Breaking the Capacity Bottleneck in Model-Heterogeneous Federated Learning via Gradual Model Restoration (FedGMR)

Chengjie Ma[1], Seungeun Oh[1,2], Jihong Park[2], Seong-Lyun Kim[1]

[1]School of Electrical and Electronic Engineering, Yonsei University, Seoul, South Korea
[2]Information Systems Technology and Design Pillar, Singapore University of Technology and Design, Singapore

Talk Overview

1. **Background and Problem**
2. **FedGMR**
3. **Theoretical Analysis**
4. **Empirical Results**

Talk Overview

Background / Client-Heterogeneous FL

- In practical FL, clients have very different communication and computation resources.
- **Bandwidth-constrained clients (BCCs)** are especially hard to keep effective during training.
- In heterogeneous deployments, the key challenge is sustained contribution from weak clients.

$$\text{heterogeneous clients} \implies \text{uneven participation and uneven optimization influence}$$

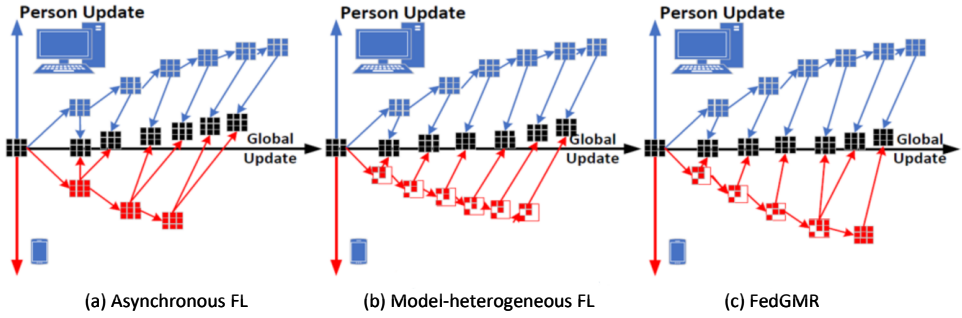
- The key challenge is that weak clients are hard to keep contributing effectively to the global model.

Background / Traditional MHFL and Our Problem

- In client-heterogeneous FL, BCCs have limited communication and computation resources.
- Traditional **model-heterogeneous federated learning (MHFL)** addresses this by assigning smaller sub-models to weak clients.
- This improves efficiency, but most MHFL methods use **fixed** model density.

- **Our problem:** model capacity limitation in MHFL.
  - Small sub-models learn quickly early, but become under-parameterized later and lose effective contribution.

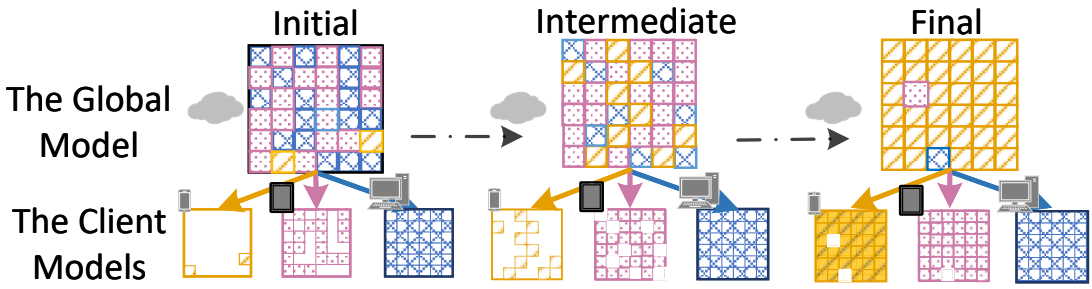
Background / Motivation



(a) Asynchronous FL (b) Model-heterogeneous FL (c) FedGMR

- AFL relaxes synchronization, so fast clients do not need to keep waiting for slow ones, but weak clients can easily be marginalized; fixed MHFL improves early participation, but later suffers from limited model capacity.
- **Motivation:** when a sub-model is no longer sufficient, restoring its capacity can reactivate effective training.

