河海大学
HOHAI UNIVERSITY

Local Performance Trade-Off in Heterogeneous Federated Learning with Dynamic Client Grouping

Yingchi Mao[a,b], Jun Wu[b], Yangkun Cheng[b], Ping Ping[a,b], and Jie Wu[c]

[a] Key Laboratory of Water Big Data Technology of Ministry of Water Resources, Hohai University, Nanjing, China
[b] School of Computer and Information, Hohai University, Nanjing, China
[c] Center of Networked Computing, Temple University, Philadelphia, USA

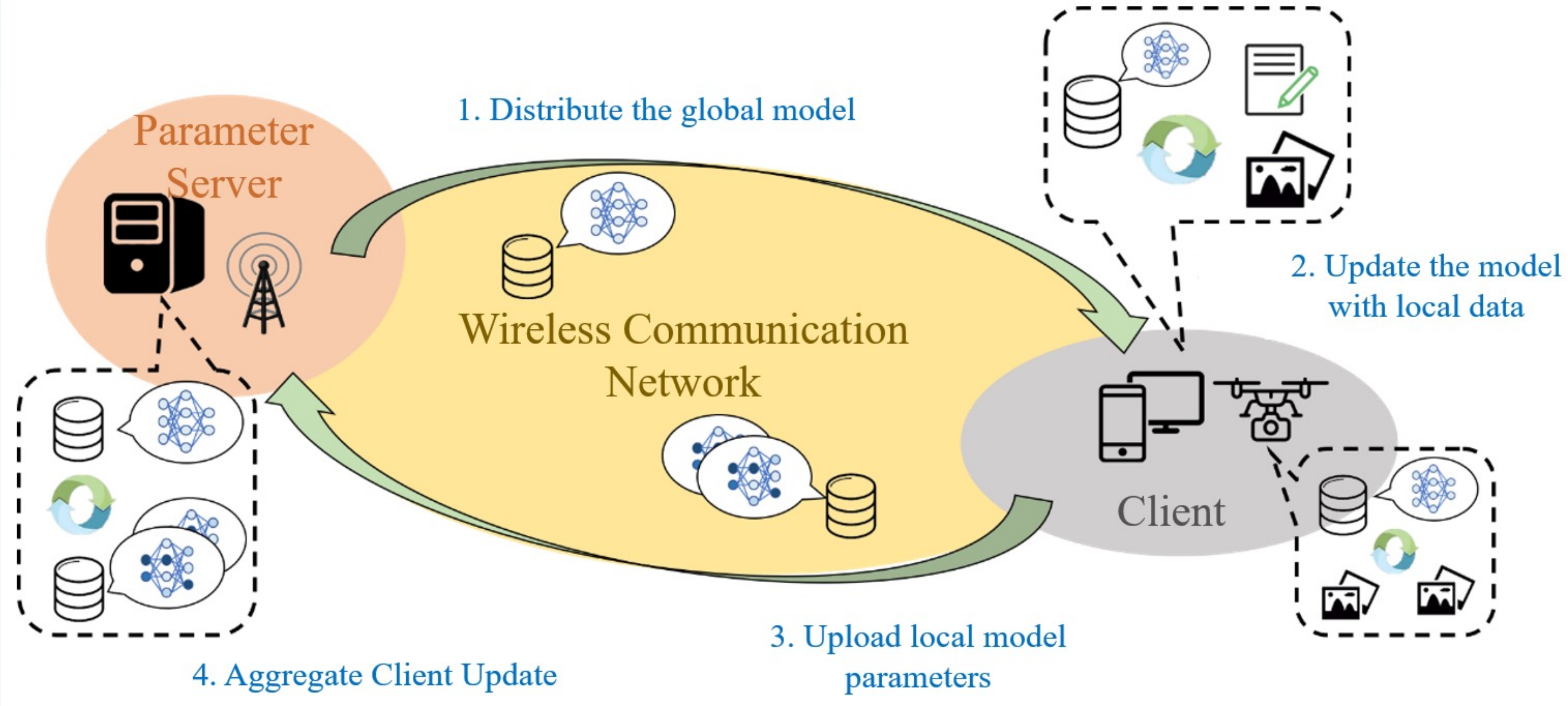
IEEE MASS
The 19th IEEE International Conference on Mobile Ad-Hoc and Smart Systems
Oct. 20, 2022

Content

Introduction

■ Background

Although client selection can coordinate large-scale clients for efficient training, the inherent characteristic of the scheme, i.e., the partial client participation can lead to **a performance bias** of the global model among clients.



