CASA

2024-08-25

KDD2024
BARCELONA, SPAIN

CASA: Clustered Federated Learning with Asynchronous Clients

Boyi Liu[1], Yiming Ma[1], Zimu Zhou[2],

Yexuan Shi[1], Shuyuan Li[1], Yongxin Tong[1]

[1]Beihang University

[2]City University of Hong Kong

Outline

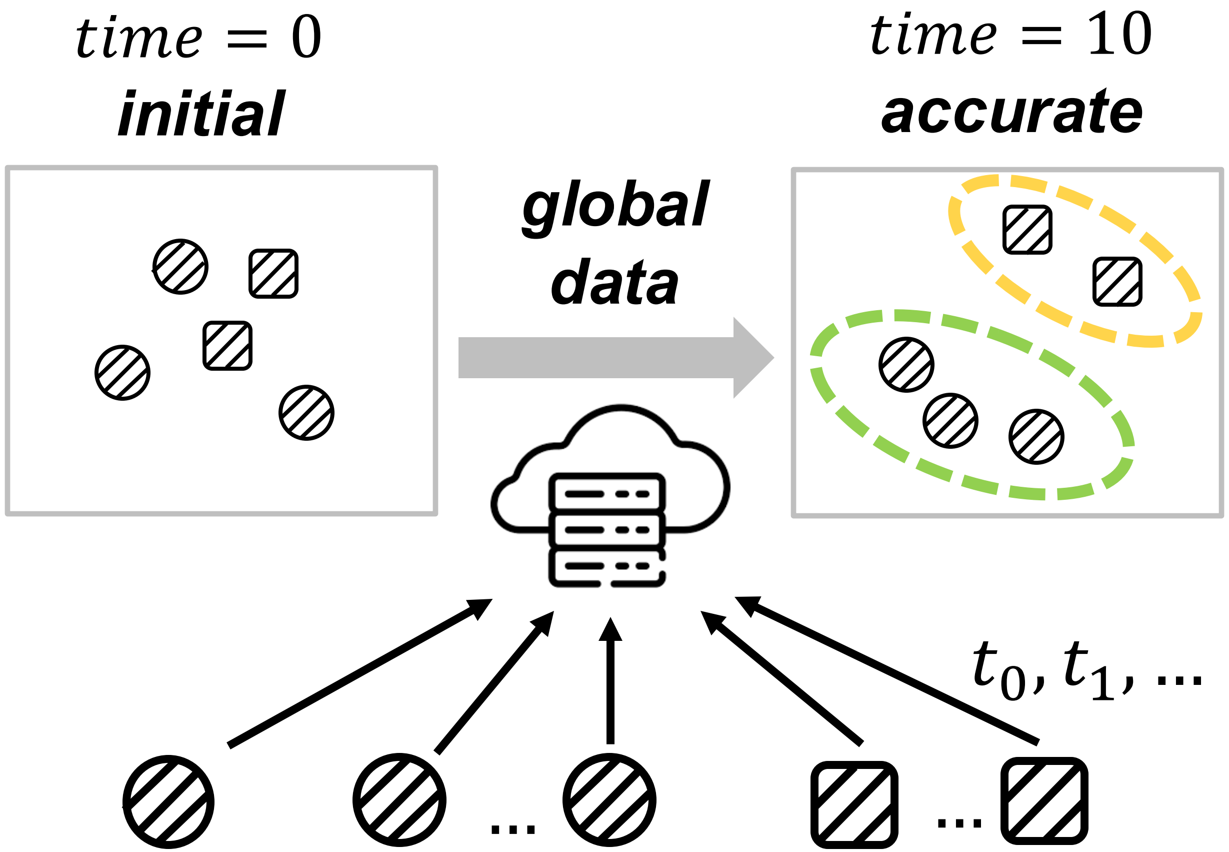
- Background & Motivation
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

