SHARE: Shaping Data Distribution at Edge for Communication-Efficient Hierarchical Federated Learning

谢礼安 12.29.2021

Introduction

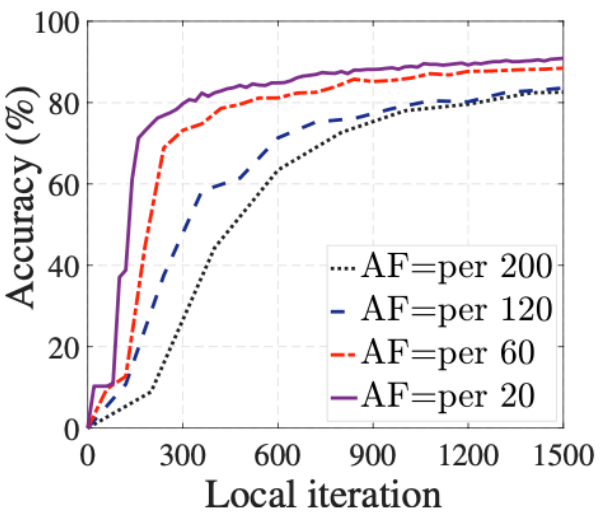
- 联邦学习(FL)中，为了保证模型的精度，需要终端与云端频繁通信，造成很大的通信开销
- 采用分层联邦学习(HFL)，引入边端节点，可以有效地降低终端与云端的直接通信的开销
- 但不同边端的数据分布不一致，需要更多轮的边-云聚合，这又会导致更多的开销

Introduction

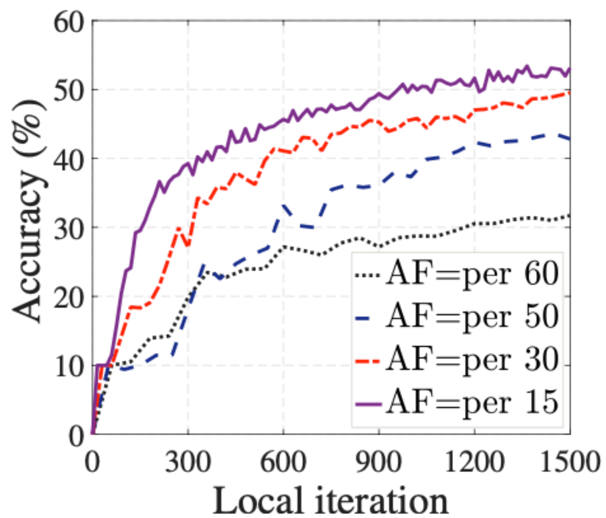
- 作者认为在HFL中要同时考虑最小通信开销和边端数据分布的平均
- 提出了CCM问题，并用GoA和LoS算法进行求解