Introduction

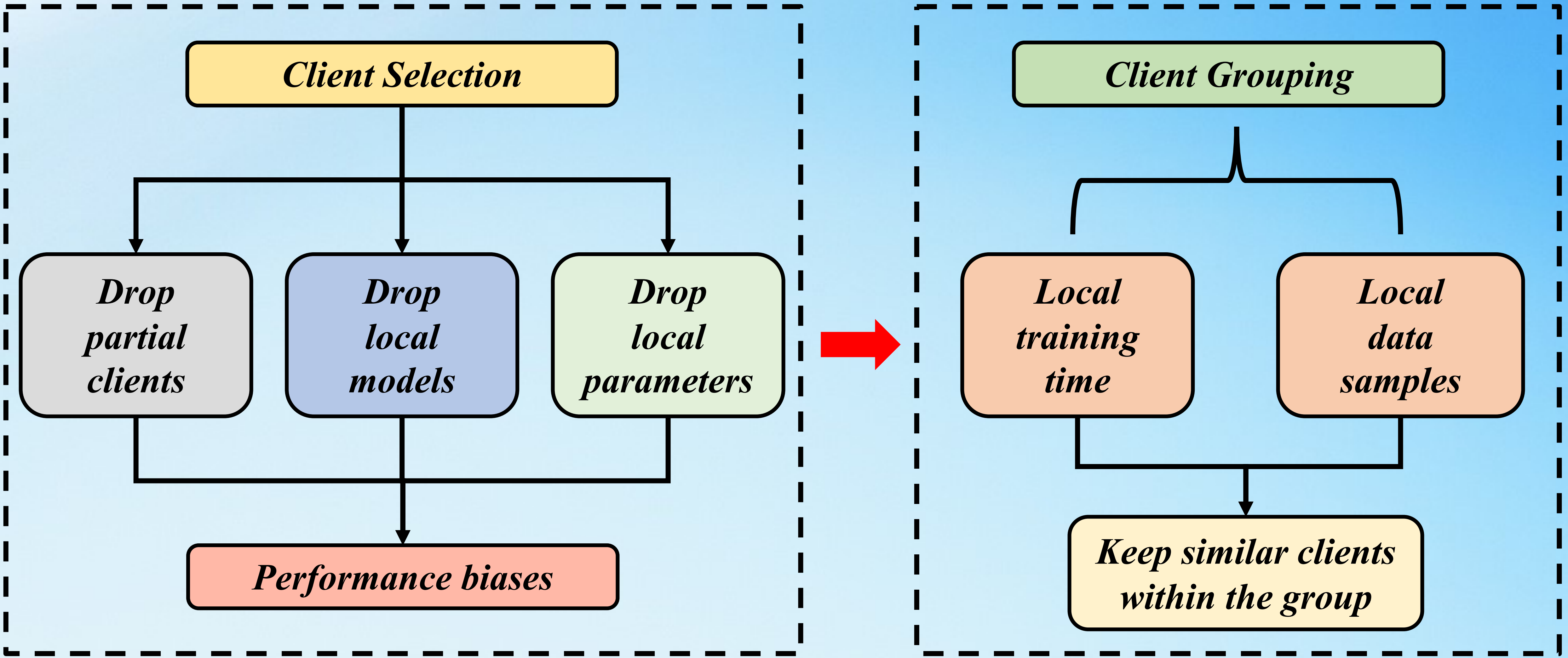
- **Optimization object**

Improve **unbalanced local performances** on different clients of the global model caused by client selection

