Exploring the Impact of Client Mobility on Decentralized Federated Learning Performance

Author: Santiago de Heredia Tenorio
Responsible Professor: Jérémie Decouchant
Supervisor: Bart Cox

1) Introduction

- **Federated Learning** (FL) is a privacy-preserving machine learning paradigm that permits multiple clients to benefit from a shared model trained from clients' data, sharing model parameters instead of client data.
- **Decentralized Federated Learning** (DFL) is a branch of FL that deals with clients directly communicating with each other and aggregating each other's models as opposed to using a central server.
- **Client mobility** describes how users may move within a FL system.
- For Hierarchical Federated Learning (HFL) systems, it has been shown that user mobility can affect learning performance [3].
- However, the effects of client mobility on DFL systems have **not been studied**.

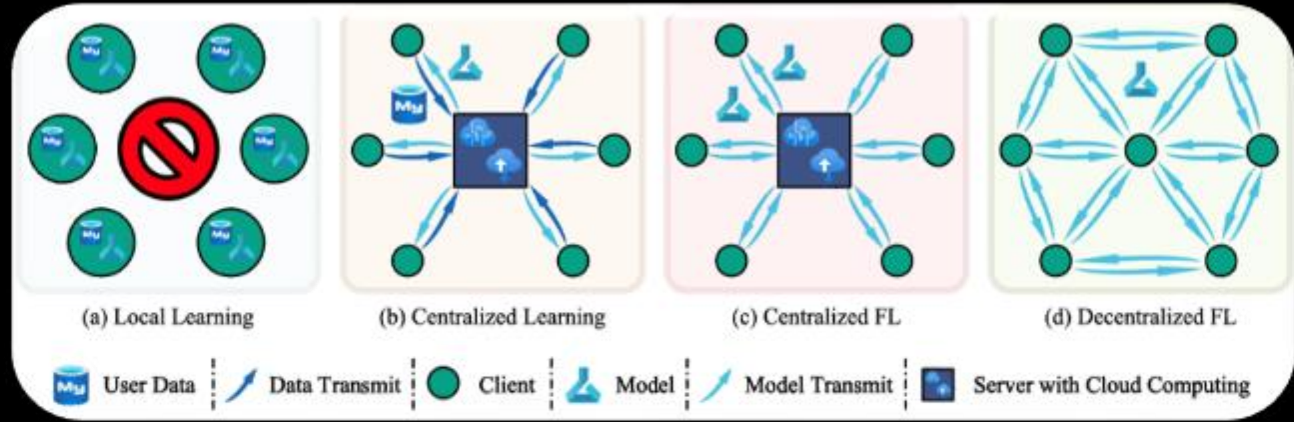

Fig 1: Illustration of different types of learning architectures [1]

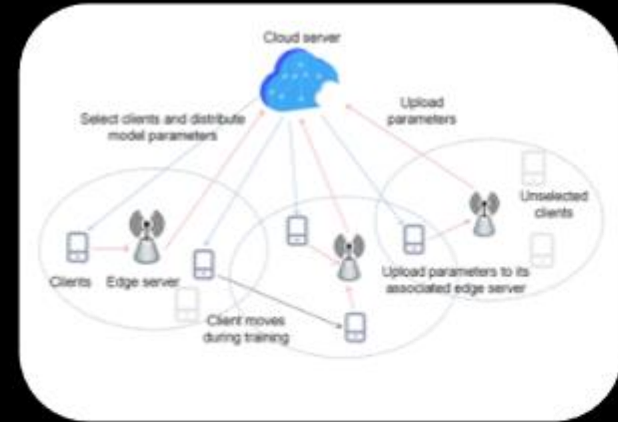

Fig 2: HFL with client mobility [2]

2) Research Questions

- **How does varying client mobility affect the global learning performance of DFL systems?**
- **How does learning performance differ between high-mobility and low-mobility clients?**
- **How can said performance difference between clients be decreased?**
- **How can said difference be used to improve global learning performance?**

3) Methodology

- We develop a **theoretical model** describing client mobility in a DFL system.
  - In this model, we define ***high-mobility*** (HM) and ***low-mobility*** (LM) clients.
  - From here, we can vary the proportion p of HM clients
- We extend the **DecentralizePy** [4] framework to be able to simulate the developed client mobility model.
- We develop a **mobility-aware model aggregation algorithm** (see Alg. 2) to decrease learning performance disparities between LM and HM clients compared to the baseline (see Alg. 1)
  - A client aggregates neighbours' models by taking the **neighbour's speed into account**, with models of faster neighbours having greater weighting.
  - The extent of which speed is taken into account is controlled by a **hyperparameter** α.
- We conduct experiments using:
  - N = 48 clients.
  - CIFAR-10 dataset (non-IID and IID partitioning).
- For each model aggregation algorithm, we analyze:
  - How **global learning performance** is affected.
  - How the **performance gap** between high-mobility and low-mobility clients is affected.

$$W_{ij} = \frac{1}{N}$$

Alg 1: Baseline model aggregation method (plain). W_{ij} is the weight that client i gives to the model of client j (one of the neighbours of client i). N is the number of neighbours of client i at iteration k

Alg 2: Mobility-aware aggregation for a given iteration k. X_i is a vector representing the normalised speeds of each of client i's neighbours

$$W_{ij} = \frac{1}{N} + \alpha\left(X_{ij} - \frac{1}{N}\right)$$

where

$$X_{ij} = \frac{s_j}{\sum_{x \in \mathcal{M}_i(t_k)} s_x}$$

4) Results & Conclusions

Here we present **non-IID** results. For IID data partitioning, results are less exciting, with performance advantages and advantages between HM and LM clients being significantly smaller.

