

One-Shot Federated Model Editing for Device Dynamics at the Edge

Youming Tao, Weijun Yin, Huanyi Xie, Yanqing Yao, Dongxiao Yu, *Senior Member, IEEE*,
Xiuzhen Cheng, *Fellow, IEEE*, Di Wang, *Member, IEEE*, Falko Dressler *Fellow, IEEE*

Abstract—Standard federated learning (FL) requires persistent device connectivity across multiple communication rounds, making it unsuitable for dynamic environments such as vehicular networks, where devices frequently join and leave. One-shot FL offers a promising alternative by completing training in a single communication round. However, one-shot FL suffers from model solidification and cannot readily adapt to subsequent client dynamics, including new client contributions and the withdrawal of previous contributions. We therefore consider exact model editing, which removes withdrawn contributions through unlearning or incorporates newly contributed ones through incremental learning without retraining from scratch. We identify that efficient editing fundamentally hinges on learning stability. To this end, we propose F-SOSA, the first stable one-shot FL algorithm that enables rapid and exact model editing. F-SOSA leverages sub-sampling of local models with carefully tuned rate for stable aggregation and performs curvature-aware model fusion to preserve utility. We prove that F-SOSA achieves vanishing training error and reasonably small generalization error bounds. Building on this stability, we further develop two single-round editing algorithms: F-SOSA-U for unlearning and F-SOSA-I for incremental learning. They ensure that the edited model, along with all intermediate algorithmic states, is statistically indistinguishable from retraining while incurring only an expected $O(\rho)$ fraction of its cost, where ρ quantifies stability. Experiments demonstrate that our approach matches retraining performance while reducing cumulative time and communication overhead by at least 36% and 22%, respectively, under realistic urban mobility simulations.

Index Terms—Federated Learning, Dynamic Participation, Model Editing, Single-Round Communication

I. INTRODUCTION

Federated learning (FL) [1] enables collaborative model training across distributed devices without sharing raw data.

This work was supported by the National Natural Science Foundation of China (Grant No. 62572025), Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing (Grant No. GJJ-25-xxx), and the Fundamental Research Funds for the Central Universities, China.

Youming Tao and Falko Dressler are with the School of Electrical Engineering and Computer Science, TU Berlin, 10623 Berlin, Germany (e-mail: tao@ccs-labs.org; dressler@ccs-labs.org).

Weijun Yin and Yanqing Yao are with the School of Cyber Science and Technology, Beihang University, Beijing 100191, China, and also with the Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing, Beihang University, Beijing 100083, China (e-mail: yinweijun@buaa.edu.cn; yaoyq@buaa.edu.cn).

Huanyi Xie and Di Wang are with the Division of Computer, Electrical and Mathematical Sciences and Engineering, King Abdullah University of Science and Technology, Saudi Arabia (e-mail: huanyi.xie@kaust.edu.sa; di.wang@kaust.edu.sa).

Dongxiao Yu and Xiuzhen Cheng are with the School of Computer Science and Technology, Shandong University, Qingdao 266237, China (e-mail: dxyu@sdu.edu.cn; xzcheng@sdu.edu.cn).

The first two authors contributed equally.

Corresponding author: Y. Yao and D. Wang.

In mobile edge computing (MEC) scenarios such as vehicular networks, client participation is inherently dynamic. Devices frequently join or leave due to mobility or network instability. This challenges standard FL algorithms like FedAvg [1], FedProx [2], and SCAFFOLD [3]. These methods rely on a largely fixed client population and multiple rounds of communication. Such assumptions rarely hold in practice.

To remove reliance on persistent availability, one-shot FL [4] has emerged, which completes training in a single round. Recent studies [5]–[7] show that one-shot FL can achieve competitive performance over non-iid data. This makes it well-suited for dynamic systems. A representative use case is vehicular micro-clouds (VMCs) [8]. In a VMC, nearby vehicles form a temporary edge collaboration, making persistent multi-round participation difficult to maintain. One-shot FL is therefore a natural solution.

However, one-shot FL suffers from model solidification. Once trained, the global model cannot adapt to changes in the client population. When clients arrive or depart, the set of participants contributing to the collaborative model changes. A newly arrived client can contribute its local model and benefit from the resulting global model, whereas a departing client may withdraw its previous contribution for privacy before disconnecting from the coordinator. Supporting such contribution additions and withdrawals requires the one-shot model to accommodate incremental learning and unlearning after its initial training. This issue is particularly relevant in dynamic edge applications such as VMCs, where vehicles participate only temporarily and repeated retraining after every arrival or departure would incur substantial communication and computation overhead. Approximate editing methods such as fine-tuning or model calibration can reduce this cost, but successive approximate updates may deviate from the model produced by retraining and propagate this deviation to later edits. Exact editing instead restores the same distribution as retraining after every change, preventing such history-dependent drift. These challenges motivate the following question:

Can we efficiently edit a one-shot trained FL model to reflect an updated client set without retraining?

We formalize this goal as *exact federated model editing*, which includes two tasks: (1) *client unlearning* to remove a client's previous contribution upon a withdrawal request, and (2) *client incremental learning* to incorporate the contribution of a newly participating client. By exactness, we require statistical indistinguishability between editing and retraining, which applies

to both the final model and all intermediate algorithmic states. To achieve this goal, we propose the first exact model editing framework for one-shot FL. The framework consists of a stable one-shot FL algorithm, F-SOSA, and two editing algorithms: F-SOSA-U for client unlearning and F-SOSA-I for client incremental learning. Our main contributions are summarized as follows:

- 1) We formalize the definition of exact federated model editing through statistical indistinguishability between editing and retraining. Specifically, we require that the joint distributions of the obtained model and all intermediate algorithmic states produced by editing match those generated by retraining on the updated client set.
- 2) We design F-SOSA as an edit-friendly one-shot FL algorithm that combines two complementary mechanisms. First, we reformulate curvature-aware model fusion as a projection characterized by client-additive sufficient statistics, so that a single-client change can be incorporated by updating only the corresponding contribution rather than rerunning the full fusion procedure. Second, we introduce randomized client subsampling to make the one-shot output stable in total variation, thereby allowing many membership changes to reuse the existing model and states without recomputation.
- 3) We provide theoretical utility guarantees for F-SOSA. Specifically, we establish upper bounds on both training and generalization errors for over-parameterized neural networks. The training error vanishes as network width and local training increase. The generalization error converges to a constant governed by data non-degeneracy and becomes favorably small as the numbers of data samples and clients grow.
- 4) Building upon F-SOSA, we develop F-SOSA-U and F-SOSA-I for client unlearning and incremental learning. Thanks to F-SOSA's total variation stability quantified by $\rho < 1$, not all editing requests need execution. We design a lightweight verification to determine the necessity. As a result, only an expected $O(\rho)$ -fraction of requests trigger the editing process.
- 5) We conduct extensive experiments on standard benchmarks to evaluate model utility, editing exactness, and efficiency. F-SOSA achieves strong accuracy compared with state-of-the-art one-shot FL methods. F-SOSA-U/I match retraining performance while significantly reducing time and communication costs. We verify the exactness of editing using membership inference attacks. We also present a case study with simulated vehicular scenarios to demonstrate the practical applicability of our proposed approach.

II. RELATED WORK

We review three related research directions in FL: unlearning, continual learning, and dynamic client participation.

Federated unlearning. Federated unlearning was first studied in [9], which removes a client approximately by reversing its updates. While faster than retraining, it remains computationally expensive. Subsequent works improve efficiency through noise perturbation [10], [11], knowledge distillation [12],

class-discriminative pruning [13], or gradient ascent [14]. However, all these methods provide only approximate removal. As a result, the model remains vulnerable to membership inference and gradient leakage [15]–[20]. Recent studies explore exact unlearning. Liu et al. [21] propose a Newton-style retraining method that uses Fisher information for curvature approximation. However, their retraining path deviates from the original trajectory and relies on convexity assumptions. Tao et al. [22] introduce a stabilized FedAvg variant called FATS for exact unlearning via resampling and rapid retraining. Their one-shot variant, however, suffers from degraded utility due to simple averaging and does not support client addition. **Federated continual learning.** Federated continual learning assumes a fixed set of clients whose local data evolve over time. Existing work considers different settings including task-incremental [23]–[26], class-incremental [27]–[29], and domain-incremental [30]–[32], which correspond to sequential tasks, new classes, and domain shifts, respectively. The primary goal is to mitigate catastrophic forgetting and preserve past knowledge. As they prioritize knowledge retention over exact alignment with new data, these methods cannot guarantee statistical equivalence to retraining. In addition, they rely on multi-round training and continued access to past clients. In contrast, we study dynamic client populations where new clients may join after others have left.

Federated Learning with dynamic participation. Recent studies consider federated learning with dynamic participants. Ruan et al. [33] and Chang et al. [34] allow clients to join or leave during training and provide convergence guarantees. Their setting differs fundamentally from ours. They study multi-round federated training where client changes occur across rounds and the goal is to adapt during training. In contrast, we focus on one-shot FL, which involves only a single round. In this setting, client dynamics arise after training rather than during it. Our goal is therefore to enable exact model editing of the trained model to accommodate changes in the client population.

Summary. Existing studies address related but distinct challenges. Federated unlearning focuses on removing clients but typically provides only approximate guarantees or suffers from utility degradation in one-shot settings. Federated continual learning assumes a fixed client set and prioritizes knowledge retention rather than exact alignment with updated data. Work on dynamic participation adapts to client churn during multi-round training and does not apply to post-training model editing in one-shot FL. Consequently, none of these directions directly address our setting, where client additions and withdrawals occur after one-shot training. This challenge is amplified under repeated edits, since any residual deviation may propagate to subsequent updates. The goal is therefore not merely post-training adaptation, but efficient and exact editing that remains valid across successive client-set changes.

III. PRELIMINARIES

A. System Model

We consider an edge system consisting of a trusted infrastructure-based coordinator and a dynamic set of devices.

The coordinator can, for example, be an RSU or MEC server in a vehicular micro-cloud (VMC), with vehicles serving as participating devices. Time proceeds in discrete rounds. Devices may join or leave at each round. We do not impose a particular distribution on client arrivals or departures; their occurrence may be non-uniform across devices and time. Let $\mathcal{M}(t)$ denote the set of active devices at round t , and let $M(t) = |\mathcal{M}(t)|$. The coordinator maintains knowledge of the active device set. When a new device joins, it establishes a connection with the coordinator and may submit its local model for inclusion in the collaborative model. Before an active device departs and loses connectivity to the coordinator, it may issue an unlearning request to withdraw its previous contribution. Standard authenticated communication between devices and the coordinator is assumed.

B. Federated Learning

Let M denote the number of active devices when the initial one-shot learning is performed. Each device $i \in [M]$ holds a local dataset $\mathcal{D}_i = \{(\mathbf{x}_{ij}, y_{ij})\}_{j=1}^n$, where $\mathbf{x}_{ij} \in \mathbb{R}^p$ denotes the feature vector and $y_{ij} \in \mathbb{R}$ denotes the label. The global dataset is $\mathcal{D} = \bigcup_{i=1}^M \mathcal{D}_i$ with total size $N = Mn$. The learning objective can be formulated as the following distributed optimization problem:

$$\min_{\mathbf{W} \in \mathbb{R}^d} L_{\mathcal{D}}^{f,\ell}(\mathbf{W}) := \frac{1}{M} \sum_{i=1}^M L_{\mathcal{D}_i}^{f,\ell}(\mathbf{W}), \quad (1)$$

where $L_{\mathcal{D}_i}^{f,\ell}(\mathbf{W}) := \frac{1}{n} \sum_{(x,y) \in \mathcal{D}_i} \ell(f(\mathbf{W}, \mathbf{x}), y)$, $f(\mathbf{W}, \cdot)$ denotes the model parameterized by $\mathbf{W}$, and $\ell(\cdot, \cdot)$ denotes the loss function. Following the standard streaming-edit setting [35], we consider editing sequences in which the active population remains within a constant factor of its initial size, i.e., $M/2 \leq M(t) \leq 2M$.

C. Federated Model Editing

To support dynamic client participation, our goal is to enable the global model to adapt efficiently to changes in the client set. Specifically, we aim to (i) remove the influence of departed clients that withdraw their previous contributions (client unlearning), and (ii) incorporate newly arrived clients (client incremental learning), while preserving model utility and avoiding full retraining.

Let $\mathfrak{M}$ denote the space of devices, $\mathcal{W}$ the space of model parameters, and $\mathcal{S}$ the space of internal algorithmic states, which will be specified later. A one-shot FL algorithm is a randomized map $\mathcal{A} : \mathfrak{M}^* \rightarrow \mathcal{W} \times \mathcal{S}$. Given a set of devices, it produces a global model together with internal algorithmic states. A model editing algorithm is a map $\mathcal{U} : \mathcal{W} \times \mathcal{S} \times \mathfrak{M}^* \rightarrow \mathcal{W} \times \mathcal{S}$. It takes the current model, the associated states, and a set of client addition or removal requests, and outputs an updated model and states. Examples of internal states include sketches of device information or intermediate computations that can be reused during editing.

Definition 1 (Exact Federated Model Editing). Let $\mathcal{M}$ and $\mathcal{M}'$ denote the device sets before and after a change. Given a

one-shot FL algorithm $\mathcal{A}$, a model editing algorithm $\mathcal{U}$ satisfies exact model editing if for any measurable event $\mathcal{E} \subseteq \mathcal{W} \times \mathcal{S}$,

$$\mathbb{P}[\mathcal{A}(\mathcal{M}') \in \mathcal{E}] = \mathbb{P}[\mathcal{U}(\mathcal{A}(\mathcal{M}), \mathcal{M} \oplus \mathcal{M}') \in \mathcal{E}],$$

where $\mathcal{M} \oplus \mathcal{M}'$ denotes the symmetric difference, i.e., $\mathcal{M} \oplus \mathcal{M}' = (\mathcal{M} \setminus \mathcal{M}') \cup (\mathcal{M}' \setminus \mathcal{M})$.

This definition requires that both the output model and all internal algorithmic states after editing be statistically indistinguishable from those obtained by retraining on the updated device set. Requiring exactness for the internal states is important for repeated editing: these states are reused by subsequent editing operations, and any approximation retained in them may be propagated to later model updates. If one edit produces exactly the same joint distribution as retraining, the resulting state can be treated as a fresh execution of the learning algorithm on the updated client set. The guarantee thus extends to an arbitrary sequence of edits by induction, without accumulating approximation error across requests.

D. Total Variation (TV) Stability

Tao et al. [22] showed that exact federated unlearning can be achieved efficiently when the FL algorithm is stable in total variation (TV) distance. We extend this insight to general model editing. Let $P = \mathcal{A}(\mathcal{M})$ and $Q = \mathcal{A}(\mathcal{M}')$ denote the distributions of the one-shot FL output, including the model and internal states, before and after editing. Model editing can be viewed as transporting probability mass from P to Q .

Retraining corresponds to a trivial transport plan that samples independently from Q , but it incurs high computational cost. Efficient editing instead exploits the overlap between P and Q by reusing existing computations. Let $\Pi(P, Q)$ denote the set of couplings between P and Q . To quantify editing cost, we follow [22] and adopt the 0-1 cost function $c(x, y) = \mathbb{1}\{x \neq y\}$, $\forall x, y \in \Omega := \mathcal{W} \times \mathcal{S}$. Let $\text{TV}(P, Q)$ denote the total variation distance between P and Q :

$$\text{TV}(P, Q) := \sup_{\text{measurable } Z} |P(Z) - Q(Z)| = \frac{1}{2} \|\phi_P - \phi_Q\|_1,$$

where ϕ_P and ϕ_Q are the probability densities of P and Q . The minimal expected editing cost is then

$$\inf_{\pi \in \Pi(P, Q)} \mathbb{E}_{(x,y) \sim \pi} [\mathbb{1}\{x \neq y\}] = \text{TV}(P, Q).$$

Thus, only an $O(\text{TV}(P, Q))$ fraction of editing requests requires actual recomputation. In practice, the future device changes are unknown at training time. When generating P , we do not know the corresponding target distribution Q . To enable efficient editing under arbitrary future changes, a one-shot FL algorithm must guarantee uniformly bounded TV distance across all possible single-device edits. This motivates the following notion of stability.

Definition 2 (ρ -TV-Stable One-Shot FL Algorithm). A one-shot FL algorithm $\mathcal{A}$ is ρ -TV-stable if for any device sets $\mathcal{M}$ and $\mathcal{M}'$ with $|\mathcal{M} \oplus \mathcal{M}'| = 1$, we have $\text{TV}(\mathcal{A}_1(\mathcal{M}), \mathcal{A}_1(\mathcal{M}')) \leq \rho$, where $\mathcal{A}_1(\cdot)$ denotes the model output of $\mathcal{A}$.

Algorithm 1: F-SOSA: Fisher-Informed Subsample One-Shot Averaging

Input: Initial model $\mathbf{W}^{(0)}$, number of local iterations K , learning rate η , stability level ρ

- 1 **Coordinator** sets $P = \rho M$ and samples a client subset $\mathcal{P} \subset [M]$ of size P uniformly at random;
- 2 **Coordinator** broadcasts $\mathbf{W}^{(0)}$ to clients;
- 3 **for every client** $i \in \mathcal{P}$ **in parallel do**
- 4 Set $\mathbf{W}_i^{(0)} \leftarrow \mathbf{W}^{(0)}$;
- 5 **for** $k \leftarrow 1, \dots, K$ **do**
- 6 $\mathbf{W}_i^{(k)} \leftarrow \mathbf{W}_i^{(k-1)} - \eta \nabla L_{\mathcal{D}_i}(\mathbf{W}_i^{(k-1)})$;
- 7 Set $\tilde{\mathbf{W}}_i \leftarrow \mathbf{W}_i^{(K)}$;
- 8 Compute the Fisher $\mathbf{F}_i \leftarrow \frac{1}{n} \sum_{j=1}^n \mathbb{E}_{\mathbf{y}} [\nabla \log \mathbb{P}(\mathbf{y} | \mathbf{x}_{ij}, \tilde{\mathbf{W}}_i) \nabla \log \mathbb{P}(\mathbf{y} | \mathbf{x}_{ij}, \tilde{\mathbf{W}}_i)^\top]$;
- 9 Communicate $\tilde{\mathbf{W}}_i$ and $\mathbf{F}_i$ to the coordinator;
- 10 **Coordinator** computes and stores $\bar{\mathbf{W}}_{\mathcal{P}}$, $\mathbf{F}_{\mathcal{P}}$ and $\mathbf{b}_{\mathcal{P}}$:
 $\bar{\mathbf{W}}_{\mathcal{P}} \leftarrow \sum_{i \in \mathcal{P}} \tilde{\mathbf{W}}_i / P$, $\mathbf{F}_{\mathcal{P}} \leftarrow \sum_{i \in \mathcal{P}} \mathbf{F}_i$, $\mathbf{b}_{\mathcal{P}} \leftarrow \sum_{i \in \mathcal{P}} \mathbf{F}_i \tilde{\mathbf{W}}_i$;
- 11 **Coordinator** solves the linear system in $\Lambda \in \mathbb{R}^d$:
 $\mathbf{F}_{\mathcal{P}}^2 \Lambda = 2(\mathbf{F}_{\mathcal{P}} \bar{\mathbf{W}}_{\mathcal{P}} - \mathbf{b}_{\mathcal{P}})$;
- 12 **Coordinator** sets the global model as
 $\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}} \leftarrow \bar{\mathbf{W}}_{\mathcal{P}} - \frac{1}{2} \mathbf{F}_{\mathcal{P}} \Lambda$;

Output: $\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}}$

Note that Definition 2 only requires stability of the marginal distribution of model output. The internal states need not be considered. For multiple device changes, suppose $|\mathcal{M} \setminus \mathcal{M}'| = k_1$ and $|\mathcal{M}' \setminus \mathcal{M}| = k_2$. By the triangle inequality of TV distance and repeated application of Definition 2, we have $\text{TV}(\mathcal{A}_1(\mathcal{M}), \mathcal{A}_1(\mathcal{M}')) \leq (k_1 + k_2)\rho$.

IV. OUR METHOD

A. TV-Stable One-Shot Federated Learning: F-SOSA

The federated learning admits a posterior-inference interpretation. Specifically, the learning objective in (1) corresponds to finding the mode of a global posterior distribution over model parameters. This viewpoint has recently been used to derive one-shot FL algorithms that aggregate local posterior information in a single communication round [5]. We adopt this probabilistic formulation as a technical starting point, but our design goal is fundamentally different. Existing posterior-based one-shot FL methods primarily optimize a single fusion operation. In contrast, our setting requires the fused model to support repeated post-training edits as the client set changes. This creates two additional requirements. First, the fusion state should admit efficient single-client addition and removal without rerunning the entire fusion procedure. Second, the output should remain stable under a single-client change so that previously computed results can be reused whenever possible. We address these requirements in two stages: we first derive a curvature-aware projection characterized by client-additive

statistics, and then introduce randomized client subsampling to control total-variation stability.

We first recall the probabilistic formulation. Following standard treatments in posterior-based FL [5], we assume the loss corresponds to a negative log-likelihood from an exponential-family model, which holds for common choices such as squared loss and cross-entropy.

Assumption 1. Given $\mathbf{z} = f(\mathbf{W}, \mathbf{x})$, $\ell(\mathbf{z}, \mathbf{y}) \propto -\log \mathbb{P}(\mathbf{y} | \mathbf{z})$ where $\mathbb{P}(\mathbf{y} | \mathbf{z}) = h(\mathbf{y}) \exp(\mathbf{z}^\top T(\mathbf{y}) - A(\mathbf{z}))$ and $h(\mathbf{y}), T(\mathbf{y}), A(\mathbf{z})$ are some real-valued functions.

Under this assumption, the likelihood of a data point $(\mathbf{x}, \mathbf{y})$ given parameters $\mathbf{W}$ can be written as

$$\begin{aligned} \mathbb{P}((\mathbf{x}, \mathbf{y}) | \mathbf{W}) &= \mathbb{P}(\mathbf{x}) \mathbb{P}(\mathbf{y} | \mathbf{x}, \mathbf{W}) \\ &\propto q(\mathbf{x}) \exp(-\ell(f(\mathbf{W}, \mathbf{x}), \mathbf{y})), \end{aligned}$$

where $q(\cdot)$ is a prior on $\mathbf{x}$ independent of $\mathbf{W}$. Under a flat prior $\mathbb{P}(\mathbf{W}) \propto 1$, the global Maximum A Posteriori (MAP) estimator coincides with the minimizer of (1) [36]. Moreover, the global posterior factorizes as

$$\mathbb{P}(\mathbf{W} | \mathcal{D}) \propto \prod_{i=1}^M \mathbb{P}(\mathbf{W} | \mathcal{D}_i).$$

This factorization suggests a one-shot protocol where each client transmits a summary of its local posterior once and the server combines these summaries to obtain a global MAP estimate. Unfortunately, an analytical expression of $\mathbb{P}(\mathbf{W} | \mathcal{D}_i)$ is generally intractable for modern models. A standard remedy is to approximate each local posterior near its mode. Let

$$\tilde{\mathbf{W}}_i \in \arg \max_{\mathbf{W}} \mathbb{P}(\mathbf{W} | \mathcal{D}_i),$$

which is obtained by local training. A second-order expansion around $\tilde{\mathbf{W}}_i$ yields a Gaussian (Laplace) approximation:

$$\log \mathbb{P}(\mathbf{W} | \mathcal{D}_i) \approx \log \mathbb{P}(\tilde{\mathbf{W}}_i | \mathcal{D}_i) - \frac{n}{2} (\mathbf{W} - \tilde{\mathbf{W}}_i)^\top \mathbf{H}_i (\mathbf{W} - \tilde{\mathbf{W}}_i),$$

where

$$\mathbf{H}_i = -\frac{1}{n} \nabla^2 \log \mathbb{P}(\mathbf{W} | \mathcal{D}_i) \big|_{\mathbf{W}=\tilde{\mathbf{W}}_i}.$$

To ensure a tractable and well-behaved curvature surrogate, $\mathbf{H}_i$ is commonly replaced by the Fisher information at $\tilde{\mathbf{W}}_i$:

$$\mathbf{F}_i = \frac{1}{n} \sum_{j=1}^n \mathbb{E}_{\mathbf{y}} [\nabla \log \mathbb{P}(\mathbf{y} | \mathbf{x}_{ij}, \tilde{\mathbf{W}}_i) \nabla \log \mathbb{P}(\mathbf{y} | \mathbf{x}_{ij}, \tilde{\mathbf{W}}_i)^\top],$$

evaluated at $\mathbf{W} = \tilde{\mathbf{W}}_i$. $\mathbf{F}_i$ is symmetric and positive semi-definite (PSD). Computing and communicating the full Fisher matrix $\mathbf{F}_i \in \mathbb{R}^{d \times d}$ can be prohibitive for large models. Our fusion formulation does not depend on a particular Fisher representation, so structured approximations can be adopted according to the available computation, storage, and communication resources. For a layer with weight matrix of size $o_\ell \times i_\ell$ and $d_\ell = o_\ell i_\ell$ parameters, Table I summarizes representative per-client Fisher representations. A diagonal approximation retains only the diagonal curvature and has linear representation cost, whereas K-FAC preserves richer layer-wise curvature through Kronecker factors while avoiding the quadratic representation cost of a full Fisher matrix.