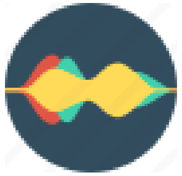
Outline

- Background & Motivation
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

Background & Motivation

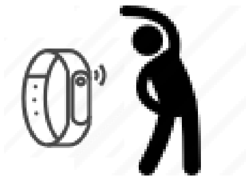
- **Clustered Federated Learning(CFL)**





Personal Voice Assistant

Smart Keyboards

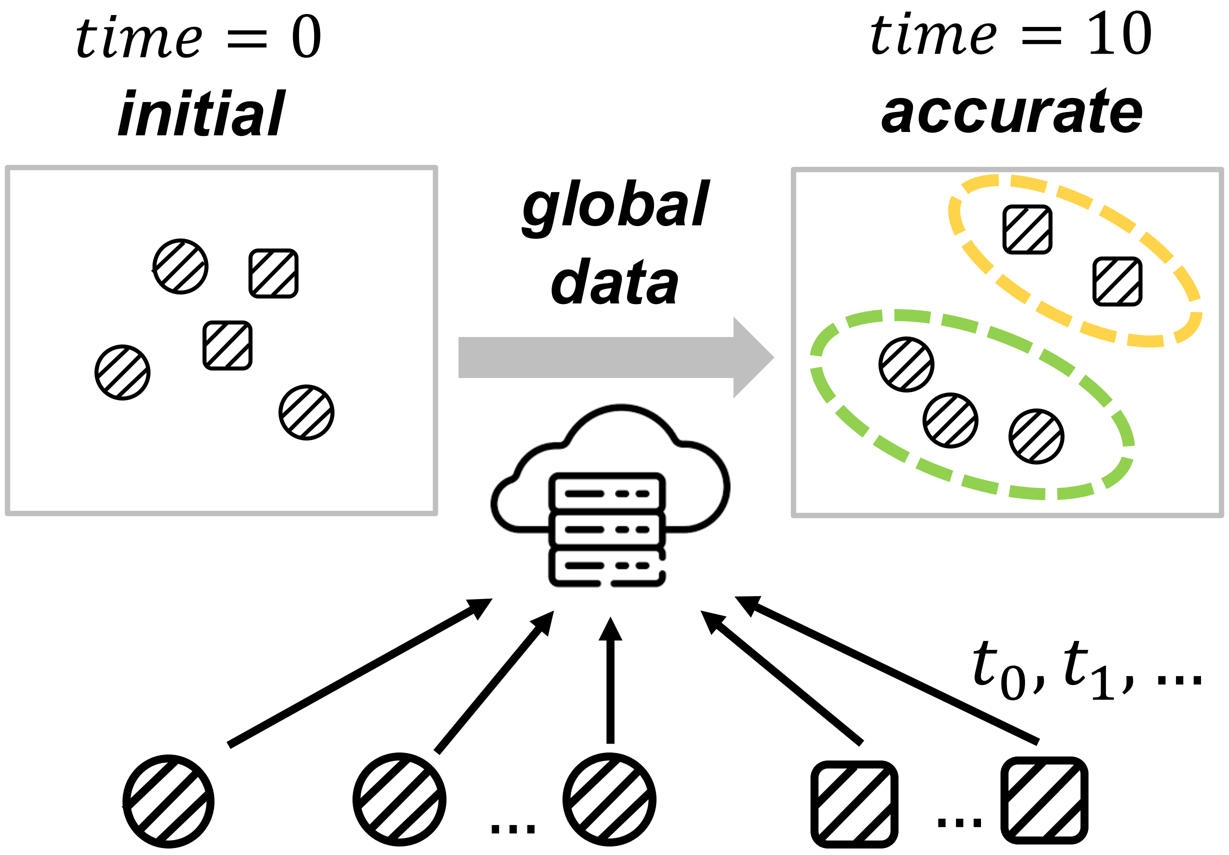


Human Activity Recognition

Data is often *heterogeneous* yet exhibits *natural clusterability*

Background & Motivation

- **Clustered Federated Learning(CFL)**



Core idea

1) Categorize clients into clusters,

2) Train cluster-wise global model,

3) Solve Non-iid problem

Training objective $\mathcal{P}$

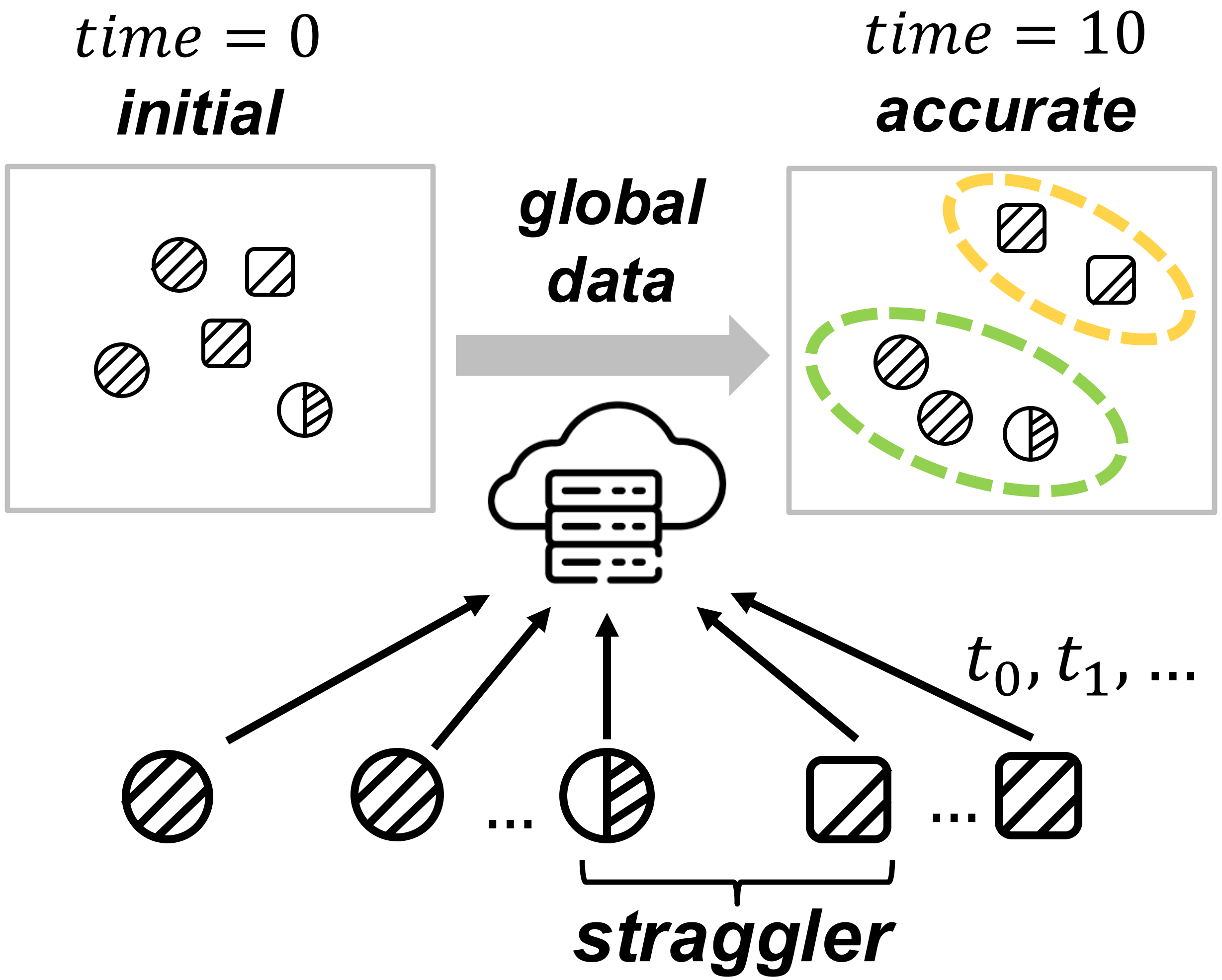
$$\mathcal{P} = \sum_{k=1}^{K} \sum_{c_i \in C_k} \frac{|D_i|}{|D|} \mathbb{E}[\mathcal{L}(w_{g,k}; D_c)]$$

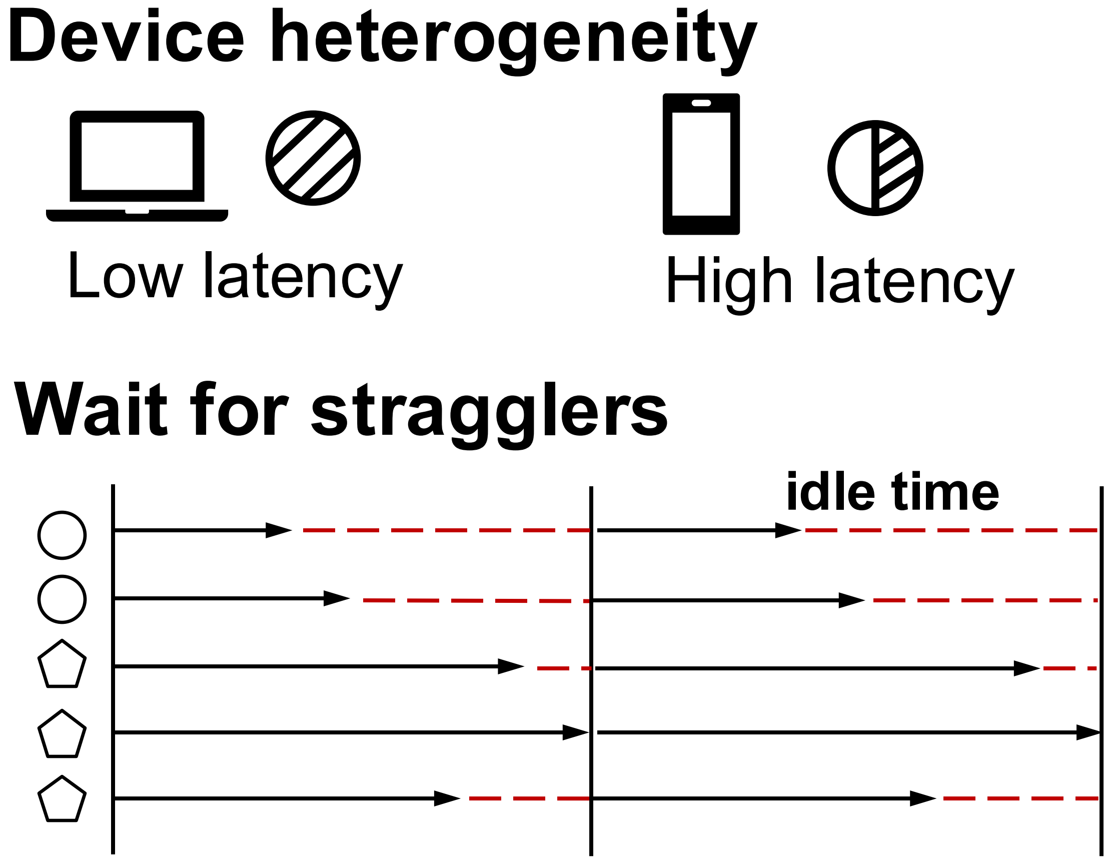
Clustering objective $\mathcal{H}$

$$\mathcal{H} = \sum_{k=1}^{K} \sum_{c_i \in C_k} \frac{|D_i|}{|D|} \|w_i - w_{g,k}\|_2^2$$