Motivation

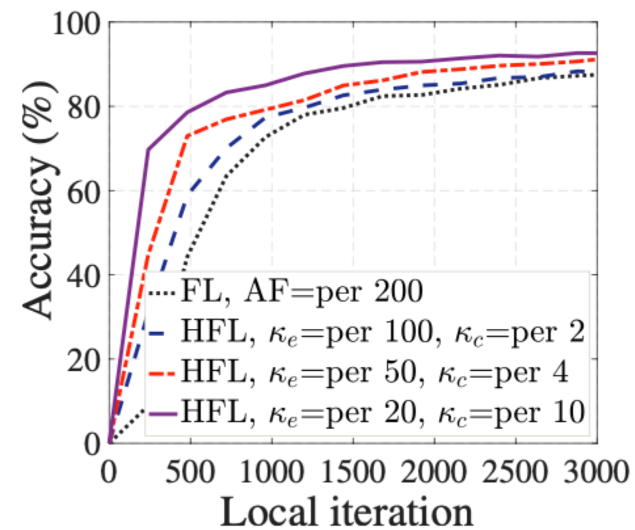
Model Aggregation Frequency Affects



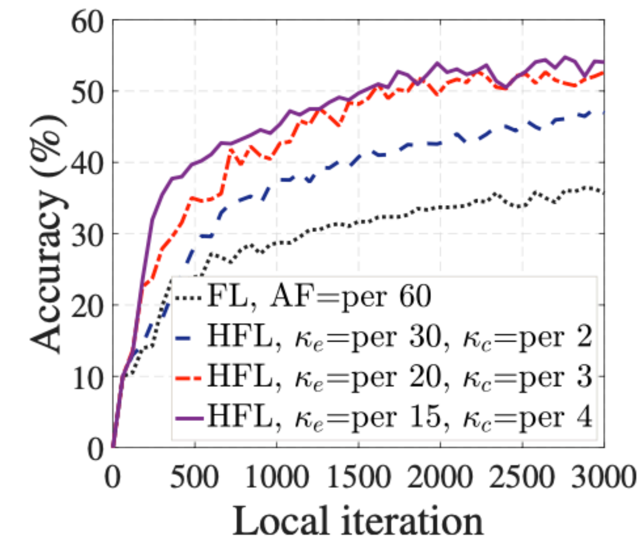
(a) MNIST



(b) CIFAR-10



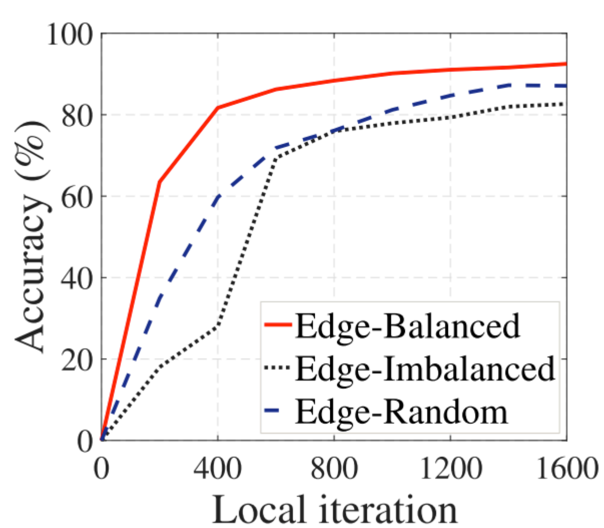
(a) MNIST



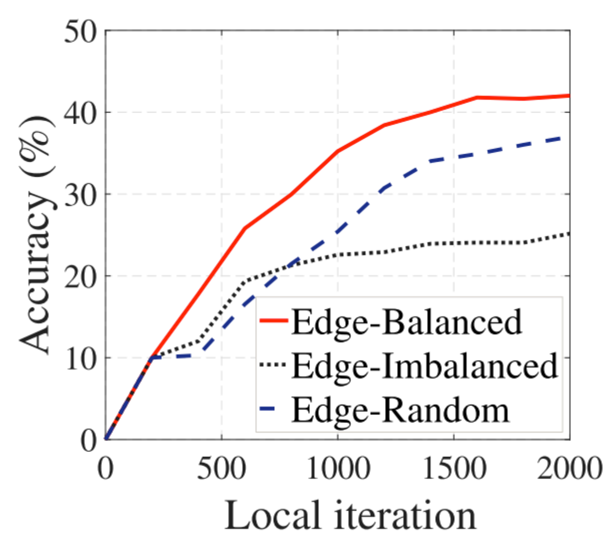
(b) CIFAR-10

κ_c表示边-云汇聚频率

κ_e表示边-终端汇聚频率

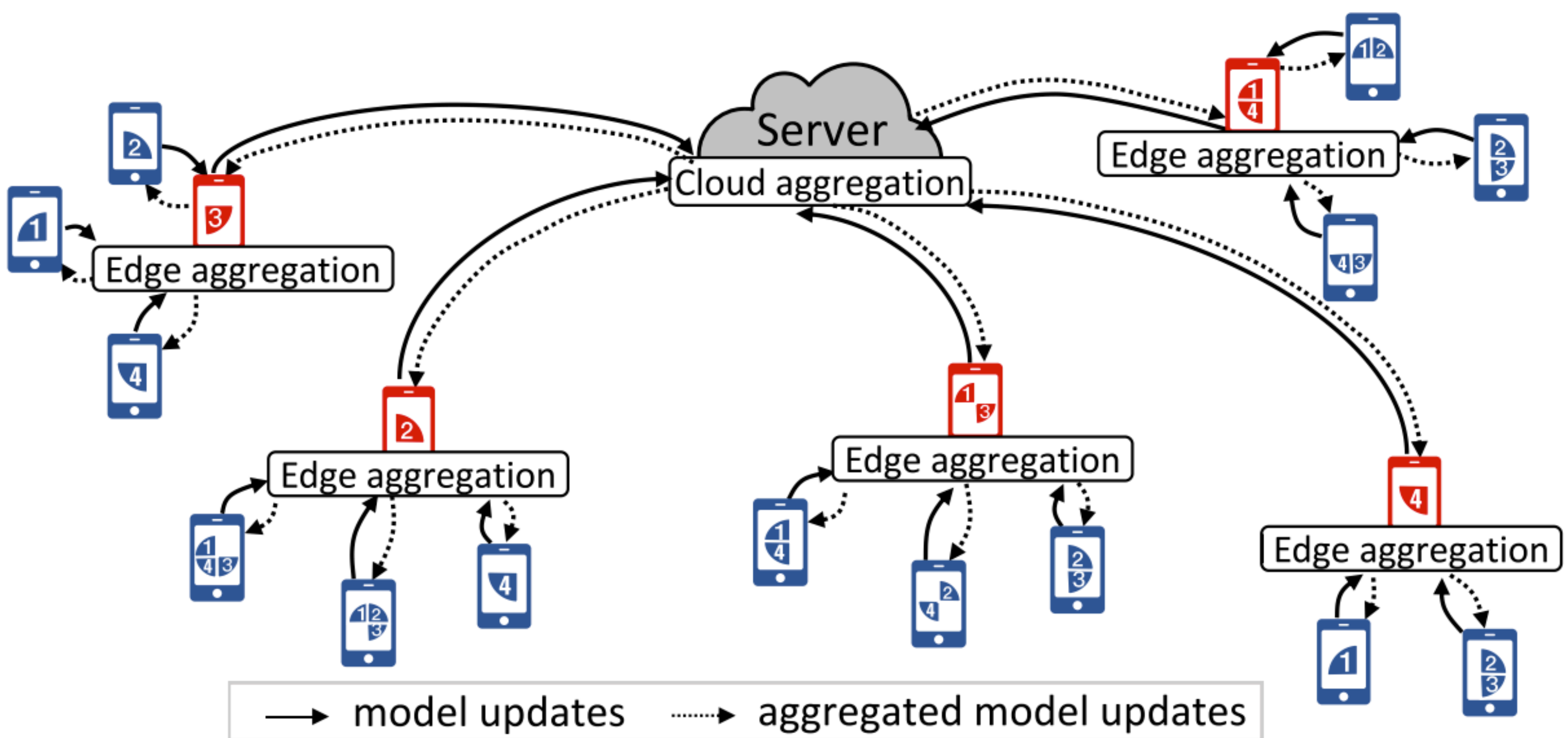


(a) Edge aggregation: MNIST



(b) Edge aggregation: CIFAR-10

Hierarchical Federated Learning System



Motivation

Problem Define

- 问题提出：给定终端节点集$\mathcal{N}$、边端候选集$\mathcal{N}_e$，如何确定边端聚合器和终端节点的关联子集，使得终端-边端和边端-云端之间的通信开销总和最小?

$$C_{ne}(Y,\kappa) = \kappa\kappa_c \sum_{n\in\mathcal{N}} \sum_{e\in\mathcal{N}_e} y_{ne}c_{ne}$$

- $x_e \in \{0,1\}$ 表示边端聚合器e是否被选中
- $y_{ne} \in \{0,1\}$ 表示终端节点n和边端聚合器e是否关联
- c_{ne}表示节点n上传到边e的通信开销
- κ表示云端汇聚的次数
- $Y = \{y_{ne}\}$表示与终端进行数据交互的边端聚合器

