Energy-Efficient Hierarchical Federated Learning in Massive Wireless IoT Networks

Renan R. de Oliveira[1,2], **Kleber V. Cardoso**[1] **and Antonio Oliveira-Jr**[1,3]

[1]Institute of Informatics – Universidade Federal de Goiás (UFG), Goiânia, Brazil

[2]Instituto Federal de Goiás (IFG), Goiânia, Brazil

[3]Fraunhofer Portugal AICOS, Porto, Portugal

renan.rodrigues@ifg.edu.br, {kleber,antoniojr}@ufg.br

Abstract. *Federated Learning (FL) enables collaborative model training without sharing raw data, preserving privacy and reducing communication overhead. On the other hand, in Internet of Things (IoT) wireless networks, FL faces issues such as limited resources, unreliable communication channels, and large delays. Hierarchical Federated Learning (HFL) addresses these issues using a tree topology with intermediate servers to reduce communication distances, improve aggregation efficiency, and mitigate transmission failures. However, current algorithms are not well-suited to address the scalability challenges posed by the massive scale of beyond-5G networks. In this context, we propose a novel HFL algorithm called HFLw$_{\mathrm{Opt}}$, which dynamically optimizes communication and computation resources in massive wireless IoT networks, maximizing successful transmissions, minimizing energy use, and reducing training latency. Our simulation with over* $1,000$ *devices, utilizing three levels of aggregation, demonstrates that HFLw$_{\mathrm{Opt}}$ surpasses baselines with fixed resource allocations. The results reveal a reduction of up to* 45.34% *in energy efficiency and* 77.01% *in training latency for the MNIST-based dataset, and* 42.23% *in energy efficiency and* 76.77% *in training latency for the FMNIST-based dataset.*

1. Introduction

The ultra-low latency demands of 5G/6G network applications, combined with strict privacy constraints, necessitate the deployment of distributed ML systems at the network edge [Stergiou and Psannis 2022]. Furthermore, traditional ML approaches that centralize data on servers for training are limited due to high communication costs and privacy risks, making them unsuitable for modern applications requiring low latency and data privacy [Nakayama and Jeno 2022].

FL has emerged as a promising solution for the distributed training of ML models, enabling multiple devices to collaboratively train a global model without the need for data sharing, thereby preserving user privacy [McMahan et al. 2016]. Transmitting model parameters instead of training data conserves energy, optimizes network resources, and reduces latency [Yang et al. 2022]. Furthermore, edge computing integrated into the network infrastructure, as proposed by the Multi-Access Edge Computing (MEC) [ETSI 2022] approach, has facilitated the deployment of FL parameter servers at the network edge, positioned closer to the devices where models are trained. However, FL in wireless networks faces challenges due to limited communication resources and the

unreliability of the wireless channel, which can lead to failures in distributed training [Chen et al. 2021]. Moreover, traditional FL systems based on a server-client architecture face significant challenges due to resource constraints in wireless networks and extended transmission distances [Xu et al. 2022]. Consequently, these traditional FL solutions may struggle to adapt effectively to large-scale scenarios, which are inevitable in massive MEC networks [You et al. 2023].

HFL has emerged as a promising solution based on a tree topology that leverages edge servers positioned closer to devices [Liu et al. 2020]. The scalable structure and communication efficiency make HFL-based solutions effective for massive networks, where direct communication between many clients with a single central server may become impractical [Wu et al. 2024]. This framework improves the scalability and efficiency of massive FL networks by organizing communication into layers, significantly reducing bandwidth overhead and central server computational complexity, and enabling shorter transmission distances [You et al. 2023]. The training process at HFL involves at least two aggregation steps: edge aggregation, conducted on edge servers near the devices, and global aggregation, performed on the central server following multiple local aggregations [Wu et al. 2024]. However, there can be additional layers between the central server and edge devices. For instance, edge servers that connect edge devices and the central server can formulate one or multiple layers, making a tree-like topology with the highest level of the tree being the central server and the lowest level being edge devices [Nakayama and Jeno 2022].

In this paper, we propose an HFL algorithm called HFLw$_{\mathrm{Opt}}$ (*Hierarchical FL Wireless Optimizer*)[1], designed to support the training of ML models in massive hierarchical wireless IoT networks collaboratively and without sharing device data. Our algorithm is formulated as an optimization problem that focuses on the efficient allocation of computing and communication resources with the objective of maximizing the number of successful transmissions while minimizing the energy consumption of devices and reducing training latency. This algorithm enhances scalability and performance by introducing an additional intermediate layer between the edge server (ES) on the first communication hop and the root ES. This extra aggregation layer enables progressive model aggregation before reaching the root ES, reducing communication overhead and improving overall learning efficiency. Moreover, HFLw$_{\mathrm{Opt}}$ is extensible to n layers of aggregation, making it highly scalable and adaptable for deployment in massive wireless IoT networks. We conducted experiments varying device selection per communication round and distributing devices and edge servers (ESs) at different distances to account for transmission errors and dynamic network conditions. We carried out our experiments to evaluate the performance of HFLw$_{\mathrm{Opt}}$ in scenarios with more than 10 hierarchically distributed ESs and more than $1,000$ devices.

The main contributions of this paper are: *(i)* **Resource awareness in massive wireless IoT networks**: HFLw$_{\mathrm{Opt}}$ uses simulation models for HFL tasks, optimizing energy efficiency, training latency, and adaptability to network conditions by considering power, bandwidth, and CPU resources; *(ii)* **Novel HFL approach**: HFLw$_{\mathrm{Opt}}$ enhances locally trained model representativeness by selecting devices that maximize data coverage, scales communication resources to minimize energy consumption, and reduces train-

[1] Available at https://github.com/LABORA-INF-UFG/HFLwOpt

ing latency while maximizing device participation; *(iii)* **Multi-level aggregation design**: HFLw$_{\mathrm{Opt}}$ algorithm supports n aggregation layers, improving scalability and enabling efficient operation in complex hierarchical wireless IoT networks; *(iv)* **Efficient exact solution**: Communication resource scaling is achieved via an optimal Mixed Integer Linear Programming (MILP) solution using the PuLP library; *(v)* **Robustness in large-scale IoT**: HFLw$_{\mathrm{Opt}}$ is validated with over $1,000$ devices, demonstrating high scalability, robustness, and efficiency in massive wireless IoT networks; *(vi)* **Performance and insights**: Experimental results show superior accuracy trends, significant energy savings, reduced training latency, and maintained global model quality.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 presents the mathematical models that were used in simulating the massive wireless IoT network for HFL tasks. Section 4 describes the formulation of HFLw$_{\mathrm{Opt}}$. Section 5 discusses the simulation results and their analysis. Finally, Section 6 presents the concluding remarks and outlines directions for future work.