TABLE I
STRUCTURAL COMPLEXITY OF REPRESENTATIVE PER-CLIENT FISHER
REPRESENTATIONS FOR A LAYER WITH $d_\ell = o_\ell i_\ell$ PARAMETERS.

Structure	Storage / Communication	Curvature Structure
Full	$O(d_\ell^2)$	Full
Diagonal	$O(d_\ell)$	Diagonal only
K-FAC	$O(i_\ell^2 + o_\ell^2)$	Kronecker-factorized

Thus, the choice of Fisher approximation provides a trade-off between curvature fidelity and resource overhead. Combining these local posterior approximations yields a Fisher-weighted quadratic form for the global log-posterior:

$$\log \mathbb{P}(\mathbf{W}|\mathcal{D}) \approx \sum_{i=1}^M \log \mathbb{P}(\tilde{\mathbf{W}}_i|\mathcal{D}_i) - \frac{n}{2} \sum_{i=1}^M (\mathbf{W} - \tilde{\mathbf{W}}_i)^\top \mathbf{F}_i (\mathbf{W} - \tilde{\mathbf{W}}_i).$$

Maximizing this approximation is equivalent to solving

$$\min_{\mathbf{W} \in \mathbb{R}^d} \sum_{i=1}^M (\mathbf{W} - \tilde{\mathbf{W}}_i)^\top \mathbf{F}_i (\mathbf{W} - \tilde{\mathbf{W}}_i). \quad (2)$$

Since each $\mathbf{F}_i \succeq 0$, any minimizer satisfies the first-order condition

$$\left(\sum_{i=1}^M \mathbf{F}_i \right) \mathbf{W} = \sum_{i=1}^M \mathbf{F}_i \tilde{\mathbf{W}}_i. \quad (3)$$

In over-parameterized regimes ($d \gg N$), the system in (3) can admit multiple solutions. To select a unique and well-behaved estimator, an effective selection rule is to choose the feasible solution that stays closest to the collection of local optima [5]. This leads to the following constrained least-squares formulation

$$\min_{\mathbf{W} \in \mathbb{R}^d} \left\{ \sum_{i=1}^M \|\mathbf{W} - \tilde{\mathbf{W}}_i\|_2^2 \text{ s.t. } \left(\sum_{i=1}^M \mathbf{F}_i \right) \mathbf{W} = \sum_{i=1}^M \mathbf{F}_i \tilde{\mathbf{W}}_i \right\}. \quad (4)$$

A solution to (2) can be obtained by iterative optimization [5]. Starting from the average initialization $\mathbf{W}^{[0]} = \frac{1}{M} \sum_{i=1}^M \tilde{\mathbf{W}}_i$, gradient descent with step size $\eta_S \leq 1/\lambda_{\max}$, where $\lambda_{\max}$ is the largest eigenvalue of $\sum_{i=1}^M \mathbf{F}_i$, generates iterates $\mathbf{W}^{[t]} \leftarrow \mathbf{W}^{[t-1]} - \tilde{\eta} \sum_{i=1}^M \mathbf{F}_i (\mathbf{W}^{[t-1]} - \tilde{\mathbf{W}}_i)$, which converge asymptotically to a minimizer $\mathbf{W}^*$. While such iterative fusion is suitable for one-time aggregation, it is poorly matched to repeated model editing. When a single client changes, the fusion objective changes accordingly, and the iterative procedure generally needs to be rerun to obtain the updated solution. Moreover, the previous optimization trajectory is not directly reusable for updating only the affected client contribution. This motivates the following projection formulation. Let

$$\bar{\mathbf{W}} := \frac{1}{M} \sum_{i=1}^M \tilde{\mathbf{W}}_i. \quad (5)$$

Using the identity

$$\sum_{i=1}^M \|\mathbf{W} - \tilde{\mathbf{W}}_i\|_2^2 = M \|\mathbf{W} - \bar{\mathbf{W}}\|_2^2 + \sum_{i=1}^M \|\bar{\mathbf{W}} - \tilde{\mathbf{W}}_i\|_2^2,$$

problem (4) is equivalent to

$$\min_{\mathbf{W} \in \mathbb{R}^d} \{ \|\mathbf{W} - \bar{\mathbf{W}}\|_2^2 \text{ s.t. } \mathbf{F}\mathbf{W} = \mathbf{b} \}, \quad (6)$$

where

$$\mathbf{F} := \sum_{i=1}^M \mathbf{F}_i, \quad \mathbf{b} := \sum_{i=1}^M \mathbf{F}_i \tilde{\mathbf{W}}_i. \quad (7)$$

Hence, the fusion state needed for subsequent editing is summarized by $(\bar{\mathbf{W}}, \mathbf{F}, \mathbf{b})$, each of which decomposes over clients. Under a single-client replacement, these statistics can be updated by subtracting the outgoing contribution and adding the incoming one, while all unchanged client contributions are reused. Since each $\mathbf{F}_i \succeq 0$, $\mathbf{b} = \sum_i \mathbf{F}_i \tilde{\mathbf{W}}_i \in \text{Range}(\mathbf{F})$, so the affine constraint $\mathbf{F}\mathbf{W} = \mathbf{b}$ in (6) is feasible. Thus, solving (6) reduces to computing the orthogonal projection of $\bar{\mathbf{W}}$ onto the affine set $\{\mathbf{W} : \mathbf{F}\mathbf{W} = \mathbf{b}\}$. This projection admits an explicit characterization via the KKT conditions. Consider the Lagrangian

$$\mathcal{L}(\mathbf{W}, \boldsymbol{\Lambda}) = \|\mathbf{W} - \bar{\mathbf{W}}\|_2^2 + \boldsymbol{\Lambda}^\top (\mathbf{F}\mathbf{W} - \mathbf{b}), \quad (8)$$

where $\boldsymbol{\Lambda} \in \mathbb{R}^d$ is the dual variable. Since the objective is strictly convex and the constraint is affine, the KKT conditions are necessary and sufficient for optimality. Taking derivatives of (8) yields the stationarity and primal feasibility conditions:

$$\begin{cases} 2(\mathbf{W} - \bar{\mathbf{W}}) + \mathbf{F}\boldsymbol{\Lambda} = \mathbf{0}, \\ \mathbf{F}\mathbf{W} - \mathbf{b} = \mathbf{0}. \end{cases} \quad (9)$$

From the first condition,

$$\mathbf{W} = \bar{\mathbf{W}} - \frac{1}{2} \mathbf{F}\boldsymbol{\Lambda}. \quad (10)$$

Substituting into the constraint yields a linear system in the dual variable $\boldsymbol{\Lambda}$:

$$\mathbf{F}^2 \boldsymbol{\Lambda} = 2(\mathbf{F}\bar{\mathbf{W}} - \mathbf{b}). \quad (11)$$

Therefore, model fusion reduces to solving the linear system (11). The linear system in (11) remains consistent even when the aggregated Fisher matrix $\mathbf{F}$ is sparse, singular, or low-rank. Since $\mathbf{b} \in \text{Range}(\mathbf{F})$ and $\mathbf{F}\bar{\mathbf{W}} \in \text{Range}(\mathbf{F})$, we have $\mathbf{F}\bar{\mathbf{W}} - \mathbf{b} \in \text{Range}(\mathbf{F})$. Moreover, because $\mathbf{F}$ is symmetric and positive semidefinite, $\text{Range}(\mathbf{F}^2) = \text{Range}(\mathbf{F})$. Therefore, (11) always admits a solution. Rank deficiency may make the dual solution $\boldsymbol{\Lambda}^*$ non-unique. However, if $\boldsymbol{\Lambda}_1^*$ and $\boldsymbol{\Lambda}_2^*$ are two solutions, then $\mathbf{F}^2(\boldsymbol{\Lambda}_1^* - \boldsymbol{\Lambda}_2^*) = \mathbf{0}$. For symmetric positive semidefinite $\mathbf{F}$, $\text{Null}(\mathbf{F}^2) = \text{Null}(\mathbf{F})$, and hence $\mathbf{F}(\boldsymbol{\Lambda}_1^* - \boldsymbol{\Lambda}_2^*) = \mathbf{0}$. Thus, all dual solutions yield the same $\mathbf{F}\boldsymbol{\Lambda}^*$ and therefore the same projected primal solution $\mathbf{W}^* = \bar{\mathbf{W}} - \frac{1}{2} \mathbf{F}\boldsymbol{\Lambda}^*$. Hence, rank deficiency may affect numerical conditioning and dual uniqueness, but not the existence or uniqueness of the projected primal solution. Once a solution $\boldsymbol{\Lambda}^*$ is obtained, the optimal model follows directly from (10):

$$\mathbf{W}^* = \bar{\mathbf{W}} - \frac{1}{2} \mathbf{F}\boldsymbol{\Lambda}^*. \quad (12)$$

This formulation makes clear that the fusion problem decomposes into two steps: solving a linear system defined by aggregated curvature statistics, and then recovering the projected model in closed form.

The linear system (11) in Λ can be solved efficiently and accurately by standard linear solvers. For example, conjugate gradient (CG) applies here because $\mathbf{F}^2$ is symmetric positive semidefinite and the linear system (11) is always consistent. In particular, CG minimizes the quadratic energy

$$\min_{\Lambda} \frac{1}{2} \Lambda^\top \mathbf{F}^2 \Lambda - \Lambda^\top 2(\mathbf{F} \bar{\mathbf{W}} - \mathbf{b}), \quad (13)$$

whose first-order optimality condition is exactly (11). CG terminates in at most $r = \text{rank}(\mathbf{F})$ iterations, which is typically much smaller than d in over-parameterized settings. This follows from the fact that CG minimizes the quadratic energy over an expanding Krylov subspaces and the Krylov subspace associated with the nonzero eigenspace has dimension at most $r = \text{rank}(\mathbf{F})$. That said, CG is only one possible solver; any method capable of solving (11) can be used within this projection-based fusion framework. The practical cost of the projection step also depends on the Fisher representation. A diagonal approximation incurs the lowest structural overhead, while K-FAC retains richer curvature information and substantially reduces representation and communication costs relative to a full Fisher matrix. However, a sum of client-wise K-FAC matrices does not generally preserve a single Kronecker factorization. Since our coordinator stores only the aggregated curvature statistic rather than individual K-FAC factors, this structure may be lost after aggregation, and the coordinator-side projection cost can approach that of a less structured representation. Hence, the Fisher approximation can be selected according to the desired trade-off among curvature fidelity, communication overhead, and coordinator-side computation.

To enable efficient exact model editing, the one-shot FL algorithm must be ρ -TV-stable (Definition 2) for a pre-specified stability level ρ . The fusion rule above uses all available clients and is deterministic. This benefits model utility, but it yields no distributional stability. In particular, for any neighboring device sets $\mathcal{M}$ and $\mathcal{M}'$ with $|\mathcal{M} \oplus \mathcal{M}'| = 1$, the outputs $\mathcal{A}_1(\mathcal{M})$ and $\mathcal{A}_1(\mathcal{M}')$ are Dirac measures supported at different points. Hence, $\text{TV}(\mathcal{A}_1(\mathcal{M}), \mathcal{A}_1(\mathcal{M}')) = 1$, which cannot meet an arbitrary target $\rho < 1$. We address this by introducing randomized client subsampling for model aggregation. The coordinator samples a client subset $\mathcal{P} \subseteq [M]$ of size P for local training and model fusion. This shared randomness induces overlap between the output distributions under neighboring client sets, which is directly controlled by the sampling rate ρ . At the initial execution of F-SOSA with M clients, the subsample size P is chosen as ρM and is fixed thereafter. Note that, for the general dynamic-device learning setting considered in this work, we assume that client contributions have comparable aggregation priority, under which uniform random subsampling is appropriate. In some safety-critical applications, such as autonomous driving, certain client updates may need to be incorporated with higher priority. These clients can be treated as mandatory participants rather than being randomly subsampled, even at the expense of reduced editing efficiency. Extending the stability analysis to priority-aware participation is beyond the current scope.

Combining Fisher-informed projection with client subsampling yields our ρ -TV-stable one-shot FL algorithm, F-

Algorithm 2: F-SOSA-U

Input: Unlearning request from client i^- with its local contribution $(\tilde{\mathbf{W}}_{i^-}, \mathbf{F}_{i^-})$, initial model $\mathbf{W}^{(0)}$, number of local iterations K , learning rate η

- 1 **if** $i^- \in \mathcal{P}$ **then**
- 2 **Coordinator** samples $i^+ \sim \text{Uniform}([M] \setminus \mathcal{P})$;
- 3 **Coordinator** updates $\mathcal{P} \leftarrow \mathcal{P} \setminus \{i^-\} \cup \{i^+\}$;
- 4 **Coordinator** communicates $\mathbf{W}^{(0)}$ to client i^+ ;
- 5 **Client** i^+ performs local training and Fisher computation following Lines 4–9 of Algorithm 1, and sends $(\tilde{\mathbf{W}}_{i^+}, \mathbf{F}_{i^+})$ to the coordinator;
- 6 **Coordinator** updates $\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{F}_{\mathcal{P}}$ and $\mathbf{b}_{\mathcal{P}}$:
$$\bar{\mathbf{W}}_{\mathcal{P}} \leftarrow \bar{\mathbf{W}}_{\mathcal{P}} + \frac{1}{P} (\tilde{\mathbf{W}}_{i^+} - \tilde{\mathbf{W}}_{i^-}),$$

$$\mathbf{F}_{\mathcal{P}} \leftarrow \mathbf{F}_{\mathcal{P}} + (\mathbf{F}_{i^+} - \mathbf{F}_{i^-}),$$

$$\mathbf{b}_{\mathcal{P}} \leftarrow \mathbf{b}_{\mathcal{P}} + (\mathbf{F}_{i^+} \tilde{\mathbf{W}}_{i^+} - \mathbf{F}_{i^-} \tilde{\mathbf{W}}_{i^-});$$
- 7 **Coordinator** solves the linear system for $\Lambda \in \mathbb{R}^d$:
$$\mathbf{F}_{\mathcal{P}}^2 \Lambda = 2(\mathbf{F}_{\mathcal{P}} \bar{\mathbf{W}}_{\mathcal{P}} - \mathbf{b}_{\mathcal{P}});$$
- 8 **Coordinator** sets the updated global model:
$$\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}} \leftarrow \bar{\mathbf{W}}_{\mathcal{P}} - \frac{1}{2} \mathbf{F}_{\mathcal{P}} \Lambda;$$

Output: $\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}}$

SOSA (Fisher-Informed Subsample One-Shot Averaging). Algorithm 1 summarizes the procedure. The coordinator samples $\mathcal{P}$ (step 1) and broadcasts the initialization $\mathbf{W}^{(0)}$ (step 2). Here, $P = \rho M$ is determined at the initial execution of F-SOSA and remains fixed during all subsequent editing operations. Each client $i \in \mathcal{P}$ runs K steps of local gradient descent to obtain $\tilde{\mathbf{W}}_i$ (steps 3–7), computes its Fisher matrix $\mathbf{F}_i$ at $\tilde{\mathbf{W}}_i$ (step 8), and uploads $(\tilde{\mathbf{W}}_i, \mathbf{F}_i)$ to the coordinator (step 9). The coordinator then forms the sufficient statistics

$$\bar{\mathbf{W}}_{\mathcal{P}} = \frac{1}{P} \sum_{i \in \mathcal{P}} \tilde{\mathbf{W}}_i, \quad \mathbf{F}_{\mathcal{P}} = \sum_{i \in \mathcal{P}} \mathbf{F}_i, \quad \mathbf{b}_{\mathcal{P}} = \sum_{i \in \mathcal{P}} \mathbf{F}_i \tilde{\mathbf{W}}_i$$

as shown in step 10. Finally, it outputs the global model by projecting $\bar{\mathbf{W}}_{\mathcal{P}}$ onto the affine set $\{\mathbf{W} : \mathbf{F}_{\mathcal{P}} \mathbf{W} = \mathbf{b}_{\mathcal{P}}\}$. This projection is computed by solving the linear system for the dual variable Λ (steps 11) and then recovering the final fused model $\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}}$ (step 12).

B. Efficient Editing Algorithms: F-SOSA-U and F-SOSA-I

F-SOSA achieves ρ -TV stability by aggregating over a uniformly sampled subset $\mathcal{P}$ of size $P = \rho M$. When the client population changes, the support of the uniform size- P subset distribution changes accordingly. Exact model editing therefore reduces to constructing a coupling between the pre- and post-change sampling procedures.

The key observation is that editing depends on the probability mass of the realized subset under the updated sampling distribution. If the realized $\mathcal{P}$ has the same probability under the post-change distribution as under the pre-change one, then we can keep $\mathcal{P}$ unchanged. In this case, the sufficient statistics $(\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{F}_{\mathcal{P}}, \mathbf{b}_{\mathcal{P}})$ and the fused model remain distributionally

Algorithm 3: F-SOSA-I

Input: new client index i^+

1 **if** Bernoulli($\frac{P}{M+1}$) **then**

2 **Coordinator** samples $i^- \sim \text{Uniform}(\mathcal{P})$;

3 **Coordinator** requests $(\tilde{\mathbf{W}}_{i^-}, \mathbf{F}_{i^-})$ from client i^-
 and $(\tilde{\mathbf{W}}_{i^+}, \mathbf{F}_{i^+})$ from client i^+ ;

4 **Coordinator** updates $\mathcal{P} \leftarrow \mathcal{P} \setminus \{i^-\} \cup \{i^+\}$;

5 **Coordinator** updates $\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{F}_{\mathcal{P}}$ and $\mathbf{b}_{\mathcal{P}}$:

$$\bar{\mathbf{W}}_{\mathcal{P}} \leftarrow \bar{\mathbf{W}}_{\mathcal{P}} + \frac{1}{P}(\tilde{\mathbf{W}}_{i^+} - \tilde{\mathbf{W}}_{i^-}),$$

$$\mathbf{F}_{\mathcal{P}} \leftarrow \mathbf{F}_{\mathcal{P}} + (\mathbf{F}_{i^+} - \mathbf{F}_{i^-}),$$

$$\mathbf{b}_{\mathcal{P}} \leftarrow \mathbf{b}_{\mathcal{P}} + (\mathbf{F}_{i^+} \tilde{\mathbf{W}}_{i^+} - \mathbf{F}_{i^-} \tilde{\mathbf{W}}_{i^-});$$

6 **Coordinator** solves the linear system for $\mathbf{\Lambda} \in \mathbb{R}^d$:
 $\mathbf{F}_{\mathcal{P}}^2 \mathbf{\Lambda} = 2(\mathbf{F}_{\mathcal{P}} \bar{\mathbf{W}}_{\mathcal{P}} - \mathbf{b}_{\mathcal{P}})$;

7 **Coordinator** updates the global model:
 $\bar{\mathbf{W}}_{\mathcal{P}}^{\mathbf{F}} \leftarrow \bar{\mathbf{W}}_{\mathcal{P}} - \frac{1}{2} \mathbf{F}_{\mathcal{P}} \mathbf{\Lambda}$;

Output: $\bar{\mathbf{W}}_{\mathcal{P}}^{\mathbf{F}}$

consistent with retraining, so no editing is required. Otherwise, we update $\mathcal{P}$ in a way that restores the correct post-change sampling distribution and then recompute the projection-based fusion using updated sufficient statistics. Thus, we construct a maximal coupling between the two uniform subset distributions for efficient editing. As shown later, under the maximal coupling, the probability that an update is needed equals the TV distance between the two sampling distributions, and is at most ρ . Based on this principle, we develop two exact editing procedures: F-SOSA-U for client unlearning (Algorithm 2) and F-SOSA-I for client incremental learning (Algorithm 3). The coordinator persistently maintains only the sampled subset $\mathcal{P}$ and the aggregate statistics $(\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{F}_{\mathcal{P}}, \mathbf{b}_{\mathcal{P}})$. Individual client contributions $(\tilde{\mathbf{W}}_i, \mathbf{F}_i)$ need not be retained at the coordinator. While connected, each active client keeps its own contribution and transmits it only when required by an editing operation; in particular, a departing client sends its contribution together with the unlearning request before disconnecting. The coordinator then updates the aggregate statistics by subtracting the departing contribution and adding the incoming one. Hence, its persistent state does not grow with the number of clients that have participated historically.

- **Client unlearning (F-SOSA-U).** When the coordinator receives an unlearning request from client i^- , the coordinator first checks whether it belongs to the sampled subset $\mathcal{P}$. If $i^- \notin \mathcal{P}$, then under the maximal coupling the subset remains unchanged, and no editing is needed. Otherwise, we must replace the removed client to preserve the subset size and sampling distribution. The coordinator therefore draws a replacement client i^+ uniformly from $[M] \setminus \mathcal{P}$ and updates $\mathcal{P} \leftarrow \mathcal{P} \setminus \{i^-\} \cup \{i^+\}$. The new client performs local training and Fisher computation, after which the coordinator updates the sufficient statistics via rank-one adjustments: $\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{F}_{\mathcal{P}}, \mathbf{b}_{\mathcal{P}}$. The edited global model is then obtained by solving the same projection system as in F-SOSA. Since

only one client contribution changes, the update cost is independent of the total number of clients.

- **Client incremental learning (F-SOSA-I).** When a new client i^+ joins, we determine whether it would have been selected under the updated F-SOSA sampling distribution. Conditional on the current client set in the learning objective containing $M+1$ clients, F-SOSA samples a size- P subset uniformly at random. Hence, each client in the updated set, including the newly arrived client i^+ , has inclusion probability $\Pr(i^+ \in \mathcal{P}') = P/(M+1)$. This probability is determined by the conditional subset-sampling rule and is independent of the external arrival distribution. If the Bernoulli trial fails, the sampled subset remains unchanged under the coupling. Otherwise, the coordinator replaces a uniformly chosen client $i^- \in \mathcal{P}$ with i^+ and updates the sufficient statistics using the same adjustments as in unlearning.

Both editing algorithms modify only a single selected-client contribution while reusing the current aggregate fusion statistics. Consequently, each triggered editing operation updates only one client contribution to the sufficient statistics and performs one projection solve. Its coordinator-side cost depends on the chosen Fisher representation, as discussed above, but does not require recomputing the contributions of all previously sampled clients. Moreover, since editing is triggered with probability at most $O(\rho)$, the expected editing cost per client change is $O(\rho)$ relative to full retraining. Together, these procedures enable provably exact and communication-efficient model adaptation to dynamic client participation in one-shot federated learning.

Finally, we demonstrate that, the proposed editing procedures extend naturally to scenarios where multiple client changes occur, either simultaneously or sequentially.

- **Simultaneous requests.** When multiple unlearning and incremental requests arrive simultaneously, we process them using the batch editing procedure shown in Algorithm 4. The coordinator first performs the same verification steps as in F-SOSA-U and F-SOSA-I to determine which requests affect the realized subset $\mathcal{P}$. For unlearning requests, this reduces to checking membership in $\mathcal{P}$; for incremental requests, this is determined by the Bernoulli inclusion test. Requests that do not affect $\mathcal{P}$ are applied directly to the client set without modifying the subset or sufficient statistics. The remaining requests are then processed sequentially to update $\mathcal{P}$ under the correct post-change sampling distribution. After the final subset $\mathcal{P}'$ is obtained, the sufficient statistics and fused model are updated in a single batch using aggregated rank-one adjustments. By the triangle inequality of total variation distance and the ρ -TV stability of F-SOSA, if a batch contains k client changes, the probability that editing is triggered is at most $k\rho$. Since all affected contributions are incorporated through a single projection solve, the expected total editing cost scales as $O(k\rho)$ relative to retraining. Consequently, the expected amortized overhead per client change remains $O(\rho)$.
- **Sequential requests.** When client changes arrive over time, we simply apply F-SOSA-U and F-SOSA-I iteratively. After

each edit, the updated subset $\mathcal{P}$ and sufficient statistics $(\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{F}_{\mathcal{P}}, \mathbf{b}_{\mathcal{P}})$ remain valid summaries of the current client set. Because exact editing restores the joint distribution of both the model and these internal states to that of retraining, each subsequent request starts from a statistically valid retraining-equivalent state. Hence, approximation errors do not propagate across successive edits. Moreover, by the triangle inequality of TV distance, the expected number of triggered updates over a sequence of edits grows linearly with the number of client changes, while the expected fraction remains $O(\rho)$, so does the expected editing cost relative to full retraining.

Overall, F-SOSA enables exact adaptation to both batched or streaming client dynamics, while ensuring that the expected editing cost per change is a $O(\rho)$ -fraction of retraining.

