Energy-Efficient Client Sampling for Federated Learning in Heterogeneous Mobile Edge Computing Networks

Jian Tang[1], Xiuhua Li[1*], Hui Li[1], Min Xiong[1], Xiaofei Wang[2], and Victor C. M. Leung[3, 4]
[1]School of Big Data & Software Engineering, Chongqing University, Chongqing, China
[2]TKLAN, College of Intelligence & Computing, Tianjin University, Tianjin, China
[3] College of Computer Science & Software Engineering, Shenzhen University, Shenzhen, China
[4]Department of Electrical & Computer Engineering, The University of British Columbia, Vancouver, Canada
*Corresponding author: Xiuhua Li
Email:kt4ngw@stu.cqu.edu.cn,{lixiuhua, h.li, xiongmin}@cqu.edu.cn, xiaofeiwang@tju.edu.cn, vleung@ieee.org

***Abstract*—To address network congestion and data privacy concerns, federated learning (FL) that combines multiple clients and a parameter server has been widely used in mobile edge computing (MEC) networks to process the abundant data generated by mobile clients. However, the existing client sampling methods do not adequately consider the data heterogeneity and system heterogeneity. Parameter server selects inappropriate clients to participate in the FL training process. This inevitably leads to slower convergence of the global model and higher energy consumption. In this paper, we design a client sampling model with the goal of selecting suitable clients to improve the energy efficiency of FL in heterogeneous MEC networks. Then we propose an energy-efficient client sampling strategy by quantifying the communication capability, computation capability and data quality of clients. Based on the quantization results, clients are assigned with a corresponding sampled probability. Simulation results show that our proposed strategy can effectively accelerate the convergence of the global model and reduce the energy consumption compared with the baseline schemes.**

I. Introduction

With the rapid development of mobile networks, the data generated by mobile clients/devices (e.g., smartphones, smartpads, and tablets) is showing an explosive growth tendency [1]–[3]. Traditional cloud computing utilizes powerful cloud servers for data processing and model training. This cloud-centric computing paradigm may bring unbearable issues such as privacy exposure and network congestion [4], [5]. To address these challenges, federated learning (FL) at mobile edge computing (MEC) networks is a feasible solution that allows clients to perform local model computation and upload updated local model parameters to the edge server for global model aggregation [6]–[8]. In this way, clients and edge server only exchange the lightweight model parameter and avoid raw data transmission, which can achieve data invisibility and relieve network congestion [5], [9].

Despite the above advantages of deploying FL at MEC networks, however, clients at the network edge are generally heterogeneous [10]–[13]. This heterogeneity is reflected in various dimensions. Firstly, there exists system heterogeneity among mobile clients, characterized by differences in central processing unit (CPU) clock speeds and transmission rates to the edge server. The laggards of FL prolong the time for each round of FL aggregation [10], [11]. Furthermore, data heterogeneity is another prevalent aspect, as different clients may exhibit diverse preferences for data collection [12], [13]. In other words, the quality of datasets from different clients is inconsistent, which negatively affects the convergence of the model. The system and data heterogeneity lead to longer model convergence time [10]–[13]. Therefore, it is necessary to study efficient client sampling strategies for FL in heterogeneous MEC networks.

Recently, common client sampling strategy in most existing FL studies is the same uniform sampling as FedAvg [6], that is, all clients are selected with the same sampled probability [12]. Although uniform sampling ensures that the contribution degree of each client to the global model is unbiased. However, the data quality of each client is not the same, and there will be a large bias among the trained local models [10]. Rizk *et al*. [14] proposed an importance sampling method based on the distribution of client data, where the sampled probability of an important client is higher. Li *et al*. [15] proposed that clients with large datasets have higher sampling probabilities. However, these strategies only considered the data heterogeneity to optimize the performance of the model, and did not fully consider the optimization of communication time or the system heterogeneity of the clients. The authors in [16] optimized the selection of clients based on the system capability, but they did not fully consider the data quality of different clients. Besides, these studies ignored the optimization of energy consumption during model training of FL in heterogeneous MEC networks.