2. Related Work

Edge-based aggregation in HFL has gained attention as a potential improvement over traditional FL. The client-edge-cloud hierarchical system by [Liu et al. 2020] reduces cloud communication costs by aggregating models in stages. Despite its benefits, it overlooks dynamic resource allocation, transmission errors, and latency variability. In contrast, [Luo et al. 2020] proposed an HFL structure to jointly minimize energy consumption and delay through resource scheduling, showing potential for low-latency and energy-efficient, but its analysis lacked scalability for large-scale networks. In [You et al. 2023], the authors introduce an HFL architecture for mobile MEC networks, enabling semi-asynchronous updates to reduce latency. However, it did not explore architectures with multiple hierarchical levels.

Table 1. Related Work

Ref.	Resource Scheduling	Multi-Level Aggregation Design	Multi-Level Aggregation Validation	Energy Efficiency	Devices (≥ 1000)
[Liu et al. 2020]	✗	✗	✗	✗	✗
[Luo et al. 2020]	✓	✗	✗	✓	✗
[Simos et al. 2022]	✓	✓	✗	✓	✗
[Xu et al. 2022]	✓	✓	✗	✓	✗
[You et al. 2023]	✓	✗	✗	✓	✗
[Su et al. 2024]	✓	✓	✗	✓	✗
HFLw$_{\mathrm{Opt}}$ (Ours)	✓	✓	✓	✓	✓

The work of [Xu et al. 2022] optimized HFL efficiency through adaptive resource allocation and dynamic aggregation control, yet simulations were limited to two levels of aggregation in small networks. Other studies addressed specific challenges but had limitations in scalability. The work by [Su et al. 2024] used strategies for dynamic client selection, improving latency but restricting applicability to small networks. In [Simos et al. 2022], the authors combined intermediate and global aggregations to minimize training delays and energy consumption, though its experiments focused on two-level hierarchies, limiting its generalizability to complex scenarios.

Table 1 highlights the HFLw$_{\mathrm{Opt}}$ algorithm. Addressing these gaps, HFLw$_{\mathrm{Opt}}$ is introduced as a scalable solution for massive wireless IoT networks. It dynamically scales computational and communication resources to maximize successful transmissions, minimize energy consumption, and reduce training latency. Tested with more than $1,000$ devices across diverse network conditions, HFLw$_{\mathrm{Opt}}$ demonstrates robustness and scalability in high-density and heterogeneous environments.

3. System Model

3.1. Network Model

Consider a wireless IoT network within a geographic area for an HFL system with L aggregation levels, where $l = 0$ represents the root ES level. The network comprises intermediate ESs with aggregator servers, as in the MEC approach. A set of IoT devices $S_z = \{i_{z,1}, i_{z,2}, \ldots, i_{z,N_z}\}$ is associated with an ES $z \in \mathcal{Z}^l$ in the first communication hop, with each ES potentially co-located with small cell base stations [Poularakis et al. 2020]. From this point on, the ES located in the first communication hop is referred to as the first-hop ES. Intermediate ESs connect to higher-level ESs, forming a hierarchical tree culminating at the root ES. Each participating device has a dataset $\mathcal{P}_{z,i}$ with $n_{z,i} = |\mathcal{P}_{z,i}|$ samples and sends local updates to its connected first-hop ES. The ES performs the first aggregation layer and transmits the aggregated model to the next higher-level ES, continuing until reaching the root ES.

3.2. HFL Model with Tree Topology

Traditional FL employs a two-tier architecture with a central server and N devices, optimizing the global model w_{global} to minimize the loss function $f(w_{global}) = \frac{1}{N}\sum_{i=1}^{N} f(w_i)$, where $f(w_i)$ evaluates the local model w_i using device data $\mathcal{P}_i$. For large-scale tasks, HFL organizes devices into geographically distributed areas, each with potentially thousands of devices [You et al. 2023], forming a hierarchical tree-like topology [Wu et al. 2024]. Devices send updates to the nearest ES z, which aggregates local models as $w_{t+1}^{z} = \frac{1}{N_z}\sum_{i=1}^{N_z} w_{t+1}^{z,i}$, where $w_{t+1}^{z,i} = w_t^{z,i} - \eta\nabla\ell_{z,i}(w_t)$. Aggregated models are forwarded up in the hierarchy, and the root ES computes the global model as $w_{t+1} = \frac{1}{|\mathcal{Z}^l|}\sum_{z=1}^{|\mathcal{Z}^l|} w_{t+1}^{z}$, providing scalability and efficiency.

3.3. Local Model Training and Transmission

Under each first-hop ES $z \in \mathcal{Z}^l$, each local device i trains its local model at round t and sends w_{t+1}^{z} to its ES z. Based on [Chen et al. 2021], the computation time and the energy required of device i in round t for local model training can be expressed by $\mathrm{Tcmp}_{z,i} = \frac{\omega_{z,i} Z(w_{z,i})}{\vartheta_{z,i}}$ and $\mathrm{Ecmp}_{z,i} = \zeta_{z,i}\omega_{z,i}\vartheta_{z,i}^2 Z(w_{z,i})$, where $\vartheta_{z,i}$, $\omega_{z,i}$, and $\zeta_{z,i}$ refer, respectively, to the clock frequency, the number of cycles of the central processing unit, and the energy consumption coefficient of device i connected to the first-hop ES z, to perform training of a model of size $Z(w_{z,i})$.

