FEDESO: EXPLORATION-TO-STABILIZATION OPTIMIZATION FOR MODEL-HETEROGENEOUS FEDERATED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Model-heterogeneous federated learning (MH-FL) enables resource-constrained clients to train and deploy submodels of different capacities extracted from a shared global model. Achieving strong submodel performance requires identifying promising parameter subsets while consistently optimizing their parameters. Existing methods either vary submodels to improve parameter coverage or adapt their selection using parameter magnitude, which provides only an indirect measure of how a component contributes to the task objective. We instead consider optimization-aware selection using task-dependent first-order importance signals. However, physical submodel execution introduces a fundamental *partial-feedback problem*: excluded components cannot receive their own first-order importance feedback and thus remain insufficiently evaluated during submodel exploration. We propose FedESO, an exploration-to-stabilization framework that learns a shared importance ranking and uses scheduled full importance feedback to directly evaluate both selected and excluded components while controlling the additional resource overhead. We further formulate dynamic MH-FL as a time-varying optimization problem and establish a nonconvex convergence guarantee governed by the cumulative variation of the induced submodel objectives. Sublinear cumulative variation is sufficient for asymptotic stationarity, while finite-time mask stabilization recovers stationary convergence for the resulting fixed aggregate submodel objective. Extensive experiments across vision and natural language processing tasks under diverse resource constraints show that FedESO consistently outperforms state-of-the-art approaches, with gains of at least 9 and 2 percentage points in training-from-scratch and fine-tuning settings, respectively.

1 INTRODUCTION

Federated learning (FL) enables distributed clients to collaboratively train models without revealing their private data (McMahan et al., 2017). In cross-device settings, however, mobile clients often have severely limited and heterogeneous computation and memory resources, making the training of a single large model infeasible. Model-heterogeneous federated learning (MH-FL) addresses this challenge by extracting resource-compatible submodels from a shared global model and assigning them to devices with different capacities. Importantly, these heterogeneous submodels are not merely computational surrogates for training: they are the models that resource-constrained clients can ultimately execute, e.g., for inference. We therefore consider MH-FL from a deployment-oriented perspective, where the objective is to optimize the aggregate performance of resource-compatible client submodels rather than solely that of the full server model.

Optimizing this objective requires jointly determining promising resource-compatible submodels and training their parameters, creating an inherent tension between exploration and consistent optimization. Static extraction methods repeatedly train predetermined nested submodels, providing a consistent optimization structure but offering limited opportunity to explore alternative parameter subsets (Diao et al., 2020; Horvath et al., 2021). Dynamic methods (e.g., FedRolex (Alam et al., 2022)) vary the extracted submodels across communication rounds to improve parameter coverage, but persistent

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

mask changes induce a sequence of mask-dependent objectives rather than consistently optimizing particular submodels. Importance-aware extraction provides an alternative by adapting submodel selection according to information acquired during training. FIARSE (Wu et al., 2024), for example, selects parameters according to their magnitudes. Parameter magnitude, however, is an indirect proxy for a component's utility: it does not explicitly characterize the sensitivity of the task objective to that component's participation.

A natural alternative is to learn submodel importance directly from task-dependent first-order signals (Sanh et al., 2020). Yet applying such optimization-aware selection to resource-constrained MH-FL exposes a fundamental challenge. Because each client physically instantiates only its resource-compatible submodel, first-order importance feedback is directly available only for selected components. Components excluded by the current mask do not participate in local computation and therefore cannot receive their own task-dependent importance signals. This *partial-feedback problem* couples submodel selection with the information available to improve that selection: the current submodel determines which components can be evaluated, potentially reinforcing existing selections without sufficiently exploring excluded alternatives. Effective optimization-aware MH-FL must therefore provide informative feedback for exploring candidate components while preserving the resource compatibility that necessitates submodel training in the first place.

We propose *FedESO*, a federated Exploration-to-Stabilization Optimization framework that addresses this challenge through scheduled full importance feedback. FedESO learns a globally shared importance ranking over structured model components from task-dependent first-order signals and extracts resource-compatible submodels according to client capacities. During scheduled full-feedback rounds, all candidate components, including those excluded from the current submodel, receive direct importance feedback, allowing the learned ranking to reconsider earlier selections. FedESO therefore explores by expanding the coverage of optimization feedback rather than by deliberately perturbing or rotating submodel masks. As importance learning progresses, increasingly consistent rankings can naturally yield stable submodel extraction and consistent optimization. Because full importance feedback incurs additional communication, computation, and memory costs, FedESO schedules these operations with geometrically diminishing frequency, making them increasingly sparse over training.

We further characterize the resulting dynamic submodel optimization theoretically. Let the masks at round t induce the aggregate submodel objective $F_t(w)$. Rather than analyzing convergence toward the full-model objective and treating heterogeneous submodels only through approximation errors (Zhou et al., 2023; Wu et al., 2024), we directly study this time-varying aggregate submodel objective. Our analysis characterizes convergence through the cumulative variation of $\{F_t\}$: sublinear cumulative objective variation is sufficient for asymptotic stationarity, without requiring the masks to become fixed in finite time. When the learned masks do stabilize, the result further specializes to stationary convergence for the resulting fixed aggregate submodel objective. The analysis therefore directly characterizes the resource-compatible client submodels that constitute the optimization and deployment targets of MH-FL.

Our contributions are summarized as follows. **Optimization-aware MH-FL with scheduled full feedback.** We introduce FedESO, which federatively learns structured submodels from task-dependent first-order importance signals. We identify the partial-feedback problem induced by physical submodel execution and introduce scheduled full importance feedback to directly evaluate both selected and excluded candidate components while controlling its additional resource overhead. **Dynamic-submodel convergence analysis.** We formulate dynamic MH-FL as time-varying optimization and establish a nonconvex convergence guarantee characterized by cumulative mask-induced objective variation. Sublinear cumulative variation is sufficient for asymptotic stationarity, while finite-time mask stabilization recovers stationary convergence for the resulting fixed aggregate client-submodel objective. **Empirical validation.** We conduct extensive experiments across vision and natural language processing tasks under diverse resource constraints. FedESO consistently outperforms state-of-the-art approaches, with gains of at least 9 and 2 percentage points in training-from-scratch and fine-tuning settings, respectively, while our ablations demonstrate the importance of sufficiently informative importance feedback for discovering high-performing resource-compatible submodels.