Background & Motivation

- **CFL Struggles with Stragglers**

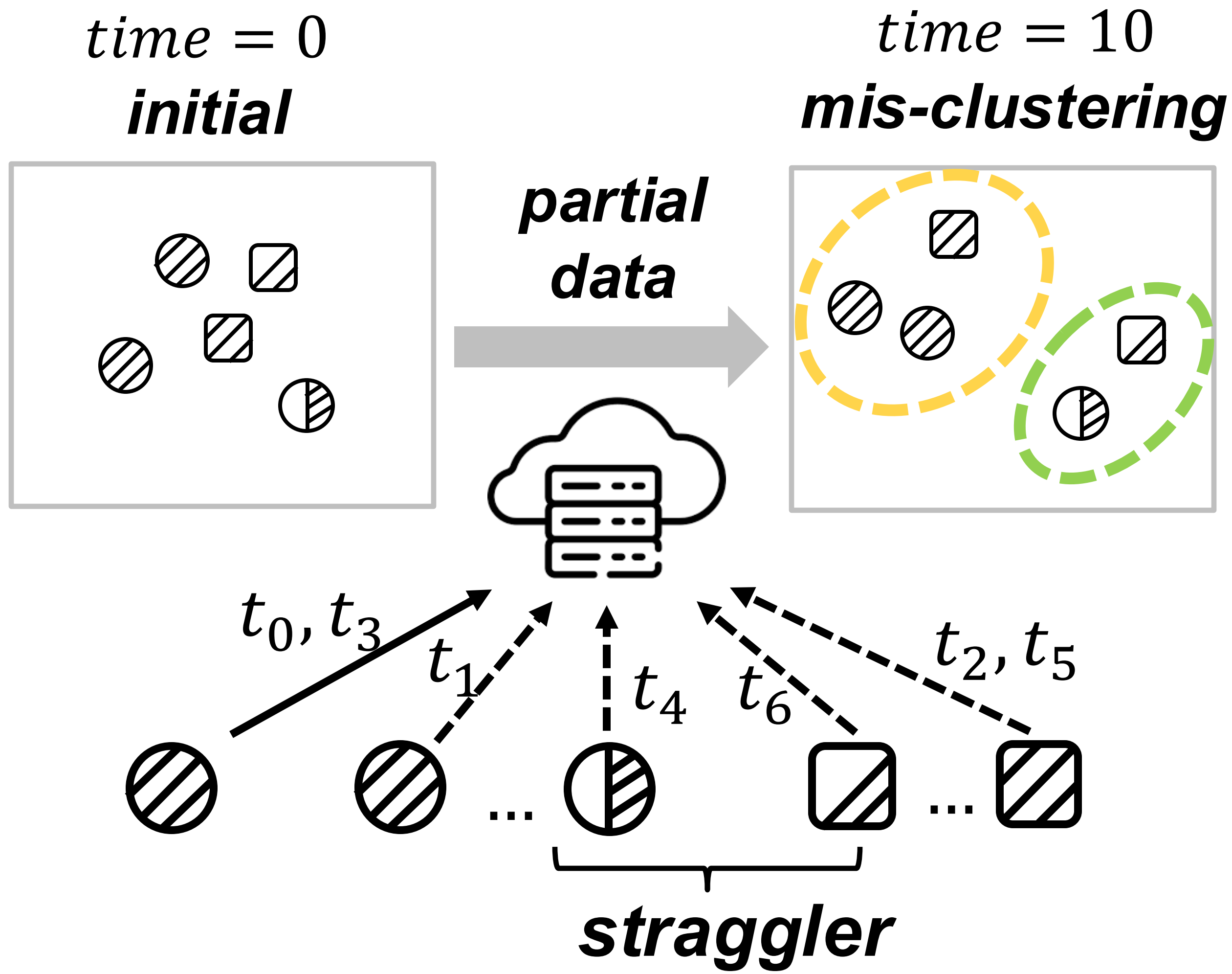




Challenge:
How to solve the inefficiency?