After each device updates its local model, it sends the update to its first-hop ES z for edge model aggregation. In this paper, we consider the Orthogonal Frequency Division Multiple Access (OFDMA) technique for uplink, where each device occupies a Resource Block (RB) with the allocation of power, bandwidth, and CPU frequency discretized with increments of non-integer values. Based on [Chen et al. 2021], the uplink

rate of device i transmitting model parameters to ES z in round t, can be formulated as $c^U_{z,i} = \sum_{k=1}^{R} r_{i,k} B^U \mathbb{E}(log_2\left(1 + \frac{P_{z,i}h_{z,i}}{I_k + B^U N_0}\right))$, where $r_{ijklm} = [r_{i1111}, ..., r_{iBRPF}]$ is an allocation vector with $r_{ijklm} \in [0, 1]$, R is the number of RBs, and B, P and F are vectors of discretized elements containing, respectively, the values for bandwidth, power and CPU frequency allocation. Furthermore, $\sum_{i=1}^{N_z} \sum_{j=1}^{|B|} \sum_{k=1}^{R} \sum_{l=1}^{|P|} \sum_{m=1}^{|F|} r_{ijklm} = 1$, with $\sum r_{ijklm} = 1$ indicating that the uplink rate of device i is $c^U_{z,i}$ using bandwidth $B^j_{z,i}$ on RB k with power $P^l_{z,i}$ and CPU frequency $F^m_{z,i}$. The channel gain between device i and the ES z is given by $h_{z,i} = o_{z,i} d^{-\alpha}_{z,i}$, where $d_{z,i}$ is the distance between device i and the ES z, $o_{z,i}$ is the Rayleigh fading parameter, and α is an exponent that affects how the channel gain varies with distance. $\mathbb{E}(\cdot)$ is the expected data rate with respect to $h_{z,i}$, N_0 is the noise power spectral density, and I_k is the interference to RB k caused by other devices.

Based on the related works discussed in Section 2, we assumed that FL models are transmitted via a single packet. Then, the communication latency and the device power consumption for transmission of local model from device i to ES z in round t can be computed by $\mathrm{Tcom}^U_{z,i} = \frac{Z}{c^U_{z,i}}$ and $\mathrm{Ecom}^U_{z,i} = P_{z,i}\mathrm{Tcom}^U_{z,i}$, where Z is the size of the uplink packet (in bits) that the devices need to transmit through the wireless link. The energy consumed by device i is given by $\mathrm{Eround}_{z,i} = \mathrm{Ecmp}_{z,i} + \mathrm{Ecom}^U_{z,i}$. The transmit power of the BS linking an ES is generally much greater than the power of the devices. Therefore, the entire downlink bandwidth can be used to transmit the global model. Thus, the downlink data rate is given by $c^D_{z,i} = B^D \mathbb{E}(log_2\left(1 + \frac{P_B h_{z,i}}{I^D + B^D N_0}\right))$, where B^D is the bandwidth used by the BS to transmit the global model to each device, P_B is the transmission power of the BS and I^D is the interference caused by other BSs.

It is assumed that the BS does not request retransmission of models from devices if they are received with errors. Therefore, whenever a packet containing a received local model has errors, the ES does not use it to update the global model. In this case, based on [Chen et al. 2021], the packet transmission error rate of the uplink from device i to ES z in round t is given by $q_{z,i} = \sum_{k=1}^{R} r_{i,k} \mathbb{E}(1 - \exp\left(-\frac{m(I_k + B^U N_0)}{P_{z,i} h_{z,i}}\right))$, where $\mathbb{E}(\cdot)$ is the expected packet error rate considering $h_{z,i}$ in RB k, with m being a threshold (*waterfall threshold*) that defines transmission quality. Thus, we can also formulate the transmission success probability of the local model from device i to ES z in round t as by $p_{z,i} = (1 - q_{z,i})$ `if` $(q_{z,i} \leq \gamma_Q)$ `else` 0, where γ_Q sets the minimum packet error rate for transmitting the local model to ES z. If transmission fails, the model is not received and does not contribute to global aggregation.

3.3.1. Model Aggregation and Transmission

After receiving models from local devices, the first-hop ES aggregates them to update the edge model. With no data training at any ES, the aggregation time is negligible. The updated model is then passed up the hierarchy until the root ES aggregates the global model. Let $B^U_{z,(l-1)}$ be the bandwidth allocated to ES z for transmitting its model to ES in the next higher level of the hierarchy in the uplink, where $l = 0$ denoting root ES. For simplicity, let us denote $p = l - 1$. In this way, the uplink rate from ES z in level l to ES in level p can be formulated by $c^U_{z,p} = B^U_{z,p} log_2\left(1 + \frac{P_{z,p} h_{z,p}}{I + B^U_{z,p} N_0}\right)$, where $B^U_{z,p}$ is the bandwidth, $P_{z,p}$ is the transmit power and $h^{z,p}_t$ is the channel gain of ES z to server

p at the next level of the hierarchy in round t. Using the formulation of $c^U_{z,p}$, we can find the downlink rate $c^D_{z,p}$ for transmitting the aggregated global model from server p to the lower level ES z. Then, the edge model upload latency and power consumption from ES z in level l to ES in level p at the next level of the hierarchy in the uplink in round t can be computed by $\mathrm{Tcom}^U_{z,p} = \frac{Z}{c^U_{z,p}}$ and $\mathrm{Ecom}^U_{z,p} = P_{z,p}\mathrm{Tcom}^U_{z,p}$, where Z is the size of the uplink packet (in bits) that the devices need to transmit through the wireless link. Based on the formulation of $\mathrm{Tcom}^U_{z,p}$ and $\mathrm{Ecom}^U_{z,p}$, we can find edge model upload latency $\mathrm{Tcom}^D_{z,p}$ and power consumption $\mathrm{Ecom}^D_{z,p}$ from ES in level p to ES z in level l.

3.3.2. HFL Communication Round Latency

The total HFL communication round latency consists of two parts: the time for downloading, training, and uploading the local model between devices and their first-hop ES. The latency of this step is determined by the slowest device within the coverage of ES z in round t, which is equal to $\mathrm{TmaxFirstHop} = \max_{z\in\mathcal{Z}_l}\{\max_{i\in\mathcal{N}_z}\{\mathrm{Tcom}^D_{z,i} + \mathrm{Tcmp}^U_{z,i} + \mathrm{Tcom}^U_{z,i}\}\}$, where Z_l is the set of ES at level l. The second part consists of the longest transmission delay from the intermediate ES at level l to the ES at level $p = (l-1)$ of the hierarchy, which is equal to $\mathrm{TmaxBridgeHop} = \max_{z\in\mathcal{Z}_l}\{\max_{q\in\mathcal{Q}_z}\{(\mathrm{Tcom}^D_{z,q} + \mathrm{Tcom}^U_{z,q})\}\}$, where Q_z is the set of ES at level p connected to intermediate ES z and $\mathrm{Tcom}^D_{z,q}$ and $\mathrm{Tcom}^U_{z,q}$ is, respectively, the communication latency between intermediate ES z at level l and ES q at level p in the downlink and uplink. As a result, the round latency in HFL can be computed by $\mathrm{Tround} = \mathrm{TmaxFirstHop} + \mathrm{TmaxBridgeHop}$. The ES energy demand for model aggregation is ignored due to its continuous power supply, and the intermediate ES aggregation delay is omitted as it is negligible.