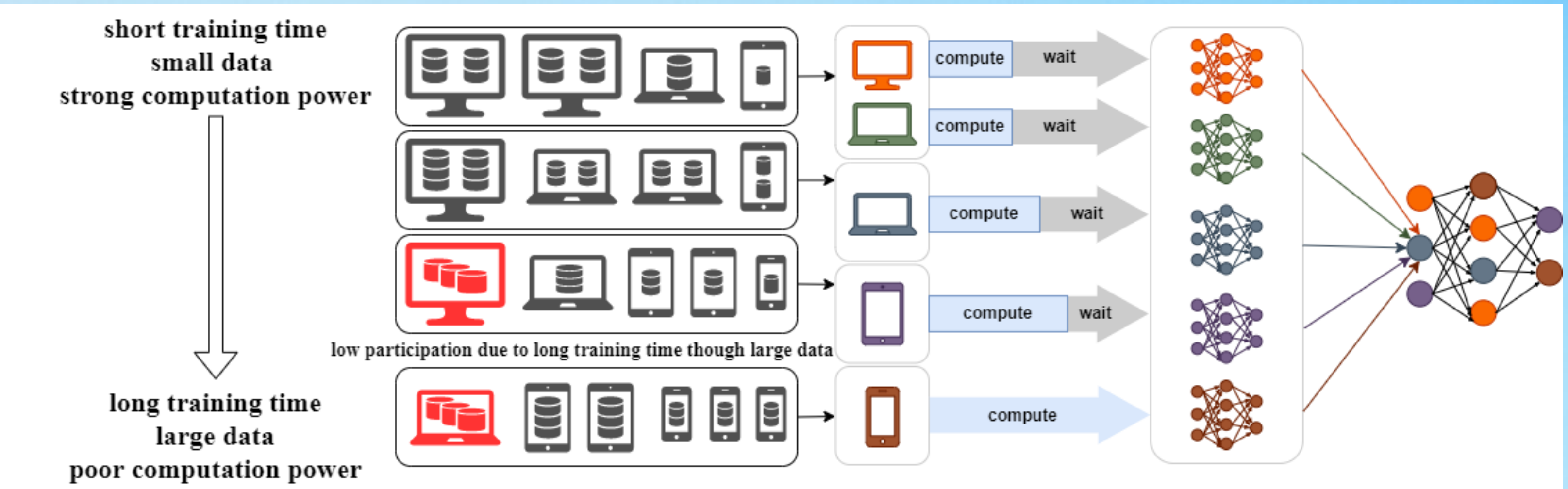
- **Goal**

Balance the client participation in local training while guaranteeing that clients are not discarded.