2 BACKGROUND AND MOTIVATION

2.1 PROBLEM FORMULATION OF MH-FL

We consider an MH-FL setting with N clients, where client i has a private dataset D_i and capacity $r_i \in (0, 1]$, representing the maximum fraction of the shared global model $w \in \mathbb{R}^d$ that it can train and deploy (Alam et al., 2022; Zhou et al., 2023; Wu et al., 2024). A binary mask $M^{(i)} \in \{0, 1\}^d$ extracts the client submodel $w \odot M^{(i)}$, subject to $\|M^{(i)}\|_1 \leq r_i d$. The MH-FL objective is

$$\min_{M,w} F(w, M) = \frac{1}{N} \sum_{i=1}^{N} \left(F_i(w \odot M^{(i)}; D_i) := \mathbb{E}_{b_i \sim D_i} \left[\ell(w \odot M^{(i)}; b_i) \right] \right). \tag{1}$$

where $M = \{M^{(i)}\}_{i \in [N]}$ and $\odot$ denotes the Hadamard product. The training goal is to jointly optimize the shared model parameters w and client masks M for the aggregate performance of the resource-compatible submodels, rather than the full global model itself. We use equally weighted client objectives for simplicity; the formulation extends directly to weighted aggregation.

2.2 MOTIVATION

The MH-FL objective in Eq. 1 reveals an inherent tension between exploring promising resource-compatible submodels and consistently optimizing them. Let $M_t = \{M_t^{(i)}\}_{i \in [N]}$ denote the masks used at communication round t, and define the corresponding objective as $F_t(w) := F(w, M_t)$. Static extraction keeps M_t fixed, providing a consistent optimization objective but limiting the exploration of alternative submodels (Diao et al., 2020; Horvath et al., 2021). Dynamic methods instead vary M_t across rounds to improve parameter coverage (Alam et al., 2022). Such persistent mask changes induce a sequence of objectives F_t rather than a fixed submodel objective. Moreover, because different submodels share parameters from the same global model w, a component w_j retained across consecutive masks can receive different optimization signals, $\nabla_{w_j} F_i(w \odot M_t^{(i)}) \neq \nabla_{w_j} F_i(w \odot M_{t+1}^{(i)})$, as its surrounding active components change. Persistent mask-induced objective variation can therefore interfere with consistent optimization of promising submodels (Xu et al., 2022). Effective submodel extraction should thus provide sufficient opportunity to explore alternative components while avoiding persistent variation that prevents promising submodels from being consistently optimized.

Importance-aware extraction provides a natural way to make submodel selection depend on information acquired during optimization. FIARSE (Wu et al., 2024), for example, ranks model parameters according to their magnitudes. While efficient, parameter magnitude $\|w_j\|$ is an indirect proxy for the utility of component j: it does not explicitly measure how its participation affects the task objective in Eq. 1. Introducing a component-level selection variable m_j, in contrast, yields the first-order signal

$$\frac{\partial F_i}{\partial m_j} = \left\langle \frac{\partial F_i}{\partial (w_j m_j)}, w_j \right\rangle, \tag{2}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product. It directly characterizes the local sensitivity of the task objective to the participation of component j (Sanh et al., 2020). Such task-dependent first-order information therefore provides a principled signal for optimization-aware submodel selection.

Obtaining sufficiently complete first-order importance information, however, is fundamentally challenging under the physical submodel execution required by resource-constrained MH-FL. If $M_{t,j}^{(i)} = 0$, component w_j is absent from client i's local computation, and its task-dependent participation sensitivity cannot be directly evaluated from the instantiated submodel. Importance learning based only on the current submodel therefore provides *partial feedback*: selected components receive direct optimization signals, whereas excluded candidates lack corresponding evidence about their potential utility. This creates a fundamental exploration challenge—how to discover promising resource-compatible submodels when the resource constraints that necessitate submodel training simultaneously restrict the optimization feedback needed to evaluate alternative components—while ultimately maintaining sufficiently consistent submodels for effective optimization.

3 FEDESO: EXPLORATION-TO-STABILIZATION OPTIMIZATION

Overview. We propose *FedESO*, an exploration-to-stabilization framework for learning resource-compatible submodels in MH-FL. FedESO jointly learns model parameters and explicit importance scores from task-dependent first-order feedback to guide structured TopK submodel extraction. To overcome the partial feedback from physical submodel execution, scheduled full importance feedback periodically evaluates all candidate components, enabling effective exploration beyond the current submodel. As optimization progresses, increasingly consistent importance rankings naturally promote stable submodel extraction, while the diminishing full-feedback frequency controls the additional resource overhead.

3.1 OPTIMIZATION-AWARE SUBMODEL EXTRACTION

We introduce explicit importance scores $s \in \mathbb{R}^{d'}$ to learn task-dependent preferences over structured components of the global model $w \in \mathbb{R}^d$. Each coordinate of s corresponds to a structured model component comprising multiple coordinates in w, such as a group of hidden dimensions or channels. Related learnable importance scores have been studied in optimization-aware pruning, e.g., movement pruning (Sanh et al., 2020). We employ them for resource-compatible submodel extraction in MH-FL. For client i with resource constraint $r_i \in [0, 1]$, we construct a binary mask $m^{(i)} = \mathrm{TopK}_{r_i}(s)$, which selects the highest-scoring r_i fraction of components, and extract the corresponding submodel $w^{(i)} = w \odot m^{(i)}$, where dimension expansion of $m^{(i)}$ is omitted for simplicity. Similar discrete functions other than TopK are also applicable.