Introduction / FedGMR Overview

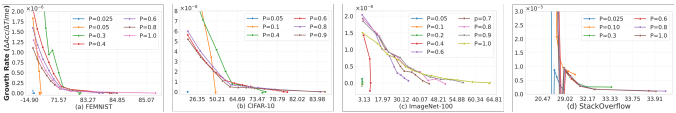


- FedGMR progressively restores client capacity so that BCCs remain effective throughout training.

Talk Overview

1. Background and Problem
2. **FedGMR**
   - Feasibility Study
   - Core Mechanism
   - Async and MaskFedAvg
3. Theoretical Analysis
4. Empirical Results

Method / Empirical Validation of Dynamic Density



FEMNIST		CIFAR-10		ImageNet-100		StackOverflow	
D.	Int.	D.	Int.	D.	Int.	D.	Int.
0.3	9.35–75.38	0.1	12.75–35.98	0.6	2.66–10.18	0.025	12.37–12.46
0.4	75.55–83.35	0.4	36.84–61.21	0.8	10.69–18.85	0.10	12.47–30.78
0.6	83.54–84.33	0.6	61.64–72.80	0.9	19.30–38.33	0.3	30.87–33.36
0.8	84.35–84.58	0.8	73.04–83.49	0.8	38.65–41.71	0.6	33.73–33.88
1.0	84.65–85.20	1.0	83.55–84.51	1.0	42.01–63.89	0.8	33.89–33.94

- Under homogeneous training, all clients use the same fixed density in each run.
- Low-density sub-models achieve the steepest gains in early training.
- Higher densities become better in later stages, when more capacity is needed.
- Across datasets, the best density moves from low to high as training progresses.
- The optimal density is dynamic, which directly motivates gradual restoration.

Method / When and How to Restore Sub-models

- Restoring too early makes MHFL closer to AFL, while restoring too late behaves similarly to fixed-density MHFL.
- Ideally, one would quantify the trade-off among model capacity, convergence speed, and training latency, but in practice this is highly task- and architecture-dependent.
- We consider an idealized but insightful scenario: when a sub-model plateaus, its growth rate is approximately zero; if restoration is beneficial, the subsequent growth rate becomes positive again.

$$\mathcal{A}(t_1, t_2) := -\frac{F(w_{t_2}) - F(w_{t_1})}{t_2 - t_1}$$

- Once a sub-model plateaus, restoring capacity yields a strictly higher subsequent speed: $\mathcal{A}(t_2, t_3) > \mathcal{A}(t_1, t_2)$ for some $t_1 < t_2 < t_3$.
- FedGMR uses a unified **Early Stopping Mechanism (ESM)** and a density ladder $\mathcal{R} = \{r_1 < \cdots < r_M\}$, e.g. $0.1 \to 0.2 \to 0.5 \to 1.0$.

Method / Workflow of FedGMR

- The server dispatches masked sub-models according to each client's current density.
- **MaskFedAvg** averages each coordinate only over active clients, avoiding zero-dilution.

$$w^{(n)} = \frac{\sum_{i \in C_k^{(n)}} w_i^{(n)}}{|C_k^{(n)}|}, \qquad C_k^{(n)} = \{i \mid m_{i,k}^{(n)} = 1\}$$

- ESM periodically checks plateaued clients and promotes them to the next density level.
- **Asynchronous coordination** avoids waiting after density restoration.
- **IMS** is an alternative option: it splits a sub-model into non-overlapping blocks and uses extra computation to reduce server-to-client communication.