Related Work

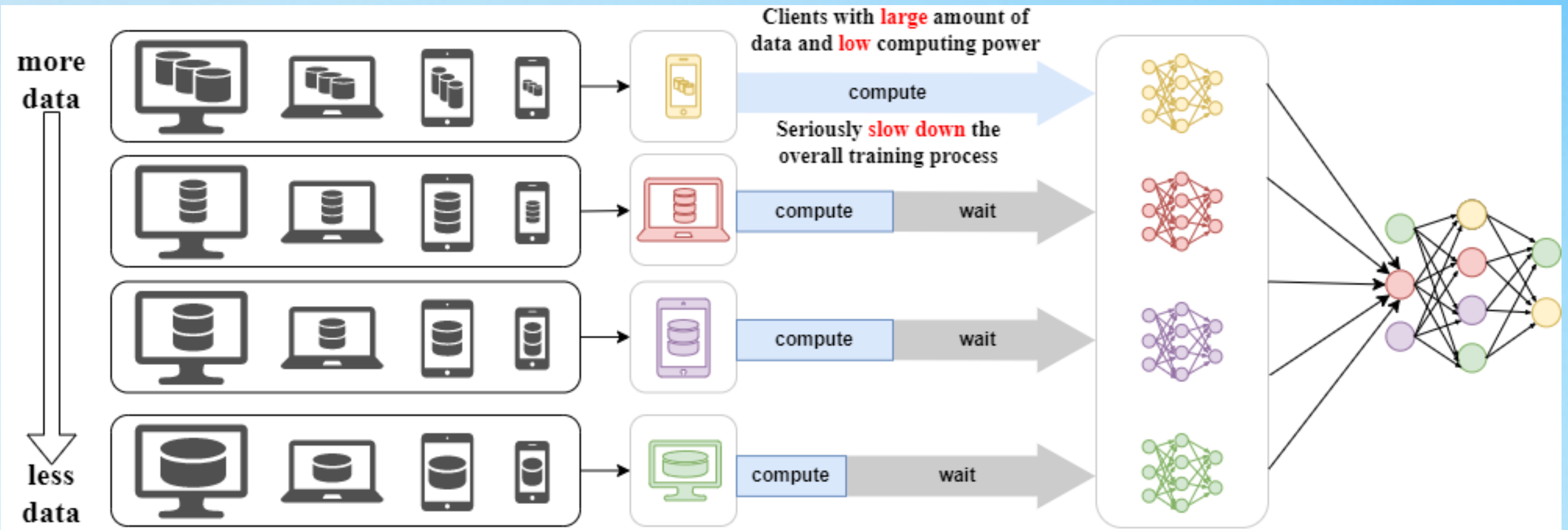


Client Grouping – local training time



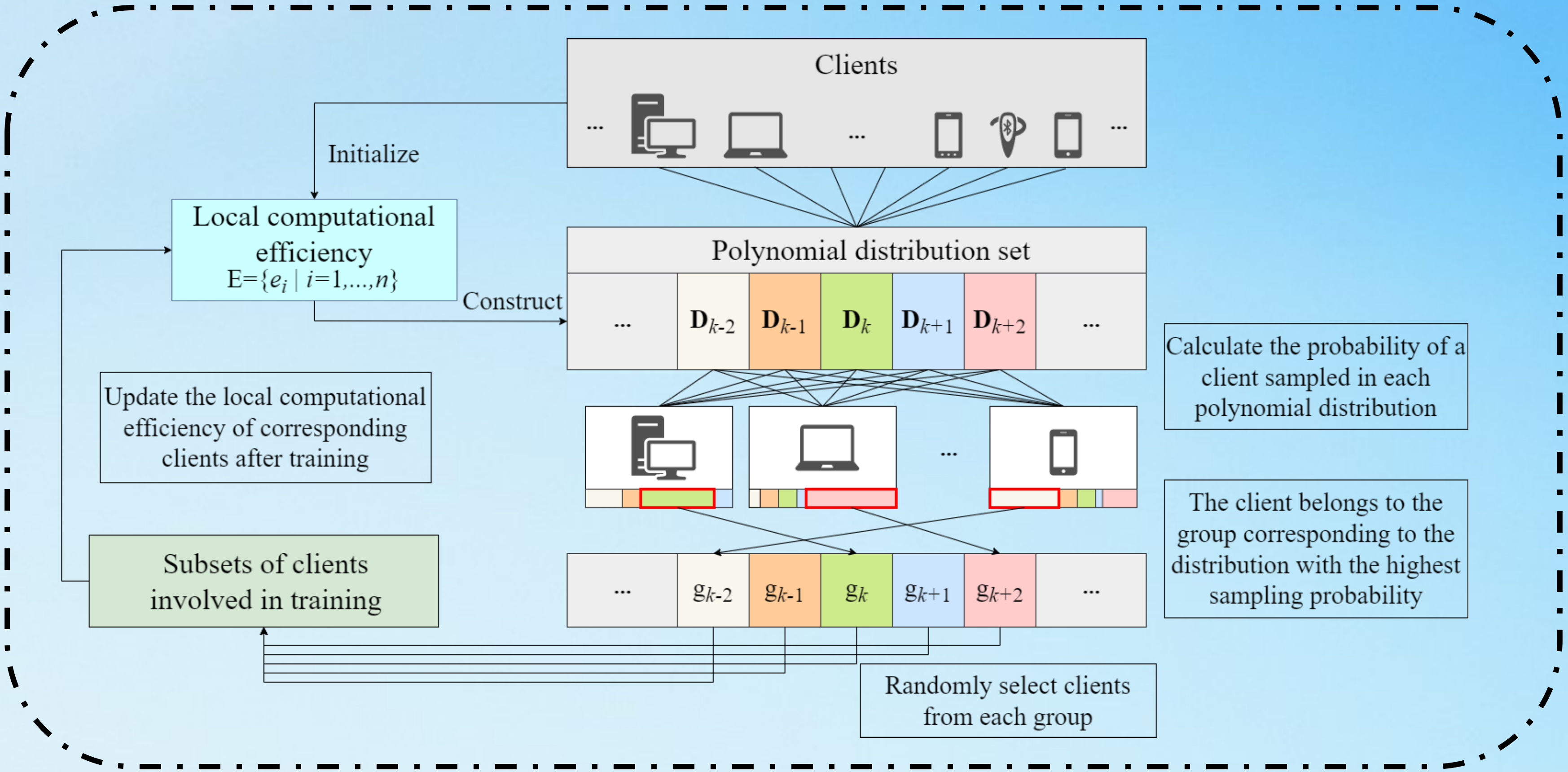
- Group clients according to local training time
- Clients with long training time due to large data volume are **despised**

Client Grouping – local data samples



- Group clients according to data volume
- Clients with large data volume seriously **slow down** the training process

Approach — Federated Learning Clients Dynamic Grouping Method based on Local Computational Efficiency



Calculate local computational efficiency → Construct polynomial distribution set → Group clients → Update local computational efficiency

Approach

Calculate local computational efficiency

Client i

$$e_i = \begin{cases} \boxed{|d_i|}, \boxed{c_i} & (r_{\text{latest}} = 0) \\ |d_i| / \boxed{t_i^{r_{\text{latest}}}}, c_i r_{\text{latest}} & (r_{\text{latest}} \neq 0) \end{cases}, i \in \{1, \ldots, n\}$$