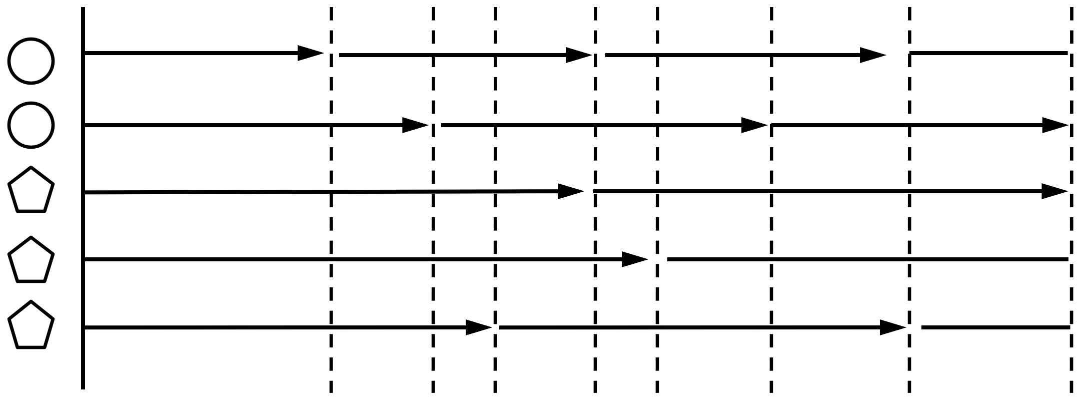
Background & Motivation

- **Integrate Asynchrony into CFL**



Device heterogeneity

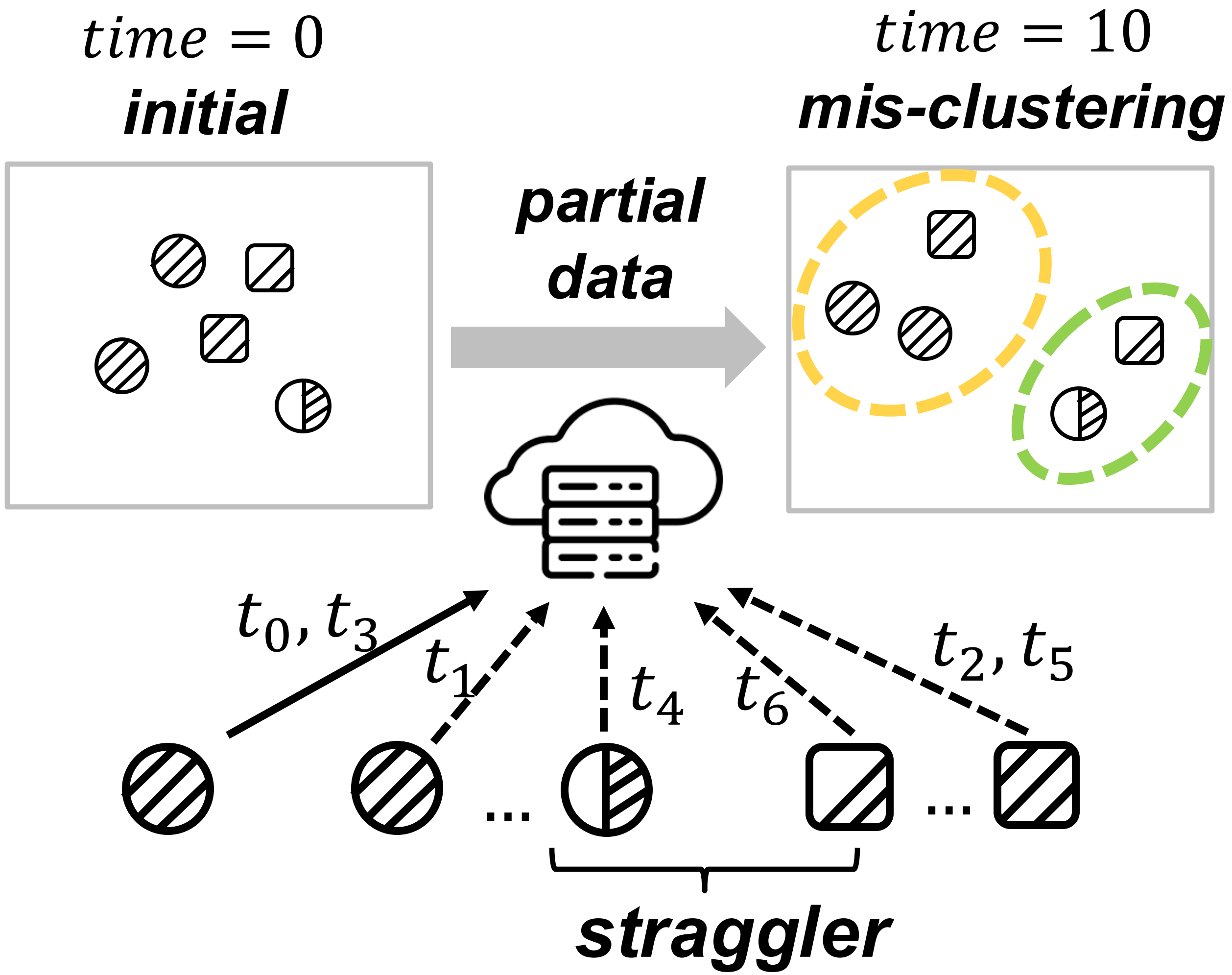
Asynchronous setup



We don't have to wait for stragglers under asynchrony!

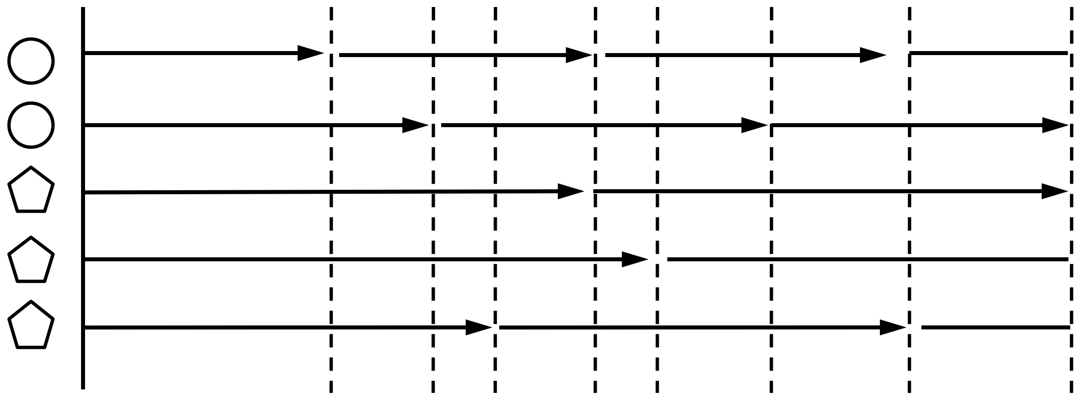
Background & Motivation

- **Integrate Asynchrony into CFL**



Device heterogeneity

Asynchronous setup



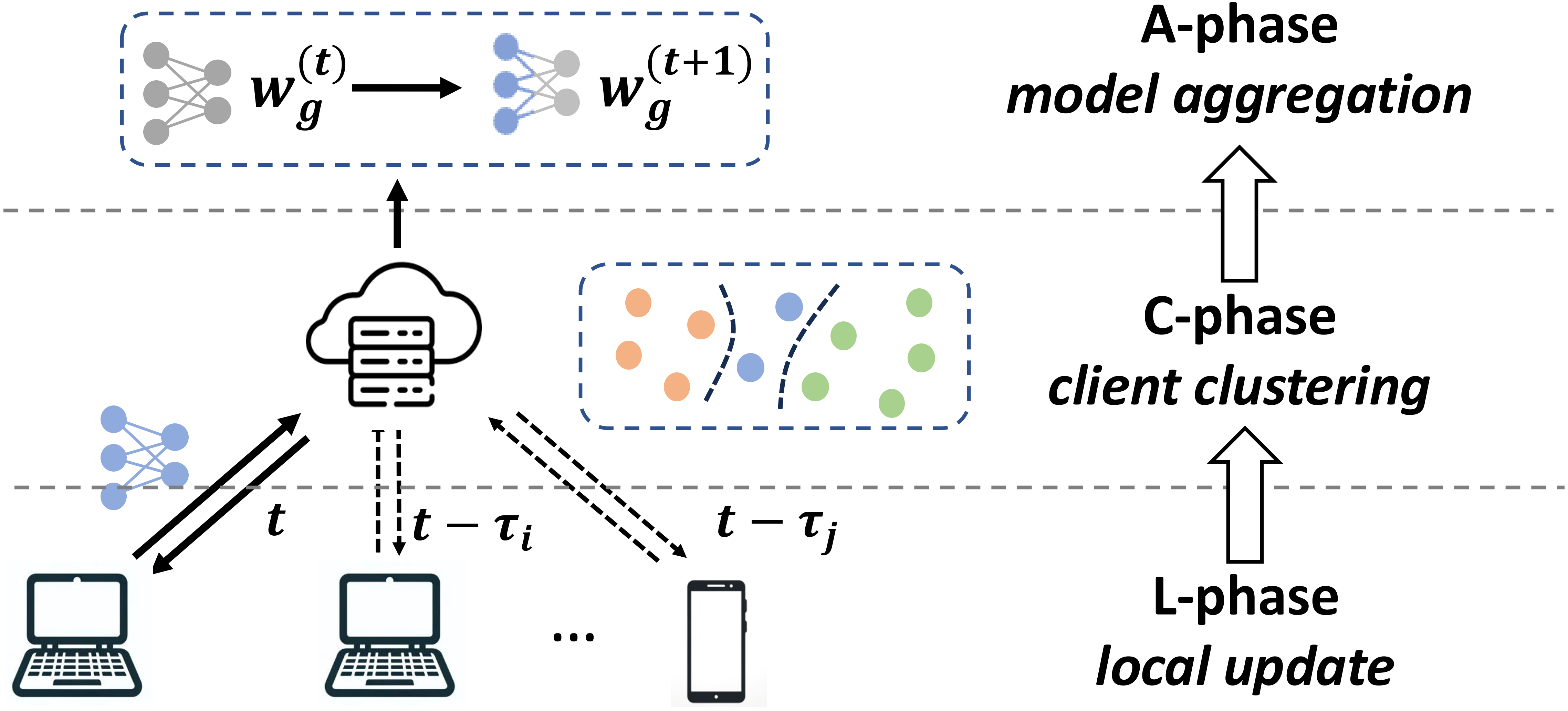
New Challenge:
Can CFL adapt to asynchrony?