Motivation

Problem Define

- 同样可以定义边-云的通信开销：

$$C_{ne}(X,\kappa) = \kappa \sum_{e \in \mathcal{N}_e} x_e c_{ec}$$

- $x_e \in \{0,1\}$ 表示边端聚合器e是否被选中
- c_{ec}表示节点n上传到边e的通信开销
- κ表示云端汇聚的次数

Motivation

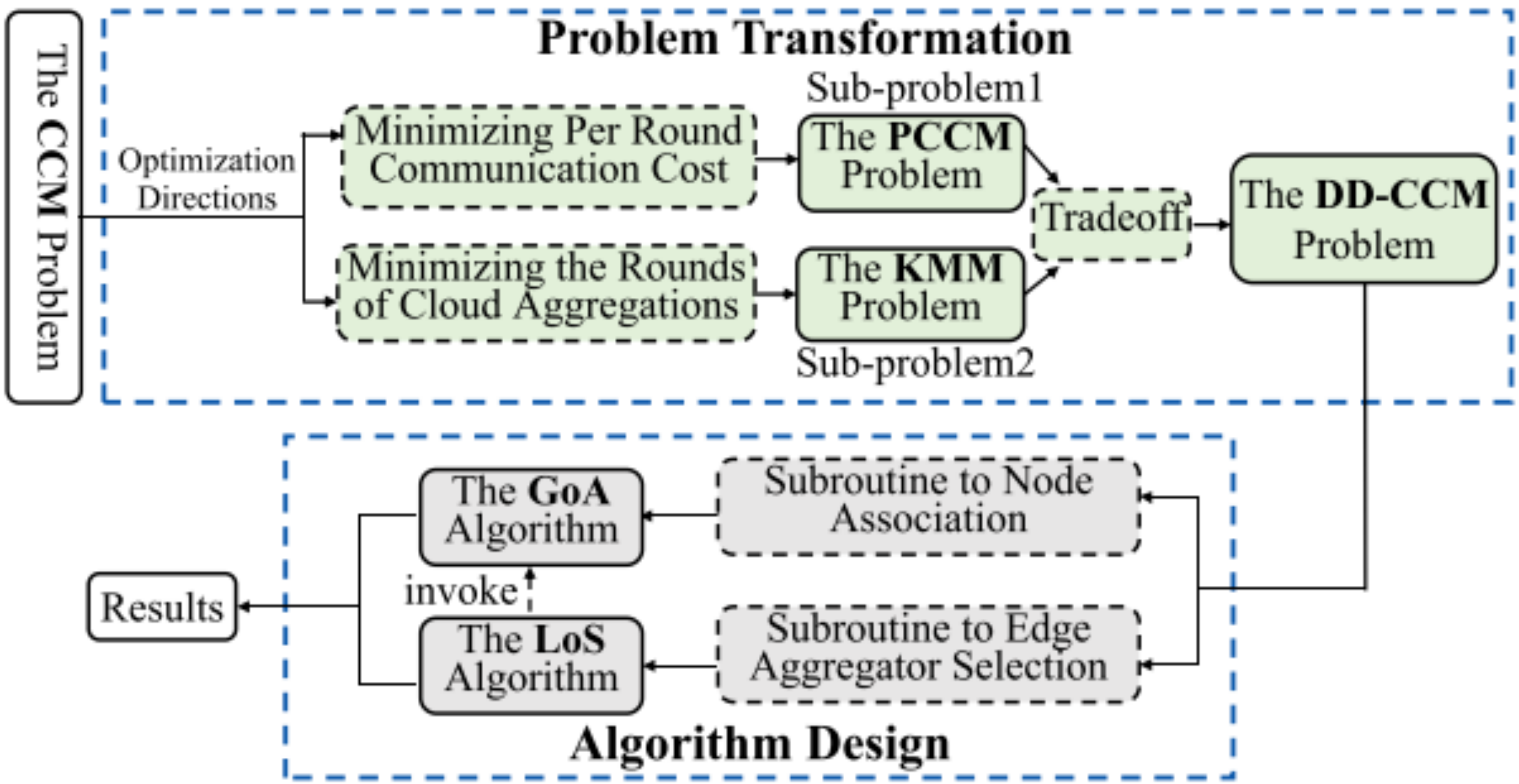
Problem Define

- CCM问题：

$$
\begin{aligned}
\min_{\boldsymbol{X},\boldsymbol{Y},\kappa} \quad & C_{ne}(\boldsymbol{Y},\kappa) + C_{ec}(\boldsymbol{X},\kappa), && && (3) \\
s.t. \quad & x_e = 0, && \forall e \notin \mathcal{N}_e, && (4) \\
& \sum_{e \in \mathcal{N}_e} y_{ne} = 1, && \forall n \in \mathcal{N}, && (5) \\
& y_{ne} \le x_e, && \forall n \in \mathcal{N}, \forall e \in \mathcal{N}_e, && (6) \\
& \sum_{n \in \mathcal{N}} y_{ne} \le B_e, && \forall e \in \mathcal{N}_e, && (7) \\
& x_e \in \{0,1\}, && \forall e \in \mathcal{N}, && (8) \\
& y_{ne} \in \{0,1\}, && \forall n \in \mathcal{N}, \forall e \in \mathcal{N}_e. && (9)
\end{aligned}
$$

Motivation

DESIGN OF SHARE



Motivation

Problem Transformation

- PCCM问题：找到通信开销最小的终端-边端集合X、边端-云端集合Y

$$\min_{\boldsymbol{X},\boldsymbol{Y}} \quad J_c(\boldsymbol{X},\boldsymbol{Y}) = \kappa_c \sum_{n\in\mathcal{N}} \sum_{e\in\mathcal{N}_e} y_{ne}c_{ne} + \sum_{e\in\mathcal{N}_e} x_e c_{ec}, \tag{10}$$

$$\begin{aligned}
s.t. \quad & x_e = 0, && \forall e \notin \mathcal{N}_e, && (4)\\
& \sum_{e\in\mathcal{N}_e} y_{ne} = 1, && \forall n \in \mathcal{N}, && (5)\\
& y_{ne} \le x_e, && \forall n \in \mathcal{N}, \forall e \in \mathcal{N}_e, && (6)\\
& \sum_{n\in\mathcal{N}} y_{ne} \le B_e, && \forall e \in \mathcal{N}_e, && (7)\\
& x_e \in \{0,1\}, && \forall e \in \mathcal{N}, && (8)\\
& y_{ne} \in \{0,1\}, && \forall n \in \mathcal{N}, \forall e \in \mathcal{N}_e. && (9)
\end{aligned}$$

Motivation

Problem Transformation

- KMM问题：选取使得边端的数据分布的KL散度最小的X、Y

$$\min_{\boldsymbol{X},\boldsymbol{Y}} \quad J_d(\boldsymbol{X},\boldsymbol{Y}) = \frac{1}{|\mathcal{E}|}\sum_{e\in\mathcal{E}} D_{KL}(P_e||P_u), \tag{11}$$

$$\begin{aligned}
s.t. \quad & x_e = 0, && \forall e \notin \mathcal{N}_e, && (4)\\
& \sum_{e\in\mathcal{N}_e} y_{ne} = 1, && \forall n \in \mathcal{N}, && (5)\\
& y_{ne} \le x_e, && \forall n \in \mathcal{N}, \forall e \in \mathcal{N}_e, && (6)\\
& \sum_{n\in\mathcal{N}} y_{ne} \le B_e, && \forall e \in \mathcal{N}_e, && (7)\\
& x_e \in \{0,1\}, && \forall e \in \mathcal{N}, && (8)\\
& y_{ne} \in \{0,1\}, && \forall n \in \mathcal{N}, \forall e \in \mathcal{N}_e. && (9)
\end{aligned}$$