In this paper, we are motivated to design an energy-efficient client sampling (ECS) strategy with the consideration of communication capability, computation capability and data quality of clients. The proposed ECS strategy can accommodate the heterogeneity of clients at MEC networks while minimizing the energy consumption during FL training process. The main contributions of this paper are summarized as follows:

- We analyze the system and data heterogeneity of clients in heterogeneous MEC networks and design a client sampling model to select suitable clients in the FL training

process for green communication orientation.
- We propose an energy-efficient client sampling strategy for FL in heterogeneous MEC networks to determine sampling probability for each client to reduce energy consumption while ensuring model convergence.
- We evaluate the proposed ECS strategy by conducting extensive experiments on typical models and datasets. Simulation results show that the proposed strategy can significantly reduce energy consumption.

The remainder of this paper is organized as follows. Section II presents the system model and formulates the optimization problem. The proposed client sampling strategy is presented in Section III. Section IV evaluates the performance of the proposed strategy. Finally, Section V concludes this paper.

II. System Model and Problem Formulation

A. FL Model in Heterogeneous MEC Networks

As shown in Fig. 1, we consider a heterogeneous MEC network of FL with a centralized parameter server (indexed as 0) and some mobile clients (denoted by $\mathcal{N} = \{1, 2, ..., k, ...\}$), in which each mobile client has a local dataset $D_{k\in\mathcal{N}}$. The data samples in D_k can be represented by (x_d, y_d), where x_d is feature vector, y_d is label vector, and d is index of sample. $|\cdot|$ denotes numbers of $\cdot$. Then, $|D_k|$ is the number of data samples in dataset D_k and $D = \sum_{k=1}^{|\mathcal{N}|} |D_k|$ denote total number of data samples from all mobile clients. To fully utilize D to train a global model in FL, parameter server and mobile clients collaborate to participate in model updates.

The FL process in the heterogeneous MEC network can be described as follows: a) At t-th round of communication of FL, parameter server randomly selects a portion of mobile clients to participate in FL; b) parameter server sends the global parameters w_0^{t-1} to selected mobile clients, and then the clients iteratively trains κ times based on D_k to get their local model; c) clients uploads their local model parameters w_k^t to parameter server; d) parameter server receives these parameters and aggregates them into global parameters w_0^t. To minimize the local loss, the local model training by mobile clients can be expressed as

$$\mathcal{L}(w_k^t) = \frac{1}{|D_k|} \sum_{(x_d, y_d)\in D_k} \ell(x_d, y_d, w). \tag{1}$$

The above local loss is generally challenging to find optimal solution through traditional methods due to its non-convex properties. However, the gradient descent algorithm can gradually approach the optimum. Therefore, the local iteration w of k-th client at t-th round of communication of FL can be given by

$$w_k^t = w_k^{t-1} - \lambda \nabla \mathcal{L}(w_k^{t-1}), \forall k \in \mathcal{N}, \tag{2}$$

where $\lambda > 0$ is the learning rate.

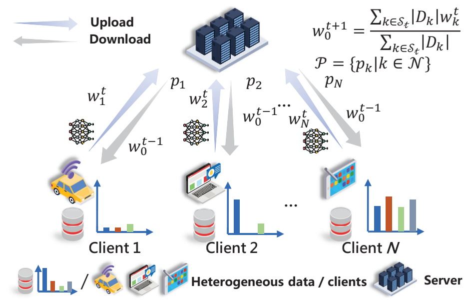


Fig. 1. Illustration of FL in a heterogeneous MEC network.

B. Mobile Clients in Heterogeneous MEC Networks

*1) **System Heterogeneity**:* System heterogeneity means that mobile clients have different computation and communication capabilities in FL. Denote f_k as the CPU clock speed of client k. The differences in CPU clock speed of mobile clients can be expressed as

$$f_k \neq f_j, \exists k, j \in \mathcal{N}, k \neq j. \tag{3}$$

Denote achievable transmission rate between client k and parameter server by $\mathcal{R}_k$. The clients get different rates with different types of communication protocols, which can be expressed as

$$\mathcal{R}_k \neq \mathcal{R}_j, \exists k, j \in \mathcal{N}, k \neq j. \tag{4}$$

In FL, there exists stragglers effect [9], that can lead to longer convergence time for global model. Specifically, clients that have weak computation or communication capability may become stragglers in the communication round, negatively affecting efficiency of FL training process.