Outline

- Background & Motivation
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

Problem Statement

- **CFL Workflow under Asynchrony**

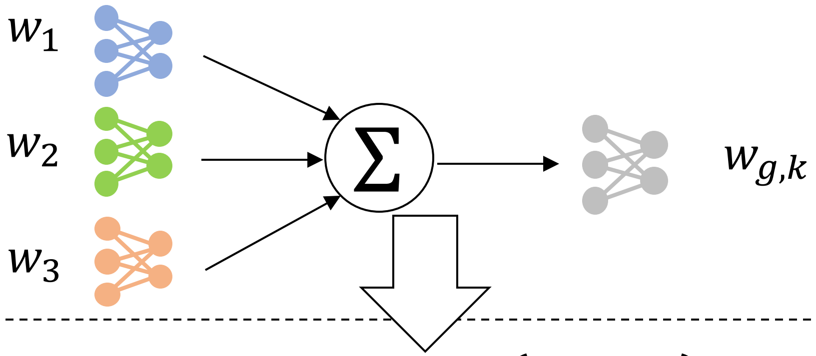


Problem Statement

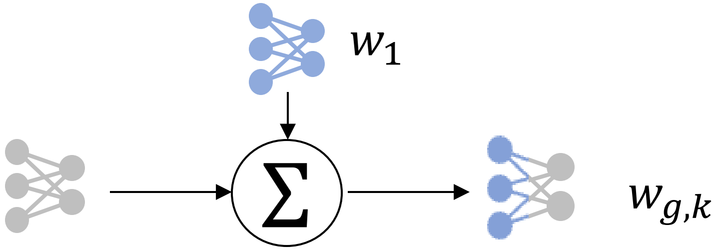
- **Direct Impact**

A-phase (aggregation)

Synchronous: $w_{g,k} \leftarrow \sum_{c_i \in \mathcal{C}_k} \frac{|D_i|}{|D|} w_i$



Asynchronous: $w_{g,k} \leftarrow (1-\alpha) w_{g,k} + \alpha w_i$



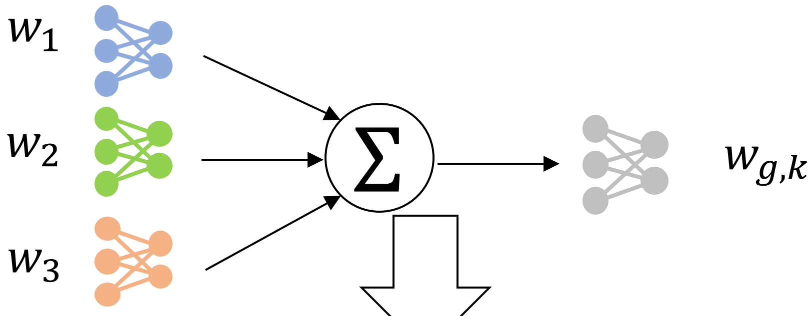
Aggregation strategy changes

Problem Statement

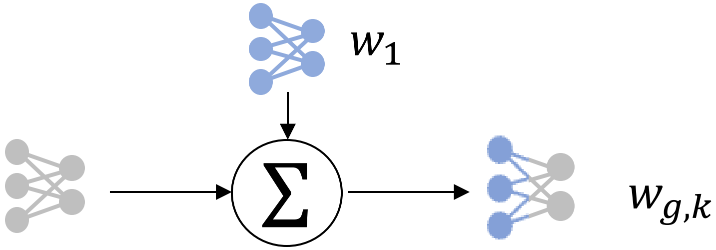
- **Direct Impact**

A-phase (aggregation)

Synchronous: $w_{g,k} \leftarrow \sum_{c_i \in \mathcal{C}_k} \frac{|D_i|}{|D|} w_i$



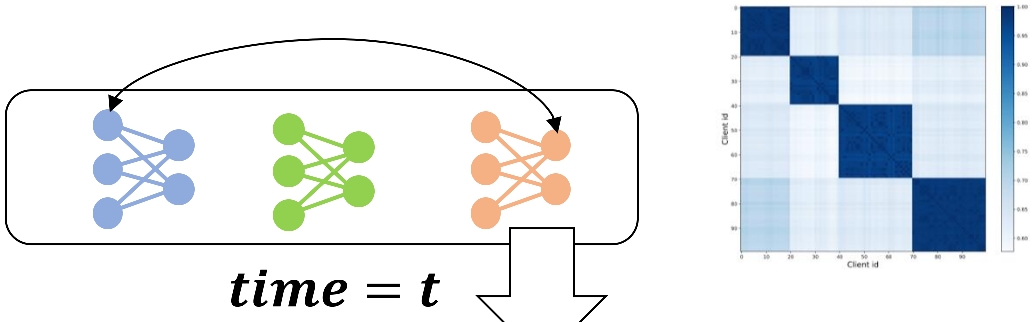
Asynchronous: $w_{g,k} \leftarrow (1-\alpha) w_{g,k} + \alpha w_i$



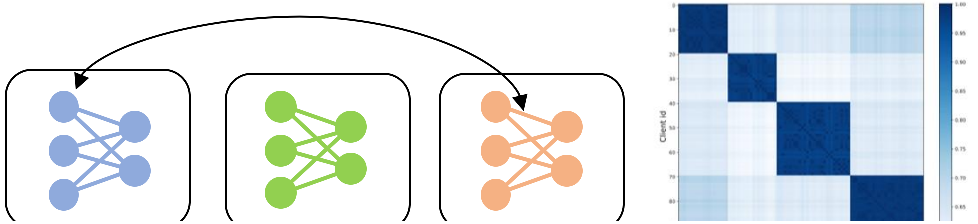
Aggregation strategy changes

C-phase (clustering)

Synchronous: $A_{ij} \leftarrow \cos(w_i^{(t)}, w_j^{(t)})$



Asynchronous: $A_{ij} \leftarrow \cos(w_i^{(t-\tau_i)}, w_i^{(t-\tau_j)})$



Large gap! not accurate!

cosine similarity ≠ data heterogeneity

Problem Statement

- **Compound Impact**

$$
\begin{tikzcd}
\begin{array}{c}\textbf{A-phase}\\ \textit{decay coefficient } \alpha\end{array} \arrow[r, bend left, "\textbf{Theorem 1}"] & \begin{array}{c}\textbf{C-phase}\\ \textit{dynamic cluster size } |\mathcal{C}_k|\end{array} \arrow[l, bend left, "\textbf{Theorem 2}"]
\end{tikzcd}
$$

Theorem 1. *(Clustering Error under Asynchrony). When clustering relies on a similarity matrix A' derived with asynchronous model parameters, the mis-clustering rate p is bounded by:*

$$p = O(\lambda\alpha\sqrt{\sum_{i=1}^{n}(\sum_{j=1}^{n}\|\tau_i - \tau_j\|^2)}) \tag{4}$$

where $\lambda = \eta Q\theta U$, and η is the learning rate, Q is the local training steps, U is the upper bound of gradient, θ is the upper bound of staleness (details in Appendix A.1.1).

Mis-clustering rate

Theorem 2. *(Convergence of Training Objective). The training objective $\mathcal{P}$ decreases monotonically, and thus the CFL framework converges under asynchrony, if the following condition is met:*

$$\alpha \leq \frac{\Omega(t)h_i}{|C_k|} \tag{5}$$

where $|C_k|$ is the size of cluster u_k, h_i is the computational capacity of c_i, and $\Omega(t)$ is a time-decreasing function (details in Appendix A.1.2).