Motivation

Problem Transformation

- DD-CCM问题

$$
\begin{aligned}
\min_{\boldsymbol{y}} \quad & \kappa_c \sum_{n\in\mathcal{N}} \sum_{e\in\mathcal{E}} y_{ne} c_{ne} + \gamma \frac{1}{|\mathcal{E}|} \sum_{e\in\mathcal{E}} D_{KL}(P_e || P_u), && \text{(13)} \\
s.t. \quad & y_{ne} = 0, \qquad \forall n \in \mathcal{N}, \forall e \notin \mathcal{E}, && \text{(14)} \\
& \sum_{e\in\mathcal{E}} y_{ne} = 1, \qquad \forall n \in \mathcal{N}, && \text{(15)} \\
& \sum_{n\in\mathcal{N}} y_{ne} \le B_e, \qquad \forall e \in \mathcal{E}, && \text{(16)} \\
& y_{ne} \in \{0, 1\}, \qquad \forall n \in \mathcal{N}, \forall e \in \mathcal{E}. && \text{(17)}
\end{aligned}
$$

Motivation

Algorithm Design

- GoA algorithm：解决PCCM问题

- 通过遍历终端节点和边端节点，计算KL散度的变化

- 更新ΔJ_{ne}

- 根据最小ΔJ_{ne}选取n和e

Algorithm 1: The workflow of *GoA* algorithm.

Input : (1) edge aggregators set $\mathcal{E}$; (2) cost c_{ne}; (3) node data distribution P_n.

Output : the node association results $\boldsymbol{Y} = \{y_{ne}\}$.

Initialize $\mathcal{M}_e \leftarrow \emptyset$, $P_e \leftarrow 0$ **for each** $e \in \mathcal{E}$;

Initialize $\mathcal{S}_n \leftarrow \mathcal{N}$;

repeat

 foreach $n \in \mathcal{S}_n$ **do**

  foreach $e \in \mathcal{E}$ **do**

   if $|\mathcal{M}_e| < B_e$ **then**

    $\Delta d \leftarrow D_{KL}(P_e + P_n || P_u) - D_{KL}(P_e || P_u)$;

    Compute $\Delta J_{ne} \leftarrow \kappa_c c_{ne} + \gamma \frac{1}{|\mathcal{E}|} \Delta d$;

   end

  end

 end

 Find the node n and the edge aggregator e with minimum ΔJ_{ne};

 $\mathcal{M}_e \leftarrow \mathcal{M}_e + \{n\}$;

 $\mathcal{S}_n \leftarrow \mathcal{S}_n - \{n\}$;

 $P_e \leftarrow P_e + P_n$;

 $y_{ne} \leftarrow 1$;

until $\mathcal{S}_n = \emptyset$

return $\{y_{ne}\}$;

Motivation

Algorithm Design

- LoS algorithm：解决KMM问题

- open操作

- close操作

- swap操作

Algorithm 2: The workflow of *LoS* algorithm.

```
Initialization: Feasible solution 𝓔_s;
repeat
    'open' operation
    foreach e ∈ 𝒩_e − 𝓔_s do
        Compute J(𝓔_s + {e});
        if J(𝓔_s + {e}) < J(𝓔_s) then
            𝓔_s ← 𝓔_s + {e};
            break
        end
    end
    'close' operation
    foreach e ∈ 𝓔_s do
        Compute J(𝓔_s − {e});
        if J(𝓔_s − {e}) < J(𝓔_s) then
            𝓔_s ← 𝓔_s − {e};
            break
        end
    end
    'swap' operation
    foreach e ∈ 𝒩_e − 𝓔_s do
        foreach e' ∈ 𝓔_s do
            Compute J(𝓔_s + {e} − {e'});
            if J(𝓔_s + {e} − {e'}) < J(𝓔_s) then
                𝓔_s ← 𝓔_s + {e} − {e'};
                break
            end
        end
    end
until No operation can reduce the total communication cost
return 𝓔_s;
```

EVALUATION

Setup

- 考虑两种拓扑：UUNET、TiNet
- 通信开销：物理距离*系数*模型参数大小

$$c_{ne} = 0.002 \cdot d_{ne} \cdot S_m \quad c_{ec} = 0.02 \cdot d_{ec} \cdot S_m$$

