation for minimizing the total 1) computation and communication costs of SIoTs, 2) hiring cost for incorporating private SIoTs [10], [11], and

[1] For ease of presentation, we will use (S)IoTs to represent (S)IoT devices.

[2] The communication cost is reduced since the local aggregation exploits short-range communications like D2D for exchanging the model with the AG.

[3] Google: https://reurl.cc/82NNlg, Jamaica eye: https://jamaicaeye.gov.jm/

[4] DP adds noise to the model weights, instead of training data, since only the weights are exchanged in FL [12]–[15]. When AG executes FedAvg, the noise will induce a biased training result, which undermines the model accuracy.

3) privacy cost of DP.[5] However, the problem introduces the following new challenging issues. 1) *Tradeoff in data coverage and privacy*. A personal private SIoT with good data coverage may induce a larger privacy cost due to the need for a higher privacy protection when it has poor social relations with others. 2) *Joint consideration of data balance and data quality*. Imbalancing data quantity between different clusters induces the bias of training models and worse convergence [18]. However, the data quality (estimated by empirical risk [19], [20], detailed later) of the SIoTs with abundant data may be diverse, and the training result will be poor when choosing those SIoTs with worse quality for ensuring the data balance. 3) *AG selection and routing with privacy*. To minimize the total communication cost, it is more favorable to choose the SIoT, with the minimum sum of the costs for communicating with the other SIoTs via the minimum-cost paths in the cluster, as an AG. However, SIoTs on a minimum-cost path may have poor social relations, leading to a larger privacy cost. Hence, it is required to jointly choose an AG and the route of each SIoT in a cluster.

To address the above issues, we formulate a new optimization problem, SIoT Training Group Construction (STGC), for HFL to minimize the total communication, computation, hiring, and privacy costs. We prove that STGC is NP-hard, and there does not have any algorithm with a finite approximation ratio for STGC unless $P = NP$. Afterward, we design an algorithm, named Privacy-aware SIoT Selection, Clustering, and Routing (PSSCR), with the ideas of *Coverage Efficiency Indicator* (CEI), *Data Balance-aware Dual Adjustment* (DBDA), and *Privacy-Aware Rerouting* (PAR) 1) to choose and cluster SIoTs with greater data coverage and quality and 2) to determine the AG and SIoT routing to minimize the total communication and privacy costs, by carefully examining the social relation between each pair of SIoTs in each cluster. Simulation results manifest that PSSCR can effectively reduce more than 60% of the total cost and convergence time compared with state-of-the-art algorithms.

The remaining of this paper is organized as follows. Section II describes the system model and STGC, and Section III presents the algorithm PSSCR. Section IV summarizes the simulation, and Section V concludes this paper.

II. System Model and Problem Formulation

A. System Model

Due to the limited space, the notation table is presented in [21]. Following [6]–[8], we consider an HFL system consisting of a single central server for global aggregation and a set of SIoTs for local training and aggregation. Let $G = (V, E)$ be an SIoT network, where V is the set of SIoTs, and E is the set of edges. An edge $e_{n,m} \in E$ exists between SIoTs n and m if they have social relations and can communicate with each other with cost $\mu_{n,m}$ [1]. Let $SR_{n,m}$ be the social relation (trust) between SIoTs n and m, which can be set according to the trust between their owners in social networks [11].[6]

In order to avoid privacy leakage when an SIoT transmits the training model to the local AG, DP is proposed to add noise to the training model weights of each SIoT before transmitting them [13], [14]. Following [16], [22], we employ a trust-to-privacy model to transform the trust $SR_{n,m}$ between SIoTs n and m into a privacy degree, and a lower trust yields a lower degree. Accordingly, the privacy degree $PD_{n,m}$ for the communication between n and m is defined as $PD_{n,m} = \frac{SR_{n,m}}{SR_{n,m}+\tau} \cdot \varepsilon$ [16], where $\tau \in [0, 1]$ and $\varepsilon > 0$ are controllable parameters for bounding the privacy degree in a certain range and for ensuring privacy intensity, respectively.[7] For each cluster, when SIoTs upload their trained model weights, DP adds noise to the weights according to the privacy degree to ensure the privacy demand of each SIoT pair on a communication path for avoiding any privacy leakage during the transmission [13], [14]. Therefore, following [14], the privacy degree in the cluster with AG a is dominated by the worst relation between the two SIoTs on a communication path toward a, i.e., $PD_a = \min_{n\in\mathbb{S}_a}\{\min_{m\in P_{n,a}}\{PD_{n,m}\}\}$, where $P_{n,a}$ is the communication path between SIoT n and AG a, and $\mathbb{S}_a$ is the set of SIoTs in cluster a.[8] Since DP may reduce the performance of training, additional training rounds are required to attain the expected accuracy [15]. Following [12], [17], the privacy degree is mapped to the privacy cost $\rho(PD_a) = (1 - PD_a/\varepsilon) \cdot \delta$, where δ is the cost for additional training rounds [15], and a higher privacy degree (due to a higher trust) induces a smaller cost.