Talk Overview

1. Background and Problem
2. FedGMR
3. **Theoretical Analysis**
   - Result 1
   - Result 2
4. Empirical Results

Theoretical Analysis / Result 1: Aggregation Effect

$$
\begin{aligned}
\mathbb{E}[F(W_{k+1})] - \mathbb{E}[F(W_k)] \leq & -\frac{G_0}{2}\,\mathbb{E}\|\nabla F(W_k)\|^2 \\
& + \frac{3G_0J_0}{2}\,\mathbb{E}\sum_{n=1}^{N}\Big\|\frac{1}{|C_k^{(n)}|}\sum_{j\in C_k^{(n)}}\nabla F_j^{(n)}(W_k)\Big\|^2 \\
& + \frac{G_0H_0}{6}\sum_g \frac{|C_g|}{c_{g,k}^*} f_1^2(\rho_{g,k})G^2 + \frac{3G_0I_0}{2}\sum_g \frac{|C_g|}{(c_{g,k}^*)^2} f_2^2(\rho_{g,k})\sigma^2
\end{aligned}
$$

- **Conclusion:** mask-aware aggregation approximates the global gradient through partial client gradients.
- It also introduces larger variance and bias terms through coverage-related factors such as $|C_g|/c_{g,k}^*$ and $|C_g|/(c_{g,k}^*)^2$.
- During model restoration, this aggregation scheme preserves more stable gradient evolution.

Theoretical Analysis / Result 2: Why GMR Helps

$$\mathcal{S}_{\text{feas}}^{(k)} = \text{span}\{\nabla F_i(W_k) \odot m_{i,k}\} \subseteq \mathcal{S}_{\text{feas}}^{(k+1)}$$

- **Conclusion:** GMR is effective because restoration expands the feasible optimization subspace over training.
- This is reflected in the gradient terms: $m_{i,k} \leq m_{i,k+1}$ leads to $\mathcal{S}_{\text{feas}}^{(k)} \subseteq \mathcal{S}_{\text{feas}}^{(k+1)}$ and increased coverage $|C_k^{(n)}| \uparrow$.
- As a result, the optimization domain becomes larger and the solution can move closer to the full-model optimum.

Talk Overview

1. Background and Problem
2. FedGMR
3. Theoretical Analysis
4. **Empirical Results**
   - Experimental Setup
   - Baseline Comparison
   - Ablation
   - Generality
   - Aggregation Analysis

Empirical Results / Experimental Setup

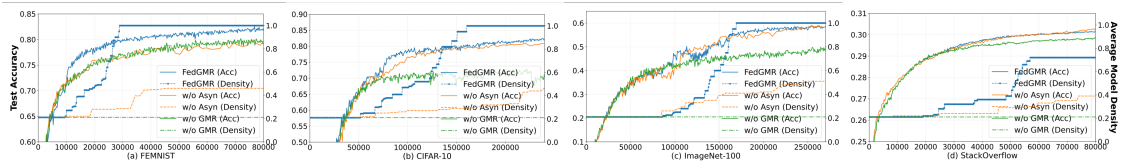
- We evaluate on **FEMNIST**, **CIFAR-10**, **ImageNet-100**, and **StackOverflow**.
- Each table reports both **IID** and **Non-IID** splits. The Non-IID setting follows label- or client-skewed partitioning used in the paper.
- Client heterogeneity is further divided into multiple levels. The slides mainly report the **high-heterogeneity** case, where bandwidth-constrained clients are most likely to be marginalized.
- **Aux** denotes the auxiliary asynchronous MHFL workflow used to make each baseline compatible with dynamic restoration.
- **Fixed-time restoration** replaces the event-driven plateau trigger with a preset restoration schedule; full configurations and additional heterogeneity settings are provided in the paper.

Empirical Results / Comparison with Baselines

Method	FEMNIST		CIFAR-10		ImageNet-100		StackOverflow	
	IID	Non-IID	IID	Non-IID	IID	Non-IID	IID	Non-IID
SynFL (FedAvg)	75.62	74.64	70.75	68.46	48.65	47.46	24.38	24.38
FedAsyn (FedAsync)	81.34	81.03	82.86	79.28	58.19	55.36	25.86	25.83
HeteroFL	82.22	79.80	81.06	75.64	41.90	28.01	29.42	29.17
FedGMR	**82.67**	**81.86**	**84.52**	**81.68**	**60.84**	**58.01**	**30.00**	**30.07**

- Results are shown under **high heterogeneity**; each entry reports **IID / Non-IID**.
- Client heterogeneity is further stratified in the full paper; this slide shows the strongest-stress regime for a concise comparison.
- FedGMR consistently improves over **SynFL**, **FedAsyn**, and **HeteroFL**, with the clearest gains on harder datasets such as ImageNet-100.