3.2 FEDERATED LEARNING OF MODEL PARAMETERS AND IMPORTANCE SCORES

For MH-FL, we extend the classic Federated Averaging (FedAvg) framework (McMahan et al., 2017) by integrating submodel extraction and local alternating optimizations for submodel weights and importance scores. A server initializes and maintains the global model weights and the importance scores. At the beginning of each communication round t, the server samples a subset of clients $\mathcal{A}_t$. Given the global importance scores s_t, the server determines the resource-compatible mask for client i as $m_t^{(i)} = \mathrm{TopK}_{r_i}(s_t)$, and physically extracts the corresponding submodel $w_t^{(i)} = w_t[m_t^{(i)}]$, where $[\cdot]$ denotes physical extraction of the selected model components rather than dense masking. The server sends the extracted submodel $w_t^{(i)}$, its mask $m_t^{(i)}$, and the full importance-score vector s_t to each client in $\mathcal{A}_t$. Since the importance scores are defined over model components and incur substantially lower memory and communication costs than the corresponding model parameters, the complete score vector can be maintained locally even though the full model cannot.

FedESO alternates between local submodel optimization and importance-score optimization, with server aggregation after each phase:

$$\text{Client weight update}: \quad w_{t,k+1}^{(i)} \leftarrow w_{t,k}^{(i)} - \eta_w \nabla_w F_i(w_{t,k}^{(i)}; m_t^{(i)}), k \in [K_w], \tag{3}$$

$$\text{Server weight aggregation}: \quad w_{t+1} \leftarrow w_t + \mathrm{Avg}_{i\in\mathcal{A}_t}\left(w_{t,K_w+1}^{(i)} - w_t^{(i)}\right), \tag{4}$$

$$\text{Client score update}: (s_{t+1}^{(i)})_j \leftarrow (s_t^{(i)})_j - \eta_s \widetilde{\nabla}_{s_j} F_i\Big((w_{t+1}^{(i)})_j \odot m_j\Big), j \in \mathrm{supp}(m_t^{(i)}), \tag{5}$$

$$\text{Server score aggregation}: \quad s_{t+1} \leftarrow s_t + \mathrm{Avg}_{i\in\mathcal{A}_t}\left(s_{t+1}^{(i)} - s_t^{(i)}\right). \tag{6}$$

In the weight-update phase, each client i conducts K_w descent steps, as Eq. 3, with initial weights $w_{t,1}^{(i)} := w_t^{(i)}$ and learning rate η_w. The computation and memory costs are resource-compatible as $O(r_i d)$. Then, the server updates the global weights, as Eq. 4, using a mean function Avg, after which it extracts and transmits the updated submodel $w_{t+1}^{(i)}$ to each client in $\mathcal{A}_t$. In the subsequent score-update phase, each client i conducts one descent step as Eq. 5. Since the gradient of TopK is zero almost everywhere, we adopt the straight-through estimator (STE) (Bengio et al., 2013) to compute an approximate value. W.r.t. Eq. 5, for an instantiated component $j \in \mathrm{supp}(m_t^{(i)})$ and by denoting $m_j := (\mathrm{TopK}_{r_i}(s))_j$, the STE-based surrogate score gradient $\widetilde{\nabla}_{s_j} F_i$ can be written as

$$\widetilde{\nabla}_{s_j} F_i(\underbrace{(w_{t+1}^{(i)})_j \odot m_j}_{w'_j}) = \frac{\partial F_i}{\partial m_j} = \left\langle \frac{\partial F_i}{\partial w'_j}, (w_{t+1}^{(i)})_j \right\rangle, \tag{7}$$

where the inner product is used as $(w_{t+1}^{(i)})_j$ may represent a component with multiple coordinates. This signal captures the local first-order sensitivity of the task objective to changing the participation of component j. Since only the resource-compatible submodel participates in local computation, the computation and memory costs remain $O(r_i d)$. The server subsequently aggregates the locally updated importance scores according to Eq. 6 to complete the communication round.

The resource-constrained importance-score update in Eqs. 5 and 7 exposes a *partial feedback problem*. Following the full-model importance-learning mechanism of movement pruning (Sanh et al., 2020), obtaining task-dependent first-order feedback for all score coordinates $j \in [d']$ would require the corresponding model components to participate in local computation. In resource-constrained MH-FL, however, client i physically instantiates only $w_{t+1}^{(i)} \subset w_{t+1}$. Hence, although the complete score vector is available locally, the model component associated with s_j is absent whenever $j \notin \mathrm{supp}(m_t^{(i)})$, and its corresponding task-dependent first-order feedback cannot be directly evaluated by client i. We note that an unselected component may still enter a future TopK set as the scores of currently selected components change, and it may also receive feedback from other clients whose resource-compatible submodels include it. However, while component j remains unselected by client i, that client cannot acquire new task-dependent evidence about the component.

3.3 Scheduled Full Importance Feedback

To address the partial-feedback problem discussed above, we introduce *full importance feedback*, which periodically evaluates all candidate components when updating their importance scores. During a full-feedback round, the server provides the full model for importance evaluation, allowing every component $j \in [d']$, including those excluded from the current TopK submodel, to receive the task-dependent first-order signal in Eq. 7. More frequent full feedback therefore provides broader coverage of candidate components and enables the learned importance ranking to more effectively reconsider earlier TopK selections as model parameters evolve. In the limiting case, full feedback at every round provides the most complete importance information, but incurs full-model communication, computation, and memory overhead throughout training. FedESO instead schedules full-feedback rounds with diminishing frequency to control this exploration–resource tradeoff.