For HFL, each SIoT $n \in V$ senses its surroundings to collect training data, and it has a computation cost κ_n for local training (and aggregation).[9] If n is chosen as AG, it further requires a communication cost ν_n for uploading the training model to the central server [3]. Let $\mathbb{L}$ be the set of training data labels (classes), and $\mathcal{D}_n^l \subseteq \mathcal{D}_n$ is the set of data with label $l \in \mathbb{L}$ collected by SIoT n, where $\mathcal{D}_n$ is the set of total data in SIoT n and $|\mathcal{D}_n| = \sum_{l\in\mathbb{L}} |\mathcal{D}_n^l|$. We follow [19], [20] to employ the historical *Empirical Risk* (ER) of an SIoT to evaluate the quality of its collected data, and the ER of SIoT n is denoted by γ_n.[10]

B. Problem Formulation

Equipped with the above model, we formulate STGC as follows. The objective is to minimize the total 1) computation

[5] According to [12], [17], the cost is related to the model convergence since DP adds noise to the model weights according to the required privacy of SIoTs, which can be set based on SIoT social relations [16], and it will introduce additional training rounds to restore a certain accuracy.

[6] We consider public and private SIoTs and assume the owner of public SIoTs is the fully trust government (i.e., $SR_{n,m} \to \infty$ if n or m is public).

[7] A smaller ε leads to a lower privacy degree, and DP will add more noise to the model weights for a higher privacy. The privacy demand of an SIoT n can also be decided by the owner, and we can directly set the privacy degree $PD_{n,m}, \forall m \in V$ according to the owner's demand.

[8] Since each cluster includes an AG, unless stated otherwise, we use the same index of AG to represent a cluster.

[9] The computation cost of an SIoT can be set according to the computing power and the number of iterations for training the SIoT's local data [3].

[10] Data quality is affected by the precision of data collection and the correctness of data labeling, which are related to the sensing and computation abilities of SIoTs. ER finds the outliers of data and measures the average training error over the training set in an SIoT, which can reflect the data quality, and a higher ER indicates worse quality [19], [20].

and communication costs of SIoTs, 2) hiring cost for hiring private SIoTs, and 3) privacy cost for DP. Specifically, let binary variables x_n and y_n represent if SIoT n is selected as a cluster member and an AG for model training, respectively. We denote by $\mathbb{A}$ the set of chosen AGs, and each member n needs to communicate with an AG for local aggregation (i.e., they are grouped into the same cluster). Let $z_{n,a}$ be a binary variable indicates if SIoT n and AG a are in the same cluster. The total computation and communication costs of SIoTs are $C_C = \sum_{n\in V} x_n \cdot \kappa_n + \sum_{n\in V} y_n \cdot \nu_n + \sum_{n\in V, a\in\mathbb{A}} z_{n,a} \cdot c(P_{n,a})$, where $c(P_{n,a}) = \sum_{e_{n,m}\in P_{n,a}} w_{n,m} \cdot \mu_{n,m}$ is the communication cost between SIoTs n and a on $P_{n,a}$, and $w_{n,m}$ indicates if edge $e_{n,m}$ is chosen. The total hiring cost for hiring private SIoTs is $C_H = \sum_{n\in V^{pr}} x_n \cdot \psi \cdot |\mathcal{D}_n|$,[11] where $V^{pr} \subseteq V$ is the set of private SIoTs, ψ is the hiring cost of unit data [3], [10], and $|\mathcal{D}_n|$ is the quantity of data provided by SIoT n. The total privacy cost is $C_P = \sum_{n\in V, a\in\mathbb{A}} z_{n,a} \cdot \rho(PD_a)$. The objective of STGC is to minimize the total cost $C_C + C_H + C_P$.