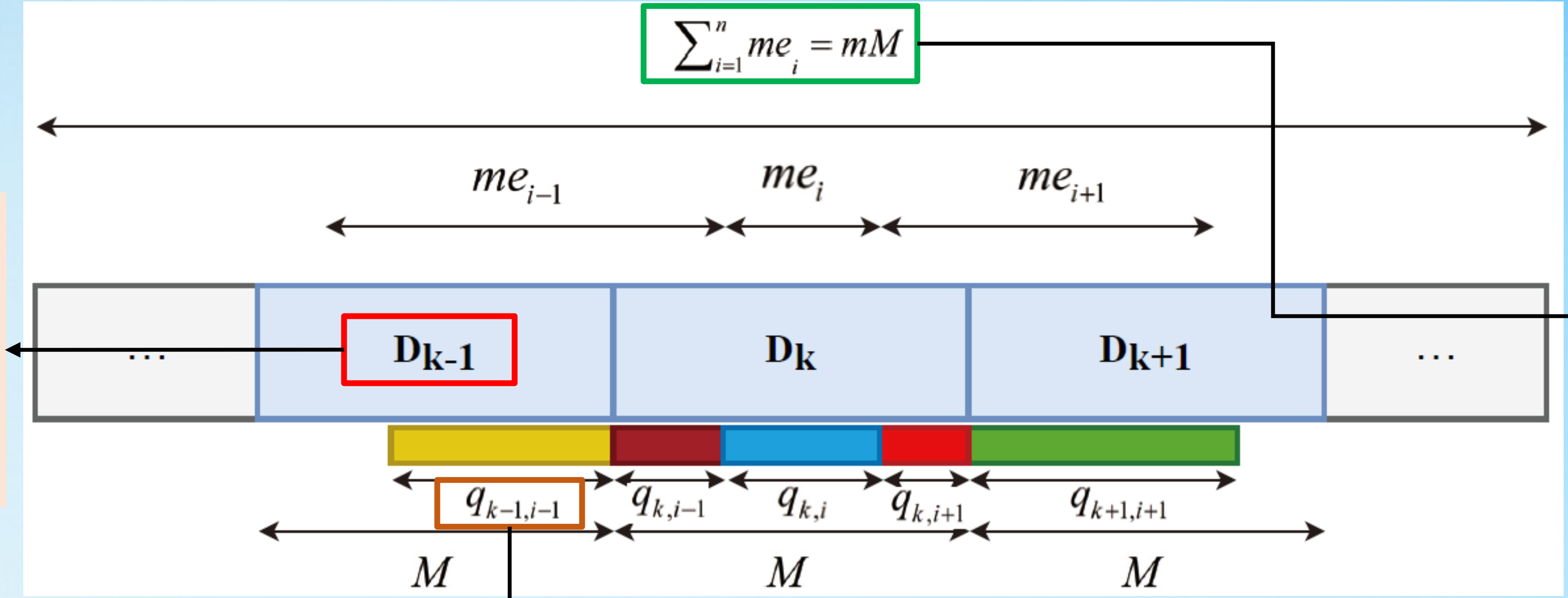
Local data quantity

Training time in the most recent iteration round

Approach

Construct Polynomial Distribution Set

A polynomial distribution set of length m



The priority of assigning clients' local computation efficiency is offered to the polynomial distribution

$$\forall k \in \{1,\ldots,m\}, \sum_{i=1}^{n} q_{i,k}^{r} = 1, q_{i,k}^{r} \geq 0$$

Sampling probability

Approach

Group Clients

Algorithm 1 Client Grouping

Input: n: number of clients, m: number of client groups, r: current iteration round, r_u: iteration round scheduled for updating client grouping results, $\{e_i|i = 1, ..., n\}$: client local computational efficiency set

Output: $\{q^{r_u}_{i,k}|i = 1, ..., n; \ k = 1, ..., m\}$: probability that a client belongs to each client group

```
if r%r_u == 0 then
    define k = 1
    define count = 0
    define M = Σ_{i=1}^{n} e_i
    for i = 1 to n do
        count = count + m e_i
        count = M α_i + β_i
        if α_i > k then
            (q^{r_u}_{i,k})' = M − β_{i−1}
            ∀l ≥ k + 1 s.t.(α_i − 1) − l ≥ 0, (q^{r_u}_{i,k})' = M
        end if
        (q^{r_u}_{i,α_i})' = β_i
        k = α_i
    end for
    return {q^{r_u}_{i,k} = (q^{r_u}_{i,k})'/M | i = 1, ..., n; k = 1, ..., m}
end if
```