*2) **Data Heterogeneity**:* Data heterogeneity refers to the different data distribution among mobile clients. The problem of non-independent and identical distribution (Non-IID) is widespread among mobile clients due to the different preferences for gathering data [12]. Non-IID can be measured by the size of dataset among clients and the degree of imbalance between labels. Dataset sizes of different mobile clients may vary greatly, which can be denoted by

$$|D_k| \neq |D_j|, \exists k, j \in \mathcal{N}, k \neq j. \tag{5}$$

Similarly, in some mobile clients, the number of samples in different labels may differ widely, denoted as

$$|D_{k,m}| \neq |D_{k,n}|, \exists m, n \in |y|, m \neq n, \exists k \in \mathcal{N}, \tag{6}$$

where $|D_{k,m}|$ denotes number of samples that belong to label m in data of client k and $|y|$ denotes total number of labels.

C. Client Sampling Model in Heterogeneous MEC Networks

Currently, client sampling in most FL studies is consistent with FedAvg with equal sampling probability. However, there exists heterogeneity in data and system among mobile clients. Clients with good or bad data quality, strong or weak computation and communication capabilities should not be selected with equal probability. We need to design a suitable client sampling model for FL in the heterogeneous MEC network.

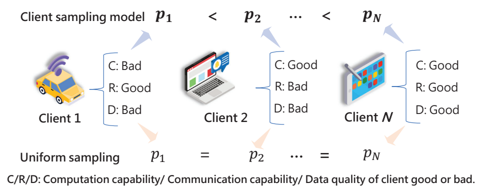


Fig. 2. Difference between client sampling model and uniform sampling.

More specifically, as shown in Fig. 2, clients with higher computation capability, lower energy consumption, and better data quality should be more involved in FL training with a greater probability.

Denote the set of clients participating in t-th round as $\mathcal{S}_t$. In uniform sampling, $\mathcal{S}_t$ is sampled by probability distribution $\mathcal{P}_{us} = \{p_k | k \in \mathcal{N}\}$, where $p_k = \frac{1}{|\mathcal{N}|}$. In this paper, we aim to reduce the energy consumption by FL training in the case of system and data heterogeneity by improving the probability distribution of client sampling. We set $\mathcal{P} = \{p_k | k \in \mathcal{N}\}$ as the probability distribution for client sampling and select a group of clients $\mathcal{S}_t$ to participate in FL training at t-th round communication. The global model parameter at t-th round communication w_0^t can be calculated as

$$w_0^t = \frac{\sum_{k \in \mathcal{S}_t} |D_k| w_k^t}{\sum_{k \in \mathcal{S}_t} |D_k|}. \tag{7}$$

And, the loss function $\mathcal{L}(w_0^t)$ of t-th round communication global model is recorded as

$$\mathcal{L}(w_0^t) = \frac{\sum_{k \in \mathcal{S}_t} |D_k| \mathcal{L}(w_k^t)}{\sum_{k \in \mathcal{S}_t} |D_k|}. \tag{8}$$

D. Latency and Energy Consumption Model

1) Latency Model: The latency of FL is mainly reflected in model parameter upload and local iterative training. Since the bandwidth resources of the parameter server are relatively abundant, thus, we ignore the latency of sinking global model to the mobile clients. In t-th communication, the latency of uploading model parameters for client k can be calculated as

$$t_k^{upload} = \frac{\mathcal{S}}{\mathcal{R}_k}, \forall k \in \mathcal{N}, \tag{9}$$

where $\mathcal{S}$ denotes size of model parameters. Denote $\mathcal{C}$ as the CPU cycles required to compute one sample, therefore, the latency of local iterative training is given by

$$t_k^{comp} = \frac{\mathcal{C} |D_k| \kappa}{f_k}, \forall k \in \mathcal{N}. \tag{10}$$

2) Energy Consumption Model: Since the energy consumption of downloading model parameters is much smaller than that of uploading them, we ignore the energy consumption of downloading parameters by mobile clients. The total energy consumption depends on the energy consumption of uploading models and local iterative training. The energy consumption in the local iterative training can be calculated as

$$E_k^{comp} = \rho f_k^2 \mathcal{C} |D_k| \kappa, \forall k \in \mathcal{N}, \tag{11}$$

where ρ denotes effective capacitance of mobile client k. Furthermore, the energy consumption of model uploading for each mobile client can be expressed as

$$E_k^{upload} = q_k t_k^{upload}, \forall k \in \mathcal{N}, \tag{12}$$

where q_k is the transmit power of mobile client k.