STGC has the following constraints. 1) *Data balance constraint*. Following [18], [23], the difference of data quantity between each pair of clusters is required to be limited to prevent the biases of the models trained by different clusters,[12] i.e., $|\mathcal{D}_a| - |\mathcal{D}_b| \leq D, \forall a, b \in \{1, 2, \ldots, |\mathbb{A}|\}$, where $|\mathcal{D}_a| = \sum_{n\in V} z_{n,a} \cdot |\mathcal{D}_n|$ is the total data quantity of cluster a, and D is the maximum difference of data quantity. 2) *Data coverage constraint* [7], [9]. To ensure training quality with various data labels, the total data quantity of each label must be at least Q [24], i.e., $\sum_{n\in V} x_n \cdot |\mathcal{D}_n^l| \geq Q,\ \forall l \in \mathbb{L}$, where Q is the least data quantity of each label, which can be set by the empirical nonlinear classification error model [24]. 3) *Data quality constraint*. Following [19], [20], the average ER of the selected SIoTs cannot exceed a certain threshold to ensure data quality for training, i.e., $\frac{\sum_{n\in V} x_n \cdot \gamma_n \cdot |\mathcal{D}_n|}{\sum_{n\in V} x_n \cdot |\mathcal{D}_n|} \leq R$,[13] where R is the tolerance of ER. 4) *SIoT connectivity constraint* [1], [11]. The SIoTs chosen in the same cluster need to be connected to ensure that they can communicate with each other. 5) *SIoT clustering constraint* indicates that each SIoT is assigned to only one cluster [6], [7], i.e., $\sum_{a\in\mathbb{A}} z_{n,a} \leq 1, \forall n \in V$.

Definition 1. Given an SIoT network $G = (V, E)$ with 1) computation cost κ_n, cost ν_n for communicating with the central server, hiring cost $\psi \cdot |\mathcal{D}_n|$, data set $\mathcal{D}_n$, and ER γ_n for each SIoT $n \in V$, 2) communication cost $\mu_{n,m}$ for each edge $e_{n,m} \in E$, and 3) the trust $SR_{n,m}$ between SIoTs n and m, STGC is to select and cluster a subset of SIoTs V and to determine an AG and the routing of SIoTs in each cluster such that the *data balance*, *data coverage*, *data quality*, *SIoT connectivity*, and *SIoT clustering* constraints are met. The objective is to minimize the total cost $C_C + C_H + C_P$.

Theorem 1. *STGC is NP-hard and cannot to be approximated by any factor unless $P = NP$.*

Proof. Due to the limited space, the proof is provided in [21].

III. Algorithm

To address STGC, an intuitive approach is to iteratively extract SIoTs with the maximum total data quantity of labels until the data coverage constraint is met, and then the SIoTs are clustered by the K-means method [8], [25], where the SIoT with the minimum cost to communicate with the central server is chosen as the AG in each cluster. However, the above approach ignores the data balance requirement when clustering SIoTs, and it may choose the SIoTs with good data coverage but diverse quantity, inducing the bias of training models and worse model convergence. In the following, we propose PSSCR to address the challenges listed in Section I. For the first challenge, we introduce *Coverage Efficiency Indicator* (CEI) to evaluate the data coverage per unit computation, hiring and privacy costs for each SIoT, and PSSCR, to iteratively select the SIoT with the maximum CEI. For the second challenge, *Data Balance-aware Dual Adjustment* (DBDA) iteratively removes or adds an SIoT into a cluster by evaluating its data quality for improving model training and data balance simultaneously. For the last challenge, PSSCR chooses the AG with the minimum communication and privacy costs for communicating with the cluster members, and *Privacy-Aware Rerouting* (PAR) iteratively reroutes SIoTs to traverse the paths with better social relations (i.e., higher privacy degree) for SIoTs to further reduce the privacy cost of DP. Due to the limited space, the pseudocode of PSSCR is presented in [21].

1) *SIoT Selection and Clustering (SSC)*: To ensure the data coverage and SIoT connectivity, SSC iteratively chooses an SIoT pair (or an SIoT) with the maximum *Coverage Efficiency Indicator* (CEI) for creating a new cluster (or incorporating the SIoT into an existing cluster). Note that only the pair of adjacent SIoTs (and the SIoTs that can connect to the cluster by one-hop) will be considered in SSC in order to ensure connectivity. In the following, we first define CEI of an SIoT pair and an SIoT for incorporating it into an existing cluster.

For ensuring the data coverage by the minimum total cost, CEI is defined as the ratio of the effective data coverage (EDC) of SIoTs to their induced costs, where EDC of an SIoT n is the sum of data coverage of different labels increased by n until reaching the least data quantity Q.[14]

$$EDC(\mathcal{D}_n) = \sum_{l\in\mathbb{L}} E_n^l, E_n^l = \begin{cases} |\mathcal{D}_n^l|, & d^l < Q, \\ 0, & d^l > Q, \end{cases} \qquad (1)$$

where E_n^l is the number of effective data with label l, and d^l is the total data quantity of label l provided by currently selected SIoTs. Therefore, CEI of an SIoT pair (n, m) is

$$CEI(n, m) = \frac{EDC(\mathcal{D}_n \cup \mathcal{D}_m)}{C(n, m)}, \qquad (2)$$

[11] The hiring cost is zero if SIoT n is public.