Build a set of polynomial distributions for each client group

Output the probability that a client belongs to each polynomial distribution (Client Grouping)

Approach

Update Local Computational Efficiency

Algorithm 2 Local Computational Efficiency Update

Input: r: current iteration round, $\{|d_i| \, | i = 1, ..., n\}$: number of local data on each client, S^r: subset of clients involved in training in iteration round r

Output: $\{e_i | i = 1, ..., n\}$: Local computational efficiency set of each client

```
define E = {e_i = 0 | i = 1, ..., n}
if r == 1 then
    for i = 1 to n do
        e_i = d_i
    end for
else
    S^r = ClientGrouping(E)
    {t^r_{i'} | c_{i'} ∈ S^r} = Train()
    for c_{i'} in S^r do
        e_{i'} = |d_{i'}| / t^r_{i'}
    end for
    Descend(E)
end if
return E = {e_i | i = 1, ..., n}
```

Initialize clients’ local computational efficiency using their local data quantity

Update the local computational efficiency of clients participating in training

Experiment

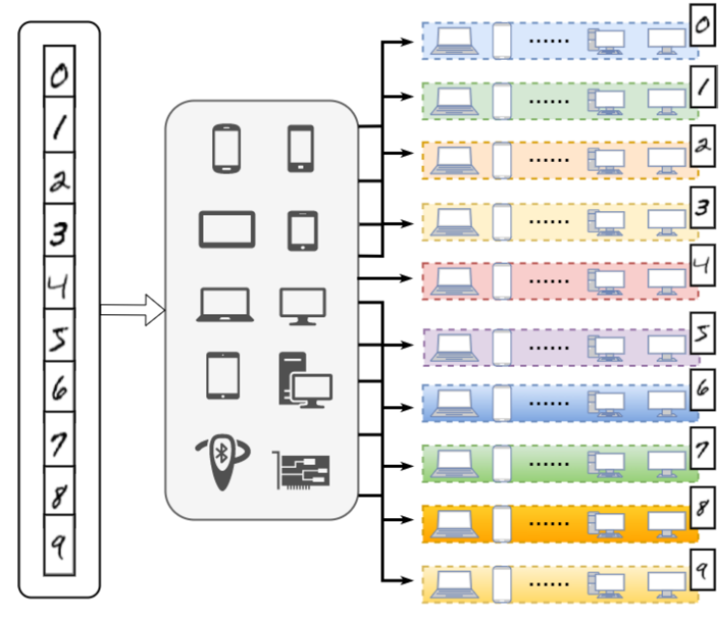
Learning Environment

Parameter Server			
CPU	AMD Ryzen5-3600@1.8GHz 12 cores		
GPU	NVIDIA GeForce RTX 2060 CUDA 1920 cores		
RAM	16GB		
Client			
Type	computational Resources	RAM	Number
Raspberry Pi 3B+	Cortex-A53@1.4GHz CPU×1	1GB	5
Nvidia Jetson Nano	Maxwell CUDA 128 cores GPU×1	4GB	10
0.8 CPU Docker	Ryzen5-3600@1.8GHz CPU×0.8	3GB	15
1.6 CPU Docker	Ryzen5-3600@1.8GHz CPU×1.6	3GB	15
2.4 CPU Docker	Ryzen5-3600@1.8GHz CPU×2.4	3GB	10
3.2 CPU Docker	Ryzen5-3600@1.8GHz CPU×3.2	3GB	10
4.0 CPU Docker	Ryzen5-3600@1.8GHz CPU×4.0	3GB	10
4.8 CPU Docker	Ryzen5-3600@1.8GHz CPU×4.8	3GB	10
5.6 CPU Docker	Ryzen5-3600@1.8GHz CPU×5.6	3GB	10
Nvidia Jetson TX2	Pascal CUDA 256 cores GPU×1	8GB	5

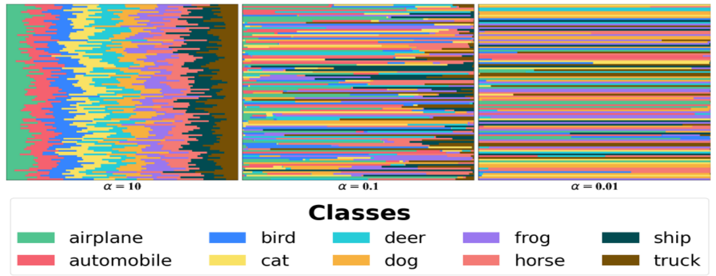
The clients and server are both on the same LAN and communicate each other via the PySyft's WebSocket protocol.