Geometrically diminishing feedback frequency. We schedule full-feedback rounds with progressively increasing intervals. Specifically, let $T_0 > 0$ determine the first full-feedback round and $a > 1$ control the rate at which the feedback frequency decreases. The full-feedback rounds t_k are:

$$t_0 := \lfloor T_0 \rfloor, \quad \text{and} \quad t_{k+1} := \max\left\{t_k + 1, \left\lfloor T_0 a^{k+1} \right\rfloor\right\}, \qquad k \in \{0, 1, \ldots\}. \tag{8}$$

Denote their collection as $\mathcal{T}_{\mathrm{F}}(T) := \{t_k \mid 0 \le t_k < T\}$, given T communication rounds. The term $t_k + 1$ in Eq. 8 guarantees strictly increasing feedback rounds despite integer discretization. Full feedback is therefore relatively frequent during early training, while the intervals between successive feedback rounds progressively increase. At round t, the set of components receiving task-dependent importance feedback on client i is

$$\mathcal{J}_t^{(i)} := \begin{cases} [d'], & t \in \mathcal{T}_{\mathrm{F}}(T), \\ \mathrm{supp}\left(m_t^{(i)}\right), & t \notin \mathcal{T}_{\mathrm{F}}(T). \end{cases} \tag{9}$$

Accordingly, the importance-score update in Eq. 5 is evaluated for $j \in \mathcal{J}_t^{(i)}$. At regular rounds, importance learning remains restricted to the physically instantiated resource-compatible submodel. At full-feedback rounds, all components receive direct first-order importance feedback. The full model is used only for importance evaluation; regular model-parameter optimization remains restricted to the client's resource-compatible submodel.

Optimization-aware exploration and stabilization. FedESO explores candidate submodels through the information available to its learned importance ranking rather than by explicitly perturbing or varying the mask. Full feedback provides first-order evidence for both selected and excluded components, allowing TopK extraction to change when the learned importance of an alternative component becomes sufficiently high. As optimization progresses, the relative score ranking can become increasingly consistent, naturally reducing changes in the TopK support and leading to stable submodel extraction. Thus, exploration and stabilization are compatible: broad importance feedback enables informed exploration, while stabilization can emerge from the convergence of the learned ranking rather than from explicitly suppressing mask variation.

Resource control. The diminishing schedule controls the additional cost of obtaining full importance feedback. For fixed $T_0 > 0$ and $a > 1$, the geometric term eventually dominates the $t_k + 1$ correction, yielding

$$|\mathcal{T}_{\mathrm{F}}(T)| = \mathcal{O}\left(\log_a \frac{T}{T_0}\right), \qquad \text{and} \qquad \frac{|\mathcal{T}_{\mathrm{F}}(T)|}{T} \longrightarrow 0, \quad T \to \infty. \tag{10}$$

Hence, full-feedback operations become asymptotically sparse, with T_0 and a providing explicit control over the full-feedback budget. Each full-feedback round incurs additional communication and computation from transmitting and evaluating the full model, as well as increased peak activation memory during backpropagation. We can employ activation checkpointing (Chen et al., 2016) to reduce this memory overhead, while the increasing feedback intervals amortize the additional costs over resource-compatible training rounds.

4 Convergence Analysis

FedESO learns importance rankings that may induce changing TopK masks across communication rounds. Let $M_t = \{M_t^{(i)}\}_{i\in[N]}$ denote the client masks at round t, and define the corresponding time-varying MH-FL objective as $F_t(w) := F(w, M_t)$. We characterize mask-induced objective variation by

$$F_{t+1}(w) - F_t(w) \le \rho_t, \qquad V_T := \sum_{t=0}^{T-1} \rho_t, \tag{11}$$

where V_T is the cumulative objective variation. We further assume standard L-smoothness, unbiased stochastic gradients with bounded variance σ^2, bounded client heterogeneity ζ^2, and a uniform lower bound $F_{\inf}$. Detailed assumptions and proofs are provided in the Appendix. Unlike standard federated optimization, changes in M_t cause the objective itself to vary across rounds. Bounding this additional variation by Eq. 11 yields the following result.

Theorem 1 (Dynamic-mask convergence). Under the assumptions above and a sufficiently small learning rate η_w, FedESO with K_w local updates per round satisfies

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\big[\|\nabla F_t(w_t)\|^2\big] \le \frac{F_0(w_0) - F_{\inf} + V_T}{c\eta_w K_w T} + C\eta_w K_w L(\sigma^2 + \zeta^2), \tag{12}$$

for constants $c, C > 0$. Balancing the terms gives

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\big[\|\nabla F_t(w_t)\|^2\big] = O\left(\sqrt{\frac{1+V_T}{T}}\right). \tag{13}$$

Hence, $V_T = o(T)$ is sufficient for asymptotic stationarity. Importantly, this condition does not require finite-time mask stabilization: the masks may continue to change provided that their induced cumulative objective variation grows sublinearly.

Finite-time stabilization is a special case. Suppose that, for some T_s,

$$M_t = M_\star, \qquad \forall t \ge T_s. \tag{14}$$

Then $F_t(w) = F(w, M_\star)$ and $\rho_t = 0$ for all $t \ge T_s$.

Corollary 1 (Convergence after stabilization). Under the conditions of Theorem 1 and Eq. 14, with an appropriate learning rate,

$$\frac{1}{T-T_s}\sum_{t=T_s}^{T-1}\mathbb{E}\big[\|\nabla F(w_t, M_\star)\|^2\big] = O\bigg(\frac{1}{\sqrt{T-T_s}}\bigg). \tag{15}$$

Thus, after mask stabilization, FedESO recovers the standard nonconvex stationary convergence rate for the resulting fixed aggregate submodel objective. Together, Theorem 1 and Corollary 1 characterize both dynamically evolving and stabilized submodels: sublinear cumulative objective variation is sufficient for asymptotic stationarity, while finite-time stabilization recovers convergence for a fixed resource-compatible submodel objective.

5 Experiments

5.1 Setup

Datasets and Models. We evaluate FedESO on representative computer vision (CV) and natural language processing (NLP) tasks. For image classification, we train ViT-Small/16 from scratch on CIFAR-100 in FP32 precision (approximately 88.4 MB) (Dosovitskiy, 2020). For text classification, we fine-tune a pre-trained BERT-Small model on Banking-77 in FP32 precision (approximately 116.3 MB) (Turc et al., 2019; Casanueva et al., 2020).