[12] A cluster with large data quantity will dominate the training process and dilute the training results of the other clusters. Moreover, the cluster with insufficient (small) data quantity will undermine the convergence [18]. Therefore, it is desired to balance the data quantity among different clusters.

[13] Note that DP adds noise to the model weights instead of training data. Hence, it affects the training convergence time (i.e, requiring additional training rounds to achieve a certain accuracy) instead of data quality.

[14] Recall that the data coverage constraint in Section II is to ensure that the total data quantity of each label can exceed Q, i.e., $\sum_{n\in V} x_n \cdot |\mathcal{D}_n^l| \geq Q$.

TABLE I: An example of data quantity for labels

	$\lvert\mathcal{D}_n^1\rvert$	$\lvert\mathcal{D}_n^2\rvert$	$\lvert\mathcal{D}_n^3\rvert$	$\lvert\mathcal{D}_n^4\rvert$	ω_n		$\lvert\mathcal{D}_n^1\rvert$	$\lvert\mathcal{D}_n^2\rvert$	$\lvert\mathcal{D}_n^3\rvert$	$\lvert\mathcal{D}_n^4\rvert$	ω_n
n_1	4	5	4	5	1.2	n_8	4	7	6	4	1.1
n_2	4	6	5	4	0.7	n_9	4	4	5	4	0.7
n_3	5	4	5	6	0.6	n_{10}	6	7	7	6	0.6
n_4	4	5	5	4	0.9	n_{11}	3	5	5	6	1.0
n_5	6	7	7	6	1.0	n_{12}	5	5	4	6	0.6
n_6	7	6	6	7	0.8	n_{13}	3	5	7	5	0.7
n_7	5	5	4	4	1.0	n_{14}	6	5	3	6	0.9

where $C(n,m) = (\kappa_n + \kappa_m) + (\mu_{n,m} + \nu) + \psi \cdot (|\mathcal{D}_n| + |\mathcal{D}_m|) + 2\rho(PD_{n,m})$ is the total cost after choosing SIoTs (n,m), and $\nu = \max\{\nu_n, \nu_m\}$ is the communication cost for uploading the model to the central server.[15] For the EDC of an SIoT chosen for an existing cluster, we can regard the cluster as a macro SIoT m with the computation cost $\kappa_m = 0$, communication cost for returning the model $\nu_m = 0$, and hiring cost $\psi \cdot |\mathcal{D}_m| = 0$, since they have been examined when constructing the cluster. In addition, $\mu_{n,m}$ is set to the communication cost for connecting SIoT n to the nearest SIoT in cluster m, which is the neighbor of n for SIoT connectivity, and $PD_{n,m}$ is set according to the worst social relations between the two SIoTs in cluster m with n joined.[16]

At the beginning, SSC first chooses an SIoT pair (n,m) with the maximum CEI to create the first cluster. In each iteration afterward, SSC simultaneously considers two cases: 1) choosing an SIoT pair for creating a new cluster and 2) incorporating an SIoT into an existing cluster. SSC finds the case with the maximum CEI. To build the communication topology, SSC directly connects the SIoT pair for the first case, while it connects the chosen SIoT to its nearest neighbor in the cluster via one-hop transmission to ensure SIoT connectivity for another case. SSC stops when the total data quantity of each label is at least Q.

Example 1. Fig. 1(a) presents an illustrative example. Each triangle is an SIoT, and the number beside each triangle and edge is the computation and communication cost, respectively. We set ψ, δ, τ, ε, ν_n, Q, D, and R to 0.2, 30, 0.3, 15, 9, 45, 20, and 0.8, respectively, and Table I summarizes the other parameters. Assume that $SR_{n_5,n_6} = 0.9$ and $SR_{n_{13},n_{14}} = 1$, and therefore $PD_{n_5,n_6} = \frac{0.9}{0.9+0.3} \times 15 = 11.25$ and $PD_{n_{13},n_{14}} = \frac{1}{1+0.3} \times 15 = 11.54$. Since the pair (n_5, n_6) has the largest $CEI(n_5,n_6) = \frac{52}{8+2+0.2\times52+2\times7.5+9} = 1.17$, SSC selects n_5 and n_6 to create the first cluster and updates (d^1, d^2, d^3, d^4) to $(13, 13, 13, 13)$. SSC then finds n_3 with $CEI(n_3, m_1) = 1.06$, where $m_1 = \{n_5, n_6\}$ is the first cluster. Similarly, SSC chooses (n_{13}, n_{14}) with $CEI(n_{13}, n_{14}) = 1.00$. Since n_3 has the largest CEI, SSC extracts n_3 to and adds it to m_1. The final result of SSC is shown in Fig. 1(b).