To sum up, the energy consumption of t-th round communication of FL is expressed as

$$E^t = \sum_{k \in \mathcal{S}_t} (E_k^{comp} + E_k^{upload}). \tag{13}$$

E. Problem Formulation

In the considered heterogeneous MEC network, our goal is to find a suitable probability distribution $\mathcal{P}$ under the client sampling model to sample the clients for minimizing energy consumption during the FL training process. The number of clients involved in each round of FL is fixed, and we only sampling clients based on the probability of each client being selected, so the optimization problem can be formulated as

$$\min_{\{\mathcal{P}\}} \sum_{t=1}^{T} E^t, \tag{14a}$$

$$\begin{aligned} s.t. \quad & \mathcal{L}(w_0^T) \leq \mathcal{L}, && (14b)\\ & |\mathcal{S}_t| = C \times |\mathcal{N}|, \forall t \in T, && (14c)\\ & \sum_{k=1}^{N} p_k = 1, && (14d)\\ & \mathcal{P} = \{p_k | k \in \mathcal{N}\}, p_k > 0, && (14e)\\ & f_k, \mathcal{R}_k, D_k, \forall k \in \mathcal{N}. && (14f) \end{aligned}$$

The constraints are explained as follows. Constraint (14b) shows that the loss function of the global model must reach the convergence value $\mathcal{L}$ at the end of communication. Constraint (14c) shows that FL needs to receive $|\mathcal{S}_t|$ clients for global model aggregation at t-th round of communication. Constraint (14d) denotes that the sum of the sampling probabilities is equal to 1. Constraint (14e) represents the sampling probability distribution of clients. Constraint (14f) denotes the system and data heterogeneity of clients.

III. Proposed Framework Design

A. Sampling Strategy for FL in Heterogeneous MEC Networks

To minimize energy consumption during the FL training process, our sampling strategy takes into account the computation capability communication capability, and data quality of mobile clients. The probability of each mobile client being selected is determined on the basis of these three aspects. Then, the parameter server selects mobile clients to participate in FL training based on this probability. The procedure of the proposed client sampling strategy is shown in Algorithm 1 as

- *Step* 1: Obtain clients information. The parameter server sends FL request information to the mobile clients, and the clients willing to participate in FL send their encrypted information of data to the server as well as client system information, etc.

- *Step* 2: Calculate the score. After the server collects information from mobile clients, it will quantify their computation capability C_k^{Score}, communication capability B_k^{Score} and data quality D_k^{Score}.
- *Step* 3: Calculate the probability of clients being selected. The probability of each client being selected is weighted after normalizing the computation capability, communication capability, and data quality scores of the clients in *Step* 2, respectively.
- *Step* 4: FL training process. Clients are selected to participate in FL training based on the probability distribution calculated in *Step* 3, which is in lines 8 - 22 of Algorithm 1. When the loss or communication round reaches a preset value training is aborted.

Since the work in this paper is to study client sampling strategy, we will next describe *Step* 1-3 in detail.

B. Details of Client Sampling Strategy

1) ***Obtain Clients Information****:* In the strategy design, in order to calculate the scores of clients in *Step* 2, parameter server collects client's information about data, system, etc. Among them, system information of clients refers to its CPU clock speed f_k, achievable transmission rate $\mathcal{R}_k$, etc. Data information, on the other hand, is used to measure the heterogeneity among different clients. We define the category imbalance D^{imb}, $|D_k|$, and D^{dis} to denote the category imbalance, dataset size, and data distribution of the clients, respectively. This is because the degree of data heterogeneity is generally determined by these three aspects.

2) ***Calculate the Score****:* After the parameter server obtains the information of clients, it calculates the data quality, communication capability, and computation capability scores of clients. The data quality score is calculated as

$$D_k^{Score} = D_k^{dis} \times |D_k| \times D_k^{imb}, \forall k \in \mathcal{N}. \tag{15}$$

We give the definitions of D_k^{dis} and D_k^{imb} as below.

The data distribution can be defined in terms of the mean and standard deviation of the client's data, and for simplicity, we utilize the client's data mean D_k^{dis} to represent it (Note that mobile clients can use vector rotation to protect privacy and not degrade performance [17]). The server computes the global mean D_0^{dis} after collecting D_k^{dis} from each client. Subsequently, D_k^{dis} is compared with D_0^{dis}, with smaller differences indicating better data quality. The degree of category balance of the data can be measured by the Gini coefficient (16), where a smaller Gini coefficient indicates that the client's dataset tends to be more homogeneous in terms of categories, and vice versa for a relatively balanced distribution of categories.

$$D_k^{imb} = 1 - \sum_{m=1}^{|y|} \left(\frac{|D_{k,m}|}{\sum_{m=1}^{|y|} |D_{k,m}|}\right)^2, k \in \mathcal{N}. \tag{16}$$

The computation capability score is calculated as

$$C_k^{Score} = 1/\left(\gamma \frac{t_k^{comp}}{t_k^{compM}} + (1-\gamma) \frac{E_k^{comp}}{E_k^{compM}}\right), \forall k \in \mathcal{N}, \tag{17}$$

Algorithm 1 Energy-efficient Client Sampling (ECS) Strategy

Input: Mobile clients set $\mathcal{N} = \{1, 2, ...k, ...\}$, Parameter Server.