- Benchmarks

Cloud-based FL

Cost only CPLEX：不考虑数据分布的HFL

Data only greedy：只考虑数据分布的HFL

EVALUATION

SHARE vs. Cloud-based FL

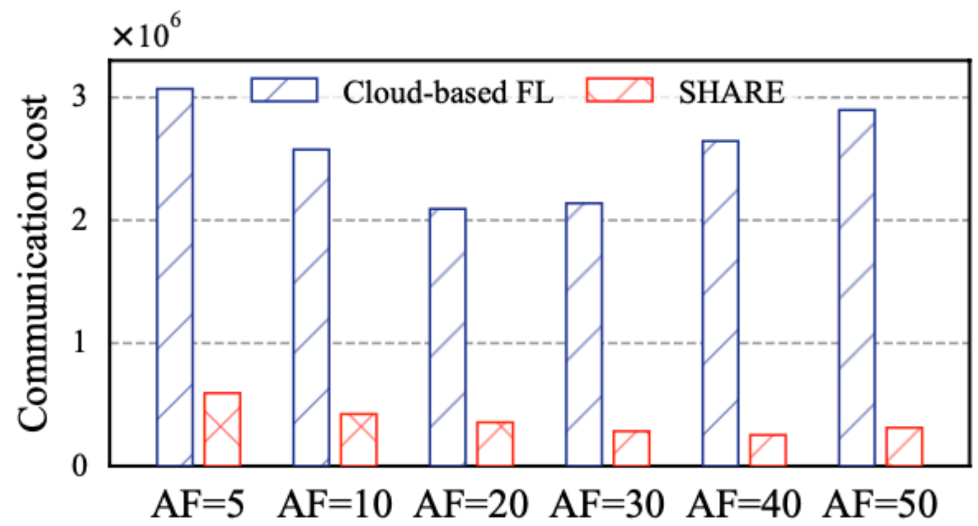


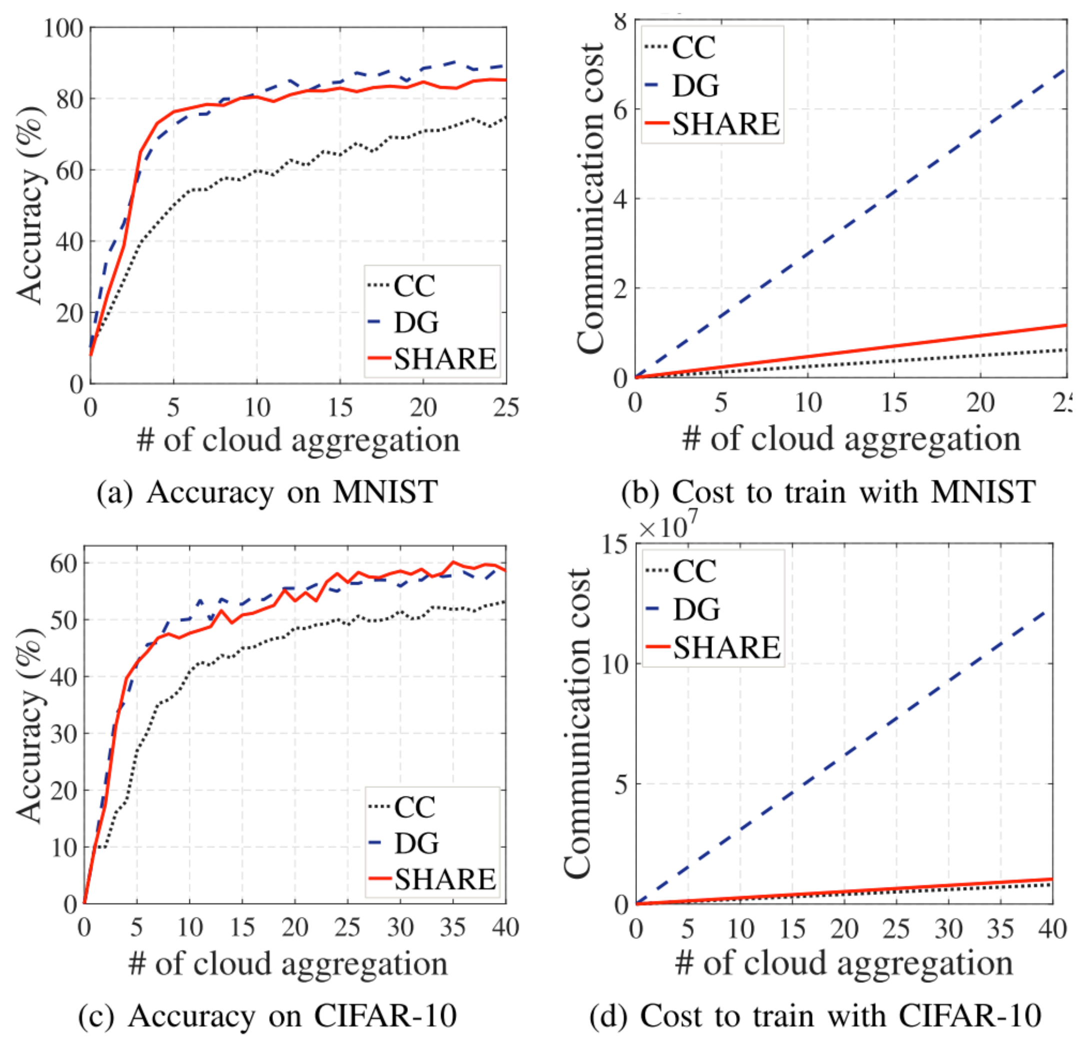
Fig. 6: *SHARE* vs. Cloud-based FL with AF $= \kappa_e \cdot \kappa_c$.