4. Algorithm Design

4.1. Device Selection

By directing the strategy towards local training that covers more data, is expected to improve the representativeness of locally trained models. Therefore, by denoting S^z_t as a fraction f_t of N_z devices and b_z as a binary vector indicating the selection of n^z_p devices, the problem of maximizing the quantity of data from $n^z_p \leq |S^z_t|$ devices associated in each first-hop ES z can be expressed as

$$\max \sum_{i\in S^z_t} |\mathcal{P}_{z,i}| b_{z,i}, \quad (1) \qquad \text{s.t.} \sum_{i=1}^{|S^z_t|} b_{z,i} = n^z_p, \ (1a) \qquad b_{z,i} \in \{0,1\}, \qquad (1b)$$

where $|\mathcal{P}_{z,i}|$ is the number of data samples from device $i \in S_z$ of ES z and n^z_p is the partial number of devices selected for the resource scaling step of communication. The constraint (1a) guarantees that the number of selected devices is equal to n^z_p, and the constraint (1b) indicates that b_z is a binary vector, where $b_{z,i} = 1$ indicates that the device was selected from the subset S^z_t and $b_{z,i} = 0$ indicates otherwise.

4.2. Communication Resource Scheduling

This section presents the objective function of the HFLw$_{\mathrm{Opt}}$ algorithm for massive wireless IoT networks with a hierarchical topology. The function is formulated as an optimization problem to efficiently allocate computing and communication resources, aiming

to maximize successful transmissions, minimize device energy consumption, and reduce federated training latency through optimized resource allocation, as follows

$$\max \left(\sum p_{z,i} - \lambda \sum \text{Eround}_{z,i} \right) \times c_{z,ijklm}, \tag{2}$$

$$\text{s.t.} \quad \sum_{i=1}^{n_p^z} \left(\sum_{j=1}^{|B|} \sum_{k=1}^{R} \sum_{l=1}^{|P|} \sum_{m=1}^{|F|} c_{z,ijklm} \right) = 1, \tag{2a}$$

$$\sum_{k=1}^{R} \left(\sum_{j=1}^{|B|} \sum_{i=1}^{n_p^z} \sum_{l=1}^{|P|} \sum_{m=1}^{|F|} c_{z,ijklm} \right) = 1, \tag{2b}$$

$$\sum B_{z,i}^{j} \leq B_z^T, \tag{2c}$$

$$P_z^{min} \leq P_{z,i}^{l} \leq P_z^{max}, \tag{2d}$$

$$F_z^{min} \leq F_{z,i}^{m} \leq F_z^{max}, \tag{2e}$$

$$q_{z,i} \leq \gamma_Q, \tag{2f}$$

$$\sum c_{z,ijklm} \leq n_f^z, \tag{2g}$$

$$c_{z,ijklm} \in \{0, 1\}, \tag{2h}$$

where λ is a weight that controls the importance of the objective of minimizing the energy consumption of each device i, with $n_f^z \leq n_p^z$ being the maximum number of devices that should be selected for the next round of communication in each ES z. The constraint (2a) guarantees that each device is allocated a single bandwidth $B_{z,i}^{j}$, power $P_{z,i}^{l}$, CPU frequency $F_{z,i}^{m}$ and assigned to exactly one RB k, and the constraint (2b) guarantees that each RB is allocated to a single device. The constraint (2c) ensures that the total sum of $B_{z,i}^{j}$ does not exceed B_z^T, which is the total bandwidth budget. Constraints (2d) and (2e) set power and CPU frequency bounds. Constraint (2f) sets the packet error rate for model transmission. The constraint (2g) guarantees that the final number of selected devices is at most n_f^z, and the constraint (2h) defines that c_z is a binary matrix, where $c_{z,ijklm} = 1$ indicates that RB k in ES z has been allocated to device i with bandwidth $B_{z,i}^{j}$, power $P_{z,i}^{l}$ and CPU frequency $F_{z,i}^{m}$, while $c_{z,ijklm} = 0$ indicates otherwise.

For the effectiveness of large-scale HFL, a robust mechanism for adding and propagating models is crucial. The next section details the implementation of HFLw$_{\text{Opt}}$ algorithm, which integrates device selection and resource allocation in optimized federated training for massive wireless IoT networks.

4.3. HFLw$_{\text{Opt}}$ Algorithm

Algorithm 1 outlines the HFLw$_{\text{Opt}}$ strategy for wireless IoT networks, based on Equations (1) and (2), designed for scalable and efficient handling of n-level HFL aggregation. The algorithm initializes model w_0 and sets L aggregation levels. AC defines the number of aggregations performed at each level of the hierarchy. The model w_0 propagates through the hierarchy via wireless links. At level $L-1$, first-hop ESs optimize resources to maximize local model aggregation, minimizing delays and energy consumption. Devices

are selected using matrix c_z, considering bandwidth, CPU, power, and delay γ_T. Local models are trained, aggregated using data-weighted averages, and sent to level $L-2$ after AC_{L-1} iterations. Intermediate ESs (level $1 \leq l \leq L-2$) aggregate models from lower levels and transmit to higher levels. The root ES performs global aggregation, generating w_{t+1}^{global} after AC_0 iterations and distributing it back through the hierarchy.