2) *Data Balance-aware Dual Adjustment (DBDA)*: To avoid the bias of training models in different clusters and ensure data quality, DBDA iteratively adjusts the clusters with the largest and the smallest data quantity for balancing the data quantity among these clusters according to *Data Quality Improvement Indicator* (DQI).[17] Specifically, let $\mathbb{S}$ be the set of selected SIoTs, and $ER(\mathbb{S}) = \frac{\sum_{n\in\mathbb{S}} \gamma_n \cdot |\mathcal{D}_n|}{\sum_{n\in\mathbb{S}} |\mathcal{D}_n|}$ is the average ER of SIoT set $\mathbb{S}$ to evaluate its data quality [19].[18] DQI of SIoT n is defined as the difference in overall data quality before and after removing (or adding) SIoT n. If an SIoT n is removed from $\mathbb{S}$, $DQI(n) = ER(\mathbb{S}) - ER(\mathbb{S} \setminus \{n\})$. Otherwise, $DQI(n) = ER(\mathbb{S}) - ER(\mathbb{S} \cup \{n\})$ if n is added to $\mathbb{S}$.

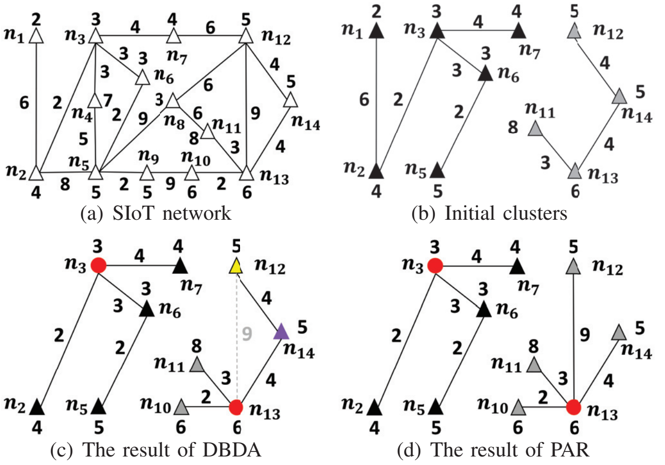

Fig. 1: An illustrative example of PSSCR.

To avoid disconnectivity, DBDA iteratively examines the cluster $k \in \mathbb{K}$ with the largest data quantity, and it removes the leaf SIoT with the largest DQI on the communication topology of k, where $\mathbb{K}$ is the set of clusters constructed in SSC. DBDA then updates the data quantity of k. If there is no SIoT that can be removed due to the data coverage constraint, DBDA examines the cluster k with the smallest data quantity to add the SIoT (that can directly connect to an SIoT in cluster k) with the largest DQI for reducing the data quantity difference. The above procedure stops when the data balance and data quality constraints are met. Afterward, for each cluster, DBDA chooses the SIoT with the minimum communication and privacy costs for communicating with other cluster members as the AG. Let $G_k = (V_k, E_k)$ be the communication topology of cluster k, and $C(m, G_k) = \sum_{n\in V_k} c(P_{n,m}^{G_k}) + \nu_m + |V_k| \cdot \rho(PD_k)$ is the total communication and privacy costs of cluster k with SIoT m being the AG, where V_k and E_k are respectively the set of SIoTs and the set of communication edges in cluster k, and $P_{n,m}^{G_k}$ is the path between n and m on G_k. For each cluster k, DBDA extracts the SIoT m with the minimum $C(m, G_k)$ as the AG.

Example 2. Following Example 1, we denote by TQ_k the total data quantity of cluster k, and cluster 1 and 2 are respectively $\{n_1, n_2, n_3, n_5, n_6, n_7\}$ with $TQ_1 = 124$ and $\{n_{11}, n_{12}, n_{13}, n_{14}\}$ with $TQ_2 = 81$. Since no SIoT can be

[15] We use the worst-case communication cost to evaluate CEI since DBDA will adjust each cluster for data balance before choosing the AG.

[16] AG will be decided in DBDA, and we use the worst-case social relation to evaluate $PD_{n,m}$ here.

[17] To satisfy the data balance constraint, it is sufficient to ensure the difference between the largest and smallest data quantity, since the data quantity difference between any two clusters must be smaller than the difference between the largest and smallest ones.