Experiment

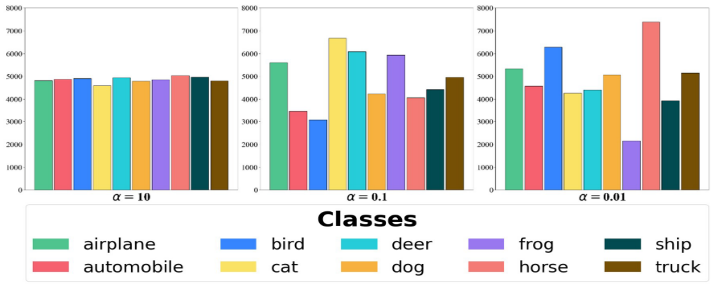
Dataset & Model



The generation of MNIST-Fed dataset



The percentage of local data belonging to categories for 100 clients in CIFAR-10-Fed dataset



The distribution of samples in each category in CIFAR-10-Fed dataset

Two different CNN models are chosen as experimental models for the MNIST-Fed dataset and the CIFAR-10-Fed dataset.

Experiment

Analysis of Hyperparameter Selection for FedGLCE

TABLE III
EXPERIMENTAL RESULTS OF FEDGLCE IN DIFFERENT GROUP UPDATE PERIODS (MNIST-FED, 5 GROUPS)

Group update period r_u	Variance of participation	Variance of local accuracy	Accuracy of global test
5	19.54	262,48	81.82%
10	17.92	61.41	86.99%
20	16.54	**37.47**	**91.14%**
50	**11.76**	80.41	90.74%

TABLE IV
EXPERIMENTAL RESULTS OF FEDGLCE IN DIFFERENT GROUP UPDATE PERIODS (MNIST-FED, 10 GROUPS)

Group update period r_u	Variance of participation	Variance of local accuracy	Accuracy of global test
5	60.00	24.81	94.93%
10	58.40	24.93	95.08%
20	53.94	19.71	94.51%
50	**43.42**	**16.36**	**95.46%**

Experiment

Analysis of Hyperparameter Selection for FedGLCE

TABLE V
EXPERIMENTAL RESULTS OF FEDGLCE IN DIFFERENT GROUP UPDATE PERIODS (CIFAR-10-FED, 5 GROUPS)

Group update period r_u	CIFAR-10-Fed($\alpha = 10$)			CIFAR-10-Fed($\alpha = 0.1$)			CIFAR-10-Fed($\alpha = 0.01$)		
	Var (par)	Var (local)	Acc (global)	Var (par)	Var (local)	Acc (global)	Var (par)	Var (local)	Acc (global)
5	**249.92**	**35.80**	**71.40%**	264.68	109.23	62.77%	267.72	526.39	48.56%
10	262.56	38.94	70.79%	259.04	144.64	62.19%	250.56	406.60	49.75%
20	265.04	38.59	69.84%	255.08	125.86	**63.47%**	269.62	405.93	**50.87%**
50	270.86	44.25	69.83%	**237.14**	**96.54**	63.36%	**241.34**	**387.73**	50.38%

TABLE VI
EXPERIMENTAL RESULTS OF FEDGLCE IN DIFFERENT GROUP UPDATE PERIODS (CIFAR-10-FED, 10 GROUPS)

Group update period r_u	CIFAR-10-Fed($\alpha = 10$)			CIFAR-10-Fed($\alpha = 0.1$)			CIFAR-10-Fed($\alpha = 0.01$)		
	Var (par)	Var (local)	Acc (global)	Var (par)	Var (local)	Acc (global)	Var (par)	Var (local)	Acc (global)
5	**1096.90**	**29.94**	**71.12%**	1172.14	137.26	64.38%	1196.06	385.10	53.05%
10	1126.62	36.00	70.54%	1133.98	103.86	65.32%	1168.56	**321.12**	53.04%
20	1134.32	35.29	71.02%	**1022.92**	**94.08**	**65.44%**	1116.88	373.48	53.72%
50	1165.22	41.05	70.94%	1115.62	113.21	65.10%	**1094.64**	353.47	**53.77%**