Extra decay coefficient bound

Outline

- Background & Motivation
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

Our Solutions

- **Bi-level Asynchronous Aggregation**
  - **Rationale:** Meet **Theorem 2** to ensure convergence, let decay relevant to *time, computation and cluster scale*
  - **Cluster & Client-level Decay**
    - ***Cluster-level Decay + Personalized Information = Client-level Decay***

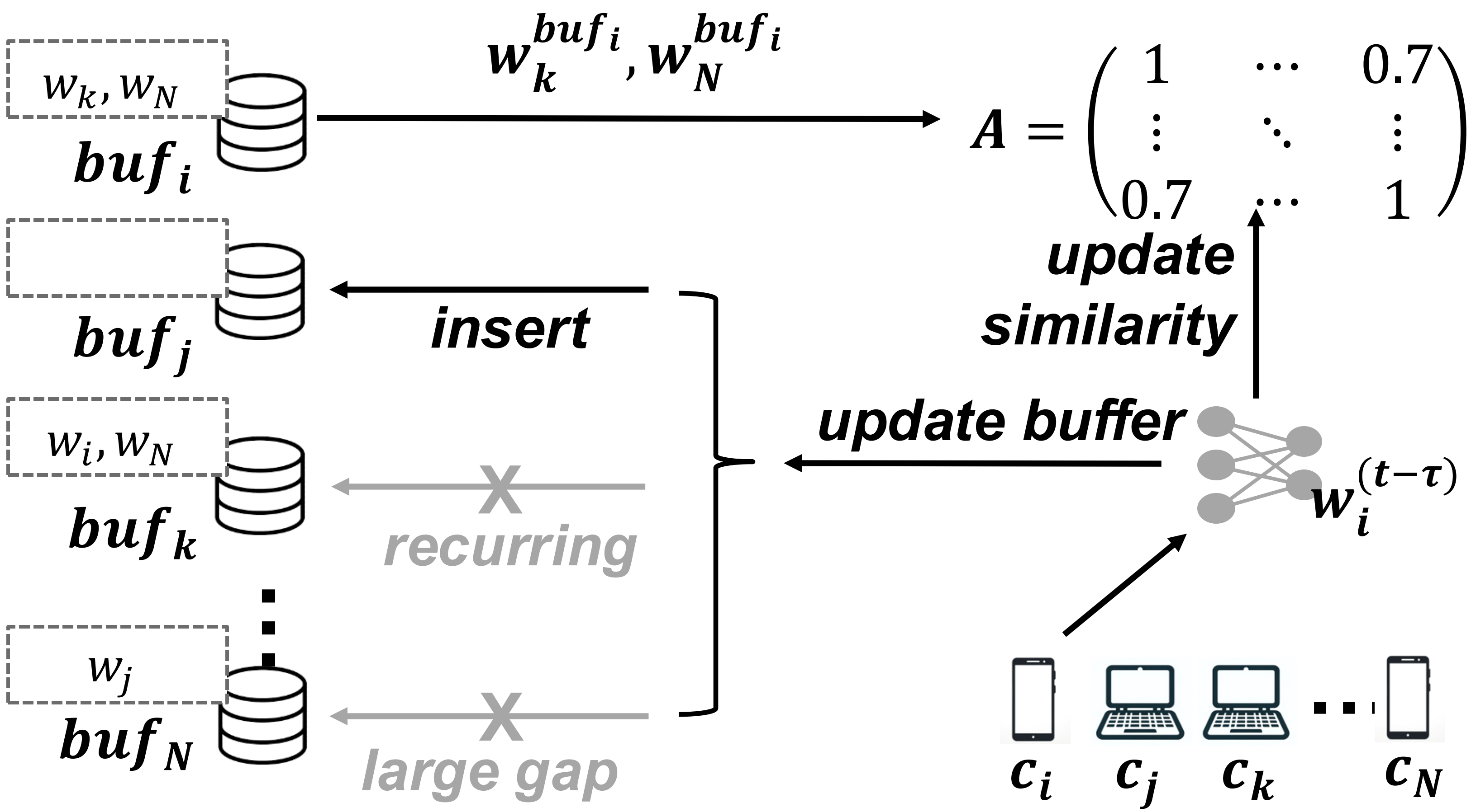
$$\alpha_{c,k}^{(t)} = \frac{\alpha_0 \Omega(t)}{\log(|C_k|)} \qquad \alpha_i^{(t)} = \begin{cases} \alpha_{c,k}^{(t)}, & \text{if } \tau_i \leq r_c^{(t)} \\ \alpha_{c,k}^{(t)} / \sqrt{\tau_i}, & \text{if } \tau_i > r_c^{(t)} \end{cases}$$

  - **Why we decouple?**
    - The cluster-level decay is not only a parameter, but **a representation of cluster information**, which we will discuss later

Problem unsolved: how to accurately cluster?

Our Solutions

- **Buffer-Aided Dynamic Clustering**
  - **Rationale**: Meet **Theorem 1** to limit $\tau_i - \tau_j$, clustering via *buffered* model parameters instead of *fresh* model parameters



Our Solutions

- **Buffer-Aided Dynamic Clustering**
  - **An interesting question: when to cluster?**
    - We compare the largest eigengap $\lambda_{k+1} - \lambda_k$ of similarity matrix and cluster-wise decay $\alpha_{c,k}^{(t)}$
    - We cluster only when $\alpha_{c,k}^{(t)} < (\lambda_{k+1} - \lambda_k)^\gamma$
  - Why cluster-wise decay?
    - Meet **Theorem 1** to limit α, clustering only when α is small is beneficial for accurate clustering
    - Once clustered, the $\alpha_{c,k}^{(t)}$ will be larger due to decreasing of $|\mathcal{C}_k|$, making it more difficult for clustering again

Our Solutions

- **CASA+: Mitigating Staleness with Sparse Training**
  - We apply a mask to sparse the local model
  - The sparse rate is relevant with divergence of decay $\alpha_{c,k}^{(t)} - \alpha_i^{(t)}$
    - The *higher staleness*, the larger sparse rate!
    - The *larger cluster scale*, the larger sparse rate!
    - The *more round*, the larger sparse rate!
  - **Rationale**
    - **Efficiency**: partial training helps to reduce computation cost
    - **Staleness Robustness**: we only asynchronously aggregate under the masked area, larger mask could limit the influence of staleness

$$w_{g,k}^{(t+1)} \odot m_i^{(t-\tau_i)} = ((1-\alpha_i^{(t)})w_{g,k}^{(t)} + \alpha_i^{(t)} w_i^{(t-\tau_i)}) \odot m_i^{(t-\tau_i)}$$

Our Solutions

• Summary of our solutions

Compound impact

A-phase — *decay coefficient α*

C-phase — *dynamic cluster size $|\mathcal{C}_k|$*

explain

Two theorems

THEOREM 1. *(Clustering Error under Asynchrony). When clustering relies on a similarity matrix A' derived with asynchronous model parameters, the mis-clustering rate p is bounded by:*

$$p = O\left(\lambda\alpha\sqrt{\sum_{i=1}^{n}\left(\sum_{j=1}^{n}\|\tau_i - \tau_j\|^2\right)}\right) \tag{4}$$

where $\lambda = \eta Q\theta U$, and η is the learning rate, Q is the local training steps, U is the upper bound of gradient, θ is the upper bound of staleness (details in Appendix A.1.1).

+

THEOREM 2. *(Convergence of Training Objective). The training objective $\mathcal{P}$ decreases monotonically, and thus the CFL framework converges under asynchrony, if the following condition is met:*

$$\alpha \leq \frac{\Omega(t)h_i}{|C_k|} \tag{5}$$

where $|C_k|$ is the size of cluster u_k, h_i is the computational capacity of c_i, and $\Omega(t)$ is a time-decreasing function (details in Appendix A.1.2).

motivate

Effective solutions

Bi-level asynchronous aggregation $\leftarrow \alpha_{c,k}^{(t)} \rightarrow$ **Buffer-aided dynamic clustering**

the three modules are unified by parameter $\alpha_{c,k}^{(t)}$

Extra solutions

CASA+: sparse training

Outline

- Background & Motivation
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

Experiments

- **Setup**
  - **Dataset**
    - **MNIST, CIFAR10, FEMNIST, IMU, HARBox**
  - **Simulation**
    - **Different non-IID settings are simulated, including**
      - **Dirichlet distribution-based setting**
      - **Realistic setting**
- **Running Information**
  - **CPU: AMD Ryzen 9 5950X 16-Core Processor**
  - **GPU: NVIDIA GeForce RTX 3090**