Data and Resource Heterogeneity. We partition the data among clients using a Dirichlet distribution with concentration parameter $\alpha = 0.3$, following common non-IID FL settings (Jhunjhunwala et al., 2023; Wu et al., 2024). The same client partitions are used across all methods. We consider three client capacity tiers, $r_i \in \{1/4, 1/16, 1/64\}$, evenly assigned across clients (Wu et al., 2024). Here, r_i denotes the fraction of global-model parameters that client i can train, realized through structured width reduction. We assume that clients can store the full model, while their training-time resources constrain the instantiated submodel size.

Baselines. We compare FedESO with three representative MH-FL methods: HeteroFL (Diao et al., 2020) with static submodel extraction, FedRolex (Alam et al., 2022) with rolling extraction, and FIARSE (Wu et al., 2024) with magnitude-based importance-aware extraction. FIARSE uses unstructured masking, which does not directly provide the practical training-memory reduction of structured physical submodels.

FL Configuration. We simulate $N = 12$ clients with full participation in every communication round. Full participation eliminates variability from client subsampling, allowing performance differences to be more directly attributed to the submodel extraction and optimization strategies. All methods use identical data partitions, resource-tier assignments, and training configurations unless otherwise specified.

Evaluation Metrics. Following prior MH-FL studies (Alam et al., 2022; Wu et al., 2024), we report local top-1 test accuracy separately for the $1/4$, $1/16$, and $1/64$ capacity tiers, as well as the average across clients. Each submodel is evaluated on its corresponding client's local test partition.

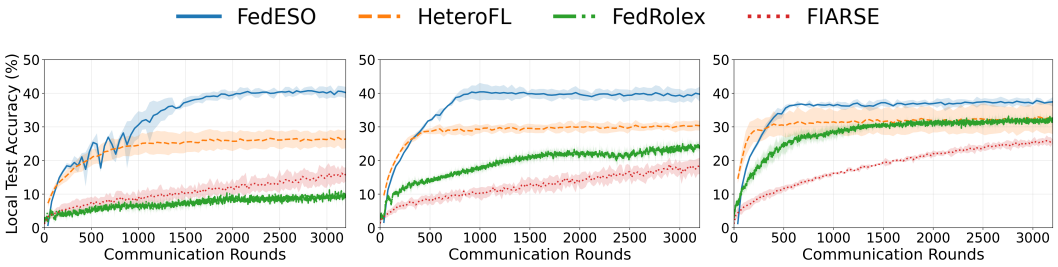
5.2 Comparison with State-of-the-Art Methods

Table 1 compares FedESO with representative MH-FL baselines. On CIFAR-100 with ViT-Small trained from scratch, FedESO achieves the highest accuracy across all capacity tiers, improving the average accuracy from 29.46% for the strongest baseline to 39.09%. The improvement is particularly pronounced for the $1/64$ submodel, where FedESO achieves 40.17%, compared with 26.47% for HeteroFL. These results demonstrate the benefit of optimization-aware submodel selection when task-specific representations must be learned from scratch.

The convergence curves in Figure 1 further reveal different exploration–stabilization behaviors. HeteroFL converges rapidly and smoothly but to lower accuracy, suggesting that stable optimization alone may lead to suboptimal submodels when alternative components are insufficiently explored. FedRolex exhibits persistent accuracy variation under rolling extraction. In contrast, FedESO exhibits greater variation primarily during early and intermediate training, followed by increasingly stable

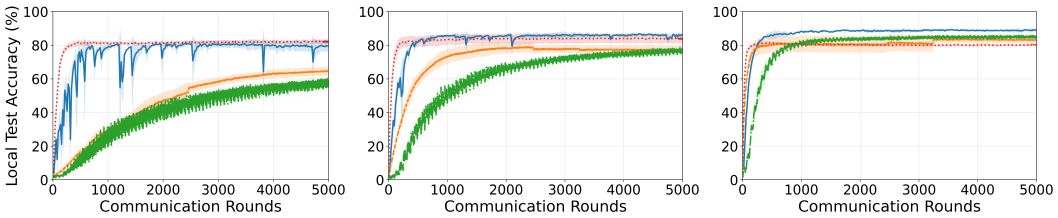
Table 1: Comparison of local test accuracy, FLOPs and communication costs across methods. The columns under (1/64, 1/16, 1/4) Model are average test accuracy of clients w.r.t. their submodel sizes.

	CIFAR-100 / ViT-small (Train-From-Scratch)						Banking-77 / BERT-small (Fine-Tuning)					
	Test Accuracy (%)				FLOPs	Comm.	Test Accuracy (%)				FLOPs	Comm.
Method	**1/64 Model**	**1/16 Model**	**1/4 Model**	**Avg.**	**($\times 10^{15}$)**	**(GB)**	**1/64 Model**	**1/16 Model**	**1/4 Model**	**Avg.**	**($\times 10^{15}$)**	**(TB)**
HeteroFL	26.47 ± 2.67	29.89 ± 1.18	32.01 ± 3.30	29.46 ± 1.27	**47.97**	**676.43**	63.30 ± 4.53	79.02 ± 3.03	80.13 ± 4.47	73.99 ± 1.43	26.50	1.50
FedRolex	9.34 ± 0.03	23.80 ± 1.18	31.85 ± 0.77	21.66 ± 0.64	53.97	760.98	56.80 ± 1.84	76.86 ± 1.24	83.32 ± 2.32	72.32 ± 0.81	**26.00**	**1.49**
FIARSE	15.81 ± 2.15	17.69 ± 1.84	25.62 ± 0.86	19.70 ± 1.13	418.87	668.79	**82.05 ± 1.16**	83.86 ± 2.87	80.17 ± 0.57	82.03 ± 1.21	487.61	2.21
FedESO	**40.17 ± 1.69**	**39.69 ± 2.15**	**37.40 ± 1.35**	**39.09 ± 1.73**	49.28	867.55	79.73 ± 3.03	**86.05 ± 1.13**	**89.07 ± 0.92**	**84.95 ± 0.94**	28.92	1.90



(a) 1/64 Model (b) 1/16 Model (c) 1/4 Model

Figure 1: Comparison of convergence for training ViT-small on CIFAR-100.