Table I: *SHARE* vs. Cloud-based FL with AF $= \kappa_e$.

$\times 10^5$	C-FL	κ_c=1	κ_c=2	κ_c=3	κ_c=4	κ_c=5	κ_c=6
AF=κ_e=5	30.70	5.93	4.23	3.74	3.55	3.32	2.82
AF=κ_e=10	25.75	5.07	3.85	2.91	3.29	2.91	2.86
AF=κ_e=20	20.92	4.04	3.36	3.30	2.65	2.21	2.84
AF=κ_e=30	21.38	4.58	3.67	3.30	3.19	2.60	2.22
AF=κ_e=40	26.44	4.56	3.30	3.11	3.08	2.50	2.37
AF=κ_e=50	28.97	4.82	4.06	3.08	3.23	2.84	2.07

EVALUATION

SHARE vs. HFL



(a) Accuracy on MNIST (b) Cost to train with MNIST

(c) Accuracy on CIFAR-10 (d) Cost to train with CIFAR-10

Fig. 7: *SHARE* vs. HFL benchmarks.

EVALUATION

SHARE vs. HFL

- 超参数的影响：

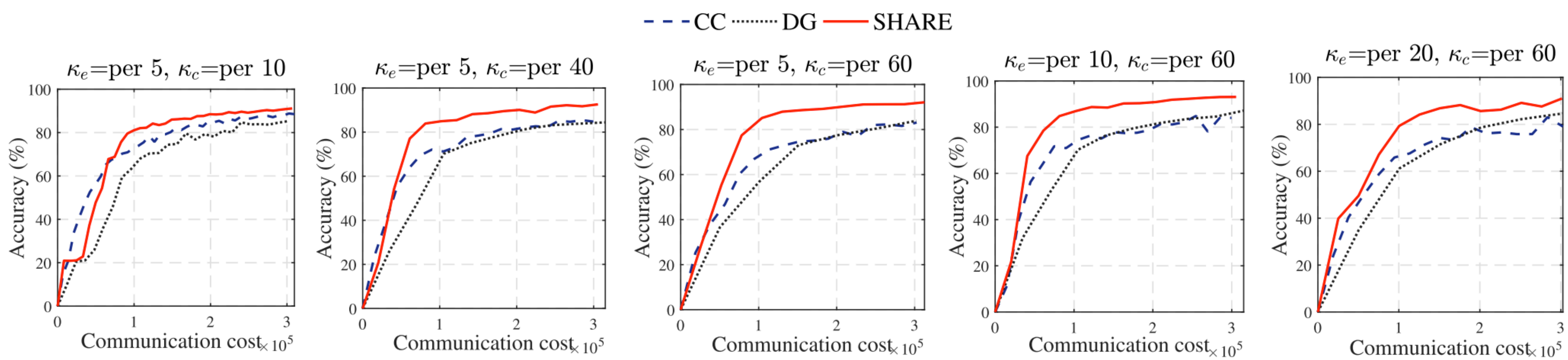
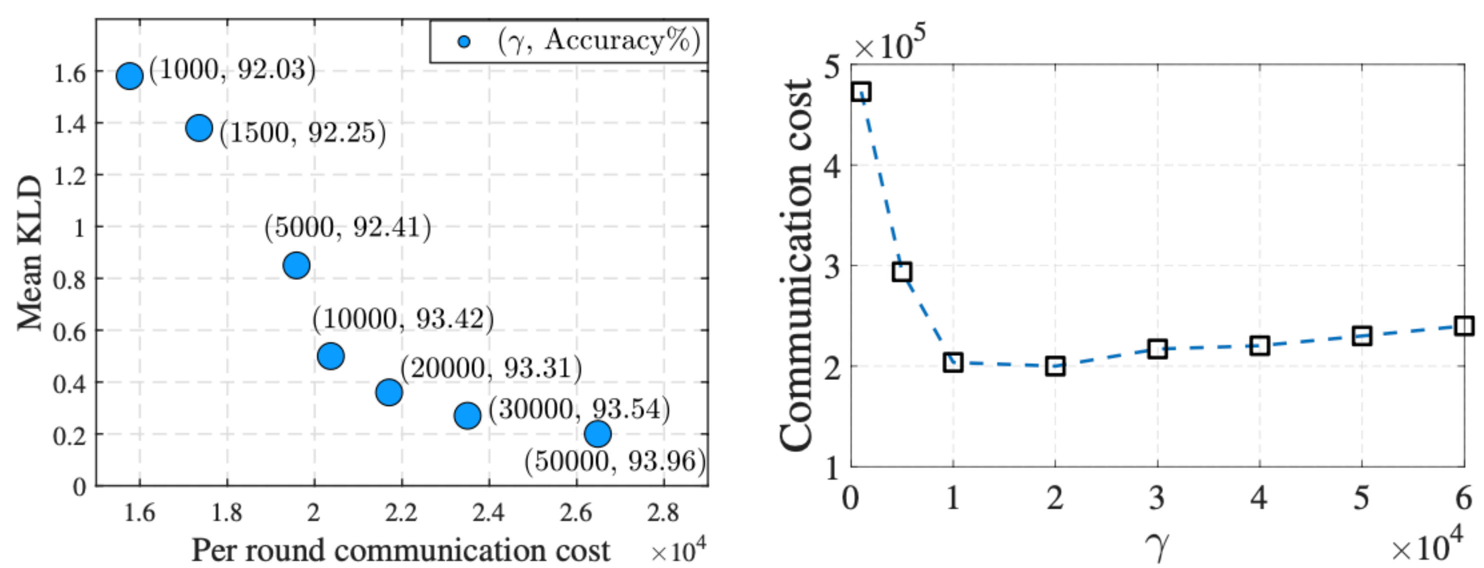