V. THEORETICAL RESULTS

A. Stability of F-SOSA

We consider a general device set $\mathcal{M}$ with $|\mathcal{M}| = M$, and note that all results apply to any round t with device set $\mathcal{M}(t)$. We first prove that F-SOSA is ρ -TV-stable for any given stability level ρ , and then analyze its learning utility. Our stability analysis uses the concept of the push-forward measure: for a measurable map $f : \mathcal{X} \rightarrow \mathcal{Y}$ and a measure μ on $\mathcal{X}$, the push-forward $f\#\mu$ is defined by $(f\#\mu)(\mathcal{Z}) = \mu(f^{-1}(\mathcal{Z}))$ for any measurable set $\mathcal{Z} \subseteq \mathcal{Y}$.

Theorem 1. F-SOSA is ρ -TV-stable for any $\rho \in (0, 1)$.

Proof. Let $\mathcal{M}$ be original device set and $\mathcal{M}'$ be the update device set such that $|\mathcal{M} \oplus \mathcal{M}'| = 1$. We first consider $\mathcal{M}$ contains $M-1$ devices, where w.l.o.g. we assume device M leaves, i.e., $\mathcal{M} = \{1, 2, \dots, M\}$ and $\mathcal{M}' = \{1, 2, \dots, M-1\}$. Let the sigma-algebra on these sets be the power sets of $\mathcal{M}$ and $\mathcal{M}'$ respectively, denoted by $\text{Pow}(\mathcal{M})$ and $\text{Pow}(\mathcal{M}')$ respectively. Let $\mu_{(P,M)}$ be the sub-sampling probability measure on M devices in $\mathcal{M}$, i.e., sub-samples a subset $\mathcal{P}$ of size P out of M elements in $\mathcal{M}$ uniformly at random. Similarly we can define $\mu_{(P,M-1)}$ for $\mathcal{M}'$. We first extend the sigma-algebra for the probability spaces so that the random variables $\mu_{(P,M)}$ and $\mu_{(P,M-1)}$ are defined on a common probability space. For this, we will just add an event where the device M can be sampled under $\mu_{(P,M-1)}$ with probability 0. We define $\mu'_{(P,M)}$ as follows: for any set $\mathcal{P} \in \text{Pow}(\mathcal{M})$, $\mu'_{(P,M)}(\mathcal{P}) := \begin{cases} \mu_{(P,M-1)}(\mathcal{P}), & \text{if } M \notin \mathcal{P} \\ 0, & \text{otherwise} \end{cases}$. Note that, for fixed initialization $\mathbf{W}^{(0)}$ and other parameters, Algorithm $\mathcal{A}(\mathcal{M})$ and $\mathcal{A}(\mathcal{M}')$ is the same deterministic map from $\mathcal{P} \in [M]^P$ to $\mathcal{W}$. They only differ because of different measure on the input space. Hence TV distance between $\mathcal{A}(\mathcal{M})$ and $\mathcal{A}(\mathcal{M}')$ is just the TV distance between the push-forward measures $\mathcal{A}(\mathcal{M})\#\mu_{(P,M)}$ and $\mathcal{A}(\mathcal{M}')\#\mu'_{(P,M)}$, which by using the fact that $\mathcal{A}(\mathcal{M}) \equiv \mathcal{A}(\mathcal{M}')$ and data processing inequality is at most the TV distance between $\mu_{(P,M)}$ and $\mu'_{(P,M)}$. Therefore,

$$\begin{aligned} \text{TV}(\mathcal{A}(\mathcal{M}), \mathcal{A}(\mathcal{M}')) &\leq \text{TV}(\mu_{(P,M)}, \mu'_{(P,M)}) \\ &\leq \mu_{(P,M)}(S \text{ containing } M) = \binom{M-1}{P-1} / \binom{M}{P} = P/M. \end{aligned}$$

Algorithm 4: F-SOSA-Batch: Batch Editing for Mixed Unlearning and Incremental Requests

Input: request list $\mathcal{R}$

- 1 **Coordinator** initializes queue $\mathcal{Q} \leftarrow []$;
- 2 **Coordinator** **foreach** request $r \in \mathcal{R}$ (in order) **do**
- 3 **if** $r = \text{Unlearn}(i^-)$ **then**
- 4 **if** $i^- \in \mathcal{P}$ **then**
- 5 Append r to $\mathcal{Q}$;
- 6 **else**
- 7 Update $\mathcal{M} \leftarrow \mathcal{M} \setminus \{i^-\}$;
- 8 **else if** $r = \text{Insert}(i^+)$ **then**
- 9 **if** $\text{Bernoulli}(\frac{P}{|\mathcal{M}|+1})$ **then**
- 10 Append r to $\mathcal{Q}$;
- 11 **else**
- 12 Update $\mathcal{M} \leftarrow \mathcal{M} \cup \{i^+\}$;
- 13 **Coordinator** stores $\mathcal{P}_{\text{old}} \leftarrow \mathcal{P}$;
- 14 **Coordinator** **foreach** request $r \in \mathcal{Q}$ (in order) **do**
- 15 **if** $r = \text{Unlearn}(i^-)$ **then**
- 16 Sample $j \sim \text{Uniform}(\mathcal{M} \setminus \mathcal{P})$;
- 17 **if** $(\bar{\mathbf{W}}_j, \mathbf{F}_j)$ not available **then**
- 18 **Coordinator** sends $\mathbf{W}^{(0)}$ to client j ;
- 19 **Client** j executes Lines 4–9 of Algorithm 1
- 20 and sends $(\bar{\mathbf{W}}_j, \mathbf{F}_j)$ to the coordinator;
- 21 Update $\mathcal{P} \leftarrow \mathcal{P} \setminus \{i^-\} \cup \{j\}$;
- 22 Update $\mathcal{M} \leftarrow \mathcal{M} \setminus \{i^-\}$;
- 23 **else if** $r = \text{Insert}(i^+)$ **then**
- 24 **if** $(\bar{\mathbf{W}}_{i^+}, \mathbf{F}_{i^+})$ not available **then**
- 25 **Coordinator** sends $\mathbf{W}^{(0)}$ to client i^+ ;
- 26 **Client** i^+ executes Lines 4–9 of
- 27 Algorithm 1 and sends $(\bar{\mathbf{W}}_{i^+}, \mathbf{F}_{i^+})$ to the
- 28 coordinator;
- 29 Sample $i^- \sim \text{Uniform}(\mathcal{P})$;
- 30 Update $\mathcal{P} \leftarrow \mathcal{P} \setminus \{i^-\} \cup \{i^+\}$;
- 31 Update $\mathcal{M} \leftarrow \mathcal{M} \cup \{i^+\}$;
- 32 **Coordinator** sets $\mathcal{P}' \leftarrow \mathcal{P}$;
- 33 **Coordinator** sets $S_- \leftarrow \mathcal{P}_{\text{old}} \setminus \mathcal{P}'$ and $S_+ \leftarrow \mathcal{P}' \setminus \mathcal{P}_{\text{old}}$;
- 34 **Coordinator** updates statistics:

$$\bar{\mathbf{W}}_{\mathcal{P}'} \leftarrow \bar{\mathbf{W}}_{\mathcal{P}} + \frac{1}{P} \left(\sum_{i \in S_+} \tilde{\mathbf{W}}_i - \sum_{i \in S_-} \tilde{\mathbf{W}}_i \right),$$

$$\mathbf{F}_{\mathcal{P}'} \leftarrow \mathbf{F}_{\mathcal{P}} + \left(\sum_{i \in S_+} \mathbf{F}_i - \sum_{i \in S_-} \mathbf{F}_i \right),$$

$$\mathbf{b}_{\mathcal{P}'} \leftarrow \mathbf{b}_{\mathcal{P}} + \left(\sum_{i \in S_+} \mathbf{F}_i \tilde{\mathbf{W}}_i - \sum_{i \in S_-} \mathbf{F}_i \tilde{\mathbf{W}}_i \right).$$

35 **Coordinator** solves the linear system for $\mathbf{\Lambda} \in \mathbb{R}^d$:

$$\mathbf{F}_{\mathcal{P}'}^2 \mathbf{\Lambda} = 2(\mathbf{F}_{\mathcal{P}'} \bar{\mathbf{W}}_{\mathcal{P}'} - \mathbf{b}_{\mathcal{P}'}), \text{ and set}$$

$$\bar{\mathbf{W}}_{\mathcal{P}'}^{\mathbf{F}} \leftarrow \bar{\mathbf{W}}_{\mathcal{P}'} - \frac{1}{2} \mathbf{F}_{\mathcal{P}'} \mathbf{\Lambda}.$$

Output: $\bar{\mathbf{W}}_{\mathcal{P}'}^{\mathbf{F}}$

A similar argument works when $\mathcal{M}' = \{1, 2, \dots, M, M+1\}$, yielding a TV bound of $\frac{P}{M+1} \leq \frac{P}{M}$. Taking a uniform bound over all neighbouring set $\mathcal{M}'$, we get that $\sup_{|\mathcal{M} \oplus \mathcal{M}'|=1} \text{TV}(\mathcal{A}(\mathcal{M}), \mathcal{A}(\mathcal{M}')) \leq P/M$. Since in F-SOSA, P is set to ρM , thus $\text{TV}(\mathcal{A}(\mathcal{M}), \mathcal{A}(\mathcal{M}')) \leq \rho$. $\square$

B. Learning Utility of F-SOSA

We analyze the utility guarantee of $\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}}$ obtained by F-SOSA under the following standard data assumptions:

Assumption 2 (Data Normalization). For any $(\mathbf{x}, y) \in \mathcal{D}$, $\|\mathbf{x}\|_2 = 1$ and $|y| \leq C$ for some constant $C > 0$.

Assumption 3 (Data Uniqueness). Any two samples $(\mathbf{x}, y)$ and $(\mathbf{x}', y')$ in $\mathcal{D}$ satisfy $\|\mathbf{x} - \mathbf{x}'\|_2 \geq \phi > 0$.

Assumption 4 (Data Non-Degeneracy). The minimum eigenvalue of the Gram matrix $\mathbf{H}^\infty \in \mathbb{R}^{N \times N}$ defined by $(\mathbf{H}^\infty)_{k,l} = \mathbb{E}_{\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} [\mathbf{x}_k^\top \mathbf{x}_l \mathbb{1}\{\mathbf{w}^\top \mathbf{x}_k \geq 0\} \mathbb{1}\{\mathbf{w}^\top \mathbf{x}_l \geq 0\}]$, is bounded below: $\lambda_{\min}(\mathbf{H}^\infty) = \lambda_0 > 0$.

No bounded heterogeneity assumption is made, so our results hold under arbitrary non-i.i.d. client data. We consider a two-layer ReLU neural network (NN) where the first-layer consists of m trainable weights $\{\mathbf{w}_r\}_{r=1}^m$ and the second-layer weights $\mathbf{a} = \{a_r\}_{r=1}^m$ are drawn i.i.d. from $\{+1, -1\}$ at random and then fixed [5], [37]. Let $\mathbf{W} = \text{vec}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m)$. The NN is modeled as

$$f_{\mathbf{a}}(\mathbf{W}, \mathbf{x}) = \frac{1}{\sqrt{m}} \sum_{r=1}^m a_r \gamma(\mathbf{x}^\top \mathbf{w}_r), \quad (14)$$

where $\gamma(x) = \max\{x, 0\}$ is the ReLU activation function. This model can be equivalently written in a linearized form:

$$f_{\mathbf{a}}(\mathbf{W}, \mathbf{x}) = \phi_{\mathbf{a}}(\mathbf{W}, \mathbf{x})^\top \mathbf{W}, \quad (15)$$

where

$$\phi_{\mathbf{a}}(\mathbf{W}, \mathbf{x}) := \frac{1}{\sqrt{m}} \begin{bmatrix} a_1 \mathbf{x} \mathbb{1}\{\mathbf{x}^\top \mathbf{w}_1 \geq 0\} \\ a_2 \mathbf{x} \mathbb{1}\{\mathbf{x}^\top \mathbf{w}_2 \geq 0\} \\ \vdots \\ a_m \mathbf{x} \mathbb{1}\{\mathbf{x}^\top \mathbf{w}_m \geq 0\} \end{bmatrix} \in \mathbb{R}^{mp}. \quad (16)$$

The neural-network objective is non-convex. We analyze it using the two-layer ReLU model, a standard canonical setting for NTK-style analysis of over-parameterized neural networks, where the training dynamics can be characterized through activation stability and the associated Gram matrix. In this regime, gradient descent remains within a controlled neighborhood of the random initialization, the relevant Gram matrices remain well conditioned, and the empirical fitting error decreases geometrically with the number of local iterations.

We first record a basic bound on the feature representation that will be used repeatedly in the subsequent analysis. It follows directly from the definition of $\phi_{\mathbf{a}}(\mathbf{W}, \mathbf{x})$, since $a_r \in \{-1, +1\}$ and each activation indicator is at most one.

Lemma 1. For $\forall \mathbf{W} \in \mathbb{R}^d$ and $\|\mathbf{x}\|_2 \leq 1$, $\|\phi_{\mathbf{a}}(\mathbf{W}, \mathbf{x})\|_2 \leq 1$.

We initialize $\mathbf{w}_r^{(0)} \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$. To quantify activation pattern stability, given a real $R \geq 0$, define event $\mathcal{E}_{ijr}(R) :=$

$\{\exists \mathbf{w} : \|\mathbf{w} - \mathbf{w}_r^{(0)}\|_2 \leq R, \mathbb{1}\{\mathbf{x}_{ij}^\top \mathbf{w}_r^{(0)} \geq 0\} \neq \mathbb{1}\{\mathbf{x}_{ij}^\top \mathbf{w} \geq 0\}\}$. We next characterize the stability of ReLU activation patterns around the random initialization.

Lemma 2 ([5, Lemma 3]). The random initialization of $\mathbf{w}_r^{(0)} \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$ ensures that $\mathbb{P}(\mathcal{E}_{ijr}(R)) \leq 2R/(\sqrt{2\pi}\sigma)$.

We then invoke a standard over-parameterized training result to control both the local fitting error and the distance traveled by each local model.

Lemma 3 ([5, Lemma 2]). Let $m = \Omega(\frac{N^6}{\lambda_0^6 \sigma^2 \delta^3})$ and $\eta = O(\frac{\lambda_0}{N^2})$. With probability at least $1 - \delta$ over the initialization of $\mathbf{W}^{(0)}$ and $\mathbf{a}$, we have simultaneously for all $i \in [M]$:

- 1) $\sum_{j=1}^n (y_{ij} - f_{\mathbf{a}}(\mathbf{W}^{(0)}, \mathbf{x}_{ij}))^2 = O(\frac{N}{\delta})$;
- 2) $\sum_{j=1}^n (y_{ij} - f_{\mathbf{a}}(\tilde{\mathbf{W}}_i, \mathbf{x}_{ij}))^2 \leq (1 - \frac{\eta \lambda_0}{2})^K \cdot O(\frac{N}{\delta})$;
- 3) For $\forall r \in [m]$, $\|\tilde{\mathbf{w}}_{i,r} - \mathbf{w}_r^{(0)}\|_2 \leq R_0$, where $R_0 = \frac{4}{\lambda_0} \sqrt{\frac{N}{m}} (1 - (1 - \frac{\eta \lambda_0}{4})^K) \sqrt{\sum_{j=1}^n (y_{ij} - f_{\mathbf{a}}(\mathbf{W}^{(0)}, \mathbf{x}_{ij}))^2}$.

Combining 1) and 3), we have $R_0 \leq O(\frac{N}{\lambda_0 \sqrt{\delta m}})$.

Building on these local training properties, we next bound the deviations of the trained local models and their subsampled average from the common initialization. Proofs of all auxiliary lemmas established in this paper are deferred to Appendix A.

Lemma 4. Let $m = \frac{\text{poly}(N, \delta^{-1}, \lambda_0^{-1})}{\sigma^2}$, $\eta = O(\frac{\lambda_0}{N^2})$, and $\sigma = O(\frac{\lambda_0 \delta}{N})$. With probability at least $1 - \delta$ over random initialization:

- 1) for all $i \in [M]$, $\|\tilde{\mathbf{W}}_i - \mathbf{W}^{(0)}\|_2 \leq \sqrt{\mathbf{y}_i^\top (\mathbf{H}_i^\infty)^{-1} \mathbf{y}_i} + O(\frac{N\sigma}{\lambda_0 \delta}) + \frac{\text{poly}(N, \lambda_0^{-1}, \delta^{-1})}{m^{1/4} \sigma^{1/2}}$;
- 2) $\|\bar{\mathbf{W}}_{\mathcal{P}} - \mathbf{W}^{(0)}\|_2 \leq \sqrt{\frac{2\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}}{\rho}} + O(\frac{N\sigma}{\lambda_0 \delta}) + \frac{\text{poly}(N, \lambda_0^{-1}, \delta^{-1})}{m^{1/4} \sigma^{1/2}}$.

These parameter-space bounds further allow us to control the prediction discrepancy between the averaged model and the collection of local models. Let $\mathbf{y} = \text{vec}(\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_M) \in \mathbb{R}^N$ be the true label vector and $\tilde{\mathbf{y}} = \text{vec}(\tilde{\mathbf{y}}_1, \tilde{\mathbf{y}}_2, \dots, \tilde{\mathbf{y}}_M) = [f_{\mathbf{a}}(\tilde{\mathbf{W}}_1, \mathbf{x}_{11}), f_{\mathbf{a}}(\tilde{\mathbf{W}}_1, \mathbf{x}_{12}), \dots, f_{\mathbf{a}}(\tilde{\mathbf{W}}_M, \mathbf{x}_{Mn})]^\top \in \mathbb{R}^N$ be the predictions of trained local models. We define the proxy output $\tilde{f}(\mathbf{W}, \mathbf{x}_{ij}) = \phi_{\mathbf{a}}(\tilde{\mathbf{W}}_i, \mathbf{x}_{ij})^\top \mathbf{W}$ for $\forall \mathbf{W} \in \mathbb{R}^d$ and $\mathbf{x}_{ij}$.

Lemma 5. Let $m = \text{poly}(N, \delta^{-1}, \lambda_0^{-1})/\sigma^2$, $\eta = O(\lambda_0/N^2)$, and $\sigma = O(\lambda_0 \delta/N)$. With probability at least $1 - \delta$ over the random initialization, we have

$$\|\tilde{f}(\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{x}_{ij}) - \tilde{\mathbf{y}}\|_2^2 \leq O\left(\frac{N}{\rho} \mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}\right).$$

We define four matrices associated with the initial model $\mathbf{W}^{(0)}$ and participant set $\mathcal{P} = \{i_1, \dots, i_P\}$: $\mathbf{A}_{\mathcal{P}}^{(0)}, \tilde{\mathbf{A}}_{\mathcal{P}} \in \mathbb{R}^{(Pn) \times (mp)}$ and $\mathbf{H}_{\mathcal{P}}^{(0)}, \tilde{\mathbf{H}}_{\mathcal{P}} \in \mathbb{R}^{(Pn) \times (Pn)}$:

$$\mathbf{A}_{\mathcal{P}}^{(0)} := \begin{bmatrix} \phi_{\mathbf{a}}(\mathbf{W}^{(0)}, \mathbf{x}_{i_1 1})^\top \\ \phi_{\mathbf{a}}(\mathbf{W}^{(0)}, \mathbf{x}_{i_1 2})^\top \\ \vdots \\ \phi_{\mathbf{a}}(\mathbf{W}^{(0)}, \mathbf{x}_{i_P n})^\top \end{bmatrix}, \quad \tilde{\mathbf{A}}_{\mathcal{P}} := \begin{bmatrix} \phi_{\mathbf{a}}(\tilde{\mathbf{W}}_1, \mathbf{x}_{i_1 1})^\top \\ \phi_{\mathbf{a}}(\tilde{\mathbf{W}}_1, \mathbf{x}_{i_1 2})^\top \\ \vdots \\ \phi_{\mathbf{a}}(\tilde{\mathbf{W}}_M, \mathbf{x}_{i_P n})^\top \end{bmatrix},$$

$$\mathbf{H}_{\mathcal{P}}^{(0)} := \mathbf{A}_{\mathcal{P}}^{(0)} \mathbf{A}_{\mathcal{P}}^{(0)\top}, \quad \tilde{\mathbf{H}}_{\mathcal{P}} := \tilde{\mathbf{A}}_{\mathcal{P}} \tilde{\mathbf{A}}_{\mathcal{P}}^\top.$$

We next study the conditioning of the Gram matrices associated with the sampled clients, which is needed to control the projection step.

Lemma 6. Let $\mathbf{H}_P^\infty \in \mathbb{R}^{(Pn) \times (Pn)}$ be the principal submatrix of $\mathbf{H}^\infty$ formed by $\mathcal{P}$. If $m = \Omega(\frac{(\rho N)^2 \log(\rho N/\delta)}{\lambda_0^2})$, with probability at least $1 - \delta$, $\|\mathbf{H}_P^{(0)} - \mathbf{H}_P^\infty\| \leq \lambda_0/4$, $\lambda_{\min}(\mathbf{H}_P^{(0)}) \geq 3\lambda_0/4$.

Lemma 7. If $m = \Omega(\frac{N^6}{\lambda_0^4 \sigma^2 \delta^3})$, then with probability at least $1 - \delta$, we have $\lambda_{\min}(\tilde{\mathbf{H}}_P) \geq \lambda_0/2$ and $\lambda_{\max}(\tilde{\mathbf{H}}_P) \leq \rho N$.

With these spectral bounds in place, we can quantify the additional displacement introduced by Fisher-based projection.

Lemma 8. Let $m \geq \frac{\sigma^2}{\text{poly}(\rho, N, \delta^{-1}, \lambda_0^{-1})}$, $\sigma = O(\frac{\lambda_0 \delta}{N})$. With probability at least $1 - \delta$ over the initialization, we have

$$\|\bar{\mathbf{W}}_P^F - \bar{\mathbf{W}}_P\|_2 = O\left(\sqrt{\frac{N}{\rho \lambda_0}} \mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}\right).$$

Lemma 9. Let $m = \text{poly}(N, \delta^{-1}, \lambda_0^{-1})/\sigma^2$, $\eta = O(\lambda_0/N^2)$, and $\sigma = O(\lambda_0 \delta/N)$. With probability at least $1 - \delta$ over the random initialization, we have

$$\|\bar{\mathbf{w}}_{P,r}^F - \bar{\mathbf{w}}_{P,r}\|_2 \leq R_1 \leq O\left(\frac{N}{\sqrt{m} \lambda_0} \sqrt{\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}}\right).$$

The resulting neuron-wise displacement bound allows us to control activation-pattern mismatches between the fused and local models. Given any (i, j) pair, we define $S_{ij}^F := \{r \in [m] : \mathbb{1}\{\mathbf{x}_{ij}^\top \bar{\mathbf{w}}_{P,r}^F \geq 0\} = \mathbb{1}\{\mathbf{x}_{ij}^\top \tilde{\mathbf{w}}_{i,r} \geq 0\}\}$ as the set of neurons with matching activation signs between the global and local models. Let $S_{ij}^{F\perp} := [m] \setminus S_{ij}^F$ denote the mismatch set.

Lemma 10. With probability at least $1 - \delta$ over the initialization, $|S_{ij}^{F\perp}| = O(\frac{mN(R_0 + R_1)}{\delta \sigma})$ for all i, j simultaneously.

Note that the mismatch size $|S_{ij}^{F\perp}|$ is independent of the client subsampling rate ρ , since R_0 bounds the deviation of each local trained model $\tilde{\mathbf{w}}_{i,r}$ from initial $\mathbf{w}_r^{(0)}$ uniformly across $i \in [M]$, regardless of which clients are sampled.

We are now ready to establish the empirical fitting guarantee of F-SOSA. Lemma 3 controls the local fitting error, while Lemmas 8–10 bound the additional error induced by Fisher-based fusion through the resulting activation-pattern mismatches. Combining these two components yields the following training-error guarantee.