Output: The decision of $\{[\mathcal{S}_t] | t \in T\}$ and the final FL model.

1: The server collects information on the CPU clock speed f_k, achievable transmission rate $\mathcal{R}_k$, data quality D^{imb}, $|D_k|$ and D^{dis} of mobile clients.
2: Calculate $\{D_k^{Score} | k \in \mathcal{N}\}$ by (15).
3: Calculate $\{C_k^{Score} | k \in \mathcal{N}\}$ by (17).
4: Calculate $\{B_k^{Score} | k \in \mathcal{N}\}$ by (18).
5: The $\{[D_k^{Score}, C_k^{Score}, B_k^{Score}] | k \in \mathcal{N}\}$ are normalized by (19), respectively.
6: Compute the probability of the mobile clients being selected $\mathcal{P} = \{p_k | k \in \mathcal{N}\}$ by (20).
7: Initialize global model parameters w_0^0.
8: **for** t in 1 to T **do**
9:    **—[Processing at the parameter server]**.
10:    $m \Leftarrow max(C \times |\mathcal{N}|, 1)$, C set to 0.1.
11:    $\mathcal{S}_t \Leftarrow$ Random selection of m clients by the probability of the clients being selected $\{p_k | k \in \mathcal{N}\}$.
12:    **for** each client $k \in S_t$ **in paralled do**
13:       Execute local model updates via line 17.
14:    **end for**
15:    Global aggregation after receive local parameters.
16:    **—[Processing at the mobile clients]**.
17:    Receive the global model from parameter server.
18:    **for** $e = 1$ to κ **do**
19:       $w_k^{t+1} \Leftarrow w_k^t - \lambda \nabla L(w_k^{t-1}), \forall k \in \mathcal{S}_t$.
20:    **end for**
21:    Upload the updated model to parameter server.
22: **end for**

where t_k^{compM} and E_k^{compM} refer to the maximum value of local computing latency and energy efficiency consumption in $|\mathcal{N}|$ clients respectively.

The communication capability score is calculated as

$$B_k^{Score} = 1/\left(\beta \frac{t_k^{uoload}}{t_k^{uoloadM}} + (1-\beta) \frac{E_k^{uoload}}{E_k^{uoloadM}}\right), \forall k \in \mathcal{N}, \tag{18}$$

where $t_k^{uploadM}$ and $E_k^{uploadM}$ refer to the maximum value of parameter upload latency and energy efficiency consumption among $|\mathcal{N}|$ clients respectively.

3) ***Calculate the Probability of Clients Being Selected****:* The scores are normalized and then weighted to derive the probability of each selected client, calculated as

$$\mathcal{Z}_k^{nor} = \frac{\mathcal{Z}_k^{Score}}{\sum_{k=1}^{|\mathcal{N}|} \mathcal{Z}_k^{Score}}, \mathcal{Z} \in \{D, C, B\}, \forall k \in \mathcal{N}, \tag{19}$$

where $\mathcal{Z}_k^{nor}$ refers to the normalized value of $\mathcal{Z}_k^{Score}$.

$$p_k = (\omega_1 D_k^{nor} + \omega_2 C_k^{nor} + \omega_3 B_k^{nor}) / \mathcal{T}, \forall k \in \mathcal{N}, \tag{20}$$

where p_k refers to the probability that the k-th client is selected, $\omega_1, \omega_2, \omega_3$ are the set hyperparameters, $\mathcal{T} = \omega_1 \sum_{k \in \mathcal{N}} D_k^{nor} + \omega_2 \sum_{k \in \mathcal{N}} C_k^{nor} + \omega_3 \sum_{k \in \mathcal{N}} B_k^{nor}$.