Experiments

- **Comparing methods**
  - **Local Training:**
    - **Each client trains its model only with its local data**
  - **Sync FL Algorithms:**
    - **FedAvg, FedProx, CFL, IFCA, ICFL**
  - **Async FL Algorithms:**
    - **FedAsync, FedBuff, CFL-Async, IFCA-Async, ICFL-Async**
  - **Ours:**
    - **CASA, CASA+ (CASA** with sparse training**)**
- **Evaluation metrics**
  - **Time to Convergence**
  - **Time to Given Accuracy**
  - **Accuracy**

Experiments

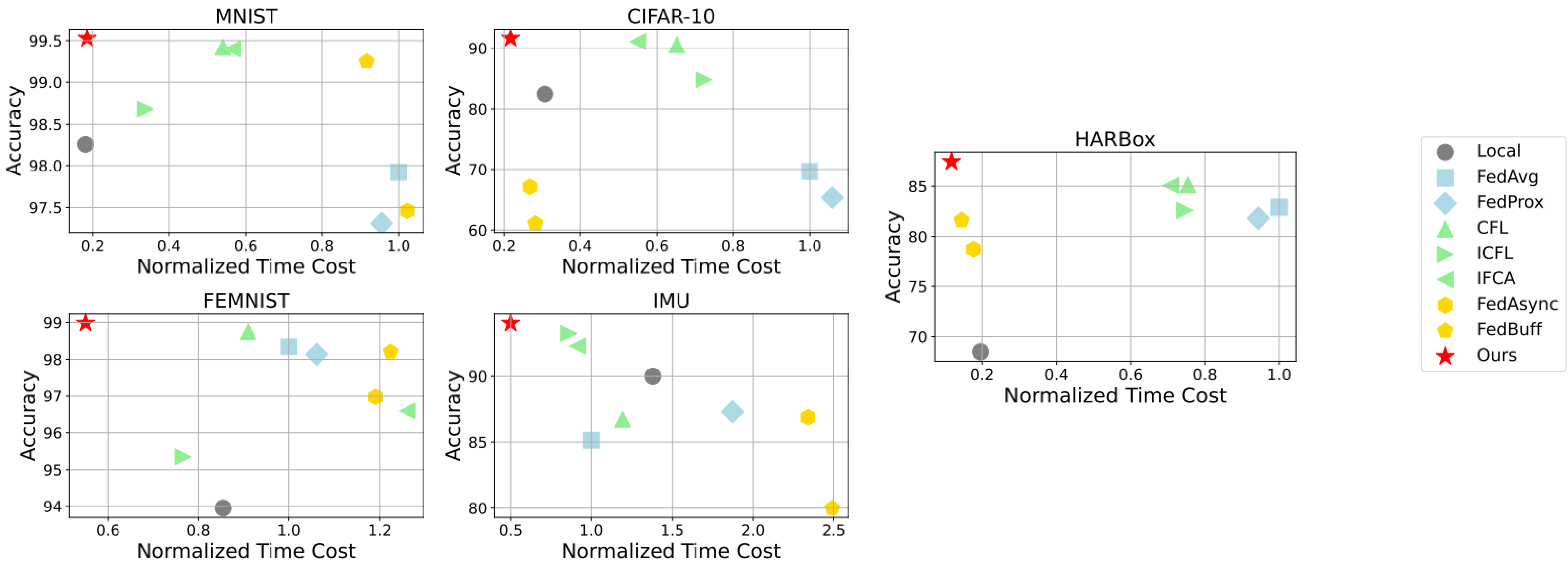
- ## Time-to-Accuracy

Type	Method	MNIST		CIFAR-10		FEMNIST		IMU		HARBox	
		Acc	Time	Acc	Time	Acc	Time	Acc	Time	Acc	Time
N/A	Standalone	98.26	4.9	82.6	35.2	93.95	/	90.00	41.9	69.48	/
	FedAvg	97.92	80.24	69.49	/	98.34	85.01	85.71	89.08	82.90	251.42
	FedProx	97.31	104.68	65.83	/	98.14	114.34	87.57	96.05	81.80	390.96
	CFL	99.42	30.63	90.50	209.55	98.75	75.14	86.29	94.93	85.06	170.99
Sync		99.40(3)	11.21	89.10(3)	209.62	97.77(2)	145.34	94.28(2)	80.73	83.73(2)	334.34
	IFCA(k)	99.50(4)	12.8	90.88(4)	77.88	96.59(5)	237.46	92.67(3)	73.28	85.06(4)	226.53
		99.48(5)	5.61	91.14(5)	80.39	95.37(8)	279.85	91.81(4)	148.6	87.09(6)	220.98
	ICFL	98.68	12.18	84.19	36.49	95.35	121.65	93.23	30.45	82.58	126.43
	FedAsync	97.46	109.53	67.66	/	96.97	183.57	86.86	64.9	78.72	158.97
	FedBuff	99.25	53.31	61.11	/	98.21	82.63	80.00	282.9	81.62	55.7
	CFL-Async	99.23	9.54	89.97	145.8	98.68	36.67	87.71	69.3	82.43	59.8
Async		99.28(3)	9.53	83.41(3)	195.30	98.39(2)	81.07	89.61(2)	143.1	77.58(2)	252.60
	IFCA-Async(k)	98.88(4)	8.83	88.99(4)	80.70	97.78(5)	118.27	85.62(3)	143.1	78.13(4)	215.20
		99.32(5)	8.57	87.98(5)	83.20	97.62(8)	126.77	89.14(4)	87.77	76.81(6)	236.30
	ICFL-Async	98.82	5	83.30	25.3	94.66	/	91.52	81.83	79.65	92.00
Ours	CASA	**99.52**	**2.80**	**91.45**	23.4	**98.97**	36.2	**95.33**	37.47	**87.38**	54.8
	CASA+	99.34	4.80	90.64	**20.3**	98.53	**35.03**	94.57	**22.71**	87.21	**53.6**

CASA outperform existing Sync & Async CFL algorithms under both Accuracy and Time-to-Accuracy

Experiments

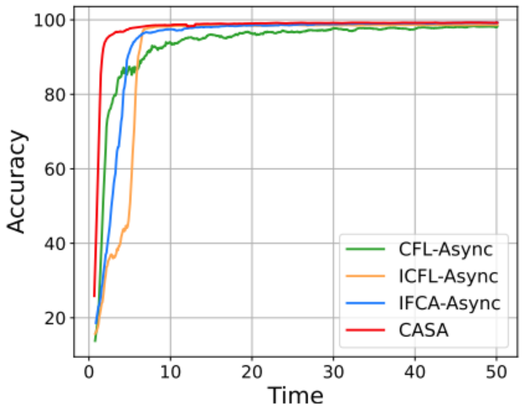
- **Convergence Time & Accuracy**



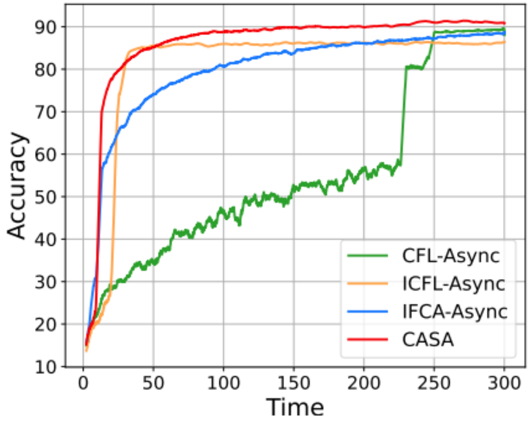
CASA outperform existing baselines under both Accuracy and Convergence Time