Theorem 2 (Training Error for Fitting the Data). Under Assumptions 2 – 4, let $m = \Omega(N^6/(\lambda_0^4 \sigma^2 \delta^3))$, $\eta = \lambda_0/N^2$, $\mathbf{w}_r^{(0)} \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$, $a_r \sim \text{Uniform}(\{-1, +1\})$, then with probability at least $1 - \delta$, we have

$$L_D^{f,\ell}(\bar{\mathbf{W}}_P^F) \leq O\left(\left(1 - \frac{\eta \lambda_0}{2}\right) K \cdot \frac{N}{\delta} + \frac{N^6 (\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y})^2}{m \delta^4 \sigma^2 \lambda_0^4}\right).$$

Proof. We decompose the loss $L_D^{f,\ell}(\bar{\mathbf{W}}_P^F)$ as follows

$$\begin{aligned} L(\bar{\mathbf{W}}_P^F) &= \frac{1}{N} \sum_{i=1}^M \sum_{j=1}^n (f(\bar{\mathbf{W}}_P^F, \mathbf{x}_{ij}) - y_{ij})^2 \\ &= \frac{1}{N} \sum_{i=1}^M \sum_{j=1}^n (\phi(\bar{\mathbf{W}}_P^F, \mathbf{x}_{ij})^\top \bar{\mathbf{W}}_P^F - \tilde{y}_{ij} + \tilde{y}_{ij} - y_{ij})^2 \end{aligned}$$

$$\leq \underbrace{\frac{2}{N} \sum_{i=1}^M \sum_{j=1}^n (\phi(\bar{\mathbf{W}}_P^F, \mathbf{x}_{ij})^\top \bar{\mathbf{W}}_P^F - \tilde{y}_{ij})^2}_{T_1} + \underbrace{\frac{2}{N} \sum_{i=1}^M \sum_{j=1}^n (\tilde{y}_{ij} - y_{ij})^2}_{T_2}.$$

T_2 measures how well the local models fit their local data and can be bounded via Lemma 3 as $T_2 = O\left(\left(1 - \frac{\eta \lambda_0}{2}\right)^K \frac{N}{\delta}\right)$.

By taking $t \rightarrow \infty$ in (21), $\tilde{y}_{ij} = \phi_a(\tilde{\mathbf{W}}_i, \mathbf{x}_{ij})^\top \bar{\mathbf{W}}_P^F$, we have

$$\begin{aligned} T_1 &= \frac{2}{N} \sum_{i=1}^M \sum_{j=1}^n (\phi(\bar{\mathbf{W}}_P^F, \mathbf{x}_{ij})^\top \bar{\mathbf{W}}_P^F - \phi(\tilde{\mathbf{W}}_i, \mathbf{x}_{ij})^\top \bar{\mathbf{W}}_P^F)^2 \\ &= \frac{2}{N} \sum_{i=1}^M \sum_{j=1}^n \frac{(\sum_{r \in S_{ij}^{F\perp}} a_r \mathbf{x}_{ij}^\top \bar{\mathbf{w}}_{P,r}^F (\mathbb{1}\{\mathbf{x}_{ij}^\top \bar{\mathbf{w}}_{P,r}^F \geq 0\} - \mathbb{1}\{\mathbf{x}_{ij}^\top \tilde{\mathbf{w}}_{i,r} \geq 0\}))^2}{m} \\ &\leq \frac{2}{N} \sum_{i=1}^M \sum_{j=1}^n \frac{|S_{ij}^{F\perp}|}{m} \sum_{r \in S_{ij}^{F\perp}} (a_r \mathbf{x}_{ij}^\top \bar{\mathbf{w}}_{P,r}^F)^2 \\ &\leq \frac{2}{N} \sum_{i=1}^M \sum_{j=1}^n \frac{|S_{ij}^{F\perp}|}{m} \sum_{r \in S_{ij}^{F\perp}} (\mathbf{x}_{ij}^\top \bar{\mathbf{w}}_{P,r}^F - \mathbf{x}_{ij}^\top \tilde{\mathbf{w}}_{i,r})^2 \\ &\leq \frac{2}{N} \sum_{i=1}^M \sum_{j=1}^n \frac{|S_{ij}^{F\perp}|^2}{m} \max_{r \in [m]} \|\bar{\mathbf{w}}_{P,r}^F - \tilde{\mathbf{w}}_{i,r}\|_2^2 \\ &= O\left(\frac{mN^2(R_0^4 + R_1^2)}{\delta^2 \sigma^2} (R_0^2 + R_1^2)\right) \leq O\left(\frac{mN^2(R_0^4 + R_1^4)}{\delta^2 \sigma^2}\right) \\ &\leq O\left(\frac{mN^2}{\delta^2 \sigma^2} \left(\frac{N^4}{\lambda_0^4 \delta^2 m^2} + \frac{N^4 (\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y})^2}{\lambda_0^4 m^2}\right)\right) \\ &\leq O\left(\frac{N^6 (\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y})^2}{m \delta^4 \sigma^2 \lambda_0^4}\right) \end{aligned}$$

where the second-to-the-last inequality uses upper bounds on R_0 and R_1 from Lemma 3 and Lemma 9. $\square$

Corollary 1. Under the setting of Theorem 2, given a target training error ϵ , if we set $m = \Omega(\text{poly}(N, \delta^{-1}, \lambda_0^{-1})/\sigma^2)$, $\eta = \lambda_0/N^2$, $\sigma = O(\lambda_0 \delta/N)$, and $K = \Omega((1/\eta \lambda_0) \log(N/\delta \epsilon))$, then with probability at least $1 - \delta$, we have $L_D^{f,\ell}(\bar{\mathbf{W}}_P^F) \leq \epsilon$.

Corollary 1 shows that the empirical risk can be made vanishingly small in the over-parameterized regime by choosing the network width m and the number of local iterations K sufficiently large. The above width conditions are sufficient analytical conditions for establishing the guarantee in this tractable regime, rather than architectural requirements of F-SOSA itself. The learning and editing procedures are not restricted to two-layer networks and can be instantiated with finite-width deep architectures, as further evaluated with ResNet18 and YOLOv8s in our experiments.

Having established the empirical fitting guarantee, we next study generalization to unseen data. For this purpose, we adopt a two-level data distribution framework [38]. Let $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$ denote the data space, $\mathfrak{D}$ the set of distributions over $\mathcal{Z}$, and $\mathcal{T}$ a meta-distribution over $\mathfrak{D}$. Each client i samples a distribution $D_i \sim \mathcal{T}$ and obtains n i.i.d. samples $\mathcal{D}_i \sim D_i^n$. Let $\mathcal{D} = \bigcup_{i=1}^M \mathcal{D}_i$ be the union of local datasets. We consider

the squared loss $\ell(z, y) = \frac{1}{2}(z - y)^2$. The empirical risk over training data is

$$L_{\mathcal{D}}^{f,\ell}(\mathbf{W}) = \frac{1}{M} \sum_{i=1}^M \frac{1}{n} \sum_{j=1}^n \frac{1}{2} (f_{\mathbf{a}}(\mathbf{W}, \mathbf{x}_{ij}) - y_{ij})^2.$$

The population risk is defined as

$$L_{\mathcal{T}}^{f,\ell}(\mathbf{W}) = \mathbb{E}_{D \sim \mathcal{T}} \mathbb{E}_{\mathbf{z} \sim D} \left[\frac{1}{2} (f_{\mathbf{a}}(\mathbf{W}, \mathbf{x}) - y)^2 \right].$$

We also define the semi-population risk:

$$L_S^{f,\ell}(\mathbf{W}) = \frac{1}{M} \sum_{i=1}^M \mathbb{E}_{\mathbf{z} \sim D_i} \left[\frac{1}{2} (f_{\mathbf{a}}(\mathbf{W}, \mathbf{x}) - y)^2 \right].$$

To extend the empirical analysis to unseen data, we adopt the following federated Rademacher complexity.

Definition 3 (Federated Rademacher complexity). Let $\mathcal{F}$ be a class of functions $g : \mathcal{X} \rightarrow \mathbb{R}$. For M clients with n samples each, the federated Rademacher complexity is

$$\mathcal{R}_n^M(\mathcal{F}) = \mathbb{E}_{\{\iota_{ij}\}} \left[\sup_{g \in \mathcal{F}} \frac{1}{M} \sum_{i=1}^M \frac{1}{n} \sum_{j=1}^n \iota_{ij} \cdot g(\mathbf{x}_{ij}) \right],$$

where $\{\iota_{ij}\}$ are i.i.d. Rademacher variables.

When $M = 1$, this reduces to the classical Rademacher complexity $\mathbb{E}_{\{\iota_j\}} \left[\sup_{g \in \mathcal{F}} \frac{1}{n} \sum_{j=1}^n \iota_j g(\mathbf{x}_j) \right]$. We define a neural network class with weight constraints relative to initialization: $\mathcal{F}_{\mathbf{W}^{(0)}, \mathbf{a}, R, B} := \{f_{\mathbf{a}}(\mathbf{W}, \cdot) : \|\mathbf{w}_r - \mathbf{w}_r^{(0)}\| \leq R, \forall r \in [m], \|\mathbf{W} - \mathbf{W}^{(0)}\|_2 \leq B\}$. The following existing result bounds the Rademacher complexity of the network class determined by the parameter neighborhood above.

Lemma 11 ([5, Lemma 8]). Given $R > 0$, with probability at least $1 - \delta$ over the initialization, uniformly over all $B > 0$,

$$\begin{aligned} \mathcal{R}_n^M(\mathcal{F}_{\mathbf{W}^{(0)}, \mathbf{a}, R, B}) \\ \leq \frac{B}{\sqrt{2N}} \left(1 + \left(\frac{2 \log(2/\delta)}{m} \right)^{1/4} \right) + \frac{2R^2 \sqrt{m}}{\sigma} + R \sqrt{2 \log(2/\delta)}. \end{aligned}$$

We next recall the empirical-to-semi-population generalization bound under the two-level federated data model.

Lemma 12 ([38, Lemma 1]). Let the loss function $\ell(f(\mathbf{W}, \mathbf{x}), y)$ be bounded in $[0, c]$. Under the two-level distribution setting, with probability at least $1 - \delta$, we have

$$\sup_{f \in \mathcal{F}} \{L_S^{f,\ell} - L_{\mathcal{D}}^{f,\ell}\} \leq 2\mathcal{R}_n^M(\mathcal{F}) + 3c \sqrt{\frac{\log(2/\delta)}{2N}}.$$

The remaining step is to control the gap between the semi-population and population risks.

Lemma 13. Let the loss $\ell(f(\mathbf{W}, \mathbf{x}), y)$ be bounded in $[0, c]$. With probability at least $1 - \delta$,

$$\sup_{f \in \mathcal{F}} \{L_{\mathcal{T}}^{f,\ell} - L_S^{f,\ell}\} \leq 2\mathcal{R}_n^M(\mathcal{F}) + 2c \sqrt{\frac{\log(2/\delta)}{2N}} + c \sqrt{\frac{\log(2/\delta)}{2M}}.$$

Lemmas 4 and 8–9 control the fused model around the initialization, while Lemmas 11–13 bound the generalization gaps. Together with Theorem 2, these results yield the following population-risk bound.

Theorem 3 (Generalization Error for Unseen Data). Under Assumptions 2, 3 and 4, let $\sigma = O(\lambda_0 \delta / N)$, $m = \Omega(\text{poly}(N, \delta^{-1}, \lambda_0^{-1}) / \sigma^2)$, $\eta = \lambda_0 / N^2$, $\mathbf{w}_r^{(0)} \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$, $\mathbf{a}_r \sim \text{Uniform}(\{-1, +1\})$ and $K = \Omega((1/\eta \lambda_0) \log(\rho N / (\delta(1 + \rho) \mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y})))$, with probability at least $1 - \delta$, we have

$$L_{\mathcal{T}}^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}^F) \leq O\left(\frac{1}{\rho} \sqrt{\frac{\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}}{\lambda_0}} + \sqrt{\frac{\log \frac{1}{\delta}}{N}} + \sqrt{\frac{\log \frac{1}{\delta}}{M}}\right).$$

Proof. We first bound $\|\bar{\mathbf{W}}_{\mathcal{P}}^F - \bar{\mathbf{W}}_{\mathcal{P}}\|_2$ via Lemmas 8 and Lemma 4, which gives

$$\|\bar{\mathbf{W}}_{\mathcal{P}}^F - \bar{\mathbf{W}}_{\mathcal{P}}\|_2 \leq O(\sqrt{N/(\rho \lambda_0)} \cdot \sqrt{\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}}). \quad (17)$$

Similarly, we bound $\|\bar{\mathbf{w}}_{\mathcal{P},r}^F - \bar{\mathbf{w}}_{\mathcal{P},r}\|_2$ via Lemmas 3 and Lemma 9, which gives

$$\|\bar{\mathbf{w}}_{\mathcal{P},r}^F - \mathbf{w}_r^{(0)}\|_2 \leq O(\text{poly}(N, \delta^{-1}, \lambda_0^{-1}) / \sqrt{m}) \quad (18)$$

Using above bounds, we can conclude that $f_{\mathbf{a}}(\bar{\mathbf{W}}_{\mathcal{P}}^F, \cdot) \in \mathcal{F}_{\mathbf{W}^{(0)}, \mathbf{a}, R, B}$ with R and B set by Equation (18) and (17) respectively. Using Lemma 12 and Lemma 13, we have

$$\begin{aligned} & \sup_{f \in \mathcal{F}_{\mathbf{W}^{(0)}, \mathbf{a}, R, B}} \{L_{\mathcal{T}}^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}^F) - L_{\mathcal{D}}^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}^F)\} \\ & \leq \sup_{f \in \mathcal{F}_{\mathbf{W}^{(0)}, \mathbf{a}, R, B}} \{L_{\mathcal{T}}^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}^F) - L_S^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}^F)\} \\ & \quad + \sup_{f \in \mathcal{F}_{\mathbf{W}^{(0)}, \mathbf{a}, R, B}} \{L_S^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}^F) - L_{\mathcal{D}}^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}^F)\} \\ & \leq O(\mathcal{R}_n^M(\mathcal{F}_{\mathbf{W}^{(0)}, \mathbf{a}, R, B}) + \sqrt{\log(1/\delta)/N} + \sqrt{\log(1/\delta)/M}). \end{aligned}$$

By Lemma 11, the Rademacher complexity satisfies $\mathcal{R}_n^M(\mathcal{F}_{\mathbf{W}^{(0)}, \mathbf{a}, R, B}) \leq O(\frac{1}{\rho} \sqrt{\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}} / \lambda_0)$. Since $L_{\mathcal{T}}^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}^F) = L_{\mathcal{T}}^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}) - L_{\mathcal{D}}^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}}) + L_{\mathcal{D}}^{f,\ell}(\bar{\mathbf{W}}_{\mathcal{P}})$, by combining with the training error established in Theorem 2, we obtain the result by union bound. $\square$

Theorem 3 shows that, with sufficiently many clients and local samples, the generalization of F-SOSA is governed by a Rademacher complexity term of order $\frac{1}{\rho} \sqrt{\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}} / \lambda_0$. As shown in [37], the quantity $\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}$ is bounded by a constant independent of sample size for a broad class of target functions, including linear, quadratic, and cosine models. The factor $1/\rho$ reflects a trade-off between editing efficiency and generalization: larger ρ includes more clients and improves the utility/generalization bound, but weakens TV stability and increases the expected editing cost.

C. Exactness and Efficiency of Model Editing

F-SOSA-U/I are valid transports and efficient for editing.

Theorem 4. (F-SOSA, F-SOSA-U/I) satisfies exact federated model editing. For R editing requests, F-SOSA-U/I triggers at most $O(\rho R)$ recomputation in expectation, with total expected runtime bounded by $O(\max\{\rho R \cdot \text{Training Time}, R\})$.

Proof. Let $\mu_{(P,M)}$ be the uniform distribution over size- P subsets of $[M]$. Let $\mathcal{P} \sim \mu_{(P,M)}$ be a sampled client subset. Assume M leaves or $M + 1$ joins, and consider a deletion

map $\text{DEL} : \mathcal{P} \mapsto \mathcal{P}' \subseteq [M-1]$ (for client removal), and an insertion map $\text{INS} : \mathcal{P} \mapsto \mathcal{P}' \subseteq [M+1]$ (for new client addition). These induce pushforward distributions $\mu_{(P,M)}^{\text{DEL}} := \text{DEL} \# \mu_{(P,M)}$ and $\mu_{(P,M)}^{\text{INS}} := \text{INS} \# \mu_{(P,M)}$. To show F-SOSA-U/I are valid transports, we prove that for any $\mathcal{P} \in [M]^P$, $\mu_{(P,M)}^{\text{DEL}}(\mathcal{P}) = \mu_{P,M-1}(\mathcal{P})$ and for any $\mathcal{P} \in [M+1]^P$, $\mu_{(P,M)}^{\text{INS}}(\mathcal{P}) = \mu_{(P,M+1)}(\mathcal{P})$.

For F-SOSA-U, we consider two cases based on whether the membership verification succeeds. **Case 1 (verification succeeds).** The pushforward measure $\mu_{(P,M)}^{\text{DEL}}$ is simply $\mu_{(P,M)}$ conditioned on the event $M \notin \mathcal{P}$,

$$\mu_{(P,M)}^{\text{DEL}}(\mathcal{P}) = \mu_{(P,M)}(\mathcal{P} | M \notin \mathcal{P}) = \frac{\mu_{(P,M)}(\mathcal{P} \cap \{M \notin \mathcal{P}\})}{\mu_{(P,M)}(\{M \notin \mathcal{P}\})}.$$

The denominator $\mu_{(P,M)}(\{M \notin \mathcal{P}\}) = 1 - \mu_{(P,M)}(\{M \in \mathcal{P}\}) = 1 - \binom{M-1}{P-1} / \binom{M}{P} = 1 - P/M$. For $\mu_{(P,M)}(\mathcal{P} \cap \{M \notin \mathcal{P}\})$, it equals 0 if $M \in \mathcal{P}$, and $\mu_{(P,M)}(\mathcal{P})$ otherwise. In both branches, we have $\mu_{(P,M)}^{\text{DEL}}(\mathcal{P}) = \mu_{P,M-1}(\mathcal{P})$. **Case 2 (verification fails).** In this case, M would have been replaced by some point in $\mathcal{P}$ and there are M possibilities. Let ν be the uniform probability measure over $[M]$. For each $k \in [M]$, let $\mathcal{E}_k$ be the event that M was replaced by i_k . The events $\mathcal{E}_k$'s are disjoint and $\mathcal{P}$ is $\cup_{k=1}^M \mathcal{E}_k$. Then $\mu_{(P,M)}^{\text{DEL}}(\mathcal{P}) = \mu_{(P,M)}^{\text{DEL}}(\sum_{k=1}^M \mathcal{E}_k) = \sum_{k=1}^M \mu_{(P,M)}(\mathcal{P} \setminus \{i_k\} \cup \{M\}) \nu(\{i_k\}) = 1 / \binom{M-1}{P} = \mu_{(P,M-1)}(\mathcal{P})$.

For F-SOSA-I, given $\mathcal{P}$, we consider two cases based on whether the inserted $M+1$ lies in $\mathcal{P}$ or not. **Case 1 ($M+1 \in \mathcal{P}$).** In this case, Bernoulli($P/(M+1)$) must be 1, $M+1$ would have replaced some other points in $[M+1] \setminus \mathcal{P}$ and there are $M+1-P$ possibilities. Let ν be a uniform probability measure over $[M+1-P]$. For each $k \in [M+1-P]$, let $\mathcal{E}_k$ be the event that $M+1$ replaced i_k , then $\mathcal{E}_k$'s are disjoint and the event $\mathcal{P}$ is $\cup_{k=1}^{M+1-P} \mathcal{E}_k$. Then $\mu_{(P,M)}^{\text{INS}}(\mathcal{E}_k)$ equals

$$\begin{aligned} \mu_{(P,M)}^{\text{INS}}(\mathcal{P} \setminus \{M+1\} \cup \{i_k\} | i_k \text{ replaced}) \mu_{(P,M)}^{\text{INS}}(i_k \text{ replaced}) \\ = \mu_{(P,M)}(\mathcal{P} \setminus \{M+1\} \cup \{i_k\} | \{i_k\}) \nu(i_k) = \frac{1}{\binom{M}{P-1} (M+1-P)}. \end{aligned}$$

Thus, $\mu_{(P,M)}^{\text{INS}}(\mathcal{P}) = \frac{P}{M+1} \mu_{(P,M)}^{\text{INS}}(\cup_{k=1}^{M+1-P} \mathcal{E}_k) = \mu_{(P,M+1)}(\mathcal{P})$. **Case 2 ($M+1 \notin \mathcal{P}$).** Bernoulli($P/(M+1)$) = 0, there is no replacement, $\mu_{(P,M)}^{\text{INS}}(\mathcal{P}) = (1 - \frac{P}{M+1}) / \binom{M}{P} = 1 / \binom{M+1}{P} = \mu_{(P,M+1)}(\mathcal{P})$.

Therefore, F-SOSA-U/I are both valid transports. Let $\mathcal{P}^{\text{EDT}}$ be the post-editing sampled subset. By ρ -TV-stability of F-SOSA, $\mathbb{P}(\mathcal{P} \neq \mathcal{P}^{\text{EDT}}) \leq \rho$, and $M/2 \leq M(t) \leq 2M$, so each request triggers recomputation with probability at most 2ρ . For R total edits, the expected number of recomputations is $O(\rho R)$, with total expected runtime bounded by $O(\max\{\rho R \cdot \text{Training Time}, R\})$. $\square$

VI. EXPERIMENTS

We empirically evaluate our framework with focus on: (1) one-shot learning performance, (2) model editing accuracy and efficiency, (3) trade-offs controlled by the stability level ρ , and (4) resilience to membership inference attacks. We also include a realistic case study based on vehicle mobility simulation.

A. Experimental Setup

1) *Datasets and Models:* We use MNIST [39], SVHN [40], CIFAR-10 [41], and FEMNIST from LEAF [42]. For MNIST/SVHN/CIFAR-10, we simulate client heterogeneity using Dirichlet partitioning ($\alpha \in \{0.1, 0.5, 0.8\}$; default $\alpha=0.5$). FEMNIST, as a FL dataset, naturally exhibits non-IID characteristics; we follow [6], sampling 10% of users and grouping into M clients. Unless otherwise specified, we use the two-layer CNN from [5] with K-FAC Fisher approximation. To evaluate F-SOSA beyond the default classification setting, we additionally consider ResNet18 on CIFAR-10 and YOLOv8s on the real-world driving dataset BDD100K; detailed results are reported in the Appendix.

2) *Baselines:* We compare F-SOSA with state-of-the-art one-shot FL baselines: FedFisher [5] leverages client-specific Fisher information to improve utility under heterogeneity; FedLPA [6] applies layer-wise Laplace approximation to construct local posteriors; Dense [7] uses GAN-generated samples for data-free distillation at the server. For model editing, we evaluate both unlearning and incremental learning settings. In unlearning: Retrain simply does re-execution; FedEraser [9] reconstructs the target model via calibrated gradient updates; FedRR [21] uses diagonal Fisher information with momentum correction; FATS-CU [22] stabilizes FedAvg for rapid unlearning. In incremental learning, we compare with Retrain and Fine-Tuning, where the latter adapts the global model locally on the new client using a tuned number of epochs selected from $\{5, 10, \dots, 50\}$. All methods are evaluated under the one-shot constraint for fair comparison.

3) *Hyperparameters:* We follow the settings in [5], using learning rate $\eta = 0.01$ and default local epoch number $K = 50$. For learning experiments, we set $M = 20$ clients; for model editing, we use at most 100 clients and initialize with 50 active clients, while the scalability experiments in the Appendix consider client populations from 20 to 500. The stability level is set to $\rho = 0.5$ unless otherwise specified.

4) *Evaluation Metrics:* (1) *test accuracy*, measuring model generalization; (2) *training loss*, reflecting the model's fit to training data; (3) *time cost*, the total wall-clock time of execution; (4) *communication cost*, measuring the total bytes transmitted during training and model editing, including global-model broadcasts, local-model uploads, and Fisher-information uploads; (5) *time efficiency* and (6) *communication efficiency*, defined as the average accuracy gain per unit of time and communication, respectively, for efficiency evaluation.

B. Experimental Results

1) *Performance of F-SOSA in Heterogeneous Settings:* We evaluate F-SOSA under varying data heterogeneity levels ($\alpha \in \{0.1, 0.5, 0.8\}$) on MNIST/SVHN/CIFAR-10, and include FEMNIST as an inherently heterogeneous FL dataset. We compare against three state-of-the-art one-shot FL baselines. As shown in Table II and Table III, F-SOSA consistently achieves the lowest communication cost across all datasets and heterogeneity levels, leading to the highest communication efficiency. This benefit arises from its client subsampling mechanism, which reduces aggregation overhead. Time cost

TABLE II
PERFORMANCE COMPARISON OF F-SOSA AND BASELINES ACROSS MNIST, SVHN, AND CIFAR-10 UNDER VARYING DIRICHLET HETEROGENEITY α .
BEST RESULTS ARE IN BOLD; SECOND-BEST RESULTS ARE UNDERLINED.