Empirical Results / Ablation of GMR and Asynchrony



- This figure isolates the two key factors: **gradual restoration** and **asynchronous coordination**.
- The dominant gain comes from **GMR**: removing restoration leads to earlier saturation and a larger final gap across datasets.
- **Asynchrony** is not always effective, but it helps guarantee a stronger lower bound under latency imbalance.

Empirical Results / Generality of the Restoration Mechanism

GMR on Other MHFL Methods

Method	Base	+Aux	+Aux+GMR
HeteroFL	79.80	80.42	82.29
FjORD	81.76	82.16	82.20
FedRolex	77.83	80.36	81.58
FIARSE	78.77	79.56	81.97

- **Aux** is the auxiliary asynchronous workflow that makes each MHFL baseline compatible with gradual restoration.
- GMR remains beneficial across different pruning and sub-model construction rules.

Fixed-Time Restoration

Dataset	ES	Fixed	w/o Asyn	w/o GMR
FEMNIST	81.71	80.07	78.80	79.51
CIFAR-10	81.68	80.42	80.53	71.93
ImageNet-100	58.01	56.38	57.85	48.28
StackOverflow	30.04	30.15	30.11	29.76

- **Fixed-time restoration** uses a preset restoration schedule instead of ESM-based plateau detection.
- Even with a fixed-time trigger, gradual restoration still preserves the main gain.

Empirical Results / Different Aggregation Schemes

FA: zero-padded FedAvg over all clients. **GA:** aggregate active gradients but still divide by the total number of clients. **MA:** average each coordinate only over the active clients.
w/o GMR: the same training pipeline without gradual model restoration, i.e., fixed sub-model density throughout training.

Method	FEMNIST		CIFAR-10		ImageNet-100		StackOverflow	
	IID	Non-IID	IID	Non-IID	IID	Non-IID	IID	Non-IID
GMR + MA	**82.67**	**81.86**	**84.52**	**81.68**	**60.84**	**58.01**	**30.00**	**30.07**
GMR + GA	81.27	80.95	83.73	78.83	58.36	57.32	29.56	29.71
GMR + FA	80.55	79.79	10.00	10.00	59.68	55.22	25.62	28.24
w/o GMR + MA	**82.18**	**79.91**	**81.65**	71.93	53.49	48.28	**29.68**	**29.76**
w/o GMR + GA	80.60	78.59	80.66	**74.37**	**60.74**	56.66	29.00	29.11
w/o GMR + FA	79.16	78.09	10.00	10.00	60.35	**58.65**	25.95	25.94

- This page studies the **joint effect** of aggregation choice and model restoration.
- With **GMR**, **MA** is the most compatible choice: coordinate-wise normalization keeps restored parameters stable and preserves GMR's gains.
- Without restoration, **GA** can outperform **MA** in some cases, while **FA** is usually the weakest because zero padding causes weight shrinkage under structural heterogeneity.

Conclusion / Takeaways

- The key issue in MHFL is not only making weak clients train faster in early rounds.
- More importantly, their model capacity should be restored when fixed sparse models become ineffective later.
- Mask-aware aggregation reconstructs the global gradient from partial local gradients, but introduces additional variance and bias terms; meanwhile, it preserves gradient stability during model restoration.
- Theoretical analysis and experiments support the same message: **late-stage capacity restoration matters.**


raiio


ICML
International Conference
On Machine Learning


SINGAPORE UNIVERSITY OF
TECHNOLOGY AND DESIGN

Thanks