Experiments

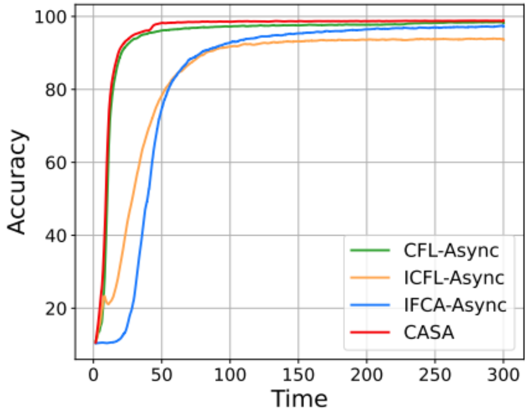
- **Time & Accuracy of async baselines**



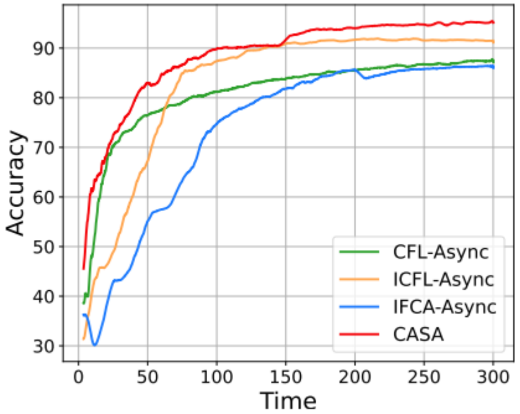
(a) MNIST



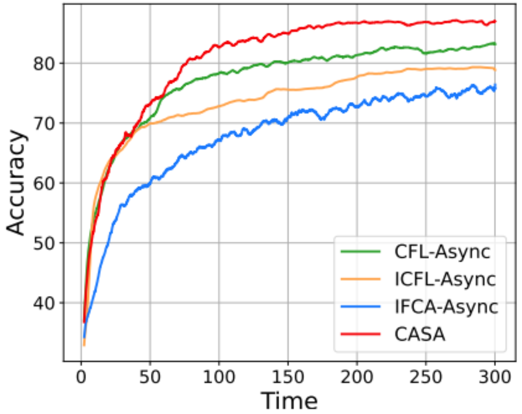
(b) CIFAR-10



(c) FEMNIST



(d) IMU

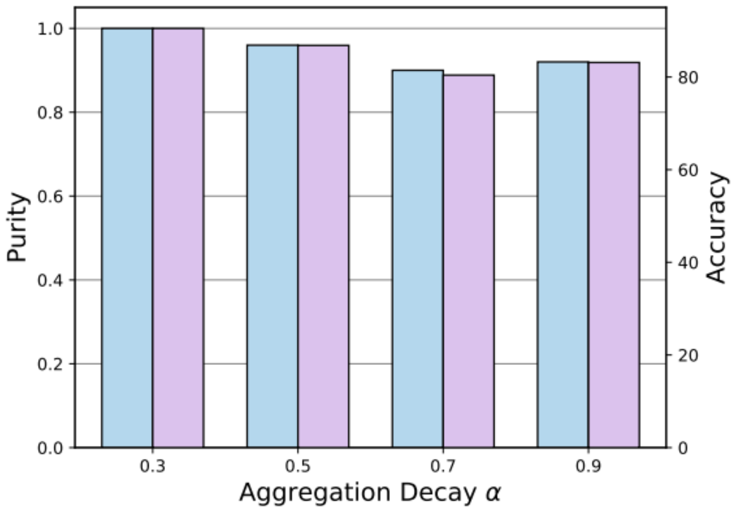


(e) HARBox

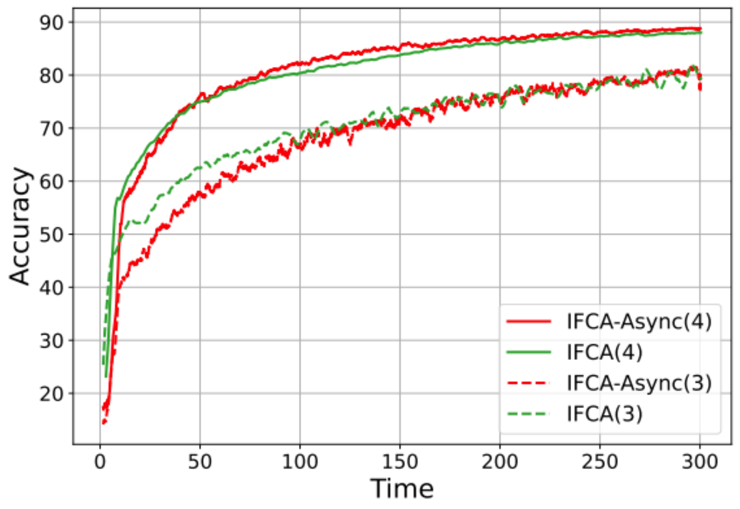
CASA outperform async version of existing baselines

Experiments

- **Impact on asynchrony on clustering**
  - For Hierarchical clustering (as CFL), aggregation decay influences the accuracy
  - For Dynamic clustering (as IFCA), asynchrony will not bring convergence boost



(a) Hierarchical clustering

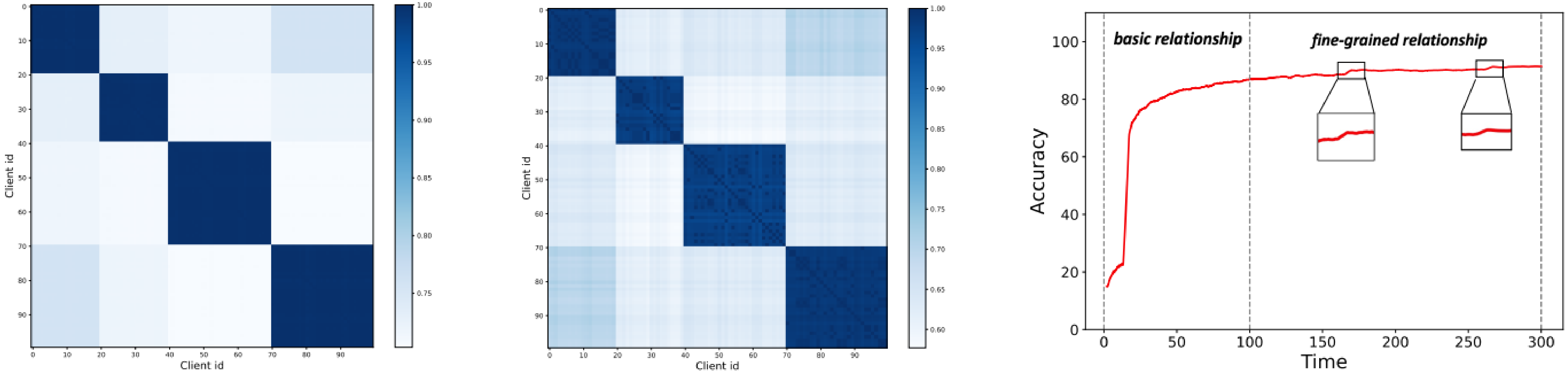


(b) Dynamic clustering

Asynchrony exerts impact on both hierarchical and dynamic clustering!

Experiments

- **Effectiveness of clustering**
  - We visualize the similarity matrix of clients
  - We observe accuracy boost with the clustering in CASA



(a) Sims at $t = 100$ (b) Sims at $t = 300$ (c) Accuracy at two stages

CASA can gradually captures more detailed relationships and boost accuracy

Outline

- **Background & Motivation**
- **Problem Statement**
- **Our Solutions**
- **Experiments**
- **Conclusion**

Conclusions

- We explore the ***asynchronous clustered federated learning***, showing that the ***compound impact*** of asynchrony and clustering

- We propose ***CASA***, a new framework that solves the compound impact simultaneously

- Extensive experiments on various datasets validate the performances on ***accuracy and efficiency***

THANK YOU

if you have problems, feel free to email
boyliu@buaa.edu.cn
or talk with me at Poster 90, 27th August