Dataset	Metrics	$\alpha = 0.1$				$\alpha = 0.5$				$\alpha = 0.8$			
		FedFisher	FedLPA	Dense	F-SOSA	FedFisher	FedLPA	Dense	F-SOSA	FedFisher	FedLPA	Dense	F-SOSA
MNIST	Accuracy (%)	90.35	87.90	85.92	91.08	95.98	94.21	96.06	95.43	96.08	92.82	96.93	95.80
	Time (s)	43.11	24.59	187.96	<u>34.63</u>	43.76	26.61	190.54	<u>34.59</u>	43.15	29.02	193.58	<u>35.56</u>
	Comm. (MB)	35.30	793.31	<u>33.76</u>	18.46	35.30	793.31	<u>33.76</u>	18.46	35.30	793.31	<u>33.76</u>	18.46
	Time Efficiency	2.10	3.57	0.52	<u>2.63</u>	2.19	3.54	0.57	<u>2.76</u>	2.23	3.20	0.58	<u>2.69</u>
	Comm. Efficiency	<u>2.56</u>	0.11	2.55	4.93	2.72	0.12	<u>2.85</u>	5.17	2.72	0.12	<u>2.87</u>	5.19
SVHN	Accuracy (%)	64.89	38.81	62.78	<u>64.88</u>	80.52	76.24	70.63	<u>80.33</u>	79.64	77.75	78.76	<u>79.40</u>
	Time (s)	<u>29.15</u>	50.50	198.85	23.54	<u>28.98</u>	51.61	184.83	21.29	<u>28.53</u>	51.33	188.70	22.14
	Comm. (MB)	77.04	362.74	<u>75.27</u>	40.31	77.04	362.74	<u>75.27</u>	40.31	77.04	362.74	<u>75.27</u>	40.31
	Time Efficiency	<u>2.23</u>	0.77	0.36	2.76	<u>2.78</u>	1.48	0.40	3.77	<u>2.79</u>	1.51	0.46	3.59
	Comm. Efficiency	0.84	0.11	0.83	1.61	<u>1.05</u>	0.21	0.94	1.99	1.03	0.21	1.05	1.97
CIFAR10	Accuracy (%)	39.99	30.49	<u>36.57</u>	32.38	51.86	40.36	50.09	<u>51.69</u>	55.18	44.32	52.58	<u>52.65</u>
	Time (s)	<u>22.76</u>	63.79	202.91	16.55	<u>23.16</u>	56.50	197.96	16.26	<u>23.22</u>	55.92	197.70	16.22
	Comm. (MB)	77.04	362.74	<u>75.27</u>	40.31	77.04	362.74	<u>75.27</u>	40.31	77.04	362.74	<u>75.27</u>	40.31
	Time Efficiency	1.76	0.48	0.19	1.96	2.24	0.72	0.26	3.18	2.38	0.79	0.28	3.25
	Comm. Efficiency	<u>0.52</u>	0.08	0.49	0.80	<u>0.67</u>	0.11	0.67	1.28	0.72	0.12	0.70	1.31

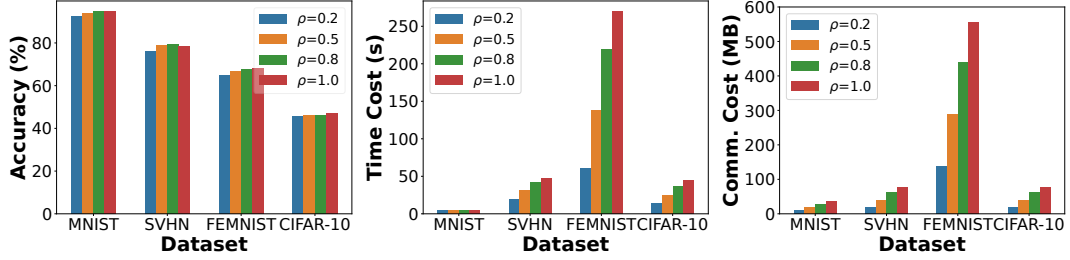


Fig. 1. Effect of the stability level ρ on F-SOSA's accuracy and time/ communication overhead across datasets.

TABLE III
PERFORMANCE COMPARISON ON FEMNIST UNDER A REAL-WORLD NON-IID SETTING. BEST RESULTS ARE IN BOLD; SECOND-BEST RESULTS ARE UNDERLINED.

Metrics	FedFisher	FedLPA	Dense	F-SOSA
Accuracy (%)	76.94	70.18	75.97	75.71
Time (s)	115.89	18.69	138.57	<u>67.07</u>
Comm. (MB)	554.12	1968.70	<u>528.83</u>	289.68
Time Efficiency	0.66	3.75	0.64	<u>1.13</u>
Comm. Efficiency	0.139	0.036	0.144	0.261

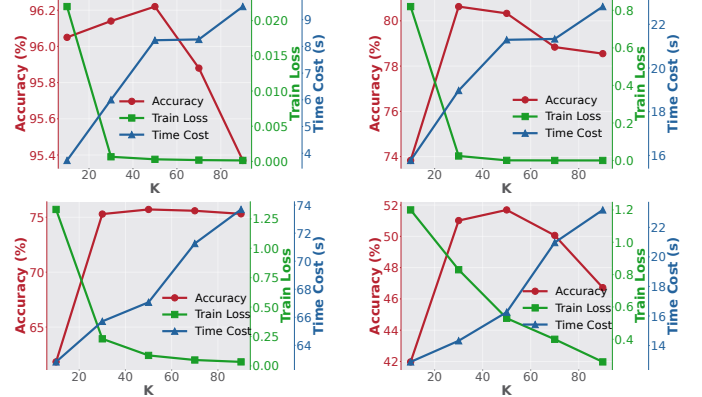


Fig. 2. Effect of local epoch number K on performance. Top-left: MNIST, top-right: SVHN, bottom-left: FEMNIST, bottom-right: CIFAR-10.

is also substantially lower, often by over 50% compared to FedLPA and Dense, while remaining comparable to FedFisher in most cases. In terms of accuracy, F-SOSA matches or exceeds FedFisher under moderate heterogeneity ($\alpha = 0.5, 0.8$), and consistently outperforms Dense and FedLPA. Under high heterogeneity ($\alpha = 0.1$), its performance remains robust with minimal degradation. These results demonstrate that F-SOSA offers strong utility-efficiency trade-offs, maintaining competitive accuracy while substantially reducing communication and computation costs.

2) *Impact and Selection of Hyperparameters:* We examine how the stability parameter ρ and local epoch number K affect F-SOSA's utility and efficiency. Figure 1 shows that increasing ρ improves accuracy, as more clients are aggregated, enhancing generalization. When $\rho = 1.0$, F-SOSA recovers FedFisher and achieves the highest accuracy. However, this comes at a

cost of increased time and communication, especially on larger datasets like FEMNIST. A moderate setting ($\rho = 0.5$) provides a strong trade-off between utility and efficiency, and is used by default. As shown in Figure 2, larger K reduces training loss, indicating better local fitting. However, test accuracy peaks around $K = 30$ or 50 , beyond which overfitting may reduce generalization. Meanwhile, time cost grows linearly with K . We adopt $K = 50$ as default, balancing generalization and efficiency across datasets.

3) *Performance of F-SOSA-I/U:* We evaluate F-SOSA-I/U on CIFAR-10 using a fixed sequence of randomly generated client join and leave requests. All methods are tested on

TABLE IV
MEMBERSHIP INFERENCE ATTACK (MIA) RESULTS ON UNLEARNED GLOBAL MODELS FROM DIFFERENT METHODS ON FEMNIST AND CIFAR-10.

Method	FEMNIST			CIFAR-10		
	Accuracy (%)	Precision (%)	AUC (%)	Accuracy (%)	Precision (%)	AUC (%)
FedEraser	50.01 $\pm$ 0.02	51.08 $\pm$ 3.23	50.01 $\pm$ 0.02	48.99 $\pm$ 0.78	40.75 $\pm$ 7.83	48.99 $\pm$ 0.78
FedRR	50.45 $\pm$ 0.49	50.37 $\pm$ 0.41	50.45 $\pm$ 0.49	45.17 $\pm$ 2.33	42.29 $\pm$ 2.87	45.17 $\pm$ 2.33
FATS-CU	50.47 $\pm$ 0.25	50.33 $\pm$ 0.18	50.47 $\pm$ 0.25	53.54 $\pm$ 0.88	61.29 $\pm$ 2.99	53.54 $\pm$ 0.88
Retrain	50.45 $\pm$ 0.21	50.31 $\pm$ 0.15	50.45 $\pm$ 0.21	51.64 $\pm$ 1.67	52.04 $\pm$ 2.09	51.64 $\pm$ 1.67
F-SOSA-U	50.92 $\pm$ 0.33	50.63 $\pm$ 0.23	50.92 $\pm$ 0.33	51.63 $\pm$ 1.80	52.17 $\pm$ 2.72	51.63 $\pm$ 1.80

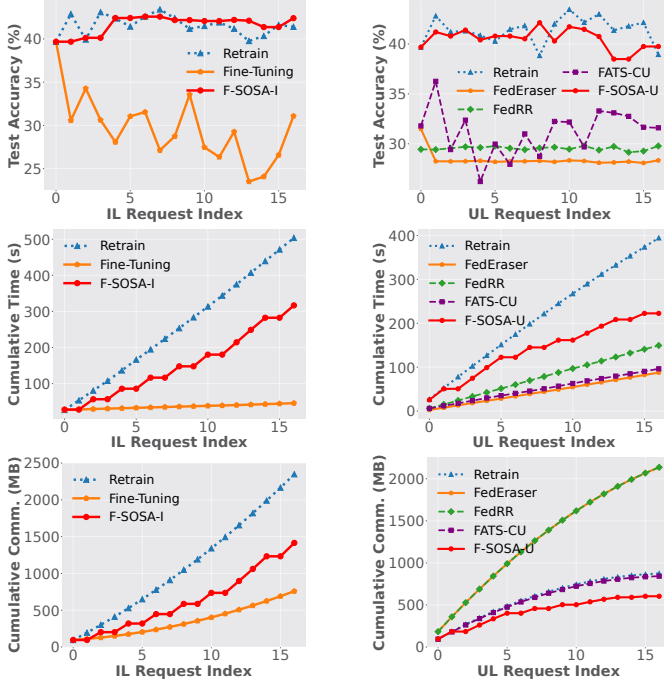


Fig. 3. Performance of F-SOSA-I/U vs. baselines on CIFAR-10. Columns: IL (left), UL (right). Rows: test accuracy (top), time cost (middle), communication cost (bottom).

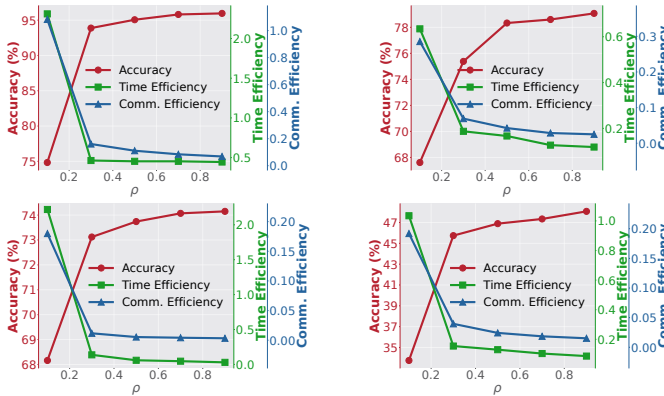


Fig. 4. Effect of stability parameter ρ on accuracy and efficiency. Top-left: MNIST, top-right: SVHN, bottom-left: FEMNIST, bottom-right: CIFAR-10.

the same sequence for fair comparison. Figure 3 presents the results. Note that initial accuracy and overhead differ across unlearning baselines due to their learning methods: FedEraser, FedRR, and FATS-CU use FedAvg, while F-SOSA-

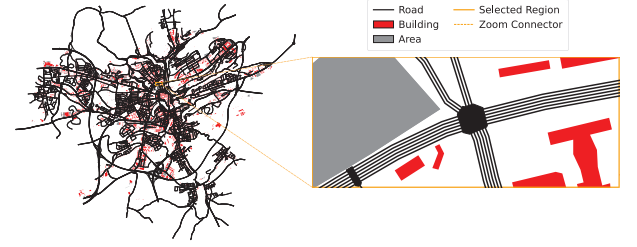


Fig. 5. The Luxembourg map and the region considered.

U and Retrain use F-SOSA. F-SOSA-I matches Retrain in accuracy while consistently outperforming Fine-Tuning by up to 18.6%. It also exhibits the lowest accuracy fluctuation (2.28%), compared to 3.20% (Retrain) and 9.12% (Fine-Tuning), benefiting from its verification mechanism that avoids unnecessary re-aggregation. In contrast, Fine-Tuning overfits to newly joined clients, degrading generalization. F-SOSA-I reduces the final cumulative communication cost by 39.73% compared with Retrain. Fine-Tuning incurs the lowest communication and time costs, but suffers from significant utility loss. F-SOSA-U reaches accuracy comparable to Retrain and surpasses FedEraser/FedRR/FATS-CU by up to 10%. This advantage arises from Fisher-aware aggregation. While FedEraser/FedRR/FATS-CU incur slightly lower time costs, their performance drops notably. F-SOSA-U also reduces accuracy fluctuation and maintains high efficiency by avoiding redundant recomputation.

4) *Trade-off Between Editing Utility and Efficiency w.r.t. Stability Parameter ρ* : We study how the stability parameter ρ influences the utility-efficiency trade-off in F-SOSA-I/U across datasets, with results in Figure 4. As ρ increases, accuracy improves due to greater client participation, which enhances generalization. However, this comes at the cost of increased time and communication overhead from larger aggregation sets. Notably, small values of ρ (e.g., 0.1–0.3) yield significant accuracy gains with minimal efficiency loss, suggesting that editing benefits from a small number of well-chosen clients. Around $\rho = 0.5$, F-SOSA-I/U achieves a favorable balance between accuracy and resource usage across all datasets.

5) *Resistance to Membership Inference Attacks*: We assess unlearning effectiveness via membership inference attacks (MIA) on the global models produced by different methods on FEMNIST and CIFAR-10 (Table IV). F-SOSA-U achieves near-random metrics (accuracy, precision, AUC around 50%), closely matching Retrain across both datasets. This suggests

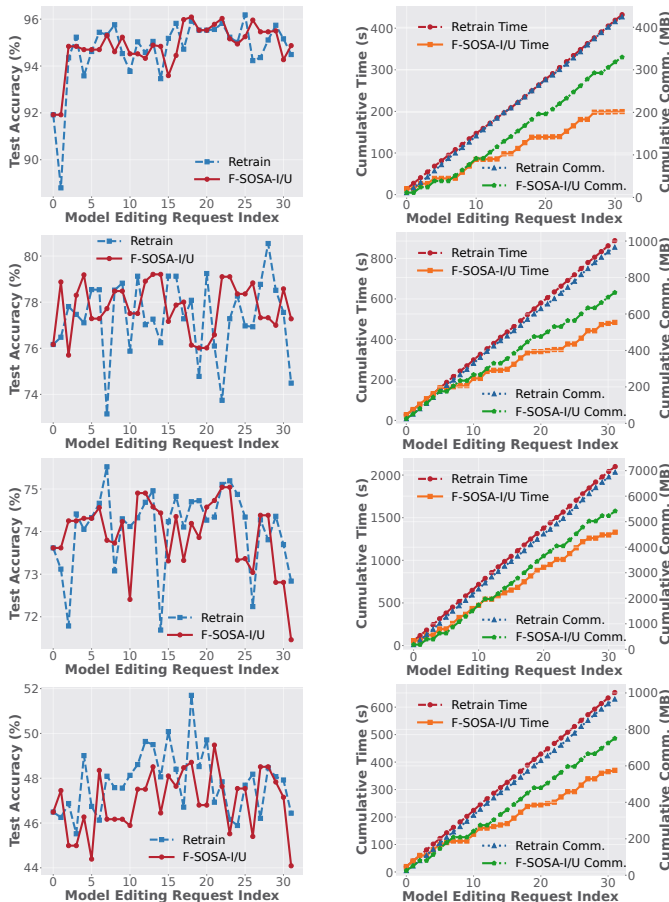


Fig. 6. Test accuracy (left) and cumulative cost (right) of F-SOSA-I/U vs. Retrain under simulator-generated model editing requests. Rows: MNIST, SVHN, FEMNIST, CIFAR-10 (top to bottom).

that the model cannot reliably distinguish whether a sample was seen during training, confirming successful unlearning. On CIFAR-10, F-SOSA-U shows even smaller deviations from random guessing compared to other baselines, some of which exceed 5%, indicating a lower risk of memorization leakage. Although Retrain is marginally stronger in a few cases, the differences are negligible. Given its much lower time and communication costs, F-SOSA-U offers an effective and efficient solution for one-shot federated unlearning.

6) *F-SOSA Performance Across Model Architectures*: We additionally evaluate F-SOSA with ResNet18 on CIFAR-10 and YOLOv8s on BDD100K, with detailed results reported in the Appendix B-B. Across both architectures and varying data heterogeneity, F-SOSA maintains competitive utility while reducing communication and computation costs. These results complement the default experiments by demonstrating the effectiveness of F-SOSA on substantially deeper network architectures.

7) *Scalability with the Number of Clients*: We evaluate F-SOSA-I/U with total client populations ranging from 20 to 500, using 15 sequential incremental-learning and 15 sequential unlearning requests at each scale. Across the evaluated range, F-SOSA-I/U maintain accuracy close to Retrain while consistently reducing communication and computation costs.

The efficiency gains remain substantial even with 500 clients, showing that selective editing continues to provide benefits as the client population grows. Detailed results are reported in the Appendix B-C.

C. Case Study Using Simulation of Urban Mobility (SUMO)

We further evaluate F-SOSA-I/U using the Simulation of Urban Mobility (SUMO) [43], focusing on a selected city center region in the Luxembourg scenario [44] (see Figure 5). Editing requests are triggered by vehicle movements during the 08:00 morning rush hour, simulating dynamic client join/leave events. For this case study, we map each vehicle arrival to a new client contribution and each vehicle departure to an unlearning request issued before disconnection, thereby generating a mobility-driven editing sequence. We apply F-SOSA-I/U and Retrain to the resulting request sequence across four datasets. Figure 6 reports the test accuracy after each request and the cumulative time/communication costs. F-SOSA-I/U match the accuracy of Retrain while maintaining smaller fluctuations. On SVHN, for instance, Retrain exhibits swings over 5%, whereas F-SOSA-I/U keep variations within 3.96% across all datasets. This robustness stems from client subsampling and Bernoulli sampling, which avoid unnecessary updates when edits do not affect the aggregation set. In terms of efficiency, F-SOSA-I/U reduce cumulative time and communication overhead by 36.66%–53.84% and 21.97%–25.76%, respectively, where the reported communication cost includes all model transfers and Fisher-information uploads. In summary, F-SOSA-I/U deliver model editing with near-optimal accuracy and significantly improved efficiency, showing their practical benefits in dynamic environments. We further replay the same mobility-driven editing sequence on a YOLOv8s object-detection model trained on BDD100K; the corresponding results are reported in the Appendix B-D.

VII. CONCLUSIONS

We present a principled framework for exact model editing in one-shot FL, enabling both client unlearning and incremental learning. At its core is F-SOSA, a stable learning algorithm that supports efficient post-training edits. Together with the editing algorithms F-SOSA-U and F-SOSA-I, our method achieves statistical exactness while preserving model utility and efficiency. We provide theoretical guarantees and comprehensive experiments demonstrating its effectiveness across standard benchmarks and dynamic edge scenarios.

REFERENCES

- [1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *20th International Conference on Artificial Intelligence and Statistics (AISTATS 2017)*, Fort Lauderdale, FL, 2017.
- [2] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *3rd Conference on Machine Learning and Systems (MLSys 2020)*, Austin, TX, 2020.
- [3] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *37th International Conference on Machine Learning (ICML 2020)*. Virtual Conference: PMLR, 2020.
- [4] N. Guha, A. Talwalkar, and V. Smith, "One-shot federated learning," arXiv, cs.LG 1902.11175, 2019.

- [5] D. Jhunjunwala, S. Wang, and G. Joshi, “FedFisher: Leveraging Fisher Information for One-Shot Federated Learning,” in *27th International Conference on Artificial Intelligence and Statistics (AISTATS 2024)*. Valencia, Spain: PMLR, 2024.
- [6] X. Liu, L. Liu, F. Ye, Y. Shen, X. Li, L. Jiang, and J. Li, “Fedlpa: One-shot federated learning with layer-wise posterior aggregation,” in *38th Conference on Neural Information Processing Systems (NeurIPS 2024)*, vol. 37. Vancouver, Canada: Curran Associates Inc., 2024.
- [7] J. Zhang, C. Chen, B. Li, L. Lyu, S. Wu, S. Ding, C. Shen, and C. Wu, “Dense: Data-free one-shot federated learning,” in *36th Conference on Neural Information Processing Systems (NeurIPS 2022)*, vol. 35. New Orleans, LA: Curran Associates Inc., 2022.
- [8] F. Dressler, G. S. Pannu, F. Hagenauer, M. Gerla, T. Higuchi, and O. Altintas, “Virtual Edge Computing Using Vehicular Micro Clouds,” in *IEEE International Conference on Computing, Networking and Communications (ICNC 2019)*. Honolulu, HI: IEEE, 2019.
- [9] G. Liu, X. Ma, Y. Yang, C. Wang, and J. Liu, “FedEraser: Enabling efficient client-level data removal from federated learning models,” in *29th IEEE/ACM International Symposium on Quality of Service (IWQOS 2021)*. Chiyoda, Japan: IEEE, 2021.
- [10] Y. Fraboni, M. Van Waerebeke, K. Scaman, R. Vidal, L. Kameni, and M. Lorenzi, “Sifu: Sequential informed federated unlearning for efficient and provable client unlearning in federated optimization,” in *27th International Conference on Artificial Intelligence and Statistics (AISTATS 2024)*. Valencia, Spain: PMLR, 2024.
- [11] H. Wu, J. Yang, S. Chen, D. Wang, D. Yu, and Y. Tao, “Trajectory-aware certified decentralized unlearning via sgd stability,” in *Forty-third International Conference on Machine Learning*.
- [12] C. Wu, S. Zhu, and P. Mitra, “Federated unlearning with knowledge distillation,” arXiv, cs.LG 2201.09441, 2022.
- [13] J. Wang, S. Guo, X. Xie, and H. Qi, “Federated unlearning via class-discriminative pruning,” in *31st ACM Web Conference (WWW 2022)*. Lyon, France: ACM, 2022.
- [14] L. Wu, S. Guo, J. Wang, Z. Hong, J. Zhang, and Y. Ding, “Federated Unlearning: Guarantee the Right of Clients to Forget,” *IEEE Network*, vol. 27, no. 5, 2013.
- [15] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership inference attacks against machine learning models,” in *38th IEEE Symposium on Security and Privacy (S&P 2017)*. San Jose, CA: IEEE, 2017.
- [16] M. Song, Z. Wang, Z. Zhang, Y. Song, Q. Wang, J. Ren, and H. Qi, “Analyzing user-level privacy attack against federated learning,” *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 10, 2020.
- [17] Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi, “Beyond inferring class representatives: User-level privacy leakage from federated learning,” in *38th IEEE Conference on Computer Communications (INFOCOM 2019)*. Paris, France: IEEE, 2019.
- [18] L. Zhu, Z. Liu, and S. Han, “Deep Leakage from Gradients,” in *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*. Vancouver, Canada: Curran Associates Inc., 2019.
- [19] C.-L. Wang, Q. Li, Z. Xiang, Y. Cao, and D. Wang, “Towards lifecycle unlearning commitment management: Measuring sample-level unlearning completeness,” in *34th USENIX Security Symposium (USENIX Security 25)*, 2025, pp. 6481–6500.
- [20] J. Chen, Z. Lin, W. Lin, W. Shi, X. Yin, and D. Wang, “Fedmua: Exploring the vulnerabilities of federated learning to malicious unlearning attacks,” *IEEE Transactions on Information Forensics and Security*, vol. 20, pp. 1665–1678, 2025.
- [21] Y. Liu, L. Xu, X. Yuan, C. Wang, and B. Li, “The right to be forgotten in federated learning: An efficient realization with rapid retraining,” in *41st IEEE International Conference on Computer Communications (INFOCOM 2022)*. Virtual Conference: IEEE, 2022.
- [22] Y. Tao, C.-L. Wang, M. Pan, D. Yu, X. Cheng, and D. Wang, “Communication Efficient and Provable Federated Unlearning,” *Proceedings of the VLDB Endowment*, vol. 17, no. 5, 2024.
- [23] Y. F. Bakman, D. N. Yaldiz, Y. H. Ezzeldin, and S. Avestimehr, “Federated orthogonal training: Mitigating global catastrophic forgetting in continual federated learning,” in *12th International Conference on Learning Representations (ICLR 2024)*. Vienna, Austria: ICLR, 2024.
- [24] Z. Zhang, B. Guo, W. Sun, Y. Liu, and Z. Yu, “Cross-FCL: Toward a Cross-Edge Federated Continual Learning Framework in Mobile Edge Computing Systems,” *IEEE Transactions on Mobile Computing*, vol. 23, no. 1, 2024.
- [25] S. Yue, J. Ren, J. Xin, S. Lin, and J. Zhang, “Inexact-ADMM based federated meta-learning for fast and continual edge learning,” in *22nd ACM International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing (MobiHoc 2021)*. Shanghai, China: ACM, 2021.
- [26] M. Ding, K. Ji, D. Wang, and J. Xu, “Understanding forgetting in continual learning with linear regression,” arXiv preprint arXiv:2405.17583, 2024.
- [27] S. Babakniya, Z. Fabian, C. He, M. Soltanolkotabi, and S. Avestimehr, “A data-free approach to mitigate catastrophic forgetting in federated class incremental learning for vision tasks,” in *37th Conference on Neural Information Processing Systems (NeurIPS 2023)*. New Orleans, LA: Curran Associates Inc., 2023.
- [28] J. Dong, L. Wang, Z. Fang, G. Sun, S. Xu, X. Wang, and Q. Zhu, “Federated class-incremental learning,” in *33rd IEEE/CVF conference on computer vision and pattern recognition (CVPR 2022)*. New Orleans, LA: IEEE, 2022.
- [29] J. Yoon, W. Jeong, G. Lee, E. Yang, and S. J. Hwang, “Federated continual learning with weighted inter-client transfer,” in *38th International Conference on Machine Learning (ICML 2021)*, vol. 139. Virtual Conference: PMLR, 2021.
- [30] W. Huang, M. Ye, and B. Du, “Learn from others and be yourself in heterogeneous federated learning,” in *33rd IEEE/CVF conference on computer vision and pattern recognition (CVPR 2022)*. New Orleans, LA: IEEE, 2022.
- [31] Z. Ma, Y. Lu, W. Li, and S. Cui, “EFL: elastic federated learning on non-IID data,” in *1st Conference on Lifelong Learning Agents (COLLAS 2022)*. Montréal, Canada: PMLR, 2022.
- [32] R. Zhang, Q. Xu, J. Yao, Y. Zhang, Q. Tian, and Y. Wang, “Federated domain generalization with generalization adjustment,” in *34th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2023)*. Vancouver, Canada: IEEE, 2023.
- [33] Y. Ruan, X. Zhang, S.-C. Liang, and C. Joe-Wong, “Towards flexible device participation in federated learning,” in *24th International Conference on Artificial Intelligence and Statistics (AISTATS 2021)*. Virtual Conference: PMLR, 2021.
- [34] Z.-L. Chang, D.-J. Han, S. Hosseinalipour, M. Chiang, and C. G. Brinton, “Federated Learning with Dynamic Client Arrival and Departure: Convergence and Rapid Adaptation via Initial Model Construction,” arXiv, cs.LG 2410.05662, 2024.
- [35] E. Ullah, T. Mai, A. Rao, R. A. Rossi, and R. Arora, “Machine unlearning via algorithmic stability,” in *34th Annual Conference on Learning Theory (COLT 2021)*. Boulder, CO: PMLR, 2021.
- [36] M. Al-Shedivat, J. Gillenwater, E. Xing, and A. Rostamizadeh, “Federated learning via posterior averaging: A new perspective and practical algorithms,” in *9th International Conference on Learning Representations (ICLR 2021)*. Virtual Conference: OpenReview.net, 5 2021.
- [37] S. Arora, S. Du, W. Hu, Z. Li, and R. Wang, “Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks,” in *36th International Conference on Machine Learning (ICML 2019)*, vol. 97. Long Beach, CA: PMLR, 2019.
- [38] X. Hu, S. Li, and Y. Liu, “Generalization bounds for federated learning: Fast rates, unparticipating clients and unbounded losses,” in *11th International Conference on Learning Representations (ICLR 2023)*. Kigali, Rwanda: OpenReview.net, 2023.
- [39] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, 1998.
- [40] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, “Reading digits in natural images with unsupervised feature learning,” in *NIPS 2011 Workshop on Deep Learning and Unsupervised Feature Learning*, Granada, Spain, 2011.
- [41] A. Krizhevsky, “Learning multiple layers of features from tiny images,” University of Toronto, Technical Report TR-2009, 2009.
- [42] S. Caldas, S. M. K. Duddu, P. Wu, T. Li, J. Konečný, B. McMahan, V. Smith, and A. Talwalkar, “Leaf: A benchmark for federated settings,” arXiv, cs.LG 1812.01097, 2018.
- [43] P. A. Lopez, E. Wiessner, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flotterod, R. Hilbrich, L. Lucken, J. Rummel, and P. Wagner, “Microscopic Traffic Simulation using SUMO,” in *21st IEEE International Conference on Intelligent Transportation Systems (ITSC 2018)*. Maui, HI: IEEE, 2018.
- [44] L. Codecá, R. Frank, S. Faye, and T. Engel, “Luxembourg SUMO Traffic (LuST) Scenario: Traffic Demand Evaluation,” *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 2, 2017.
- [45] G. Jocher, A. Chaurasia, and J. Qiu, “Ultralytics yolov8,” 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [46] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, “Bdd100k: A diverse driving dataset for heterogeneous multitask learning,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