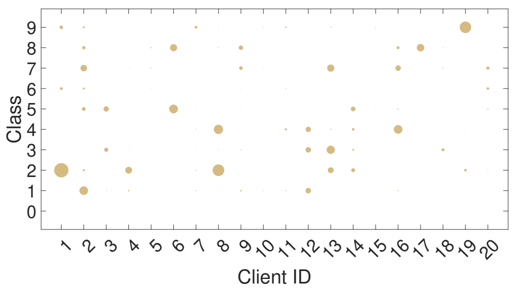


Fig. 3. Illustration of data heterogeneity on MNIST among the top 20 clients, where the x-axis denotes the client ID and the y-axis denotes the class labels, with a larger scatter denoting a greater number of samples per label assigned to each client and vice versa.

IV. Simulation Results

A. Experiment Setup

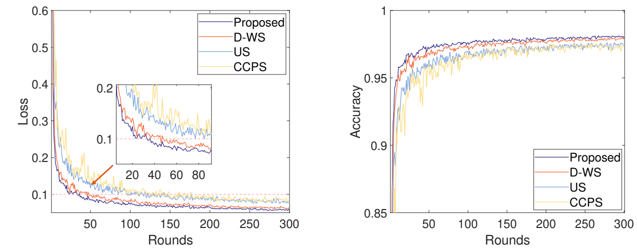
1) Datasets and Models: We evaluate the proposed ECS strategy using MNIST and one CNN model. MNIST contains 60,000 training images and 10,000 test images of handwritten digits 0-9. The CNN model consists of two convolutional layers, two pooling layers, and fully connected layers.

2) Parameter Setup: To evaluate effect of the proposed ECS strategy, we consider simulation for training one global model of a heterogeneous MEC network (system and data heterogeneity of clients are explained below) to build FL and conduct extensive experiments based on Pytorch (Intel Core i5-13600KF, NVIDIA GeForce RTX 4070 Ti). f_k is randomly distributed at [0.1, 3] GHz, $\mathcal{R}_k$ is calculated by Shannon's formula, $\mathcal{R}_k = B_k log_2(1+\frac{q_k h_k}{N_0 B_k})$, where transmission bandwidth B_k is randomly set at [1, 20] MHz. This sets as system heterogeneity of clients. Ratio of h_k to N_0 is fixed at 8 Hz/w [18], transmit power q_k is 1W. Number of mobile clients $|\mathcal{N}|$ set to 100. The CNN model parameter size $\mathcal{S}$ is 6.35MB. For localized training, it takes $\mathcal{C} = 10^4$ CPU cycles to compute one sample data. In addition, to improve convergence of global model in FL, we set number of local iterations $\kappa = 5$, learning rate $\lambda = \frac{0.1}{t+1}$, and effective capacitance $\rho = 10^{-26}$.

3) Metrics: Our goal is to minimize energy consumption during FL training while converging the global model. Therefore, the efficiency of the proposed ECS strategy is quantified as the energy consumed by the global model to converge in the heterogeneous MEC network. In addition, the stability of the ECS strategy is also worth considering in different heterogeneous MEC networks.

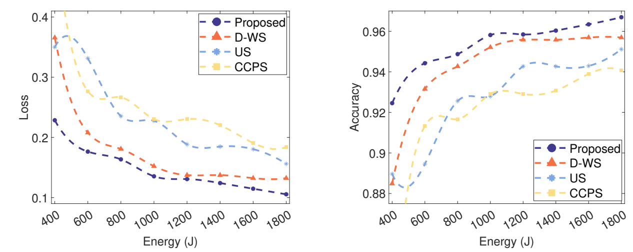
4) Baselines: To compare the advantages of the proposed ECS strategy, the following three baseline schemes are available for comparison:

- **Data-based weight sampling (D-WS)** [15]: The probability of a client being selected depends on dataset size.
- **Uniform sampling (US)** [6], [10], [12]: Each client is selected with equal probability. This approach is widely used in various research.
- **Computing and communication priority sampling (CCPS)**: Sampling strategy that do not take into account dataset quality. This allows us to compare the advantages of our proposed data quality options.



(a) Accuracy vs. Number of Rounds (b) Loss vs. Number of Rounds

Fig. 4. Accuracy and Loss with Number of Rounds ($|\mathcal{N}|$ = 100).



(a) Loss with Energy Constraint (b) Accuracy with Energy Constraint

Fig. 5. Energy Consumption of Model Training Process ($|\mathcal{N}|$ = 100).