Fig. 8: The accuracy vs. communication cost under different settings of κ_e and κ_c.



(a) Tuning tradeoff (b) Required communication cost vs. γ

Fig. 9: Impact of parameter γ.

EVALUATION

SHARE vs. HFL

- 网络拓扑的影响：

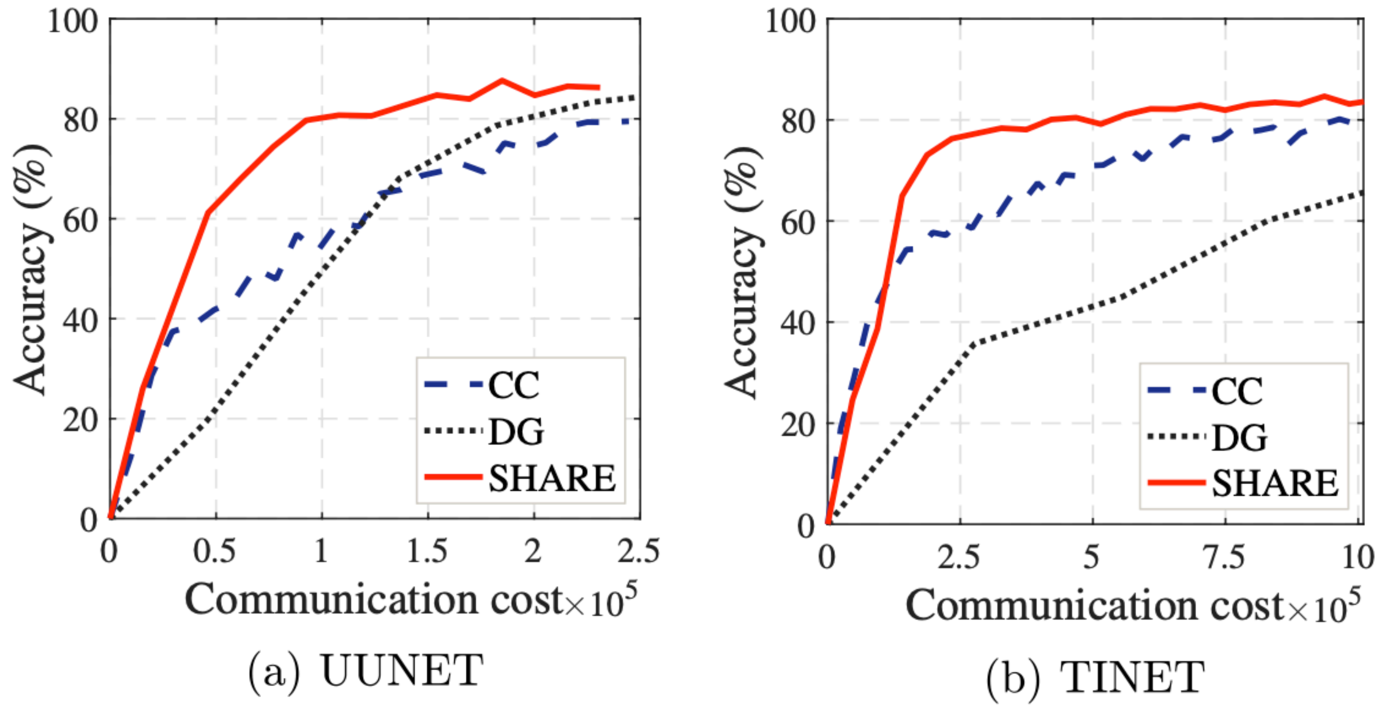


Fig. 10: The performance under different network topologies.

THINKING

- 优点：
  - 同时考虑了数据分布平均和最小通信开销
- 待改进：
  - 没有考虑终端设备的移动性，可能会打破数据分布
  - 边缘节点的可靠性