(a) 1/64 Model (b) 1/16 Model (c) 1/4 Model

Figure 2: Comparison of convergence for training BERT-small on Banking-77.

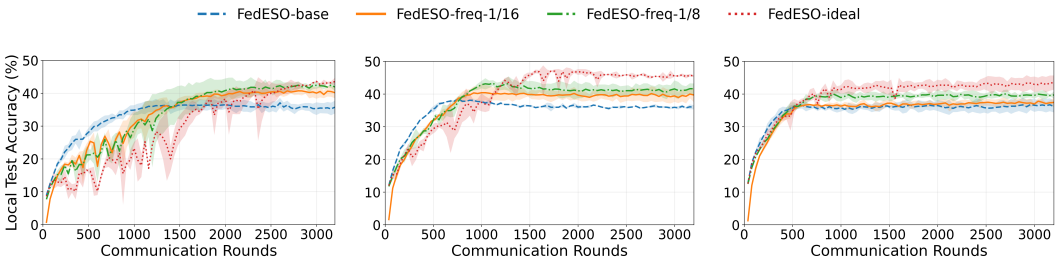
optimization at higher accuracy. This variation is most pronounced for the $1/64$ submodel, where changes in a small number of selected components constitute a larger fraction of its active support and can therefore have a greater impact on submodel performance.

On Banking-77 with pre-trained BERT-Small, the performance differences are smaller. FedESO achieves the highest average accuracy (84.95%) and performs best at the $1/16$ and $1/4$ tiers, while FIARSE performs better at the $1/64$ tier. This suggests that submodel selection matters more when training a model from scratch, whereas pre-trained models already provide strong representations and are therefore less sensitive to the choice of submodel. We also note that FIARSE employs unstructured masking, whose nominal sparsity does not directly translate into the practical training-memory reduction provided by structured physical submodels.

5.3 Impact of Full Importance Feedback

We ablate the frequency of full importance feedback on CIFAR-100. As shown in Table 2, the average accuracy consistently increases from 35.94% without full feedback to 39.09%, 41.05%, and 44.16% with full feedback in $1/16$, $1/8$, and all communication rounds, respectively. These results support our partial-feedback motivation: providing first-order feedback for excluded components allows the learned importance ranking to reconsider current selections and leads to better-performing submodels.

The convergence curves in Figure 3 provide further insight. Without full feedback, FedESO-base converges relatively smoothly but to lower accuracy. Increasing full-feedback frequency introduces greater variation during early and intermediate training while consistently yielding higher final accuracy. Notably, even the every-round full-feedback variant becomes increasingly stable later in



(a) 1/64 Model (b) 1/16 Model (c) 1/4 Model

Figure 3: Impact of full-feedback frequency for training ViT-small on CIFAR-100.

Table 2: Impact of the frequency of full-importance feedback. FedESO-ideal (or base) has full-feedback at each (or no) round.

	CIFAR-100 / ViT-small (Train-From-Scratch)					
	Test Accuracy (%)				FLOPs	Comm.
Method	**1/64 Model**	**1/16 Model**	**1/4 Model**	**Avg.**	($\times 10^{15}$)	**(GB)**
FedESO-base	35.72 ± 1.92	35.92 ± 0.65	36.17 ± 1.80	35.94 ± 1.02	**47.97**	**676.43**
FedESO-freq-1/16	40.17 ± 1.69	39.69 ± 2.15	37.40 ± 1.35	39.09 ± 1.73	49.28	867.55
FedESO-freq-1/8	41.88 ± 1.19	41.61 ± 1.58	39.67 ± 1.01	41.05 ± 1.26	50.59	1058.68
FedESO-ideal	$\mathbf{43.50 \pm 1.10}$	$\mathbf{45.49 \pm 0.29}$	$\mathbf{43.48 \pm 2.16}$	$\mathbf{44.16 \pm 1.18}$	58.76	2252.26

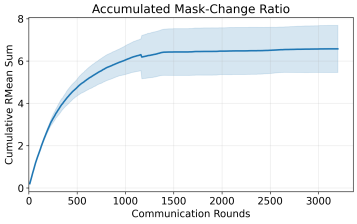


Figure 4: Accumulated mask-change across rounds.

training. This suggests that stabilization can emerge as the learned importance rankings become increasingly consistent, rather than simply from reducing the frequency of full feedback.

Full feedback nevertheless incurs additional resource costs. The $1/16$ configuration used in our main experiments improves average accuracy by 3.15 percentage points over FedESO-base while increasing computation from only 47.97 to 49.28 PFLOPs. In contrast, every-round full feedback achieves the highest accuracy but increases computation to 58.76 PFLOPs and communication to 2252.26 GB. Thus, scheduled full feedback provides a practical tradeoff between informative submodel exploration and additional resource overhead.

Finally, Figure 4 shows the cumulative variation of the learned masks. Mask variation accumulates rapidly during early training but grows substantially more slowly at later rounds, indicating a transition from active exploration toward increasingly stable submodel extraction. Small variations nevertheless persist throughout training, showing that finite-time mask stabilization is not required. Although this structural variation is distinct from the objective variation V_T in our theoretical analysis, its diminishing growth provides empirical evidence consistent with the sublinear-variation regime considered in Theorem 1.