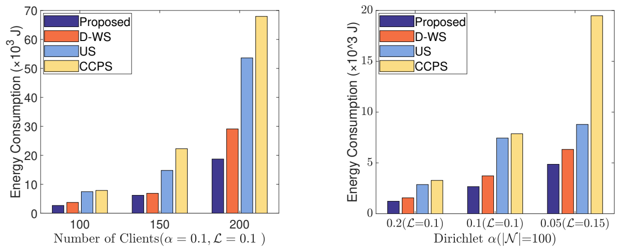
B. Experimental Results

We use Dirichlet distribution [19] with $\alpha = 0.1$ to partition the 60000 training images to Non-IID scenes as data heterogeneity of clients in the heterogeneous MEC network, where smaller α means more data heterogeneity of clients, and $\mathcal{L}$ varies with α. Fig. 3 shows the results for the top 20 clients. To ensure the reliability of our experiments, we set different random seed numbers in more than 5 comparison experiments.

1) Loss and Accuracy with Number of Rounds: Fig. 4 compares the global model under the proposed ECS, D-WS, CCPS, and US strategies in terms of accuracy and loss. As can be seen, the ECS converges ($\mathcal{L} = 0.1$) in fewer rounds (as shown in Fig 4(a)) and improves accuracy faster (as shown in Fig 4(b)) than other baseline schemes. This implies that the proposed ECS strategy can achieve better training performance on mobile clients. In addition, in contrast to ECS, it can be seen that the loss fluctuates more in training for all three baseline schemes. This is due to the fact that ECS takes into account data quality of mobile clients. This also indicates that the ECS has some stability in FL training.

2) Energy Consumption of Model Training Process: Fig. 5 illustrates the energy consumption of the FL training process. As shown in Fig. 5(a) and Fig. 5(b), the proposed ECS strategy can achieve smaller loss and higher accuracy under certain energy consumption constraints. Specifically, given an energy consumption constraint of about 800*Joule(J)*, the global model's loss can be reduced to 0.16 under the ECS strategy, compared to only 0.18, 0.23, and 0.26 for the D-WS, US, and CCPS, respectively. The global model accuracy under the ECS strategy is 94.87%, while the accuracy of D-WS, US, and CCPS are 94.27%, 92.80%, and 91.66%, respectively.

3) Energy Consumption under Different Heterogeneous MEC Networks: Fig. 6 illustrates the energy consumption of the proposed ECS strategy under different heterogeneous MEC



(a) Number of Clients (b) Dirichlet α ($|\mathcal{N}|$ = 100)

Fig. 6. Energy consumption under different heterogeneous MEC networks.

networks. Fig. 6(a) shows the energy consumption for model convergence with different numbers of clients. In contrast, the proposed ECS strategy consumes less energy than the other three strategies with $|\mathcal{N}|$ set to 100, 200 and 300. Specifically, when $|\mathcal{N}|$ is set to 100, the energy consumption of ECS is 1230J, while D-WS, US, and CCPS consume 1563J, 2871J and 3281J. Fig. 6(b) shows the energy consumption for model convergence under different Non-IID settings. As we can observe, the proposed ECS strategy consumes less energy than the other three baseline schemes in all settings, and the growth rate of energy consumption is much lower than the other baseline schemes. Therefore, we can conclude that energy consumption of FL process under the proposed ECS strategy is still lower than three baseline schemes in different heterogeneous MEC networks.

V. Conclusion and Future Work

In this paper, we have proposed a client sampling strategy for FL in heterogeneous MEC networks. This effectively reduces the energy consumption in FL training. Specifically, we quantify the communication capability, computation capability, and data quality of mobile clients, and finally obtain the probability of each mobile client being selected by weighted. In the simulation results, the proposed ECS strategy can significantly reduce energy consumption compared to the baseline schemes, and the model is stable in different heterogeneous MEC networks. In the future, we will further study the scenarios with dynamic network environments, in which the data quality and transmission rate of mobile clients may not be static.