Algorithm 1: HFLw$_{\text{Opt}}$ in Wireless IoT Networks

Initialization:
  Initialize w_0, L, $AC \leftarrow [c_0, c_1, ..., c_{(L-1)}]$
  DistributeGlobalModel(w_0)
for each global iteration AC_0 **in parallel do**
  Level $L-1$**:** ▷ *ES z in the first hop*
    for each first hop iteration AC_{L-1} **in parallel do**
      $S_t^z \leftarrow$ (random set of $max(f_t \cdot \mathcal{N}_z, 1)$ devices)
      Define b_z with $n_p^z \leq |S_t^z|$ ▷*Select n_p^z devices using Equation 1*
      Define c_z with $n_f^z \leq n_p^z$ ▷*Schedule resources to n_f^z devices using Equation 2*
      for each device $i \in S_t^z$ **in parallel do** ▷*Device Level*
        $w_{t+1}^{z,i} \leftarrow$ TrainingOnDevice(i, w_t^z)
      $w_{t+1}^z \leftarrow$ AggregationModels($w_{t+1}^{z,i}$)
    UploadLocalModels(w_{t+1}^z, $L-2$) ▷*Send models to ES at the top level*
  Level $L-2, ..., 1$**:** ▷ *ES in intermediate levels*
    for each intermediate level l **in parallel do**
      for each aggregation round AC_l **in parallel do**
        $w_{t+1}^{z,L_{l+1}} \leftarrow$ ModelsFromLevel($l+1$)
      $w_{t+1}^{z,L_l} \leftarrow$ AggregationModels($w_{t+1}^{z,L_{l+1}}$)
      UploadLocalModels(w_{t+1}^{z,L_l}, $l-1$) ▷*Send models to ES at the top level*
  Level 0**:** ▷ *Root ES*
    for each aggregation round AC_0 **in parallel do**
      $w_{t+1}^{z,L_1} \leftarrow$ ModelsFromLevel(1)
    $w_{t+1}^{global} \leftarrow$ AggregationModels(w_{t+1}^{z,L_1})
    DistributeGlobalModel(w_{t+1}^{global}) ▷*Global model is propagated back down*
Output:
  Return w_{global} after AC_0 global iteration

5. Configuration Parameters and Simulation Results

5.1. Network Topologies

The scenario in Figure 1a features a two-level aggregation hierarchy, where devices send local updates to a Middle ES for the first aggregation. Then, the Middle ES forwards the aggregated models to the Far ES for the final aggregation. In this case, consider a wireless IoT network with three distinct circular areas with a radius r of 625 meters with a BS in the center of each area. Each BS is directly associated with a FL ES aggregator, where there are $N_z = 120$ devices in each area distributed uniformly and randomly, totaling 360 devices in the total area. The distance between the devices and the first-hop Middle ES

is between 100 and 625 meters and each Middle ES is approximately $1,625$ meters from Far ES.

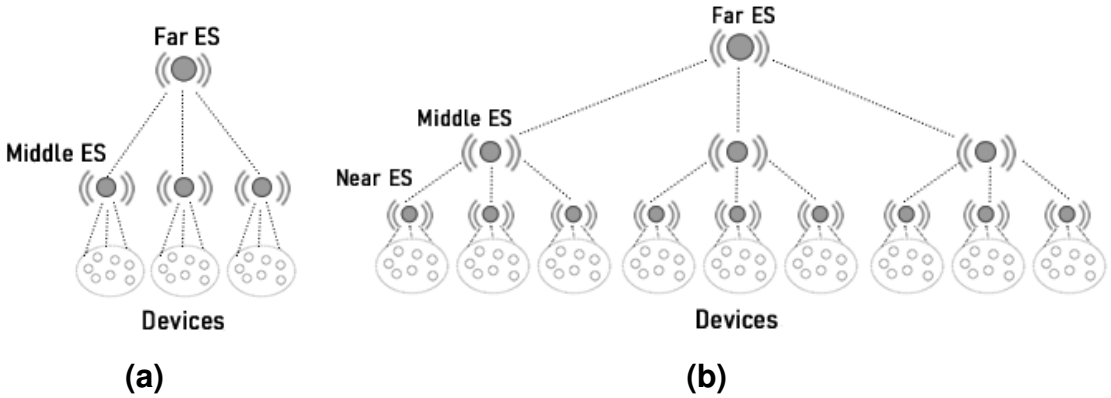

Figure 1. (a) HFL with the Middle ES, Far ES and the device level. (b) HFL with the NearES, Middle ES, Far ES and the device level.

The scenario in Figure 1b introduces a hierarchy with three levels of aggregation. A new server, called Near ES, is positioned closer to the devices in this topology. In this case, Near ES is responsible for the first level of aggregation, Middle ES is responsible for the second level of aggregation and Far ES, located at the top of the hierarchy, is responsible for the third and final level of aggregation. In this case, consider a wireless IoT network with nine distinct circular areas with a radius r of 625 meters with a BS in the center of each area. Each BS is directly associated with a FL ES aggregator, where there are $N_z = 120$ devices in each area distributed uniformly and randomly, totaling $1,080$ devices in the total area. The distance between the devices and the Near ES in the first-hop is between 50 and 250 meters. Each set of three Near ES is associated with a Middle ES. The distance from each Near ES to Middle ES and from each Middle ES to Far ES is approximately 875 meters.

5.2. Network Parameters

The uplink bandwidth $B_{z,i}$ of each RB in ES z is limited by discretized values generated by an arithmetic progression in the range of $[1, 2]$ MHz with increments and median values of 0.25 MHz and 1.5 MHz, respectively. Similarly, the transmission power $P_{z,i}$ of the devices is limited by the range of $[5, 10]$ mW with increments and median values of 1 mW and 7.5 mW and the CPU frequency $F_{z,i}$ of the devices is limited by the range of $[0.6, 1]$ GHz with increments and median values of 0.1 GHz and 0.8 GHz. The total bandwidth budget B_z^T for the uplink of each ES z is equivalent to the value of $n_f^z \times 1.5$ MHz, where n_f^z defines the maximum number of devices participating in each round of communication. Furthermore, each RB includes the attribution of a distinct and incremental interference. The downlink bandwidth is 20 MHz, and the BS transmit power is 1 W. In each communication round t, the $h_{z,i}$ modeling incorporates a fading effect that indicates that the channel gain decreases as the distance of the devices i from the ES z increases. According to [Chen et al. 2021], other parameters include $\alpha = 2$, $N_0 = -174$ dBm/Hz, $m = 0.023$ dB, $\omega_{z,i} = 40$ and $\zeta_{z,i} = 10^{-27}$.

Communication between devices and their respective first-hop ES is higher than between intermediate ESs. In our experiments, the aggregation frequencies are controlled

by the parameters. For example, in the topology of Figure 1a, 5 Middle ES aggregations trigger 1 Far ES aggregation and in the topology of Figure 1b, 5 Near ES aggregations trigger 1 Middle ES aggregation, and 2 Middle ES aggregations trigger 1 Far ES aggregation. In the communication between the intermediate ESs, the uplink and downlink bandwidth is 20 MHz and the BS transmission power is 1 W. For the assignment of uplink RBs, we use a heuristic that prioritizes the intermediate ESs that are farthest from the higher-level ES, allocating these ESs to the uplink channels with the highest SINR value.