6 Conclusion

We proposed FedESO, an exploration-to-stabilization framework for model-heterogeneous federated learning that targets the performance of resource-compatible client submodels. FedESO learns task-dependent importance rankings and addresses the partial-feedback problem of physical submodel execution through scheduled full importance feedback, enabling informed exploration while controlling additional resource costs. We further characterized dynamic submodel optimization through cumulative objective variation, showing that sublinear variation is sufficient for asymptotic stationarity, with finite-time mask stabilization recovering convergence for the resulting fixed submodel objective. Experiments across vision and NLP tasks demonstrate the effectiveness of FedESO and highlight the importance of balancing informative submodel exploration with consistent optimization.

AI USE STATEMENT

In this work, we used generative AI tools for writing assistance, specifically to polish and refine manuscript text for clarity and style. We also used AI for assistance with code implementation and organizing experimental results, as well as conducting literature surveys of related work. We have not used generative AI tools for core research idea generation, experimental design, or mathematical proof derivations, and data collection tasks are not applicable to this work.

REFERENCES

Samiul Alam, Luyang Liu, Ming Yan, and Mi Zhang. Fedrolex: Model-heterogeneous federated learning with rolling sub-model extraction. *Advances in neural information processing systems*, 35:29677–29690, 2022.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd workshop on natural language processing for conversational AI*, pages 38–45, 2020.

Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. Training deep nets with sublinear memory cost. *arXiv preprint arXiv:1604.06174*, 2016.

Enmao Diao, Jie Ding, and Vahid Tarokh. Heterofl: Computation and communication efficient federated learning for heterogeneous clients. *arXiv preprint arXiv:2010.01264*, 2020.

Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Samuel Horvath, Stefanos Laskaridis, Mario Almeida, Ilias Leontiadis, Stylianos Venieris, and Nicholas Lane. Fjord: Fair and accurate federated learning under heterogeneous targets with ordered dropout. *Advances in Neural Information Processing Systems*, 34:12876–12889, 2021.

Divyansh Jhunjhunwala, Shiqiang Wang, and Gauri Joshi. Fedexp: Speeding up federated averaging via extrapolation. *arXiv preprint arXiv:2301.09604*, 2023.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. Pmlr, 2017.

Victor Sanh, Thomas Wolf, and Alexander Rush. Movement pruning: Adaptive sparsity by fine-tuning. *Advances in neural information processing systems*, 33:20378–20389, 2020.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models, 2019.

Feijie Wu, Xingchen Wang, Yaqing Wang, Tianci Liu, Lu Su, and Jing Gao. Fiarse: Model-heterogeneous federated learning via importance-aware submodel extraction. *Advances in Neural Information Processing Systems*, 37:115615–115651, 2024.

Jin Xu, Xu Tan, Kaitao Song, Renqian Luo, Yichong Leng, Tao Qin, Tie-Yan Liu, and Jian Li. Analyzing and mitigating interference in neural architecture search. In *International Conference on Machine Learning*, pages 24646–24662. PMLR, 2022.

Hanhan Zhou, Tian Lan, Guru Prasadh Venkataramani, and Wenbo Ding. Every parameter matters: Ensuring the convergence of federated learning with dynamic heterogeneous models reduction. *Advances in Neural Information Processing Systems*, 36:25991–26002, 2023.

A Appendix: Proof of Theorem 1

We first state the assumptions used in the analysis. For a fixed set of client masks $M_t = \{M_t^{(i)}\}_{i\in[N]}$ at round t, define

$$F_{i,t}(w) := F_i(w \odot M_t^{(i)}; D_i), \tag{16}$$

$$F_t(w) := \frac{1}{N}\sum_{i=1}^{N} F_{i,t}(w). \tag{17}$$

Assumption 1 (L-smoothness)**.** For every client i, round t, and w, w',

$$\|\nabla F_{i,t}(w) - \nabla F_{i,t}(w')\| \le L\|w - w'\|. \tag{18}$$

Assumption 2 (Bounded stochastic-gradient variance)**.** Let $g_{t,k}^{(i)}$ denote the stochastic gradient computed by client i at local step k. Then

$$\mathbb{E}\Big[g_{t,k}^{(i)} \mid w_{t,k}^{(i)}, M_t\Big] = \nabla F_{i,t}(w_{t,k}^{(i)}), \tag{19}$$

$$\mathbb{E}\Big[\|g_{t,k}^{(i)} - \nabla F_{i,t}(w_{t,k}^{(i)})\|^2\Big] \le \sigma^2. \tag{20}$$

Assumption 3 (Bounded client heterogeneity)**.** For every round t and w,

$$\frac{1}{N}\sum_{i=1}^{N} \|\nabla F_{i,t}(w) - \nabla F_t(w)\|^2 \le \zeta^2. \tag{21}$$

Assumption 4 (Bounded objective variation)**.** There exists a sequence $\{\rho_t\}_{t\ge 0}$ with $\rho_t \ge 0$ such that, for every w,

$$F_{t+1}(w) - F_t(w) \le \rho_t. \tag{22}$$

Define the cumulative objective variation as

$$V_T := \sum_{t=0}^{T-1} \rho_t. \tag{23}$$

Moreover, the round-dependent objectives are uniformly lower bounded:

$$F_t(w) \ge F_{\inf}, \qquad \forall t, w. \tag{24}$$

Proof. A.1 Local Updates

At round t, client i initializes

$$w_{t,0}^{(i)} = w_t \tag{25}$$

and performs K_w local updates

$$w_{t,k+1}^{(i)} = w_{t,k}^{(i)} - \eta_w g_{t,k}^{(i)}, \qquad k = 0, \dots, K_w - 1. \tag{26}$$

The server update is written as

$$w_{t+1} = w_t + \Delta_t, \tag{27}$$

$$\Delta_t := -\frac{\eta_w}{N}\sum_{i=1}^{N}\sum_{k=0}^{K_w-1} g_{t,k}^{(i)}. \tag{28}$$

A.2 One-Round Descent

Since F_t is L-smooth, conditioned on the masks M_t,

$$F_t(w_{t+1}) - F_t(w_t) \le \langle \nabla F_t(w_t), \Delta_t\rangle + \frac{L}{2}\|\Delta_t\|^2. \tag{29}$$