Analysis of Client Participation

TABLE VII
EXPERIMENTAL RESULTS OF CLIENT PARTICIPATION DIFFERENCES FOR CLIENT GROUPING APPROACHES

Comparison approaches	Evaluation metrics	Dataset			
		MNIST-Fed	CIFAR-10-Fed		
			Dir(10)	*Dir(0.1)*	*Dir(0.01)*
FedAvg	$Var(Fre)$	64.30	1323.4	1326.86	1322.5
FedDrop		210.02	1887.22	1900.14	1818.46
TiFL		**17.02**	1537.14	1527.5	1547.14
FedSS		78.36	1258.06	1259.86	1252.56
FedGLCE		53.94	**1134.52**	**1022.92**	**1094.64**

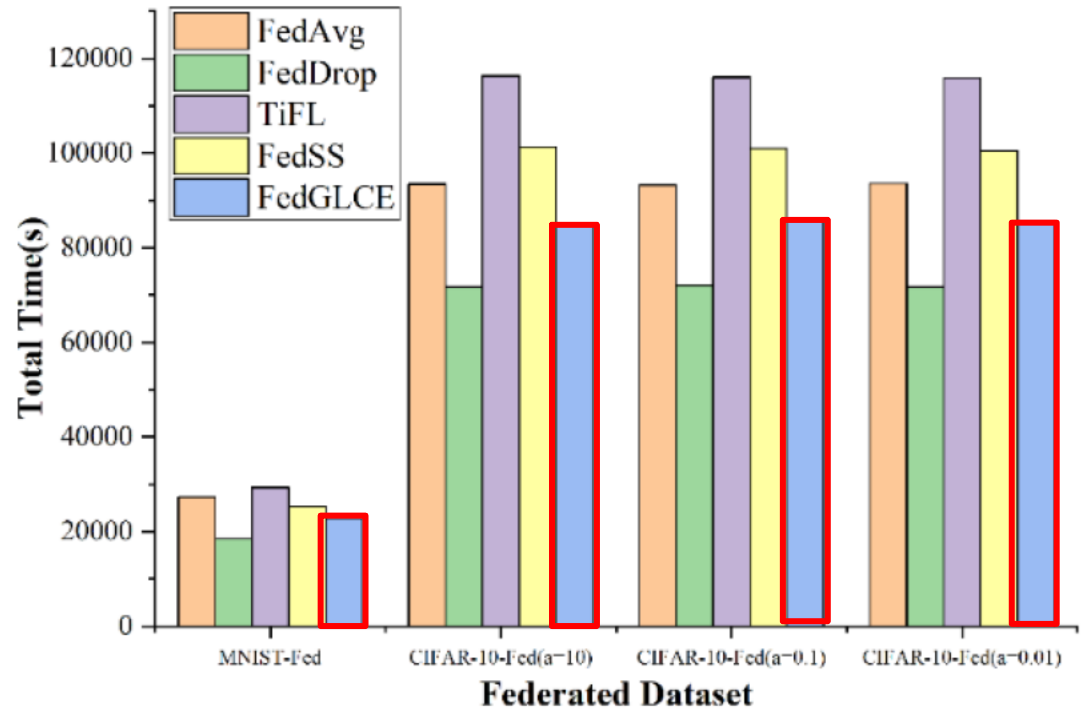
Experiment

Analysis of Local Test Accuracy

TABLE VIII
EXPERIMENTAL RESULTS OF LOCAL TEST ACCURACY DIFFERENCES FOR CLIENT GROUPING APPROACHES

Comparison approaches	Evaluation metrics	Dataset: MNIST-Fed	CIFAR-10-Fed *Dir(10)*	CIFAR-10-Fed *Dir(0.1)*	CIFAR-10-Fed *Dir(0.01)*
FedAvg	$Var(Acc)$	24.02	**34.29**	116.28	406.91
FedDrop		23.33	37.74	97.80	549.22
TiFL		**15.77**	38.79	132.16	408.10
FedSS		20.40	37.56	95.10	520.88
FedGLCE		19.71	35.29	**94.08**	**373.48**

Total Hours of Federated Training



Experimental results of total federated learning training hours for different client grouping approaches

Conclusion

- FedGLCE can **balance the participation of clients** in the local federated training through the dynamic client grouping approach based on local computing efficiency under the premise of **ensuring the local test accuracy**.

- **Performance biases** of the global model on clients are improved.

- FedGLCE reduces the **total training time** of clients in federated learning.

At Last

Thanks for your attention!