[18] Recall that we follow [19], [20] to estimate data quality by ER.

removed in cluster 1, DBDA first chooses n_{10} with the largest $DQI(n_{10}) = 0.845 - 0.818 = 0.027$ to join cluster 2. Then, it updates the clusters with the largest and the smallest data quantity to cluster 1 with $TQ_1 = 124$ and cluster 2 with $TQ_2 = 103$, respectively, and $(d^1, d^2, d^3, d^4) = (54, 60, 57, 61)$. Next, DBDA selects n_1 with the largest $DQI(n_1) = 0.032$ and removes n_1 in cluster 1. The adjustment stops since the data balance and data quality constraints are met. Then, DBDA selects n_3 as AG in cluster 1 since $C(n_3, G_1) = 14 + 8 + 5 \times 8.82 = 66.1$ and $PD_1 = 10.59$. Similarly, DBDA chooses n_{13} as AG in cluster 2 since $C(n_{13}, G_2) = 88.5$ and $PD_2 = 8.75$. The final result of DBDA is shown in Fig. 1(c), with AGs (red circle) and the total cost $C_C + C_H + C_P = 241.6$.

3) *Privacy-Aware Rerouting (PAR)*:

It is worth noting that the selected AG does not optimize the privacy cost since $G_k \subseteq G$. PAR reroutes SIoTs by a longer path composed of SIoTs with better social relations to further reduce the privacy cost. Specifically, for each cluster k, PAR first examines the social relation between each SIoT pair (n, m) to find the one with the worst social relation, where m is on the path from n to the AG.[19] Since an SIoT may be a relay for several SIoTs, rerouting an SIoT i will also incur the routing of every SIoT that exploits i as a relay. For the pair (n, m) with the worst social relation, PAR then finds the edge $e_{i,j}$ with the largest communication cost on the path from n to the nearest *branch* SIoT b (i.e., the SIoT with multiple incoming edges) to avoid rerouting excessive SIoTs. For minimizing the privacy cost, PAR reconnects i to the neighbor node (except j) that induces the smallest privacy degree $\min_{r \in \mathbb{R}_i}\{\min_{s \in P_{r,k}}\{PD_{r,s}\}\}$ if the total cost $C(k, G_k)$ can be reduced, where $\mathbb{R}_i$ is the set of SIoTs that exploit i as a relay for communicating with the AG. The above process repeats until the total cost cannot be reduced.

Example 3. Following Example 2, PAR finds the pair (n_{12}, n_{14}) with the worst social relation and the edge $e_{n_{12},n_{14}}$ with the largest communication cost. Then, PAR reconnects n_{12} to n_{13} with the smallest privacy degree, and the total cost $C(n_{13}, G_2)$ is reduced from 88.5 to 61.62. Fig. 1(d) shows the final result, where cluster 1 has $PD_1 = 10.59$ and $AG = n_3$, and cluster 2 has $PD_2 = 11.54$ and $AG = n_{13}$. The final total cost is 214.72, which is optimal in this example.

Time Complexity. The time complexity of PSSCR is $O(|V|^2 \cdot (Q|\mathbb{L}| + |\mathcal{D}|))$, where $|\mathcal{D}| = \sum_{n \in V} |\mathcal{D}_n|$ is the total data quantity of all SIoTs. Due to the space constraint, detailed complexity analysis is presented in [21].

IV. Simulation

A. Simulation Setup

Following [8], we consider an HFL system with a central server, 40 users, and 300 SIoTs, where the trust between users is generated according to [11], and each user owns 6 SIoTs in average. Following [1], the SIoT social relations and trust are established according to the ownership relations. The computation cost is assigned based on the SIoT computation ability [3], [11], and the communication cost is set according to the transmission rate and communication distance [1], [11]. For DP, parameters ε and δ are set to 25 [13] and 30 [17], respectively. For HFL, we follow [6] to train a CNN model with two 5×5 convolution layers and two fully connected layers, where the batch size and learning rate are 64 and 0.002, respectively. Each AG executes a local aggregation after 10 training rounds, and a global aggregation is executed on the central server after 5 local aggregations in each cluster [8]. Following [2], we consider *Non-Identically and Independently Distributed* (Non-IID) data setting by allocating 150 training data, which is composed of 80% data with the same label [2], to each SIoT. In addition to synthetic data, we also follow to use two standard datasets, *MNIST* and *Fashion-MNIST*, for the learning task in HFL. The constraint parameters D, Q, and R are set to 500 [23], 1500 [24], and 0.8 [19], respectively.