We first bound the inner-product term. Taking conditional expectations over the stochastic gradient noise at round t:

$$\mathbb{E}[\Delta_t] = -\eta_w K_w G_t, \quad \text{where} \quad G_t := \frac{1}{NK_w} \sum_{i=1}^{N} \sum_{k=0}^{K_w-1} \nabla F_{i,t}(w_{t,k}^{(i)}). \tag{30}$$

Using the identity $-\langle a, b\rangle = -\frac{1}{2}\|a\|^2 - \frac{1}{2}\|b\|^2 + \frac{1}{2}\|a-b\|^2 \le -\frac{1}{2}\|a\|^2 + \frac{1}{2}\|a-b\|^2$, we get:

$$\begin{aligned}\mathbb{E}\left\langle \nabla F_t(w_t), \Delta_t \right\rangle &= -\eta_w K_w \mathbb{E}\left\langle \nabla F_t(w_t), G_t \right\rangle \\ &\le -\frac{\eta_w K_w}{2}\mathbb{E}\|\nabla F_t(w_t)\|^2 + \frac{\eta_w K_w}{2}\mathbb{E}\|\nabla F_t(w_t) - G_t\|^2. \end{aligned} \tag{31}$$

By Jensen's inequality and L-smoothness of $F_{i,t}$:

$$\begin{aligned}\mathbb{E}\|\nabla F_t(w_t) - G_t\|^2 &= \mathbb{E}\left\| \frac{1}{NK_w} \sum_{i=1}^{N} \sum_{k=0}^{K_w-1} \left(\nabla F_{i,t}(w_t) - \nabla F_{i,t}(w_{t,k}^{(i)}) \right) \right\|^2 \\ &\le \frac{L^2}{NK_w} \sum_{i=1}^{N} \sum_{k=0}^{K_w-1} \mathbb{E}\|w_{t,k}^{(i)} - w_t\|^2. \end{aligned} \tag{32}$$

Applying standard local-SGD drift bounds under Assumptions 1–3, for $\eta_w K_w L \le \frac{1}{2C_1^{1/2}}$:

$$\frac{1}{NK_w} \sum_{i=1}^{N} \sum_{k=0}^{K_w-1} \mathbb{E}\|w_{t,k}^{(i)} - w_t\|^2 \le C_1 \eta_w^2 K_w^2 \left(\mathbb{E}\|\nabla F_t(w_t)\|^2 + \sigma^2 + \zeta^2 \right), \tag{33}$$

where $C_1 > 0$ is a constant. Substituting Eq. 33 into Eq. 32 and then into Eq. 31:

$$\mathbb{E}\left\langle \nabla F_t(w_t), \Delta_t \right\rangle \le -\left(\frac{\eta_w K_w}{2} - \frac{C_1 L^2 \eta_w^3 K_w^3}{2} \right) \mathbb{E}\|\nabla F_t(w_t)\|^2 + \frac{C_1 L^2 \eta_w^3 K_w^3}{2}(\sigma^2 + \zeta^2). \tag{34}$$

Similarly, for the second-order term $\mathbb{E}\|\Delta_t\|^2$:

$$\mathbb{E}\|\Delta_t\|^2 \le C_3 \eta_w^2 K_w^2 \left(\mathbb{E}\|\nabla F_t(w_t)\|^2 + \sigma^2 + \zeta^2 \right). \tag{35}$$

Substituting Eqs. 34 and 35 into Eq. 29, and picking η_w small enough such that $\left(\frac{1}{2} - \frac{C_1 L^2 \eta_w^2 K_w^2}{2} - \frac{C_3 L \eta_w K_w}{2} \right) \ge c > 0$, we establish:

$$\mathbb{E}[F_t(w_{t+1}) - F_t(w_t)] \le -c\eta_w K_w \mathbb{E}\|\nabla F_t(w_t)\|^2 + C\eta_w^2 K_w^2 L(\sigma^2 + \zeta^2). \tag{36}$$

A.3 Accounting for Mask-Induced Objective Variation

Decomposing the objective jump between successive rounds:

$$F_{t+1}(w_{t+1}) - F_t(w_t) = F_t(w_{t+1}) - F_t(w_t) + F_{t+1}(w_{t+1}) - F_t(w_{t+1}). \tag{37}$$

Using Assumption 4 ($F_{t+1}(w_{t+1}) - F_t(w_{t+1}) \le \rho_t$) along with Eq. 36:

$$\mathbb{E}[F_{t+1}(w_{t+1}) - F_t(w_t)] \le -c\eta_w K_w \mathbb{E}\|\nabla F_t(w_t)\|^2 + C\eta_w^2 K_w^2 L(\sigma^2 + \zeta^2) + \rho_t. \tag{38}$$

Telescoping over $t = 0, \ldots, T-1$:

$$\mathbb{E}[F_T(w_T)] - F_0(w_0) \le -c\eta_w K_w \sum_{t=0}^{T-1} \mathbb{E}\|\nabla F_t(w_t)\|^2 + C\eta_w^2 K_w^2 L(\sigma^2 + \zeta^2)T + V_T. \tag{39}$$

Rearranging and applying the lower bound $F_T(w_T) \ge F_{\inf}$:

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\|\nabla F_t(w_t)\|^2 \le \frac{F_0(w_0) - F_{\inf} + V_T}{c\eta_w K_w T} + \tilde{C}\eta_w K_w L(\sigma^2 + \zeta^2). \tag{40}$$

Setting $A_T := F_0(w_0) - F_{\inf} + V_T$ and choosing step size $\eta_w K_w = \Theta(\sqrt{A_T/T})$ gives:

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\|\nabla F_t(w_t)\|^2 = \mathcal{O}\left(\sqrt{\frac{1 + V_T}{T}} \right). \tag{41}$$

If $V_T = o(T)$, then $\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla F_t(w_t)\|^2 \longrightarrow 0$, establishing asymptotic stationarity. □