APPENDIX A
PROOFS OF AUXILIARY LEMMAS

Proof of Lemma 4. Part 1) is directly from [5, Lemma 10]. For Part 2), we first apply convexity of the squared norm to bound $\|\bar{\mathbf{W}}_{\mathcal{P}} - \mathbf{W}^{(0)}\|_2^2$ by the average $\|\tilde{\mathbf{W}}_i - \mathbf{W}^{(0)}\|_2^2$ over $i \in \mathcal{P}$.

$$\left\| \bar{\mathbf{W}}_{\mathcal{P}} - \mathbf{W}^{(0)} \right\|_2^2 \leq \frac{1}{\rho M} \sum_{l \in \mathcal{P}} \left\| \tilde{\mathbf{W}}_l - \mathbf{W}^{(0)} \right\|_2^2,$$

where each summation term can be bounded using Part 1. Expressing the resulting sum via the block-diagonal Gram matrix $\text{diag}(\mathbf{H}_1^\infty, \dots, \mathbf{H}_M^\infty)$, we have

$$\begin{aligned} & \left\| \bar{\mathbf{W}}_{\mathcal{P}} - \mathbf{W}^{(0)} \right\|_2^2 \\ & \leq \frac{2 \sum_{l \in \mathcal{P}} \mathbf{y}_l^\top (\mathbf{H}_l^\infty)^{-1} \mathbf{y}_l}{\rho M} + O\left(\frac{N^2 \sigma^2}{\lambda_0^2 \delta^2}\right) + \frac{\text{poly}(N, \lambda_0^{-1}, \delta^{-1})}{m^{\frac{1}{2}} \sigma} \\ & \leq \frac{2 \sum_{i=1}^M \mathbf{y}_i^\top (\mathbf{H}_i^\infty)^{-1} \mathbf{y}_i}{\rho M} + O\left(\frac{N^2 \sigma^2}{\lambda_0^2 \delta^2}\right) + \frac{\text{poly}(N, \lambda_0^{-1}, \delta^{-1})}{m^{\frac{1}{2}} \sigma} \\ & = \frac{2 \mathbf{y}^\top (M \text{dag}(\mathbf{H}_1^\infty, \mathbf{H}_2^\infty, \dots, \mathbf{H}_M^\infty))^{-1} \mathbf{y}}{\rho} \\ & \quad + O\left(\frac{N^2 \sigma^2}{\lambda_0^2 \delta^2}\right) + \frac{\text{poly}(N, \lambda_0^{-1}, \delta^{-1})}{m^{\frac{1}{2}} \sigma} \\ & \leq \frac{2 \mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}}{\rho} + O\left(\frac{N^2 \sigma^2}{\lambda_0^2 \delta^2}\right) + \frac{\text{poly}(N, \lambda_0^{-1}, \delta^{-1})}{m^{\frac{1}{2}} \sigma}. \end{aligned}$$

The proof concludes by the setting of m and σ . □

Proof of Lemma 5. We bound $\|\tilde{f}(\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{x}_{ij}) - \tilde{\mathbf{y}}\|_2^2$ by first expanding the difference as a sum of squared inner products and then applying $\|\phi_{\mathbf{a}}(\mathbf{W}, \mathbf{x})\|_2 \leq 1$ from Lemma 1.

$$\begin{aligned} \|\tilde{f}(\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{x}_{ij}) - \tilde{\mathbf{y}}\|_2^2 &= \sum_{i=1}^M \sum_{j=1}^n (\phi_{\mathbf{a}}(\tilde{\mathbf{W}}_i, \mathbf{x}_{ij})^\top (\bar{\mathbf{W}}_{\mathcal{P}} - \tilde{\mathbf{W}}_i))^2 \\ &\leq \sum_{i=1}^M \sum_{j=1}^n \|\phi_{\mathbf{a}}(\tilde{\mathbf{W}}_i, \mathbf{x}_{ij})\|_2^2 \|\bar{\mathbf{W}}_{\mathcal{P}} - \tilde{\mathbf{W}}_i\|_2^2 \leq \sum_{i=1}^M \sum_{j=1}^n \|\bar{\mathbf{W}}_{\mathcal{P}} - \tilde{\mathbf{W}}_i\|_2^2 \end{aligned}$$

Triangle inequality gives a further bound via $\|\tilde{\mathbf{W}}_i - \mathbf{W}^{(0)}\|_2$ and $\|\bar{\mathbf{W}}_{\mathcal{P}} - \mathbf{W}^{(0)}\|_2$.

$$\begin{aligned} & \|\tilde{f}(\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{x}_{ij}) - \tilde{\mathbf{y}}\|_2^2 \\ & \leq 2 \sum_{i=1}^M \sum_{j=1}^n (\|\bar{\mathbf{W}}_{\mathcal{P}} - \mathbf{W}^{(0)}\|_2^2 + \|\tilde{\mathbf{W}}_i - \mathbf{W}^{(0)}\|_2^2) \end{aligned}$$

By Lemma 4 and the setting of m, η, δ , we obtain

$$\begin{aligned} & \|\tilde{f}(\bar{\mathbf{W}}_{\mathcal{P}}, \mathbf{x}_{ij}) - \tilde{\mathbf{y}}\|_2^2 \\ & \leq \sum_{i=1}^M \sum_{j=1}^n O\left(\frac{\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}}{\rho} + \mathbf{y}_i^\top (\mathbf{H}_i^\infty)^{-1} \mathbf{y}_i\right) \\ & = O\left(N \left(\frac{\mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}}{\rho} + \mathbf{y}^\top \frac{(\text{dag}(\mathbf{H}_1^\infty, \mathbf{H}_2^\infty, \dots, \mathbf{H}_M^\infty))^{-1}}{M} \mathbf{y}\right)\right) \\ & = O\left(\frac{N}{\rho} \mathbf{y}^\top (\mathbf{H}^\infty)^{-1} \mathbf{y}\right), \end{aligned}$$

□

Proof of Lemma 6. By Assumption 4 and Cauchy's interlacing theorem, we have $\lambda_{\min}(\mathbf{H}_{\mathcal{P}}^\infty) \geq \lambda_0$. For any entry (k, l) ,

$$\mathbf{H}_{\mathcal{P}}^{(0)}(k, l) = \frac{\mathbf{x}_{k'k''}^\top \mathbf{x}_{l'l''}}{m} \sum_{r=1}^m \mathbb{1}\{\mathbf{x}_{k'k''}^\top \mathbf{w}_r^{(0)} \geq 0\} \mathbb{1}\{\mathbf{x}_{l'l''}^\top \mathbf{w}_r^{(0)} \geq 0\},$$

where $k' = i_{\lceil k/n \rceil}$, $k'' = k - \lceil k/n \rceil + 1$, $l' = i_{\lceil l/n \rceil}$, $l'' = l - \lceil l/n \rceil + 1$. Thus $\mathbf{H}_{\mathcal{P}}^{(0)}(k, l)$ is an average of m bounded independent variables. Applying Hoeffding's inequality yields

$$|\mathbf{H}_{\mathcal{P}}^{(0)}(k, l) - \mathbf{H}_{\mathcal{P}}^{\infty}(k, l)| \leq 4\sqrt{\log(\rho N/\delta)/m}.$$

for all (k, l) with probability at least $1 - \delta$. Therefore,

$$\|\mathbf{H}_{\mathcal{P}}^{(0)} - \mathbf{H}_{\mathcal{P}}^{\infty}\|_2^2 \leq \|\mathbf{H}_{\mathcal{P}}^{(0)} - \mathbf{H}_{\mathcal{P}}^{\infty}\|_F^2 \leq 16(\rho N)^2 \log(\rho N/\delta)/m.$$

Setting $m = \Omega\left(\frac{(\rho N)^2 \log(\rho N/\delta)}{\lambda_0^2}\right)$ completes the proof. $\square$

Proof of Lemma 7. The (k, l) -th entry of $\tilde{\mathbf{H}}_{\mathcal{P}}$ is given by

$$\frac{\mathbf{x}_{k'k''}^{\top} \mathbf{x}_{k'k''}}{m} \sum_{r=1}^m \mathbb{1}\{\mathbf{x}_{k'k''}^{\top} \tilde{\mathbf{w}}_{k',r} \geq 0\} \cdot \mathbb{1}\{\mathbf{x}_{l'l''}^{\top} \tilde{\mathbf{w}}_{l',r} \geq 0\},$$

where $k' = i_{\lceil k/n \rceil}$, $k'' = k - \lceil k/n \rceil + 1$, $l' = i_{\lceil l/n \rceil}$, $l'' = l - \lceil l/n \rceil + 1$. We have

$$\begin{aligned} & \mathbb{E}[|\tilde{\mathbf{H}}_{\mathcal{P}}(k, l) - \mathbf{H}_{\mathcal{P}}^{(0)}(k, l)|] \\ &= \mathbb{E}\left[\left|\frac{\mathbf{x}_{k'k''}^{\top} \mathbf{x}_{k'k''}}{m} \sum_{r=1}^m \mathbb{1}\{\mathbf{x}_{k'k''}^{\top} \tilde{\mathbf{w}}_{k',r} \geq 0\} \mathbb{1}\{\mathbf{x}_{l'l''}^{\top} \tilde{\mathbf{w}}_{l',r} \geq 0\}\right.\right. \\ & \quad \left.\left. \neq \mathbb{1}\{\mathbf{x}_{k'k''}^{\top} \mathbf{w}_r^{(0)} \geq 0\} \mathbb{1}\{\mathbf{x}_{l'l''}^{\top} \mathbf{w}_r^{(0)} \geq 0\}\right|\right] \\ & \leq \mathbb{E}\left[\frac{1}{m} \sum_{r=1}^m \mathbb{1}\{\mathbb{1}\{\mathbf{x}_{k'k''}^{\top} \tilde{\mathbf{w}}_{k',r} \geq 0\} \mathbb{1}\{\mathbf{x}_{l'l''}^{\top} \tilde{\mathbf{w}}_{l',r} \geq 0\}\right. \\ & \quad \left. \neq \mathbb{1}\{\mathbf{x}_{k'k''}^{\top} \mathbf{w}_r^{(0)} \geq 0\} \mathbb{1}\{\mathbf{x}_{l'l''}^{\top} \mathbf{w}_r^{(0)} \geq 0\}\}\right] \\ & \leq \mathbb{E}\left[\frac{1}{m} \sum_{r=1}^m (\mathbb{1}\{\mathbb{1}\{\mathbf{x}_{k'k''}^{\top} \tilde{\mathbf{w}}_{k',r} \geq 0\} \neq \mathbb{1}\{\mathbf{x}_{l'l''}^{\top} \tilde{\mathbf{w}}_{l',r} \geq 0\}\} \right. \\ & \quad \left. + \mathbb{1}\{\mathbb{1}\{\mathbf{x}_{k'k''}^{\top} \mathbf{w}_r^{(0)} \geq 0\} \neq \mathbb{1}\{\mathbf{x}_{l'l''}^{\top} \mathbf{w}_r^{(0)} \geq 0\}\})\right] \\ & \leq \mathbb{E}\left[\frac{1}{m} \sum_{r=1}^m (\mathbb{1}\{\mathcal{E}_{k'k''r}(R_0)\} + \mathbb{1}\{\mathcal{E}_{l'l''r}(R_0)\})\right] \\ &= \frac{1}{m} \sum_{r=1}^m (\mathbb{P}(\mathcal{E}_{k'k''r}(R_0)) + \mathbb{P}(\mathcal{E}_{l'l''r}(R_0))) \leq \frac{4R_0}{\sqrt{2\pi}\sigma}. \end{aligned}$$

Then, we bound the Frobenius norm,

$$\begin{aligned} \mathbb{E}[\|\tilde{\mathbf{H}}_{\mathcal{P}} - \mathbf{H}_{\mathcal{P}}^{(0)}\|_F] &\leq \mathbb{E}\left[\sum_{k,l \in \mathcal{P}} |\tilde{\mathbf{H}}_{\mathcal{P}}(k, l) - \mathbf{H}_{\mathcal{P}}^{(0)}(k, l)|\right] \\ &\leq 4(\rho N)^2 R_0/(\sqrt{2\pi}\sigma). \end{aligned}$$

Using Markov's inequality, with probability $1 - \delta$, we get

$$\|\tilde{\mathbf{H}}_{\mathcal{P}} - \mathbf{H}_{\mathcal{P}}^{(0)}\|_F \leq \mathbb{E}[\|\tilde{\mathbf{H}}_{\mathcal{P}} - \mathbf{H}_{\mathcal{P}}^{(0)}\|_F]/\delta \leq 4(\rho N)^2 R_0/(\sqrt{2\pi}\delta\sigma).$$

Thus,

$$\|\tilde{\mathbf{H}}_{\mathcal{P}} - \mathbf{H}_{\mathcal{P}}^{(0)}\|_2 \leq \|\tilde{\mathbf{H}}_{\mathcal{P}} - \mathbf{H}_{\mathcal{P}}^{(0)}\|_F \leq 4(\rho N)^2 R_0/(\sqrt{2\pi}\delta\sigma).$$

By Lemma 6 and the setting of m , we have

$$\lambda_{\min}(\tilde{\mathbf{H}}_{\mathcal{P}}) \geq \lambda_{\min}(\mathbf{H}_{\mathcal{P}}^{(0)}) - 4(\rho N)^2 R_0/(\sqrt{2\pi}\delta\sigma) \geq \lambda_0/2.$$

To get upper bound of $\lambda_{\max}(\tilde{\mathbf{H}}_{\mathcal{P}})$, we have

$$\begin{aligned} \|\tilde{\mathbf{H}}_{\mathcal{P}}\|_2 &= \|\tilde{\mathbf{A}}_{\mathcal{P}} \tilde{\mathbf{A}}_{\mathcal{P}}^{\top}\|_2 = \|\tilde{\mathbf{A}}_{\mathcal{P}}^{\top} \tilde{\mathbf{A}}_{\mathcal{P}}\|_2 \\ &= \left\| \sum_{i \in \mathcal{P}} \sum_{j=1}^n \phi_{\mathbf{a}}(\tilde{\mathbf{W}}_i, x_{ij}) \phi_{\mathbf{a}}(\tilde{\mathbf{W}}_i, x_{ij})^{\top} \right\|_2 \\ &\leq \sum_{i \in \mathcal{P}} \sum_{j=1}^n \|\phi_{\mathbf{a}}(\tilde{\mathbf{W}}_i, x_{ij}) \phi_{\mathbf{a}}(\tilde{\mathbf{W}}_i, x_{ij})^{\top}\|_2 \\ &\leq \sum_{i \in \mathcal{P}} \sum_{j=1}^n \|\phi_{\mathbf{a}}(\tilde{\mathbf{W}}_i, x_{ij})\|_2^2 \leq \rho N, \end{aligned}$$

where the last inequality uses Lemma 1. $\square$

Proof of Lemma 8. By [5, Lemma 1], $\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}}$ is equivalent to the solution obtained via a gradient descent (GD) procedure. Let $\tilde{\mathbf{W}}_{\mathcal{P}}^{[t]}$ be the t -th iterate of the GD procedure with the initialization $\tilde{\mathbf{W}}_{\mathcal{P}}^{[0]} = \bar{\mathbf{W}}_{\mathcal{P}}$. Define the vectorized output $\tilde{f}(t) \in \mathbb{R}^N$ by stacking $\tilde{f}(\tilde{\mathbf{W}}_{\mathcal{P}}^{[t]}, x_{ij})$ across all $i \in [M], j \in [n]$:

$$\tilde{f}(t) := [\tilde{f}(\tilde{\mathbf{W}}_{\mathcal{P}}^{[t]}, x_{11}), \tilde{f}(\tilde{\mathbf{W}}_{\mathcal{P}}^{[t]}, x_{12}), \dots, \tilde{f}(\tilde{\mathbf{W}}_{\mathcal{P}}^{[t]}, x_{Mn})]^\top \in \mathbb{R}^N.$$

Thus, GD produces the recurrence

$$\tilde{\mathbf{W}}_{\mathcal{P}}^{[t]} = \tilde{\mathbf{W}}_{\mathcal{P}}^{[t-1]} - \frac{\tilde{\eta}}{n} \tilde{\mathbf{A}}_{\mathcal{P}}^\top (\tilde{f}(t-1) - \tilde{\mathbf{y}}). \quad (19)$$

which implies the prediction evolution

$$\tilde{f}(t) = \tilde{f}(t-1) - \frac{\tilde{\eta}}{n} \tilde{\mathbf{H}}_{\mathcal{P}} (\tilde{f}(t-1) - \tilde{\mathbf{y}}). \quad (20)$$

Unrolling the recurrence, we obtain for any t ,

$$\tilde{f}(t) - \tilde{\mathbf{y}} = (\mathbf{I} - \frac{\tilde{\eta}}{n} \tilde{\mathbf{H}}_{\mathcal{P}}) (\tilde{f}(t-1) - \tilde{\mathbf{y}}) = (\mathbf{I} - \frac{\tilde{\eta}}{n} \tilde{\mathbf{H}}_{\mathcal{P}})^t (\tilde{f}(0) - \tilde{\mathbf{y}}). \quad (21)$$

The distance between the final global model and the initialization is expressed as a gradient series:

$$\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}} - \bar{\mathbf{W}}_{\mathcal{P}} = -\frac{\tilde{\eta}}{n} \sum_{t=0}^{\infty} \tilde{\mathbf{A}}_{\mathcal{P}}^\top (\mathbf{I} - \frac{\tilde{\eta}}{n} \tilde{\mathbf{H}}_{\mathcal{P}})^t (\tilde{f}(0) - \tilde{\mathbf{y}}). \quad (22)$$

Let $\mathbf{U} := \frac{\tilde{\eta}}{n} (\mathbf{I} - \frac{\tilde{\eta}}{n} \tilde{\mathbf{H}}_{\mathcal{P}})^t$ be the geometric series matrix. Then

$$\begin{aligned} \|\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}} - \bar{\mathbf{W}}_{\mathcal{P}}\|_2^2 &= \|\tilde{\mathbf{A}}_{\mathcal{P}}^\top \mathbf{U} (\tilde{f}(0) - \tilde{\mathbf{y}})\|_2^2 \\ &\leq (\tilde{f}(0) - \tilde{\mathbf{y}})^\top \mathbf{U} \tilde{\mathbf{H}}_{\mathcal{P}} \mathbf{U} (\tilde{f}(0) - \tilde{\mathbf{y}}). \end{aligned}$$

Using eigen-decomposition $\tilde{\mathbf{H}}_{\mathcal{P}} = \mathbf{V} \mathbf{D} \mathbf{V}^\top$, we obtain

$$\mathbf{U} = \frac{\tilde{\eta}}{n} \mathbf{V} \sum_{t=1}^{\infty} (\mathbf{I} - \frac{\tilde{\eta}}{n} \mathbf{D})^t \mathbf{V}^\top = \mathbf{V} \mathbf{D}^{-1} \mathbf{V}^\top = \tilde{\mathbf{H}}_{\mathcal{P}}^{-1},$$

The squared distance becomes

$$\|\bar{\mathbf{W}}_{\mathcal{P}}^{\text{F}} - \bar{\mathbf{W}}_{\mathcal{P}}\|_2^2 \leq (\tilde{f}(0) - \tilde{\mathbf{y}})^\top \tilde{\mathbf{H}}_{\mathcal{P}}^{-1} (\tilde{f}(0) - \tilde{\mathbf{y}}) \leq \frac{2}{\lambda_0} \|\tilde{f}(0) - \tilde{\mathbf{y}}\|_2^2,$$

where the last inequality is due to Lemma 7. Finally, Lemma 5 bounds $\|\tilde{f}(0) - \tilde{\mathbf{y}}\|_2^2$, which leads to the desired results. $\square$

