frontiers | Frontiers in Artificial Intelligence

TYPE Original Research
PUBLISHED 28 January 2026
DOI 10.3389/frai.2026.1699896

OPEN ACCESS

EDITED BY
Haifeng Chen,
NEC Laboratories America Inc, United States

REVIEWED BY
Mengmeng Ren,
Xidian University, China
Salil Bharany,
Chitkara University, India

*CORRESPONDENCE
Chundong Wang
✉ michael3769@163.com

RECEIVED 05 September 2025
REVISED 30 December 2025
ACCEPTED 05 January 2026
PUBLISHED 28 January 2026

CITATION
Xue J and Wang C (2026) EF-Feddr: communication-efficient federated learning with Douglas–Rachford splitting and error feedback. *Front. Artif. Intell.* 9:1699896.
doi: 10.3389/frai.2026.1699896

EF-Feddr: communication-efficient federated learning with Douglas–Rachford splitting and error feedback

Jiao Xue[1] and Chundong Wang[1,2]*

[1]School of Computer Science and Engineering, Tianjin University of Technology, Tianjin, China,
[2]Tianjin Police Institute, Tianjin, China

Introduction: Federated learning (FL) is a distributed machine learning paradigm that preserves data privacy and mitigates data silos. Nevertheless, frequent communication between clients and the server often becomes a major bottleneck, restricting training efficiency and scalability.
Methods: To address this challenge, we propose a novel communication-efficient algorithm, **EF-Feddr**, for federated composite optimization, where the objective function includes a potentially non-smooth regularization term and local datasets are non-IID. Our method is built upon the relaxed Douglas–Rachford splitting method and incorporates error feedback (EF)—a widely adopted compression framework—to ensure convergence when biased compression (e.g., top-k sparsification) is applied.
Results: Under the partial client participation setting, our theoretical analysis demonstrates that **EF-Feddr** achieves a fast convergence rate of $O(1/K)$ and a communication complexity of $O(1/\varepsilon^2)$. Comprehensive experiments conducted on the FEMNIST and Shakespeare benchmarks, as well as controlled synthetic data, consistently validate the efficacy of **EF-Feddr** across diverse scenarios.
Discussion: The results confirm that the integration of error feedback with the relaxed Douglas–Rachford splitting method in **EF-Feddr** effectively overcomes the convergence degradation typically caused by biased compression, thereby offering a practical and efficient solution for communication-constrained federated learning.

KEYWORDS

communication efficiency, composite optimization, data heterogeneity, error feedback, federated learning, operator splitting

1 Introduction

Federated learning (FL) (Konecný et al., 2016; McMahan et al., 2017) is a distributed framework designed to address large-scale learning problems across networks of edge clients. In this paradigm, clients update models locally on their private data, while the server aggregates these updates to refine a shared global model. This collaborative process enables the development of global or personalized models without compromising user privacy (Ezequiel et al., 2022; Saifullah et al., 2024). Despite these advantages, communication between clients and the server remains a critical bottleneck, particularly when the number of participating clients is large, bandwidth is constrained, and the models involve high-dimensional parameters (Bhardwaj et al., 2023; Talwar et al., 2021). Recent efforts to improve the communication efficiency of FL have primarily focused on two directions: (i) reducing the number of communication rounds through partial client participation or increased local computation, and (ii) lowering the number of transmitted bits per round via techniques such as quantization and residual gradient compression. While these strategies effectively cut communication costs, they also introduce additional variance, which may widen the neighborhood around the optimal solution and, in some cases, prevent convergence under biased compression. To mitigate these issues, variance-reduction techniques such as error feedback (EF) are commonly employed. In contrast to traditional distributed training, it is unrealistic to assume that data on each local device are always independent and identically distributed (IID). Prior studies have consistently shown that FL accuracy degrades significantly when faced with non-IID or heterogeneous data (Islam et al., 2024). In this study, we focus on the following federated composite optimization (FCO) problem:

$$\min_{x\in\mathbb{R}^d} F(x) = f(x) + g(x) = \frac{1}{n}\sum\nolimits_{i=1}^{n} f_i(x) + g(x), \qquad (1)$$

where n denotes the number of clients, f_i is the local loss function for the i-th client, which is L-smooth and non-convex, and g represents the regularization term, which is proper, closed, convex (possibly non-smooth). As a practical example, consider a collaborative environmental monitoring project in which multiple research institutions aim to analyze sensor data from diverse geographical locations to detect climate change patterns. Due to privacy concerns and proprietary restrictions, however, raw data cannot be shared directly. In this case, enforcing sparse regularization becomes particularly important: although the dataset may contain relatively few observations (e.g., readings from a sparse sensor network Bhardwaj et al., 2022), each observation typically involves a high-dimensional set of features such as temperature, humidity, wind speed, and pollution levels, a combination of factors that further justifies the use of sparse regularization to identify salient features and prevent overfitting.

Operator splitting constitutes a broad class of methods for solving optimization problems of the form (Equation 1). These methods decompose numerically intractable components into simpler subproblems, thereby reducing computational complexity, enhancing efficiency, and enabling modular algorithms that are naturally suited for parallelization. Operator splitting has been successfully applied to a wide range of challenging optimization problems. Among these, the Douglas–Rachford splitting method is particularly well-established due to its enhanced iterative stability and accelerated convergence rate. Furthermore, its update rule decomposes the global composite objective into local proximal steps that can be executed in a fully parallel manner. This structure inherently aligns with the distributed nature of federated learning, facilitating efficient client-side computation while also underpinning the method's enhanced iterative stability. From this perspective, many state-of-the-art FL algorithms can be interpreted within the operator splitting framework (Malekmohammadi et al., 2021). Examples include FedAvg (McMahan et al., 2017), FedProx (Li et al., 2020), FedSplit (Pathak and Wainwright, 2020), and FedDR (Tran-Dinh et al., 2021). However, for the FCO Equation 1, existing FL methods such as FedAvg and its communication-efficient variants are primarily designed for smooth, unconstrained settings $\min_{x\in\mathbb{R}^d} F(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$. In non-smooth FL settings, subgradient methods are widely used but suffer from slow convergence (Jhunjhunwala et al., 2022). Although proximal operators offer a more effective alternative with superior convergence properties (Liu et al., 2024), their seamless integration into communication-efficient FL frameworks remains limited. Moreover, while compression techniques effectively reduce communication overhead, they introduce additional variance that can enlarge the solution neighborhood and hinder convergence. Critically, existing communication-efficient methods have predominantly been designed for smooth FL problems, leaving a pronounced combined gap in addressing non-smooth federated composite optimization under compression-induced variance and communication constraints simultaneously. To bridge this multifaceted gap, this study presents EF-Feddr, a communication-efficient FL algorithm that employs the Top-k sparsification technique to compress transmitted parameters and reduce communication bits, incorporates an error feedback (Li and Li, 2023) mechanism to mitigate variance introduced by compression, and further integrates the relaxed Douglas–Rachford splitting method (He et al., 2021) along with a proximal operator to accelerate the iterative process while effectively handling the non-smoothness of the global regularization term. This integrated design enables EF-Feddr to be applicable to a wider range of scenarios and constrained settings. Leveraging the Douglas–Rachford envelope, we establish convergence guarantees for EF-Feddr in non-convex FL problems under mild assumptions.