Acknowledgment

This work is supported in part by National Key R & D Program of China (Grant No. 2022YFE0125400), National NSFC (Grants No. 62372072, 62102053, 62072060, 92067206, 61972222), Chongqing Research Program of Basic Research and Frontier Technology (Grant No. cstc2022ycjhbgzxm0058), Key Research Program of Chongqing Science & Technology Commission (Grant No. cstc2021jscx-dxwtBX0019), Haihe Lab of ITAI (Grant No. 22HHXCJC00002), the Natural Science Foundation of Chongqing, China (Grant No. CSTB2022NSCQ-MSX1104), the General Program of Chongqing Science & Technology Commission (Grant No. CSTB2023TIAD-STX0035, CSTB2022TIAD-GPX0017, CSTB2022TIAD-STX0006), Regional Innovation Cooperation Project of Sichuan Province (Grants No. 2023YFQ0028, 24QYCX0019), Regional Science and Technology Innovation Cooperation Project of Chengdu City (Grant No. 2023-YF11-00023-HZ), Xiaomi Foundation, and Guangdong Pearl River Talent Recruitment Program (Grants No. 2019ZT08X603, 2019JC01X235).

References

[1] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 869–904, Jan. 2020.

[2] B. Kar, W. Yahya, Y.-D. Lin, and A. Ali, "Offloading using traditional optimization and machine learning in federated cloud–edge–fog systems: A survey," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 2, pp. 1199–1226, Jan. 2023.

[3] H. Hellström, V. Fodor, and C. Fischione, "Federated learning over-the-air by retransmissions," *IEEE Trans. Wireless Commun.*, pp. 1–13, Apr. 2023.

[4] 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 J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Mar. 2019.

[5] W. Y. B. Lim, J. S. Ng, Z. Xiong, J. Jin, Y. Zhang, D. Niyato, C. Leung, and C. Miao, "Decentralized edge intelligence: A dynamic resource allocation framework for hierarchical federated learning," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 3, pp. 536–550, Jul. 2022.

[6] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Stat*, Apr. 2017, pp. 1273–1282.

[7] M. Skocaj, P. E. I. Rivera, R. Verdone, and M. Erol-Kantarci, "Uplink scheduling in federated learning: an importance-aware approach via graph representation learning," *arXiv preprint arXiv:2301.11903*, Jan. 2023.

[8] A. Imteaj, U. Thakker, S. Wang, J. Li, and M. H. Amini, "A survey on federated learning for resource-constrained IoT devices," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 1–24, Jul. 2022.

[9] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, Feb. 2020.

[10] Y. Xiao, X. Zhang, Y. Li, G. Shi, M. Krunz, D. N. Nguyen, and D. T. Hoang, "Time-sensitive learning for heterogeneous federated edge intelligence," *IEEE Trans. Mobile Comput.*, pp. 1–18, Jan. 2023.

[11] B. Luo, W. Xiao, S. Wang, J. Huang, and L. Tassiulas, "Tackling system and statistical heterogeneity for federated learning with adaptive client sampling," in *Proc. IEEE INFOCOM*, May 2022, pp. 1739–1748.

[12] J. Zhang, S. Guo, Z. Qu, D. Zeng, Y. Zhan, Q. Liu, and R. Akerkar, "Adaptive federated learning on non-iid data with resource constraint," *IEEE Trans. Commun.*, vol. 71, no. 7, pp. 1655–1667, Jul. 2022.

[13] J. Han, A. F. Khan, S. Zawad, A. Anwar, N. B. Angel, Y. Zhou, F. Yan, and A. R. Butt, "Heterogeneity-aware adaptive federated learning scheduling," in *Proc. IEEE BigData*, Dec. 2022, pp. 911–920.

[14] E. Rizk, S. Vlaski, and A. H. Sayed, "Federated learning under importance sampling," *IEEE Trans. Signal Process.*, vol. 70, pp. 5381–5396, Sep. 2022.

[15] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-iid data," in *Proc. ICLR*, Apr. 2020, pp. 1–12.

[16] A. Reisizadeh, I. Tziotis, H. Hassani, A. Mokhtari, and R. Pedarsani, "Straggler-resilient federated learning: Leveraging the interplay between statistical accuracy and system heterogeneity," *IEEE J. Sel. Areas Infor. The.*, vol. 3, no. 2, pp. 197–205, Jun. 2022.

[17] J. Ma, X. Sun, W. Xia, X. Wang, X. Chen, and H. Zhu, "Client selection based on label quantity information for federated learning," in *Proc. IEEE PIMRC*, Sep. 2021, pp. 1–6.

[18] F. Wu, X. Li, H. Li, Q. Fan, L. Zhu, X. Wang, and V. C. M. Leung, "Energy-time efficient task offloading for mobile edge computing in hot-spot scenarios," in *Proc. IEEE ICC*, Jun. 2021, pp. 1–6.

[19] T.-M. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," *arXiv preprint arXiv:1909.06335*, Sep. 2019.