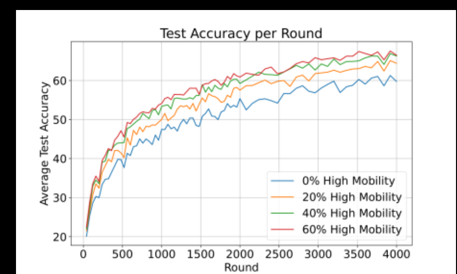

Fig 3: Test accuracy per round as p is increased from 0% to 60%

Increasing p leads to **marked improvements in learning performance**

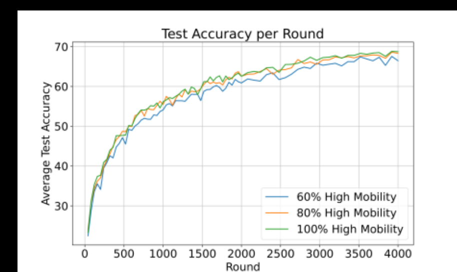

Fig 4: Test accuracy per round as p is increased from 60% to 100%

Increasing p has **diminishing returns**, with negligible differences between p values of 80% and 100%

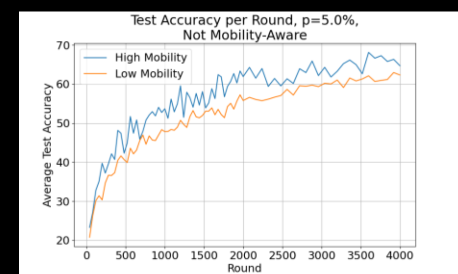

Fig 5: Test per round for high-mobility and low-mobility clients for p=5%

We see **HM clients have a significant performance advantage over LM clients** in environments with low p

	$p = 0.05$	$p = 0.2$	$p = 0.4$	$p = 0.6$	$p = 0.8$
Baseline	+4.81	+4.13	+2.53	+2.04	+0.90
Mobility-Aware ($\alpha = 0.4$)	+3.16	+3.17	+2.64	+2.28	+1.19
Advantage Reduction	+1.65	+0.96	-0.11	-0.24	-0.29

Tab 1: Test accuracy (%) advantage of HM clients over LM clients throughout experiments, for both baseline and mobility-aware aggregation (α=0.4). The advantage reduction in using α=0.4 mobility-aware aggregation compared to baseline is calculated.

Mobility-aware aggregation reduces the performance disparity between HM and LM clients in environments with low p, with best overall results at α=0.4 for the examined system.

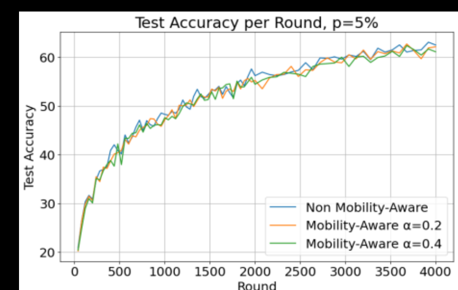

Fig 6: Test accuracy per round for p=5%, for baseline aggregation and mobility-aware aggregation (α=0.2 and α=0.4)

Mobility-aware aggregation with moderate hyperparameters (α≠1.0) **does not affect global learning performance**

5) Limitations & Future Work

- Due to the **synthetic** mobility dataset, results of this work might diverge from real-world systems.
  - An interesting direction for future work would be to use **real-life mobility traces**.
- DFL systems have a **broad taxonomy** [1], yet only one specific of system was examined
  - Investigating a DFL system with a different **communication protocol**, **aggregation paradigm** or **iteration order** is another interesting direction for future work.
- The mobility-aware model aggregation algorithm in partially **reliant on the hyperparameter** α
  - **Automating** α could be a valuable future work.

References

[1] Yuan, L., Wang, Z., Sun, L., Philip, S. Y., & Brinton, C. G. (2024). Decentralized federated learning: A survey and perspective. IEEE Internet of Things Journal.

[2] Yang, Jian, Yan Zhou, Wanli Wen, Jin Zhou, and Qingrui Zhang. 2023. "Asynchronous Hierarchical Federated Learning Based on Bandwidth Allocation and Client Scheduling" Applied Sciences 13, no. 20: 11134

[3] Feng, C., Yang, H. H., Hu, D., Zhao, Z., Quek, T. Q., & Min, G. (2022). Mobility-aware cluster federated learning in hierarchical wireless networks. IEEE Transactions on Wireless Communications, 21(10), 8441-8458.

[4] Dhasade, A., Kermarrec, A. M., Pires, R., Sharma, R., & Vujasinovic, M. (2023, May). Decentralized learning made easy with DecentralizePy. In Proceedings of the 3rd Workshop on Machine Learning and Systems (pp. 34-41).

CSE3000 Research Project Final Presentation Poster

TU Delft