Since there is no related work that explores joint SIoT selection, clustering, and routing for HFL with privacy preservation, we compare PSSCR with three conventional SIoT scheduling and clustering algorithms for FL/HFL, Traditional FL with DP (TFL) [14], Hierarchical Aggregation FL (HAFL) [8] and Semi-FL (SFL) [25]. TFL iteratively selects an SIoT at random until the data coverage constraint is met, and HAFL and SFL further cluster SIoTs by the K-means method. To evaluate PSSCR, we vary the following parameters: 1) number of SIoTs, 2) SIoT degree, and 3) δ, where SIoT degree is the average number of neighbors of each SIoT. We measure the following performance metrics: 1) total cost, 2) computation and communication costs, 3) privacy cost, 4) average trust, and 5) model accuracy. Each simulation result is averaged over 300 samples. Due to the space constraint, we provide more simulations in [21].

B. Simulation Results

In Fig. 2(a), when the number of SIoTs increases, PSSCR induces a smaller total cost since it exploits CEI to create clusters and choose SIoTs for ensuring data coverage by smaller communication and computation costs. In Fig. 2(b), TFL generates the largest computation and communication costs since it ignores SIoT selection and clustering, and every SIoT needs to upload the training model to the distant central server. In contrast, PSSCR has more opportunities to choose closer SIoTs to minimize the communication cost when balancing data quantity among different clusters by DBDA. The privacy cost of PSSCR only slightly increases in Fig. 2(c), since it exploits PAR to reroute SIoTs to the paths including the SIoTs with better social relations (i.e., they have higher trust in 2(d)) to lower down the privacy degree for minimizing the privacy cost. In contrast, the baselines choose the SIoTs with better data coverage, but they ignore the privacy demands of SIoTs when constructing training groups. It may make the SIoTs with worse social relations (lower privacy degree) being

[19] Recall that the privacy degree in a cluster is dominated by the worst social relations between the two SIoTs on the communication paths toward the AG. Therefore, we extract the one with the worst social relation to improve for reducing the privacy cost.

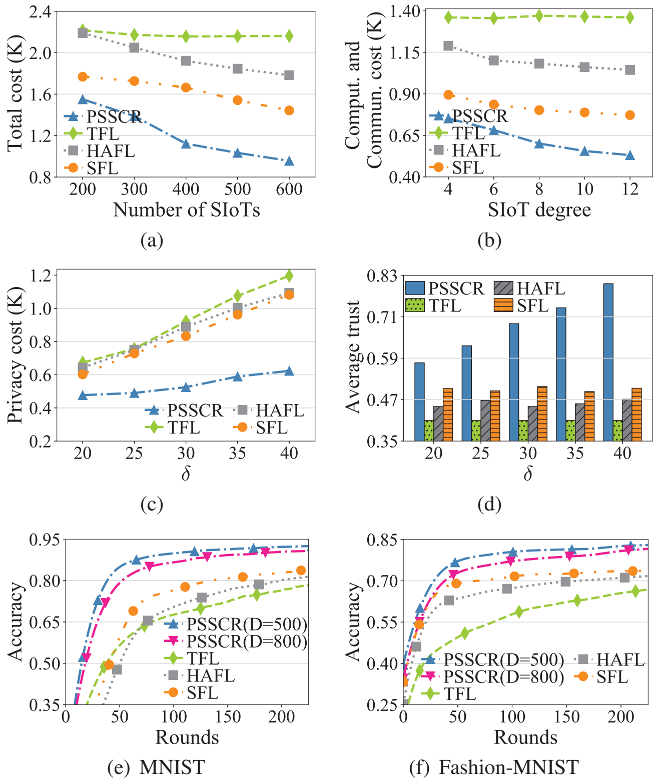

Fig. 2: Performance under different parameters.

chosen in the same group, leading to a larger privacy cost. In Figs. 2(e) and 2(f), we evaluate the training performance of different algorithms with real datasets. Since Fashion-MNIST (the clothing classification dataset) is a more complex dataset than MNIST (the handwritten digital recognition dataset), the model of MNIST is easier to be trained with high accuracy. PSSCR generates the highest accuracy with the least convergence time (i.e., the curve of PSSCR levels out after 100 rounds in Fig. 2(e)). This is because DBDA iteratively adds the SIoT with the maximum DQI to the cluster with the smallest data quantity to 1) increase total data quality and 2) reduce the data quantity difference among clusters (for decreasing the bias of the training model). In Fig. 2(f), with a more strict data balance constraint D, the data quantity difference among the clusters adjusted by DBDA is smaller, and the training model of each cluster is more consistent, leading to the higher accuracy and convergence rate. In summary, PSSCR reduces the total cost and convergence time by more than 60% compared with the state-of-the-arts.