5.3. Datasets and DNNs Architectures

The FL tasks in this work consider image classification problems using benchmark datasets used in FL research, called MNIST and Fashion-MNIST, according to the works of [McMahan et al. 2016], [Chen et al. 2021], [Simos et al. 2022], [Su et al. 2024], [Liu et al. 2020] and [Luo et al. 2020]. Unlike the cited references, this work introduces heterogeneity in device data as a core feature of wireless IoT networks, using non-Independent and Identically Distributed (non-IID) variations of MNIST and Fashion-MNIST to challenge local model accuracy and global model aggregation optimization. MNIST and Fashion-MNIST datasets were split into 10 subsets with samples from the same label, using 75% of the samples for the training set and 25% for the test set. Then, each device received a training and testing partition, where 90% of the samples belong to the same label and the remaining 10% belong equally to the other labels; each image is rotated up to 15° clockwise or counterclockwise; the final amount of data is given by a factor between $[0.25, 1]$ of the initial partition. The MNIST and Fashion-MNIST partitions used in this work are referred to as NIID R-MNIST and NIID R-FMNIST, respectively.

To evaluate HFLw$_{\mathrm{Opt}}$, we use two Deep Neural Networks (DNNs) architectures. For NIID R-MNIST, an MLP with 192 neurons, ReLU activation, and a softmax output layer was used. For NIID R-FMNIST, a CNN with three convolutional layers with 64 filters each, followed by ReLU activations, MaxPooling, and a dense layer with 192 neurons was used. The MLP and CNN architectures have $152,650$ and $187,210$ parameters, respectively, and both utilize the ADAM optimizer and the Sparse Categorical Cross-Entropy loss function.

5.4. Simulation Results

HFLw$_{\mathrm{Opt}}$ is evaluated against three baselines. H-FedAvg$_{\mathrm{SINR}}$ is based on the FedAvg algorithm [McMahan et al. 2016], which randomly selects devices and allocates communication resources, incorporating SINR-based uplink resource assignment. H-POC$_{\mathrm{SINR}}$ is derived from the POC algorithm [Cho et al. 2020], selecting devices with the highest local loss and also utilizing SINR-based uplink assignment. The third algorithm, HFLw$^{\mathrm{Fix}}_{\mathrm{Opt}}$, is a variation of HFLw$_{\mathrm{Opt}}$ that uses fixed values for uplink bandwidth, power, and CPU frequency. All these algorithms are implemented in an HFL tree topology using the same fixed parameters: 1 MHz uplink bandwidth, 10 mW transmit power, and 1 GHz CPU frequency.

Figures 2 and 3 depicts the evolution of accuracy and $f(w_{global})$ for HFL algorithms across different hierarchical aggregation topologies, analyzed as a function of successful transmissions and energy cost. The experimental setup assumes a deployment area of up to $2,000$ meters for distributing devices and ESs, where each first-hop ES has

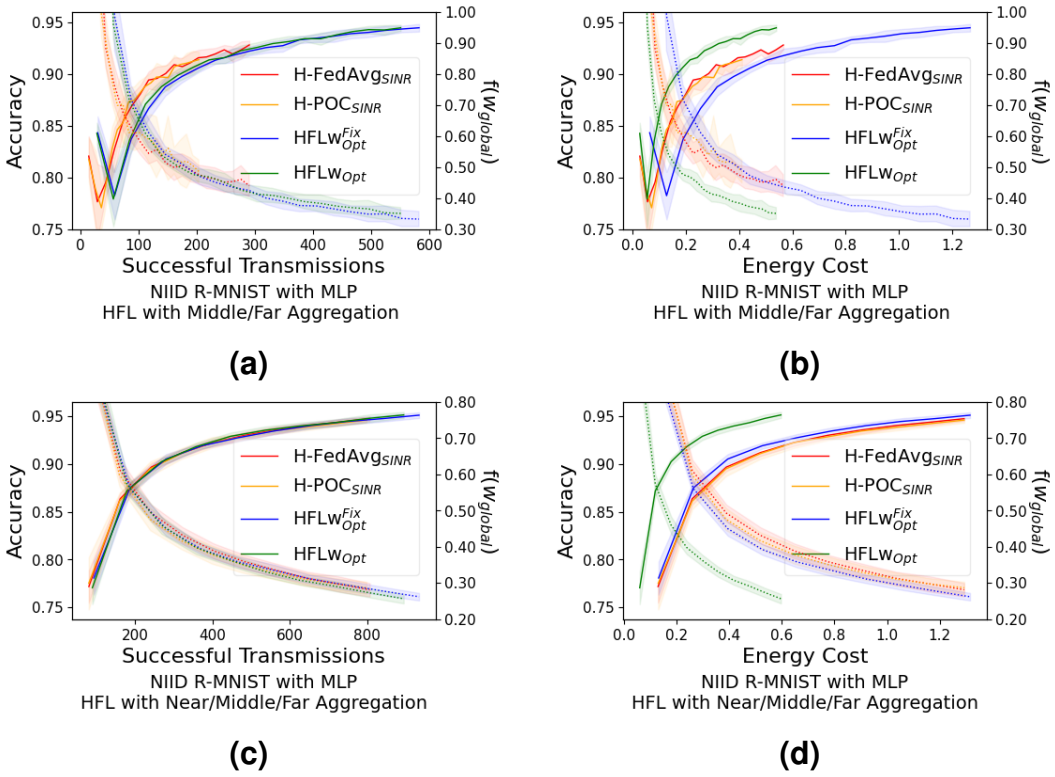

Figure 2. Evolution of accuracy and $f(w_{global})$ for HFL algorithms using the NIID R-MNIST dataset and MLP architecture, evaluated by successful transmissions and energy cost across hierarchical aggregation topologies.

120 devices. Scenarios with Middle/Far aggregation represent the topology in Figure 1a with 3 Middle ESs, 1 Far ES, and 360 devices. Similarly, scenarios with Near/Middle/Far aggregation follow the topology in Figure 1b with 9 Near ESs, 3 Middle ESs, 1 Far ES, and $1,080$ devices. The results are derived from scenarios involving 100 communication rounds at each first-hop ES z, with a maximum of $n_f^z = 10$ devices selected per communication round with the requirement of the packet transmission error rate $\gamma_Q = 0.3$. Solid curves show average performance, and shaded regions represent standard deviation across 15 executions.