Our contributions are summarized as follows:

- We propose EF-Feddr, an algorithm that combines the relaxed Douglas–Rachford splitting method with error feedback to reduce communication costs between clients and the server without sacrificing accuracy in non-IID settings. In addition, the error feedback mechanism enhances the stability of communication-compressed training in FL.
- We establish theoretical convergence guarantees for EF-Feddr based on the Douglas–Rachford envelope. Specifically, our method achieves a convergence rate of $O\left(\frac{1}{K}\right)$ and a communication complexity of $O\left(\frac{1}{\varepsilon^2}\right)$ for non-convex loss functions under partial client participation.
- Through experiments on synthetic datasets, the FEMNIST dataset, and the Shakespeare dataset, we show that EF-Feddr improves accuracy by 3.29%–12.97% over state-of-the-art FL variants, while significantly reducing communication costs compared to uncompressed FedDR.

2 Related work

2.1 Operator splitting methods

Classical operator splitting methods such as Douglas–Rachford (DR), Forward-Backward (FB), and the Alternating Direction Method of Multipliers (ADMM) have recently been adopted in FL (Godavarthi et al., 2025; Goel et al., 2025). FedAvg (McMahan et al., 2017) can be viewed as an instance of k-step FB splitting, while FedProx (Li et al., 2020) extends the backward-backward splitting method. It is another FB variant tailored for regularized FL problems. FedSplit (Pathak and Wainwright, 2020), based on Peaceman-Rachford splitting, aims to identify the correct fixed point for strictly convex FL problems. Its communication-efficient variant, Eco-FedSplit (Khirirat et al., 2022), incorporates error-compensated compression. For the FCO problem, FedDR (Tran-Dinh et al., 2021) integrates a randomized block-coordinate strategy with DR splitting to solve non-convex formulations. FedADMM (Wang et al., 2022) leverages ADMM by applying FedDR to the dual form of the FCO problem, while FedTOP-ADMM (Kant et al., 2022) generalizes FedADMM as the first three-operator method used in FL.

2.2 Communication-efficient FL

To address the communication bottleneck in FL (Sun et al., 2024), two categories of compression methods have been widely explored: unbiased compressors (e.g., stochastic quantization Alistarh et al., 2017) and biased compressors (e.g., top-k sparsification Khirirat et al., 2018). FedPAQ (Reisizadeh et al., 2020) reduces communication costs through periodic averaging, partial client participation, and quantization. However, this reduction comes at the expense of convergence accuracy, which requires additional training iterations. The authors also analyzed the trade-off between communication overhead and convergence in their experiments. The z-SignFedAvg algorithm (Tang et al., 2024), a variant of FedAvg, employs stochastic sign-based compression. It achieves accuracy comparable to uncompressed FedAvg while greatly reducing communication overhead. Building on the lazily aggregated gradient rule and error feedback, (Zhou et al., 2023) proposed two communication-efficient algorithms for non-convex FL: EF-LAG and BiEF-LAG, which adapt both uplink and downlink communications. Similarly, FedSQ (Long et al., 2024) introduces a hybrid approach combining sparsity and quantization to reduce communication costs while enhancing convergence.

2.3 Error feedback

In the realm of distributed optimization, it has been noted that employing biased compressors for direct updates may decelerate convergence, deteriorate generalization performance, or even induce divergence (Li and Li, 2023). To counteract these issues, error feedback techniques have been introduced, which can reduce the compression error compared to direct compression. The study (Seide et al., 2014) first proposed this method as a heuristic approach, which is inspired by the idea of Sigma-Delta modulation. EF21 (Richtárik et al., 2021) removes strict assumptions such as bounded gradients and bounded dissimilarity, and can handle arbitrary data heterogeneity among clients, but leads to worse computational complexity. EFSkip (Bao et al., 2025) allows arbitrary data heterogeneity and enjoys linear speedup for significantly improving upon previous results.

3 Compressed non-convex FL with error feedback

In this section, we present EF-Feddr, an algorithm that integrates error feedback into the relaxed Douglas–Rachford splitting framework to address the non-convex FCO problem. We begin with a brief introduction to the Douglas–Rachford splitting method, followed by an explanation of how error feedback is incorporated to improve communication efficiency. We then provide the detailed formulation of EF-Feddr and analyze its convergence properties. Main notations are listed in Table 1.