V. Conclusions

To the best of our knowledge, this paper makes the first attempt to explore DP for SIoT selection, clustering, and routing for HFL. We formulate a new optimization problem STGC to minimize the total computation, communication, hiring, and privacy-preservation costs of SIoTs. We prove that STGC is NP-hard and inapproximable within any ratio unless $P = NP$. Then, we propose PSSCR with the ideas of CEI, DBDA, and PAR to select and cluster SIoTs and to find the AG and SIoT routing in each cluster. Simulation results manifest that PSSCR effectively reduces the total cost and convergence time by more than 60%.

References

[1] C.-H. Wang, J.-J. Kuo, D.-N. Yang, and W.-T. Chen, "Collaborative social internet of things in mobile edge networks," *IEEE Internet Things J.*, vol. 7, no. 12, pp. 11 473–11 491, 2020.

[2] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-iid data," in *Proc. ICLR*, 2020.

[3] S. R. Pandey *et al.*, "A crowdsourcing framework for on-device federated learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3241–3256, 2020.

[4] Q. Wu, X. Chen, Z. Zhou, and J. Zhang, "Fedhome: Cloud-edge based personalized federated learning for in-home health monitoring," *IEEE Trans. Mob. Comput.*, pp. 1–1, 2020.

[5] W. Y. B. Lim *et al.*, "Dynamic contract design for federated learning in smart healthcare applications," *IEEE Internet Things J.*, pp. 1–1, 2020.

[6] L. Liu, J. Zhang, S. H. Song, and K. B. Letaief, "Client-edge-cloud hierarchical federated learning," in *Proc. IEEE ICC*, 2020.

[7] N. Mhaisen, A. Awad, A. Mohamed, A. Erbad, and M. Guizani, "Optimal user-edge assignment in hierarchical federated learning based on statistical properties and network topology constraints," *IEEE Trans. Network Sci. Eng.*, pp. 1–1, 2021.

[8] Z. Wang *et al.*, "Resource-efficient federated learning with hierarchical aggregation in edge computing," in *Proc. IEEE INFOCOM*, 2021.

[9] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-i.i.d. data," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 31, no. 9, pp. 3400–3413, 2020.

[10] Y. Zhang *et al.*, "Incentive mechanism for mobile crowdsourcing using an optimized tournament model," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 4, pp. 880–892, 2017.

[11] K.-Y. Chen *et al.*, "Collaboration between social internet of things and mobile users for accuracy-aware detection," in *Proc. IEEE ICC*, 2021.

[12] K. Wei *et al.*, "User-level privacy-preserving federated learning: Analysis and performance optimization," *IEEE Trans. Mob. Comput.*, 2021.

[13] M. Abadi *et al.*, "Deep learning with differential privacy," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2016.

[14] K. Wei *et al.*, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 3454–3469, 2020.

[15] E. Bagdasaryan, O. Poursaeed, and V. Shmatikov, "Differential privacy has disparate impact on model accuracy," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2019.

[16] L. Cui, Y. Qu, S. Yu, L. Gao, and G. Xie, "A trust-grained personalized privacy-preserving scheme for big social data," in *Proc. IEEE ICC*, 2018.

[17] M. Chen, H. V. Poor, W. Saad, and S. Cui, "Convergence time optimization for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2457–2471, 2021.

[18] H. Zhu and Y. Jin, "Multi-objective evolutionary federated learning," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 31, no. 4, pp. 1310–1322, 2020.

[19] P. Hand and V. Voroninski, "Global guarantees for enforcing deep generative priors by empirical risk," *IEEE Trans. Inf. Theory*, vol. 66, no. 1, pp. 401–418, 2020.

[20] D. Wang and J. Xu, "Differentially private empirical risk minimization with smooth non-convex loss functions: A non-stationary view," *AAAI Conf. Artif. Intell.*, vol. 33, no. 01, pp. 1182–1189, Jul. 2019.

[21] M.-S. Chung *et al.*, "SIoT selection, clustering, and routing for federated learning with privacy-preservation (full version)," Oct 2021. [Online]. Available: http://mnet.cs.nthu.edu.tw/NTHU-TechRep2021.pdf

[22] G. Xu *et al.*, "Trust2privacy: A novel fuzzy trust-to-privacy mechanism for mobile social networks," *IEEE Wireless Commun.*, vol. 27, no. 3, pp. 72–78, 2020.

[23] M. Duan *et al.*, "Self-balancing federated learning with global imbalanced data in mobile systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 1, pp. 59–71, 2021.

[24] S. Wang, Y.-C. Wu, M. Xia, R. Wang, and H. V. Poor, "Machine intelligence at the edge with learning centric power allocation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7293–7308, 2020.

[25] Z. Chen, D. Li, M. Zhao, S. Zhang, and J. Zhu, "Semi-federated learning," in *Proc. IEEE WCNC*, 2020.