The algorithms based on HFLw_{Opt} achieve high accuracy in Figures 2a and 3a, supported by the substantial number of successful transmissions. Conversely, in Figures 2c and 3c, the inclusion of Near ESs close to the devices favors SINR-based algorithms, which prioritize devices with favorable channel conditions, resulting in a high number of successful transmissions. However, this approach fails to address the optimization of global energy consumption effectively by using a fixed allocation of resources and, therefore, disregarding critical factors for energy efficiency, such as the joint optimization of bandwidth, power, and CPU frequency. This limitation is evidenced by Figures 2c and 3c, in which the SINR algorithms present significantly higher energy costs to achieve similar accuracies. Figures 2b, 2d, 3b, and 3d highlight the energy efficiency advantage of HFLw_{Opt} due to its formulation based on the Algorithm 1 that efficiently integrates resource allocation and energy consumption. By carefully balancing bandwidth, power, and CPU frequency, the HFLw_{Opt} algorithm significantly reduces energy consumption

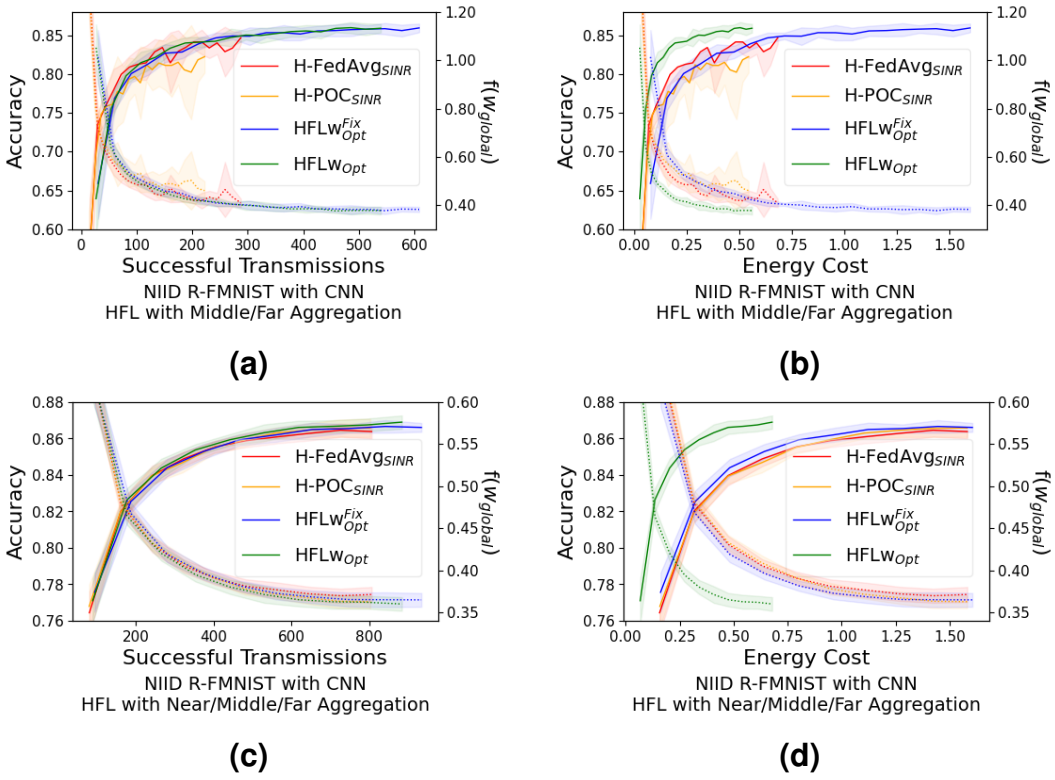

Figure 3. Evolution of accuracy and $f(w_{global})$ for HFL algorithms using the NIID R-FMNIST dataset and CNN architecture, evaluated by successful transmissions and energy cost across hierarchical aggregation topologies.

while maintaining high-quality global models, even in complex scenarios with varying hierarchical topologies.

Figure 4 presents the number of successful transmissions, energy cost, and training latency for HFL algorithms, emphasizing the influence of resource allocation and communication distances on overall performance. In Figures 4a and 4c, which analyze a topology with two levels of aggregation, HFLw_{Opt} has superior performance based on fixed resource allocation algorithms, achieving a higher number of successful transmissions with lower training latency and reduced energy cost. In simpler hierarchical configurations, such as Middle/Far, based on fixed resource allocation algorithms struggle due to their formulation and longer communication distances, leading to higher latencies and increased transmission failures. In contrast, HFLw_{Opt} minimizes these failures through dynamic resource optimization, enabling lower training latencies and superior energy efficiency.

Figures 4b and 4d examine the performance of HFL algorithms in denser hierarchical topologies with three levels of aggregation, highlighting the scalability of the HFL approach. The introduction of intermediate ESs significantly reduces communication distances and improves SINR, leading to fewer transmission failures and better overall performance. While algorithms like $\text{H-FedAvg}_{\text{SINR}}$, $\text{H-POC}_{\text{SINR}}$, and $\text{HFLw}_{\text{Opt}}^{\text{Fix}}$ show some performance improvements due to the shorter communication distances, they remain limited by their static resource allocation strategies. This constraint hampers their scalability, resulting in higher energy consumption and increased training latency as the