Proof of Lemma 9. By (22), for each neuron r , we have,

$$\begin{aligned} &\|\bar{\mathbf{w}}_{\mathcal{P},r}^{\text{F}} - \bar{\mathbf{w}}_{\mathcal{P},r}\|_2 \\ &= \left\| \frac{\tilde{\eta}}{n} \tilde{\mathbf{A}}_{\mathcal{P}} [(r-1)p+1 : rp]^\top (\mathbf{I} - \frac{\tilde{\eta}}{n} \tilde{\mathbf{H}}_{\mathcal{P}})^t (\tilde{f}(0) - \tilde{\mathbf{y}}) \right\|_2 \\ &\leq \frac{2}{\lambda_0} \|\tilde{\mathbf{A}}_{\mathcal{P}} [(r-1)p+1 : rp]^\top\|_2 \|\tilde{f}(0) - \tilde{\mathbf{y}}\|_2 \\ &\leq \frac{2}{\lambda_0} \|\tilde{\mathbf{A}}_{\mathcal{P}} [(r-1)p+1 : rp]^\top\|_{\text{F}} \|\tilde{f}(0) - \tilde{\mathbf{y}}\|_2 \\ &\leq \frac{2\sqrt{\rho N}}{\lambda_0 \sqrt{m}} \|\tilde{f}(0) - \tilde{\mathbf{y}}\|_2, \end{aligned}$$

where we use the notation $\mathbf{A}[x : y]$ to denote the submatrix of $\mathbf{A}$ having row numbers x to y , and the last inequality follows from the definition of $\tilde{\mathbf{A}}_{\mathcal{P}}$ and the data normalization assumption (Assumption 2). Combing with Lemma (5), we completes the proof. $\square$

Proof of Lemma 10. Each mismatch requires a sign flip between $\bar{\mathbf{w}}_{\mathcal{P},r}^{\text{F}}$ or $\tilde{\mathbf{w}}_{i,r}$ and the shared initialization $\mathbf{w}_r^{(0)}$.

$$\begin{aligned} \mathbb{E}[|S_{ij}^{\perp}|] &= \sum_{r=1}^m \mathbb{P}(\mathbf{1}\{\mathbf{x}_{ij}^\top \bar{\mathbf{w}}_{\mathcal{P},r}^{\text{F}} \geq 0\} \neq \mathbf{1}\{\mathbf{x}_{ij}^\top \tilde{\mathbf{w}}_{i,r} \geq 0\}) \\ &\leq \sum_{r=1}^m \mathbb{P}(\{\mathbf{1}\{\mathbf{x}_{ij}^\top \bar{\mathbf{w}}_{\mathcal{P},r}^{\text{F}} \geq 0\} \neq \mathbf{1}\{\mathbf{x}_{ij}^\top \mathbf{w}_r^{(0)} \geq 0\}\} \\ &\quad \cup \{\mathbf{1}\{\mathbf{x}_{ij}^\top \tilde{\mathbf{w}}_{i,r} \geq 0\} \neq \mathbf{1}\{\mathbf{x}_{ij}^\top \mathbf{w}_r^{(0)} \geq 0\}\}) \end{aligned}$$

$$\begin{aligned}
&\leq \sum_{r=1}^m [\mathbb{P}(\{\mathbb{1}\{\mathbf{x}_{ij}^\top \tilde{\mathbf{w}}_{\mathcal{P},r}^\mathbf{F} \geq 0\} \neq \mathbb{1}\{\mathbf{x}_{ij}^\top \mathbf{w}_r^{(0)} \geq 0\}\}) \\
&\quad + \mathbb{P}(\{\mathbb{1}\{\mathbf{x}_{ij}^\top \tilde{\mathbf{w}}_{i,r} \geq 0\} \neq \mathbb{1}\{\mathbf{x}_{ij}^\top \mathbf{w}_r^{(0)} \geq 0\}\})] \\
&\leq \sum_{r=1}^m (\mathbb{P}(\mathcal{E}_{ijr}(R_0 + R_1)) + \mathbb{P}(\mathcal{E}_{ijr}(R_0))) \leq \frac{4m(R_0 + R_1)}{\sqrt{2\pi}\sigma}.
\end{aligned}$$

By applying Markov's inequality with failure probability δ/N and taking a union bound over all (i, j) pairs, we obtain the desired bound. $\square$

Proof of Lemma 13. By [38, Lemma 2], $\sup_{f \in \mathcal{F}} \{L_J^{f,\ell} - L_S^{f,\ell}\}$ is less than

$$2\mathbb{E}_{\{D_i, \ell_i\}} \left[\sup_{f \in \mathcal{F}} \frac{1}{M} \sum_{i=1}^M \ell_i \mathbb{E}_{\mathbf{z}_i \sim D_i} [\ell(f(\mathbf{W}, \mathbf{x}_i), y_i)] \right] + c \sqrt{\frac{\log \frac{2}{\delta}}{2P}}.$$

Let $\tilde{\ell}(\mathbf{W}, \mathbf{z}) := \ell(f(\mathbf{W}, \mathbf{x}), y)$, the expectation term is

$$\begin{aligned}
&\mathbb{E}_{D_1, \dots, D_P, \ell_1, \dots, \ell_P} \left[\sup_{f \in \mathcal{F}} \frac{1}{M} \sum_{i=1}^M \ell_i \mathbb{E}_{\mathbf{z}_i \sim D_i} [\ell(f(\mathbf{W}, \mathbf{x}_i), y_i)] \right] \\
&= \mathbb{E}_{\mathbf{z}_{11}, \dots, \mathbf{z}_{Pn}, \ell_1, \dots, \ell_P} \left[\sup_{f \in \mathcal{F}} \frac{1}{M} \sum_{i=1}^M \frac{1}{n} \sum_{j=1}^n \ell_i \tilde{\ell}(\mathbf{W}, \mathbf{z}_{ij}) \right] \\
&\leq \mathbb{E}_{\mathbf{z}_{11}, \dots, \mathbf{z}_{Pn}, \ell_{11}, \dots, \ell_{Pn}} \left[\sup_{f \in \mathcal{F}} \frac{1}{P} \sum_{i \in \mathcal{P}} \frac{1}{n} \sum_{j=1}^n \ell_{ij} \tilde{\ell}(\mathbf{W}, \mathbf{z}_{ij}) \right] \\
&= \mathbb{E}_{\mathbf{z}_{11}, \dots, \mathbf{z}_{Pn}} \left[\mathbb{E}_{\ell_{11}, \dots, \ell_{Pn}} \left[\sup_{f \in \mathcal{F}} \frac{1}{P} \sum_{i \in \mathcal{P}} \frac{1}{n} \sum_{j=1}^n \ell_{ij} \tilde{\ell}(\mathbf{W}, \mathbf{z}_{ij}) \right] \right] \\
&= \mathbb{E}_{\mathbf{z}_{11}, \dots, \mathbf{z}_{Pn}} [\mathcal{R}(\mathcal{F})] \leq \mathcal{R}(\mathcal{F}) + c \sqrt{\log(2/\delta)/(2N)},
\end{aligned}$$

where the last inequality follows from McDiarmid's inequality, since replacing one point $\mathbf{z}_{ij}$ changes $\mathcal{R}(\mathcal{F})$ by at most c/N . $\square$

APPENDIX B

ADDITIONAL EXPERIMENTAL DETAILS AND RESULTS

A. Additional Experimental Details

1) *Dataset and Models:* Beyond the default two-layer CNN, we further evaluate F-SOSA using an ImageNet-pretrained ResNet18 on CIFAR-10 and YOLOv8s [45] on BDD100K [46]. BDD100K is a large-scale real-world driving dataset containing 100,000 40-second videos recorded at 720p and 30 FPS across diverse road, weather, and lighting conditions. It supports ten autonomous-driving tasks, with bounding-box annotations for ten traffic-related object categories. For both settings, we simulate client heterogeneity using the same Dirichlet partitioning scheme ($\alpha \in \{0.1, 0.5, 0.8\}$; default $\alpha=0.5$).

2) *Hyperparameters:* For experiments on CIFAR-10 with ResNet18 and BDD100K with YOLOv8s, we set the local learning rate to $\eta = 0.001$ and the total number of clients to $M = 20$. For CIFAR-10, we set the number of local epochs to $K = 50$ and the batch size to 64; for BDD100K, we use $K = 20$ local epochs, a batch size of 32, and an input resolution of 640×640 . Both settings employ SGD, with momentum 0.9. For BDD100K, we additionally apply weight decay of 5×10^{-4} , a three-epoch warm-up, and cosine learning-rate decay.

3) *Metrics:* For the object-detection task on BDD100K, we use mAP_{50-95} as the primary metric to evaluate the detection utility of F-SOSA and the baselines. It is computed by evaluating average precision (AP) at IoU thresholds from 0.50 to 0.95 in increments of 0.05 and averaging the results across all thresholds and ten object categories. This metric jointly reflects object recognition and bounding-box localization accuracy, with higher values indicating better overall detection performance.

B. F-SOSA Performance Across Model Architectures

To demonstrate that F-SOSA generalizes across models with different capacities, we evaluate network architectures beyond the two-layer CNN, including ResNet18 on CIFAR-10 and YOLOv8s on the real-world autonomous-driving dataset BDD100K. We further evaluate F-SOSA under varying data heterogeneity levels ($\alpha \in \{0.1, 0.5, 0.8\}$) to assess its robustness across both model architectures and non-IID settings. As shown in Table V, on CIFAR-10 with ResNet18, F-SOSA consistently achieves the lowest communication cost of 462.44 MB, reducing communication by 47.6% relative to FedFisher, 48.3% relative to Dense, and 94.3% relative to FedLPA. It also achieves the lowest time cost across all heterogeneity levels, with reductions of up to 43.0%, 56.0%, and 91.1% compared with FedFisher, Dense, and FedLPA, respectively. Although F-SOSA trails the

TABLE V
PERFORMANCE COMPARISON OF F-SOSA AND BASELINES ACROSS CIFAR-10 AND BDD100K UNDER VARYING DIRICHLET HETEROGENEITY α . BEST RESULTS ARE IN BOLD; SECOND-BEST RESULTS ARE UNDERLINED.

Dataset	Metrics	$\alpha = 0.1$				$\alpha = 0.5$				$\alpha = 0.8$			
		FedFisher	FedLPA	Dense	F-SOSA	FedFisher	FedLPA	Dense	F-SOSA	FedFisher	FedLPA	Dense	F-SOSA
CIFAR10 (Resnet18)	Accuracy (%)	46.60	<u>45.18</u>	28.04	35.17	69.66	<u>68.41</u>	57.48	65.98	72.61	<u>70.55</u>	63.92	69.87
	Time (s)	<u>361.02</u>	2493.05	479.82	278.22	<u>362.14</u>	2355.83	474.14	210.34	<u>365.50</u>	1713.55	473.76	208.27
	Comm. (MB)	<u>882.16</u>	8066.21	895.13	462.44	<u>882.16</u>	8066.21	895.13	462.44	<u>882.16</u>	8066.21	895.13	462.44
	Time Efficiency	0.129	0.018	0.058	<u>0.126</u>	<u>0.192</u>	0.029	0.121	0.314	<u>0.199</u>	0.041	0.135	0.335
	Comm. Efficiency	<u>0.053</u>	0.006	0.031	0.076	<u>0.079</u>	0.008	0.064	0.143	<u>0.082</u>	0.009	0.071	0.151
BDD100K (YOLOv8s)	mAP ₅₀₋₉₅ (%)	<u>20.82</u>	20.98	20.76	20.78	21.18	<u>21.17</u>	19.54	21.18	21.21	<u>21.22</u>	21.13	21.30
	Time (min)	39.88	<u>36.41</u>	37.28	32.52	37.77	<u>31.32</u>	34.67	28.96	37.69	31.10	37.11	<u>34.31</u>
	Comm. (GB)	<u>0.87</u>	9.55	0.88	0.46	<u>0.87</u>	9.55	0.88	0.46	<u>0.87</u>	9.55	0.88	0.46
	Time Efficiency	0.52	<u>0.58</u>	0.56	0.64	0.56	<u>0.68</u>	0.61	0.73	0.56	0.68	0.57	<u>0.62</u>
	Comm. Efficiency	<u>23.93</u>	2.20	23.59	45.17	<u>24.34</u>	2.22	22.20	46.04	<u>24.38</u>	2.22	24.01	46.30

best-performing baseline under high heterogeneity ($\alpha = 0.1$), it still outperforms Dense by 7.13 percentage points. At $\alpha = 0.5$ and 0.8, its accuracy increases to 65.98% and 69.87%, narrowing the gap with FedFisher to 3.68 and 2.74 percentage points, respectively, while continuing to outperform Dense. Given its consistently lowest time and communication costs, these modest accuracy gaps represent a reasonable utility-efficiency trade-off. On BDD100K with YOLOv8s, F-SOSA maintains competitive detection performance, achieving an mAP₅₀₋₉₅ only 0.20 percentage points below the best result at $\alpha = 0.1$, tying for the best result at $\alpha = 0.5$, and obtaining the highest mAP₅₀₋₉₅ of 21.30% at $\alpha = 0.8$. Meanwhile, it consistently incurs the lowest communication cost of 0.46 GB, representing reductions of 47.1% over FedFisher, 47.7% over Dense, and 95.2% over FedLPA. F-SOSA also achieves the lowest time cost at $\alpha = 0.1$ and 0.5 and the second-lowest at $\alpha = 0.8$, while delivering the highest communication efficiency under all settings. These results demonstrate that F-SOSA remains effective with larger classification and detection models, offering a favorable utility-efficiency trade-off across different model architectures, tasks, and degrees of data heterogeneity.

C. Scalability of F-SOSA-U and F-SOSA-I Across Client Populations

To assess the scalability of F-SOSA-I/U with increasing edge-device density, we consider five total client populations, $\{20, 50, 100, 200, 500\}$. For each population, we execute 15 sequential incremental-learning requests and 15 sequential unlearning requests. Table VI reports the mean accuracy, time-cost savings relative to full retraining, and communication cost normalized to Retrain, which together capture utility preservation and system efficiency.

For incremental learning, F-SOSA-I maintains accuracy comparable to Retrain across the evaluated client scales. Even at 500 clients, the accuracy gap between F-SOSA-I and Retrain remains within 2.46 percentage points across all four datasets. In the 50-client case, F-SOSA-I achieves the highest accuracy on MNIST, SVHN, and CIFAR-10, reaching 94.35%, 68.95%, and 59.99%, respectively. As the system scales to 200 clients, it achieves the best performance on FEMNIST and CIFAR-10, with accuracies of 32.33% and 46.27%, respectively. For the largest population of 500 clients, the accuracy gap between F-SOSA-I and Retrain remains within 2.46 percentage points across all four datasets, while F-SOSA-I outperforms Fine-tuning by 2.45–10.90 percentage points. This utility advantage is particularly pronounced on the inherently heterogeneous FEMNIST dataset: as the client population increases from 100 to 500, F-SOSA-I maintains an accuracy of approximately 31%, whereas Fine-tuning remains around 20%. In terms of efficiency, the normalized communication cost of F-SOSA-I never exceeds $0.87\times$ that of Retrain and decreases to $0.67\times$ with 500 clients, corresponding to a 33% reduction. Although Fine-tuning incurs marginally lower communication overhead at some scales, these savings come at the cost of substantial accuracy degradation, particularly on MNIST, SVHN, and FEMNIST. F-SOSA-I therefore provides a more favorable utility-efficiency trade-off. Its time-cost savings remain positive across all configurations, ranging from 15.84% to 49.04%. Even with a population of 500 clients, F-SOSA-I saves 23.26%–49.04% of the retraining time, indicating that its computational benefits persist as the edge system becomes increasingly dense.

Regarding unlearning, F-SOSA-U exhibits similarly strong scalability. Across all evaluated client populations, its accuracy closely tracks or exceeds that of Retrain while consistently outperforming most unlearning baselines. The results for 100 clients show that F-SOSA-U achieves the highest accuracy on MNIST, SVHN, and CIFAR-10. When the client population reaches 500, F-SOSA-U surpasses Retrain on SVHN, FEMNIST, and CIFAR-10 and remains within 0.89 percentage points of Retrain on MNIST. In this largest-scale regime, F-SOSA-U also outperforms the strongest competing unlearning baseline by 1.24–4.85 percentage points across the four datasets. Its normalized communication cost remains below that of Retrain throughout the evaluated range. With 500 clients, F-SOSA-U reduces communication by 47% relative to Retrain and FATS-CU and by approximately 60% relative to FedEraser and FedRR, whose communication costs exceed $1.33\times$ that of Retrain. Moreover, F-SOSA-U continues to save 38.92%–43.35% of the retraining time at the maximum evaluated scale. Although FedEraser and FATS-CU achieve greater time-cost savings, their accuracy degrades more noticeably as the client population

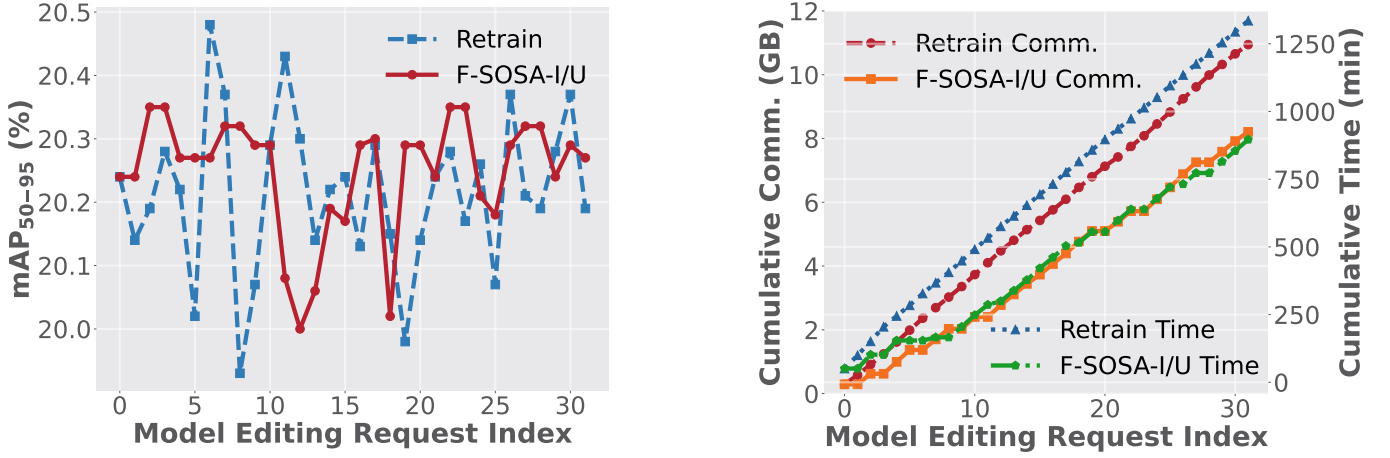


Fig. 7. mAP₅₀₋₉₅ (left) and cumulative cost (right) of F-SOSA-I/U vs. Retrain under simulator-generated model editing requests on the real driving dataset BDD100K.

increases. Consequently, F-SOSA-U provides a more favorable overall trade-off among utility, communication efficiency, and computational cost.

In addition, we note that the absolute accuracy on MNIST, SVHN, and CIFAR-10 decreases as the number of clients increases, whereas this trend is not observed on the inherently non-IID FEMNIST dataset. This behavior is a consequence of our data-partitioning strategy: increasing the number of clients reduces the amount of training data assigned to each client, thereby limiting the local data available for one-shot FL and subsequent editing. Importantly, F-SOSA-I and F-SOSA-U still achieve the best or near-best relative accuracy compared with their respective baselines across different client scales. This suggests that the decline is primarily associated with the reduced amount of local data per client, rather than a deterioration unique to F-SOSA-I/U.

Overall, these results show that the relative communication and computation benefits of F-SOSA-I/U remain substantial as the client population increases. Across the evaluated range of up to 500 clients, both editing algorithms preserve utility close to Retrain while continuing to reduce its communication and computation costs. This indicates that the efficiency advantage of selective exact editing remains robust under increasingly dense client populations.

D. Real-World Driving Case Study with BDD100K and SUMO Mobility

We further evaluate F-SOSA-I/U on a dynamic federated object-detection task by combining mobility traces generated by the Simulation of Urban MObility (SUMO) with the real-world BDD100K driving dataset. Following the mobility-trace setting described in section VI-C, the trace produces 31 sequential editing requests, comprising 16 client arrivals and 15 departures. We replay this request sequence on BDD100K using YOLOv8s and report the mAP₅₀₋₉₅ after each request, together with the cumulative computation time and communication cost, in Figure 7. F-SOSA-I/U closely tracks the utility of Retrain throughout the sequence, with a maximum mAP₅₀₋₉₅ difference of only 0.39 percentage points. Across the 31 editing requests, F-SOSA-I/U matches or outperforms Retrain after 19 requests and achieves a slightly higher average mAP₅₀₋₉₅ of 20.25%, compared with 20.21% for Retrain. After the final request, its mAP₅₀₋₉₅ reaches 20.27%, exceeding the 20.19% achieved by Retrain. F-SOSA-I/U also exhibits greater stability: its mAP₅₀₋₉₅ varies within a range of 0.35 percentage points, compared with 0.55 for Retrain, while its maximum request-to-request variation is 0.28 percentage points, versus 0.46 for Retrain.

More notably, F-SOSA-I/U provides substantial efficiency gains. As shown in the figure, its communication and time curves remain essentially flat for 8 of the 31 requests. When the client involved in a request is not selected for aggregation, Bernoulli sampling avoids unnecessary computation and communication. These skipped updates save 2.74 GB of communication and 343.73 minutes of computation, accounting for 100% and approximately 78.31% of the overall communication and time savings, respectively. The savings do not arise solely from skipped updates: across the remaining 23 requests that require model updates, F-SOSA-I/U saves an additional 95.19 minutes. This additional time saving occurs because some requests require F-SOSA-I/U to update only one client with a relatively small local dataset, whereas Retrain must wait for all participating clients to complete local training and is therefore limited by the slowest client. After the complete request sequence, F-SOSA-I/U incurs 8.21 GB of cumulative communication, compared with 10.94 GB for Retrain, corresponding to a 25.00% reduction. Its cumulative computation time decreases from 1335.21 to 896.28 minutes, yielding a 32.87% reduction. These absolute savings are particularly significant on BDD100K, since repeatedly retraining YOLOv8s and communicating its model updates over complex real-world driving data is considerably more expensive than editing lightweight classification models.

Taken together, F-SOSA-I/U preserves near-retraining detection utility while substantially reducing the cost of frequent recomputation, demonstrating its practical value in dynamic and communication-constrained edge environments.

TABLE VI

SCALABILITY OVER 15 INCREMENTAL-LEARNING OR UNLEARNING REQUESTS. WE REPORT MEAN ACCURACY (ACC.), MEAN TIME-COST SAVING (T-SAVE), AND MEAN COMMUNICATION COST NORMALIZED TO RETRAIN (COMM.; RETRAIN = 1.00). T-SAVE IS RELATIVE TO RETRAIN; HIGHER IS BETTER FOR ACC. AND T-SAVE, WHILE LOWER IS BETTER FOR COMM. BEST RESULTS ARE IN BOLD; SECOND-BEST RESULTS ARE UNDERLINED.