3.1 Problem formulation

The FCO Equation 1 is mathematically equivalent to the consensus optimization problem

$$\begin{aligned} \min_{x_1,\ldots,x_n} \; & F(x) = f(x) + g(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x_i) + g(x) \\ \text{subject to} \; & x_1 = x_2 = \cdots = x_n, \end{aligned} \tag{2}$$

where the consensus constraint set is $E = \{x = (x_1, \ldots, x_n)\,|x_1 = x_2 = \cdots = x_n\}$. Let l_E be the indicator function of E. With the indicator function, one can treat the constrained problem as unconstrained by moving the constraints

TABLE 1 Summary of main notations.

Notation	Description
N	Number of clients
n	Number of sampled clients per round
d	Dimension of model parameters
$F(\cdot)$	Global loss function
$f_i(\cdot)$	local loss function of i-th client
$g(\cdot)$	Regularizer
$C(\cdot)$	Absolute compressor
K	Total number of communication rounds between clients and server
k	Index of communication round
S_k	Set of sampled clients at k-th iteration
λ_k	Relaxation parameter
γ	Step size
y_i^k	Local auxiliary variable at the i-th client
z_i^k	Approximate proximal update for optimizing the local loss of i-th client
e_i^k	Compression-error accumulator at the i-th client
x_i^k	Local model parameters of i-th client at k-th iteration
x^k	Global model parameters at k-th iteration

into the objective function. Then Equation 1 is obviously equivalent to

$$\min \frac{1}{n}\sum_{i=1}^{n} f_i(x_i) + g(x) + l_E(x). \tag{3}$$

The first-order optimality condition is given by $0 \in \nabla f(x) + \partial g(x) + \partial l_E(x)$, where $\nabla f(x) = [\nabla f_1(x_1), ..., \nabla f_n(x_n)]$. A point x^* is a stationary point to Equation 1, if $0 \in \nabla f(x^*) + \partial g(x^*) + \partial l_E(x^*)$. Additionally, the operator splitting method encompasses a broad range of techniques to effectively address this Equation 3. A key advantage of operator splitting methods is their efficient per-iteration operations, which makes them particularly suitable for large-scale applications due to their lower computational costs (He et al., 2021), among which the DR splitting method is particularly well-known. The iteration equations for the DR splitting method are given by

$$\begin{cases} y^{k+1} = y^k + x^k - z^{k+1} \\ z^{k+1} = \mathrm{prox}_{\gamma f}(y^k) \\ x^{k+1} = \mathrm{prox}_{\gamma(g+l_E)}(2z^{k+1} - y^{k+1}). \end{cases} \tag{4}$$

Given that the DR splitting method often demonstrates favorable and stable convergence behavior in practice, we base our approach on its relaxed variant to solve Equation 1. The detailed application is presented in Section 3.3.

For convenience, we introduce the definitions of the key concepts that will be utilized. For a function f, the proximal operator at point x with a step size $\gamma > 0$ is

$$\mathrm{prox}_{\gamma f}(x) = \arg\min_y \left\{ f(y) + \frac{1}{2\gamma}\|y-x\|^2 \right\},$$

the Moreau envelope of f with a step size $\gamma > 0$ is

$$\mathrm{M}_{\gamma f}(x) = \min_y \left\{ f(y) + \frac{1}{2\gamma}\|y-x\|^2 \right\},$$

the gradient mapping of f at point x with a step size $\gamma > 0$ is

$$G_{\gamma f}(x) = \frac{1}{\gamma}(x - \mathrm{prox}_{\gamma f}(x)).$$

We observe that $\nabla \mathrm{M}_{\gamma f}(x) = G_{\gamma f}(x)$ (Liu et al., 2019). Moreover, the proximal operator update $z^k = \mathrm{prox}_{\gamma f}\left(y^k\right)$ can be written as

$$z^k = y^k - \gamma G_{\gamma f}(y^k).$$

This representation reveals that the proximal operator update is analogous to taking a gradient step applied to the gradient mapping $G_{\gamma f}(y^k)$ of f. For the composite function $F(x) = f(x) + g(x)$, the corresponding gradient mapping is given by

$$\mathcal{G}_\gamma(x) = \frac{1}{\gamma}\left(x - \mathrm{prox}_{\gamma g}(x - \gamma\nabla f(x))\right). \tag{5}$$