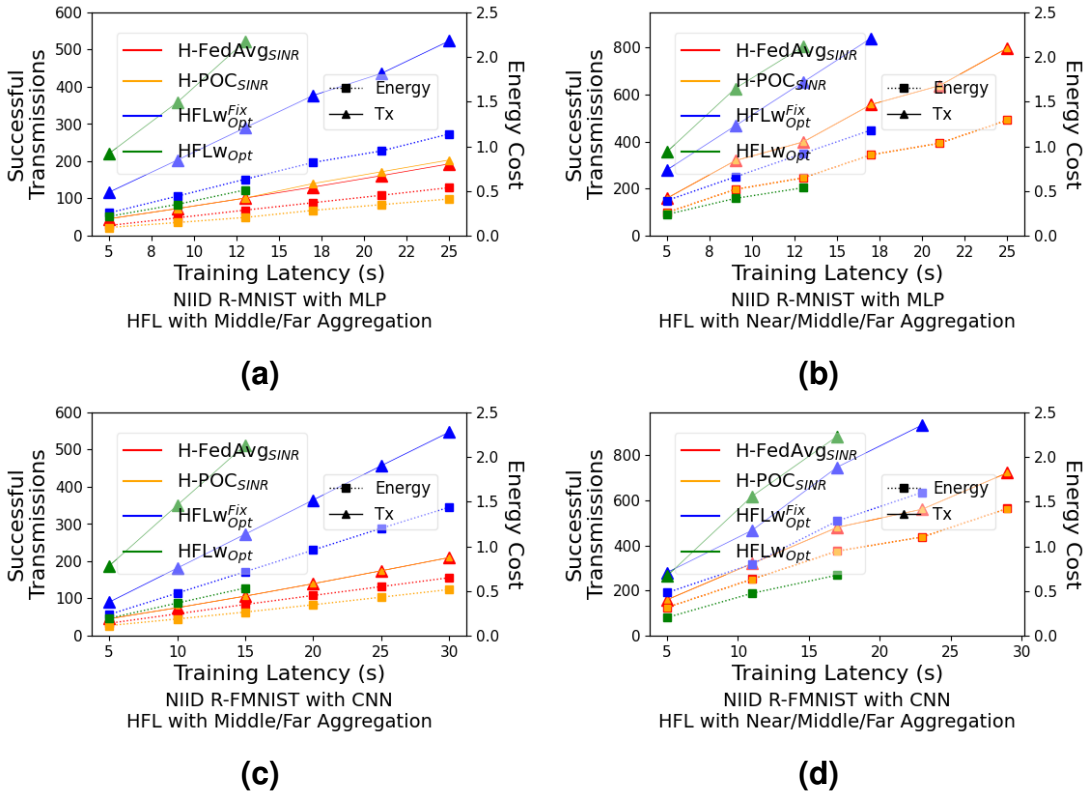

Figure 4. Successful transmissions and energy cost as a function of training latency for HFL algorithms across hierarchical aggregation topologies.

network complexity and device density grow. In contrast, HFLw$_{\mathrm{Opt}}$ demonstrates superior adaptability to hierarchical configurations, leveraging dynamic resource allocation to effectively balance energy consumption, training latency and successful transmissions. These results establish HFLw$_{\mathrm{Opt}}$ as an effective solution for massive IoT networks, offering consistent and sustainable performance.

6. Conclusion and Future Work

This work addresses device selection and communication resource allocation in wireless IoT networks for HFL tasks, proposing the HFLw$_{\mathrm{Opt}}$ algorithm. Formulated as an optimization problem, it maximizes successful transmissions, minimizes latency, and reduces energy consumption. Its hierarchical structure reduces transmission failures and supports scalability, enabling the integration of more devices without compromising efficiency. HFLw$_{\mathrm{Opt}}$ is robust in high-density scenarios and large-scale wireless IoT networks. The HFLw$_{\mathrm{Opt}}$ source code is available to allow reproduction and validation of the results. Future work will explore alternative network topologies and optimization techniques that consider device mobility.

References

Chen, M., Yang, Z., Saad, W., Yin, C., Poor, H. V., and Cui, S. (2021). A Joint Learning and Communications Framework for Federated Learning Over Wireless Networks. *IEEE Transactions on Wireless Communications*, 20(1):269–283.

Cho, Y. J., Wang, J., and Joshi, G. (2020). Client Selection in Federated Learning: Convergence Analysis and Power-of-Choice selection strategies. *CoRR*, abs/2010.01243.

ETSI (2022). Multi-access Edge Computing (MEC); Framework and Reference Architecture, ETSI GS MEC 003 V3.1.1.

Liu, L., Zhang, J., Song, S., and Letaief, K. B. (2020). Client-Edge-Cloud Hierarchical Federated Learning. In *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pages 1–6.

Luo, S., Chen, X., Wu, Q., Zhou, Z., and Yu, S. (2020). HFEL: Joint Edge Association and Resource Allocation for Cost-Efficient Hierarchical Federated Edge Learning. *IEEE Transactions on Wireless Communications*, 19(10):6535–6548.

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. (2016). Federated Learning of Deep Networks using Model Averaging. *CoRR*, abs/1602.05629.

Nakayama, K. and Jeno, G. (2022). *Federated Learning with Python: Design and implement a federated learning system and develop applications using existing frameworks*. Packt Publishing.

Poularakis, K., Llorca, J., Tulino, A. M., Taylor, I., and Tassiulas, L. (2020). Service Placement and Request Routing in MEC Networks with Storage, Computation, and Communication Constraints. *IEEE/ACM Transactions on Networking*, 28(3):1047–1060.

Simos, M., Bouzinis, P. S., Diamantoulakis, P. D., Sarigiannidis, P., and Karagiannidis, G. K. (2022). Hierarchical Federated Learning for the Next Generation IoT. In *2022 18th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 198–203.

Stergiou, K. D. and Psannis, K. E. (2022). Federated Learning Approach Decouples Clients From Training a Local Model and With the Communication With the Server. *IEEE Transactions on Network and Service Management*, 19(4):4213–4218.

Su, L., Zhou, R., Wang, N., Chen, J., and Li, Z. (2024). Low-Latency Hierarchical Federated Learning in Wireless Edge Networks. *IEEE Internet of Things Journal*, 11(4):6943–6960.

Wu, J., Dong, F., Leung, H., Zhu, Z., Zhou, J., and Drew, S. (2024). Topology-aware Federated Learning in Edge Computing: A Comprehensive Survey. *ACM Comput. Surv.*, 56(10).

Xu, B., Xia, W., Wen, W., Liu, P., Zhao, H., and Zhu, H. (2022). Adaptive Hierarchical Federated Learning Over Wireless Networks. *IEEE Transactions on Vehicular Technology*, 71(2):2070–2083.

Yang, Z., Chen, M., Wong, K.-K., Poor, H. V., and Cui, S. (2022). Federated Learning for 6G: Applications, Challenges, and Opportunities. *Engineering*, 8:33–41.

You, C., Guo, K., Yang, H. H., and Quek, T. Q. S. (2023). Hierarchical Personalized Federated Learning Over Massive Mobile Edge Computing Networks. *IEEE Transactions on Wireless Communications*, 22(11):8141–8157.