Clients	Method	MNIST			SVHN			FEMNIST			CIFAR10		
		Acc. (%)	T-save (%)	Comm. (×)	Acc. (%)	T-save (%)	Comm. (×)	Acc. (%)	T-save (%)	Comm. (×)	Acc. (%)	T-save (%)	Comm. (×)
(a) Incremental learning													
20	Retrain	96.33	0.00	1.00	73.07	0.00	1.00	27.06	0.00	1.00	68.51	0.00	1.00
	Fine-tuning	<u>95.74</u>	79.68	<u>0.78</u>	74.67	81.48	<u>0.78</u>	21.65	69.13	<u>0.78</u>	<u>67.86</u>	78.89	<u>0.78</u>
	F-SOSA-I	95.62	<u>20.83</u>	0.75	72.82	<u>37.84</u>	0.75	<u>25.64</u>	<u>19.67</u>	0.75	69.34	<u>45.26</u>	0.75
50	Retrain	94.11	0.00	1.00	66.61	0.00	1.00	30.22	0.00	1.00	57.11	0.00	1.00
	Fine-tuning	93.30	79.67	0.70	65.95	79.70	0.70	19.72	62.36	0.70	52.88	78.29	0.70
	F-SOSA-I	94.35	<u>21.64</u>	<u>0.72</u>	68.95	<u>38.63</u>	<u>0.72</u>	<u>30.09</u>	<u>20.47</u>	<u>0.72</u>	59.99	<u>46.08</u>	<u>0.72</u>
100	Retrain	92.81	0.00	1.00	62.55	0.00	1.00	31.24	0.00	1.00	56.01	0.00	1.00
	Fine-tuning	91.32	72.55	0.68	55.83	79.64	0.68	20.28	64.89	0.68	47.70	77.37	0.68
	F-SOSA-I	92.94	<u>16.93</u>	<u>0.87</u>	62.18	<u>34.86</u>	<u>0.87</u>	31.05	<u>15.84</u>	<u>0.87</u>	<u>54.22</u>	<u>41.87</u>	<u>0.87</u>
200	Retrain	91.30	0.00	1.00	36.56	0.00	1.00	30.65	0.00	1.00	43.65	0.00	1.00
	Fine-tuning	81.98	82.80	0.67	23.08	79.03	0.67	19.22	72.35	0.67	42.41	75.61	0.67
	F-SOSA-I	<u>90.22</u>	<u>18.25</u>	<u>0.87</u>	35.94	<u>36.04</u>	<u>0.87</u>	32.33	<u>17.16</u>	<u>0.87</u>	46.27	<u>43.18</u>	<u>0.87</u>
500	Retrain	85.38	0.00	1.00	19.60	0.00	1.00	31.88	0.00	1.00	35.65	0.00	1.00
	Fine-tuning	75.52	80.26	0.67	14.77	79.95	0.67	20.15	76.82	0.67	30.74	74.48	0.67
	F-SOSA-I	85.95	<u>24.17</u>	0.67	<u>17.88</u>	<u>41.63</u>	0.67	<u>31.05</u>	<u>23.26</u>	0.67	<u>33.19</u>	<u>49.04</u>	0.67
(b) Unlearning													
20	Retrain	95.73	0.00	1.00	65.04	0.00	1.00	26.12	0.00	1.00	64.75	0.00	1.00
	FedEraser	93.95	78.64	1.35	68.48	79.12	1.35	21.17	77.85	1.35	67.74	78.31	1.35
	FedRR	92.20	64.18	1.35	56.26	68.79	1.35	24.78	44.18	1.35	62.03	64.10	1.35
	FATS-CU	88.72	76.92	1.00	61.25	77.36	1.00	20.11	75.94	1.00	61.85	76.58	1.00
	F-SOSA-U	<u>94.87</u>	<u>44.73</u>	0.75	64.95	<u>45.81</u>	0.75	26.51	<u>41.62</u>	0.75	63.91	<u>43.79</u>	0.75
50	Retrain	94.87	0.00	1.00	63.73	0.00	1.00	26.49	0.00	1.00	56.93	0.00	1.00
	FedEraser	93.21	78.21	1.34	56.80	78.77	1.34	18.59	77.46	1.34	59.14	77.94	1.34
	FedRR	91.97	63.76	1.34	39.11	68.31	1.34	21.22	43.76	1.34	56.10	63.69	1.34
	FATS-CU	90.87	76.49	1.00	53.63	76.94	1.00	18.38	75.61	1.00	57.10	76.17	1.00
	F-SOSA-U	<u>94.28</u>	<u>44.16</u>	0.87	<u>62.88</u>	<u>45.22</u>	0.87	28.01	41.08	0.87	<u>57.86</u>	<u>43.25</u>	0.87
100	Retrain	<u>92.26</u>	0.00	1.00	<u>58.58</u>	0.00	1.00	30.68	0.00	1.00	<u>51.64</u>	0.00	1.00
	FedEraser	91.31	77.83	1.34	45.50	78.39	1.34	23.82	77.02	1.34	51.57	77.56	1.34
	FedRR	90.04	61.21	1.34	26.75	65.18	1.34	22.48	43.11	1.34	50.38	63.02	1.34
	FATS-CU	90.24	<u>76.12</u>	1.00	44.65	<u>76.58</u>	1.00	23.51	<u>75.18</u>	1.00	50.67	<u>75.83</u>	1.00
	F-SOSA-U	92.33	43.58	0.80	59.32	<u>44.76</u>	0.80	<u>29.81</u>	<u>40.54</u>	0.80	51.72	<u>42.61</u>	0.80
200	Retrain	91.21	0.00	1.00	40.09	0.00	1.00	28.41	0.00	1.00	48.61	0.00	1.00
	FedEraser	87.56	77.34	1.33	25.56	77.95	1.33	24.00	76.58	1.33	43.29	77.18	1.33
	FedRR	86.95	59.84	1.33	19.59	62.46	1.33	25.90	42.42	1.33	40.06	62.74	1.33
	FATS-CU	86.23	<u>75.67</u>	1.00	24.08	<u>76.11</u>	1.00	23.24	<u>74.72</u>	1.00	42.64	<u>75.41</u>	1.00
	F-SOSA-U	<u>90.28</u>	<u>42.91</u>	0.67	<u>39.62</u>	<u>44.13</u>	0.67	29.09	<u>39.86</u>	0.67	<u>46.22</u>	<u>41.94</u>	0.67
500	Retrain	86.62	0.00	1.00	19.66	0.00	1.00	30.45	0.00	1.00	33.71	0.00	1.00
	FedEraser	80.99	76.82	1.33	19.59	77.46	1.33	25.11	76.11	1.33	31.90	76.73	1.33
	FedRR	80.71	59.37	1.33	19.59	61.92	1.33	26.18	41.57	1.33	26.84	62.48	1.33
	FATS-CU	79.63	<u>75.13</u>	1.00	19.66	<u>75.62</u>	1.00	25.38	<u>74.29</u>	1.00	31.71	<u>74.96</u>	1.00
	F-SOSA-U	<u>85.73</u>	<u>41.76</u>	0.53	20.90	43.35	0.53	31.03	38.92	0.53	34.64	40.87	0.53

E. Detailed Communication-Cost Breakdown

To provide a clear breakdown of the communication costs, Table VII reports the per-round costs for selected and unselected clients in F-SOSA with $M = 20$ and $\rho = 0.5$. Under this configuration, ten clients are selected, while the remaining ten are unselected. The server first broadcasts the global model to all clients. Then, only the selected clients upload their local models and approximate Fisher information, represented by either Diag or K-FAC, whereas the unselected clients do not upload any information. Note that although four experiments in the table use two-layer CNNs, their architectures and parameter counts differ, resulting in different model sizes.

As shown in the table, the uplink cost of the Diag Fisher information is equal to that of the 16-bit quantized local model for all model architectures. Their combined size is therefore equal to the communication size of one FP32 model. This is because 16-bit quantization is applied to both the local models and the Diag Fisher information in our implementation. After receiving the quantized information, the server reconstructs the values and performs aggregation. More specifically, each local model and its diagonal Fisher information are quantized layer by layer to 16-bit precision before aggregation. They have the same communication cost because both contain one value for each model parameter. For K-FAC, the Fisher factors are further compressed using low-rank decomposition and quantization. Therefore, the K-FAC communication cost depends on the model architecture and the dimensions of its Fisher factors, and may differ slightly from the Diag cost. The table also shows that

TABLE VII

DETAILED COMMUNICATION COSTS (MB) OF F-SOSA FOR $M = 20$ AND $\rho = 0.5$, WHERE M IS THE TOTAL NUMBER OF CLIENTS AND ρ IS THE CLIENT SELECTION RATIO. THE DOWNLINK CORRESPONDS TO ONE BROADCAST OF THE GLOBAL MODEL TO ALL ACTIVE CLIENTS. MODEL UP., DIAG UP., AND K-FAC UP. DENOTE THE AGGREGATE UPLINK COMMUNICATION OF THE SELECTED CLIENTS FOR LOCAL MODELS, DIAGONAL FISHER INFORMATION, AND KRONECKER-FACTORED APPROXIMATE CURVATURE (K-FAC) FISHER INFORMATION, RESPECTIVELY. UNSELECTED CLIENTS INCUR NO UPLINK COMMUNICATION. THE FINAL COLUMN REPORTS THE TOTAL COMMUNICATION COSTS OF THE DIAG AND K-FAC VARIANTS.

Dataset	Model	Broadcast	Selected Clients			Unselected Clients	Round Total
		Downlink	Model Up.	Diag Up.	K-FAC Up.	Uplink	
MNIST	Two-Layer CNN	1.61	8.04	8.04	8.81	0	17.69/18.46
SVHN	Two-Layer CNN	3.58	17.92	17.92	18.81	0	39.42/40.31
CIFAR10	Two-Layer CNN	3.58	17.92	17.92	18.81	0	39.42/40.31
CIFAR10	ResNet18-Pretrained	42.63	213.13	213.13	206.68	0	468.89/462.44
FEMNIST	Two-Layer CNN	25.18	125.91	125.91	138.59	0	277.00/289.68
BDD100K	YOLOv8s	42.49	212.47	212.47	211.99	0	467.43/466.95

unselected clients incur no uplink cost. The final column reports the total communication costs of the Diag and K-FAC variants, respectively. The two-layer CNN has the same communication cost on SVHN and CIFAR-10 because the two settings use the same model architecture and number of parameters.

Furthermore, as shown in Tables VIII–XII, we report a detailed breakdown of the actual amount of data transmitted during the initial training round and 31 sequential editing requests. The communication cost includes three parts: global-model broadcasts, local-model uploads from selected clients, and approximate Fisher information uploads from selected clients. Unselected clients do not upload models or Fisher information. Therefore, the reported totals include all communication required to exchange the current model and Fisher information between the server and clients. With this complete accounting, F-SOSA-I/U reduces the cumulative communication cost from 424.48 to 329.61 MB on MNIST, from 966.63 to 717.63 MB on SVHN, from 6942.18 to 5416.97 MB on FEMNIST, from 966.63 to 750.59 MB on CIFAR-10, and from 11207.23 to 8405.42 MB on BDD100K. These results correspond to reductions of 22.35%, 25.76%, 21.97%, 22.35%, and 25.00%, respectively. The detailed BDD100K results further show that the F-SOSA-I/U total of 8405.42 MB consists of 1019.76 MB of global-model broadcasts, 3696.95 MB of local-model uploads, and 3688.71 MB of Fisher-information uploads. In comparison, retraining requires 1359.68, 4929.27, and 4918.28 MB for the same three parts, respectively. Thus, the reduction is obtained even after including the large Fisher-information uploads and all model transfers. When an editing request requires an update, F-SOSA-I/U has the same per-round communication components as retraining; however, it avoids the full broadcast and upload process when the request does not require updating the selected clients. This saves 94.87, 249.00, 1525.21, 216.04, and 2801.81 MB on MNIST, SVHN, FEMNIST, CIFAR-10, and BDD100K, respectively. SVHN achieves the largest relative reduction of 25.76%, while BDD100K achieves the largest absolute reduction of 2801.81 MB. Overall, F-SOSA-I/U provides consistent communication savings across datasets, ranging from 21.97% on FEMNIST to 25.76% on SVHN, while preserving performance comparable to retraining.

TABLE VIII

DETAILED COMMUNICATION COSTS (MB) OF SEQUENTIAL EDITING ON MNIST WITH $\rho = 0.5$. EACH COMMUNICATION ENTRY IS REPORTED AS F-SOSA-I/U VERSUS RETRAIN, WITH THE FIRST VALUE CORRESPONDING TO F-SOSA-I/U AND THE SECOND TO RETRAIN. MODEL UP. AND FISHER UP. DENOTE THE COMMUNICATION COSTS OF UPLOADING THE MODEL AND THE APPROXIMATE FISHER INFORMATION, RESPECTIVELY. IN THE REQUEST COLUMN, THE NUMBER IDENTIFIES THE AFFECTED CLIENT, WHILE + AND - DENOTE CLIENT ADDITION AND CLIENT REMOVAL, RESPECTIVELY.

Request	Broadcast	Selected Clients		Unselected Clients	Round Total
	Downlink	Model Up.	Fisher Up.	Uplink	
Initial	1.61/1.61	4.82/4.82	4.82/4.82	0.00/0.00	11.25/11.25
12+	0.00/1.61	0.00/4.82	0.00/4.82	0.00/0.00	0.00/11.25
13+	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
14+	0.00/1.61	0.00/5.63	0.00/5.63	0.00/0.00	0.00/12.87
15+	1.61/1.61	6.43/6.43	6.43/6.43	0.00/0.00	14.47/14.47
16+	0.00/1.61	0.00/6.43	0.00/6.43	0.00/0.00	0.00/14.47
3-	0.00/1.61	0.00/6.43	0.00/6.43	0.00/0.00	0.00/14.47
6-	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
7-	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
17+	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
18+	1.61/1.61	6.43/6.43	6.43/6.43	0.00/0.00	14.47/14.47
19+	0.00/1.61	0.00/6.43	0.00/6.43	0.00/0.00	0.00/14.47
1-	1.61/1.61	6.43/6.43	6.43/6.43	0.00/0.00	14.47/14.47
11-	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
16-	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
2-	1.61/1.61	4.82/4.82	4.82/4.82	0.00/0.00	11.25/11.25
20+	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
21+	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
22+	1.61/1.61	6.43/6.43	6.43/6.43	0.00/0.00	14.47/14.47
4-	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
8-	0.00/1.61	0.00/5.63	0.00/5.63	0.00/0.00	0.00/12.87
9-	1.61/1.61	4.82/4.82	4.82/4.82	0.00/0.00	11.25/11.25
23+	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
24+	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
25+	1.61/1.61	6.43/6.43	6.43/6.43	0.00/0.00	14.47/14.47
26+	1.61/1.61	6.43/6.43	6.43/6.43	0.00/0.00	14.47/14.47
27+	1.61/1.61	7.23/7.23	7.23/7.23	0.00/0.00	16.07/16.07
0-	1.61/1.61	6.43/6.43	6.43/6.43	0.00/0.00	14.47/14.47
10-	0.00/1.61	0.00/6.43	0.00/6.43	0.00/0.00	0.00/14.47
17-	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
18-	1.61/1.61	5.63/5.63	5.63/5.63	0.00/0.00	12.87/12.87
19-	1.61/1.61	4.82/4.82	4.82/4.82	0.00/0.00	11.25/11.25
Total	40.25/51.52	144.68/186.48	144.68/186.48	0.00/0.00	329.61/424.48

TABLE IX

DETAILED COMMUNICATION COSTS (MB) OF SEQUENTIAL EDITING ON SVHN WITH $\rho = 0.5$. EACH COMMUNICATION ENTRY IS REPORTED AS F-SOSA-I/U VERSUS RETRAIN, WITH THE FIRST VALUE CORRESPONDING TO F-SOSA-I/U AND THE SECOND TO RETRAIN. MODEL UP. AND FISHER UP. DENOTE THE COMMUNICATION COSTS OF UPLOADING THE MODEL AND THE APPROXIMATE FISHER INFORMATION, RESPECTIVELY. IN THE REQUEST COLUMN, THE NUMBER IDENTIFIES THE AFFECTED CLIENT, WHILE + AND - DENOTE CLIENT ADDITION AND CLIENT REMOVAL, RESPECTIVELY.

Request	Broadcast	Selected Clients		Unselected Clients	Round Total
	Downlink	Model Up.	Fisher Up.	Uplink	
Initial	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
12+	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
13+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
14+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
15+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
16+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
3-	0.00/3.58	0.00/14.34	0.00/15.04	0.00/0.00	0.00/32.96
6-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
7-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
17+	0.00/3.58	0.00/12.55	0.00/13.16	0.00/0.00	0.00/29.29
18+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
19+	0.00/3.58	0.00/14.34	0.00/15.04	0.00/0.00	0.00/32.96
1-	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
11-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
16-	0.00/3.58	0.00/12.55	0.00/13.16	0.00/0.00	0.00/29.29
2-	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
20+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
21+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
22+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
4-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
8-	0.00/3.58	0.00/12.55	0.00/13.16	0.00/0.00	0.00/29.29
9-	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
23+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
24+	0.00/3.58	0.00/12.55	0.00/13.16	0.00/0.00	0.00/29.29
25+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
26+	0.00/3.58	0.00/14.34	0.00/15.04	0.00/0.00	0.00/32.96
27+	3.58/3.58	16.13/16.13	16.93/16.93	0.00/0.00	36.64/36.64
0-	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
10-	0.00/3.58	0.00/14.34	0.00/15.04	0.00/0.00	0.00/32.96
17-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
18-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
19-	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
Total	85.92/114.56	308.25/415.78	323.46/436.29	0.00/0.00	717.63/966.63

TABLE X

DETAILED COMMUNICATION COSTS (MB) OF SEQUENTIAL EDITING ON FEMNIST WITH $\rho = 0.5$. EACH COMMUNICATION ENTRY IS REPORTED AS F-SOSA-I/U VERSUS RETRAIN, WITH THE FIRST VALUE CORRESPONDING TO F-SOSA-I/U AND THE SECOND TO RETRAIN. MODEL UP. AND FISHER UP. DENOTE THE COMMUNICATION COSTS OF UPLOADING THE MODEL AND THE APPROXIMATE FISHER INFORMATION, RESPECTIVELY. IN THE REQUEST COLUMN, THE NUMBER IDENTIFIES THE AFFECTED CLIENT, WHILE + AND - DENOTE CLIENT ADDITION AND CLIENT REMOVAL, RESPECTIVELY.

Request	Broadcast	Selected Clients		Unselected Clients	Round Total
	Downlink	Model Up.	Fisher Up.	Uplink	
Initial	25.18/25.18	75.55/75.55	83.15/83.15	0.00/0.00	183.88/183.88
12+	0.00/25.18	0.00/75.55	0.00/83.15	0.00/0.00	0.00/183.88
13+	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
14+	0.00/25.18	0.00/88.14	0.00/97.01	0.00/0.00	0.00/210.33
15+	25.18/25.18	100.73/100.73	110.87/110.87	0.00/0.00	236.78/236.78
16+	0.00/25.18	0.00/100.73	0.00/110.87	0.00/0.00	0.00/236.78
3-	25.18/25.18	100.73/100.73	110.87/110.87	0.00/0.00	236.78/236.78
6-	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
7-	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
17+	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
18+	25.18/25.18	100.73/100.73	110.87/110.87	0.00/0.00	236.78/236.78
19+	25.18/25.18	100.73/100.73	110.87/110.87	0.00/0.00	236.78/236.78
1-	0.00/25.18	0.00/100.73	0.00/110.87	0.00/0.00	0.00/236.78
11-	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
16-	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
2-	25.18/25.18	75.55/75.55	83.15/83.15	0.00/0.00	183.88/183.88
20+	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
21+	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
22+	25.18/25.18	100.73/100.73	110.87/110.87	0.00/0.00	236.78/236.78
4-	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
8-	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
9-	25.18/25.18	75.55/75.55	83.15/83.15	0.00/0.00	183.88/183.88
23+	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
24+	0.00/25.18	0.00/88.14	0.00/97.01	0.00/0.00	0.00/210.33
25+	25.18/25.18	100.73/100.73	110.87/110.87	0.00/0.00	236.78/236.78
26+	25.18/25.18	100.73/100.73	110.87/110.87	0.00/0.00	236.78/236.78
27+	25.18/25.18	113.32/113.32	124.73/124.73	0.00/0.00	263.23/263.23
0-	25.18/25.18	100.73/100.73	110.87/110.87	0.00/0.00	236.78/236.78
10-	0.00/25.18	0.00/100.73	0.00/110.87	0.00/0.00	0.00/236.78
17-	25.18/25.18	88.14/88.14	97.01/97.01	0.00/0.00	210.33/210.33
18-	0.00/25.18	0.00/88.14	0.00/97.01	0.00/0.00	0.00/210.33
19-	25.18/25.18	75.55/75.55	83.15/83.15	0.00/0.00	183.88/183.88
Total	629.50/805.76	2279.04/2921.20	2508.43/3215.22	0.00/0.00	5416.97/6942.18

TABLE XI

DETAILED COMMUNICATION COSTS (MB) OF SEQUENTIAL EDITING ON CIFAR10 WITH $\rho = 0.5$. EACH COMMUNICATION ENTRY IS REPORTED AS F-SOSA-I/U VERSUS RETRAIN, WITH THE FIRST VALUE CORRESPONDING TO F-SOSA-I/U AND THE SECOND TO RETRAIN. MODEL UP. AND FISHER UP. DENOTE THE COMMUNICATION COSTS OF UPLOADING THE MODEL AND THE APPROXIMATE FISHER INFORMATION, RESPECTIVELY. IN THE REQUEST COLUMN, THE NUMBER IDENTIFIES THE AFFECTED CLIENT, WHILE + AND - DENOTE CLIENT ADDITION AND CLIENT REMOVAL, RESPECTIVELY.

Request	Broadcast	Selected Clients		Unselected Clients	Round Total
	Downlink	Model Up.	Fisher Up.	Uplink	
Initial	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
12+	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
13+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
14+	0.00/3.58	0.00/12.55	0.00/13.16	0.00/0.00	0.00/29.29
15+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
16+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
3-	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
6-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
7-	0.00/3.58	0.00/12.55	0.00/13.16	0.00/0.00	0.00/29.29
17+	0.00/3.58	0.00/12.55	0.00/13.16	0.00/0.00	0.00/29.29
18+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
19+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
1-	0.00/3.58	0.00/14.34	0.00/15.04	0.00/0.00	0.00/32.96
11-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
16-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
2-	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
20+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
21+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
22+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
4-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
8-	0.00/3.58	0.00/12.55	0.00/13.16	0.00/0.00	0.00/29.29
9-	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
23+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
24+	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
25+	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
26+	0.00/3.58	0.00/14.34	0.00/15.04	0.00/0.00	0.00/32.96
27+	3.58/3.58	16.13/16.13	16.93/16.93	0.00/0.00	36.64/36.64
0-	3.58/3.58	14.34/14.34	15.04/15.04	0.00/0.00	32.96/32.96
10-	0.00/3.58	0.00/14.34	0.00/15.04	0.00/0.00	0.00/32.96
17-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
18-	3.58/3.58	12.55/12.55	13.16/13.16	0.00/0.00	29.29/29.29
19-	3.58/3.58	10.75/10.75	11.28/11.28	0.00/0.00	25.61/25.61
Total	89.50/114.56	322.59/415.78	338.50/436.29	0.00/0.00	750.59/966.63

TABLE XII

DETAILED COMMUNICATION COSTS (MB) OF SEQUENTIAL EDITING ON BDD100K WITH $\rho = 0.5$. EACH COMMUNICATION ENTRY IS REPORTED AS F-SOSA-I/U VERSUS RETRAIN, WITH THE FIRST VALUE CORRESPONDING TO F-SOSA-I/U AND THE SECOND TO RETRAIN. MODEL UP. AND FISHER UP. DENOTE THE COMMUNICATION COSTS OF UPLOADING THE MODEL AND THE APPROXIMATE FISHER INFORMATION, RESPECTIVELY. IN THE REQUEST COLUMN, THE NUMBER IDENTIFIES THE AFFECTED CLIENT, WHILE + AND - DENOTE CLIENT ADDITION AND CLIENT REMOVAL, RESPECTIVELY.

Request	Broadcast	Selected Clients		Unselected Clients	Round Total
	Downlink	Model Up.	Fisher Up.	Uplink	
Initial	42.49/42.49	127.48/127.48	127.20/127.20	0.00/0.00	297.17/297.17
12+	0.00/42.49	0.00/127.48	0.00/127.20	0.00/0.00	0.00/297.17
13+	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
14+	0.00/42.49	0.00/148.73	0.00/148.40	0.00/0.00	0.00/339.62
15+	42.49/42.49	169.97/169.97	169.60/169.60	0.00/0.00	382.06/382.06
16+	42.49/42.49	169.97/169.97	169.60/169.60	0.00/0.00	382.06/382.06
3-	0.00/42.49	0.00/169.97	0.00/169.60	0.00/0.00	0.00/382.06
6-	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
7-	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
17+	0.00/42.49	0.00/148.73	0.00/148.40	0.00/0.00	0.00/339.62
18+	42.49/42.49	169.97/169.97	169.60/169.60	0.00/0.00	382.06/382.06
19+	0.00/42.49	0.00/169.97	0.00/169.60	0.00/0.00	0.00/382.06
1-	42.49/42.49	169.97/169.97	169.60/169.60	0.00/0.00	382.06/382.06
11-	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
16-	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
2-	42.49/42.49	127.48/127.48	127.20/127.20	0.00/0.00	297.17/297.17
20+	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
21+	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
22+	42.49/42.49	169.97/169.97	169.60/169.60	0.00/0.00	382.06/382.06
4-	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
8-	0.00/42.49	0.00/148.73	0.00/148.40	0.00/0.00	0.00/339.62
9-	42.49/42.49	127.48/127.48	127.20/127.20	0.00/0.00	297.17/297.17
23+	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
24+	0.00/42.49	0.00/148.73	0.00/148.40	0.00/0.00	0.00/339.62
25+	42.49/42.49	169.97/169.97	169.60/169.60	0.00/0.00	382.06/382.06
26+	42.49/42.49	169.97/169.97	169.60/169.60	0.00/0.00	382.06/382.06
27+	42.49/42.49	191.22/191.22	190.80/190.80	0.00/0.00	424.51/424.51
0-	42.49/42.49	169.97/169.97	169.60/169.60	0.00/0.00	382.06/382.06
10-	0.00/42.49	0.00/169.97	0.00/169.60	0.00/0.00	0.00/382.06
17-	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
18-	42.49/42.49	148.73/148.73	148.40/148.40	0.00/0.00	339.62/339.62
19-	42.49/42.49	127.48/127.48	127.20/127.20	0.00/0.00	297.17/297.17
Total	1019.76/1359.68	3696.95/4929.27	3688.71/4918.28	0.00/0.00	8405.42/11207.23