In the context of general non-convex non-smooth problems, the gradient mapping $\mathcal{G}_\gamma(x)$ is commonly used to assess convergence (Liu et al., 2024). Specifically, $0 \in \nabla f(x^*) + \partial g(x^*) + \partial l_E(x^*)$ of Equation 1 is equivalent to $\mathcal{G}_\gamma(x^*) = 0$.

3.2 Error feedback

We now define a general class of compressors that will be used throughout this study

Definition 1. (Absolute compressor). A map $C: \mathbb{R}^d \to \mathbb{R}^d$ is an absolute compressor operator if there exists $\nu > 0$ such that, $\forall x \in \mathbb{R}^d, \mathbb{E}\left\|x - C(x)\right\|^2 \le \nu^2$.

Most popular compressors such as the sign compression (Bernstein et al., 2018), the Top-k sparsification (Khirirat et al., 2018) and the sparsification together with quantization (Alistarh et al., 2017) are in fact absolute compressors if the full-precision vector has a bounded norm (Khirirat et al., 2022; Sahu et al., 2021).

Error feedback (also known as error compensation) is a popular tool in FL to reduce compression error and improve convergence speed compared to direct compression (Valdeira et al., 2025). Its mechanism shares a fundamental principle with Sigma-Delta modulation in signal processing (Seide et al., 2014). Technically, when transmitting a sequence of vectors, the method incorporates an auxiliary vector that accumulates the compression error at each step. This accumulated error is then added to the current vector before it undergoes compression and transmission (Karimireddy et al., 2019). More specifically, based on the DR splitting method (Equation 4), the update steps of the direct compression scheme are as follows:

$$\begin{aligned} c^{k+1} &= C(2z^{k+1} - y^{k+1}), \quad \text{(direct compression)} \\ x^{k+1} &= \mathrm{prox}_{\gamma(g+l_E)}(c^{k+1}), \quad \text{(model update)} \end{aligned} \tag{6}$$

the update steps with error feedback compression are as follows:

$$\begin{aligned} c^{k+1} &= C(2z^{k+1} - y^{k+1} + e^k), \quad \text{(error compensation)} \\ e^{k+1} &= 2z^{k+1} - y^{k+1} + e^k - c^{k+1}, \quad \text{(compute the error)} \\ x^{k+1} &= \mathrm{prox}_{\gamma(g+l_E)}(c^{k+1}). \quad \text{(model update)} \end{aligned} \tag{7}$$

In direct compression, each vector $2z^{k+1} - y^{k+1}$ is individually compressed, and the receiver directly uses its compressed version $C(2z^{k+1} - y^{k+1})$ in place of the original. Conversely, error feedback compression employs a proxy vector c^{k+1} for $2z^{k+1} - y^{k+1}$ that integrates information from prior steps $0, 1, \ldots, k$. This proxy is refined via an auxiliary vector e^{k+1}, which is iteratively updated and stored to accumulate the compression error at each step.

3.3 EF-Feddr algorithm

In this section, we propose the following EF-Feddr algorithm. The details of EF-Feddr are presented in Algorithm 1. Specifically, applying the relaxed DR splitting method (He et al., 2021) to the Equation 3 of Equation 1 in a distributed setting yields the following iterative steps:

$$\begin{cases} y_i^{k+1} = y_i^k + \lambda\left(x^k - z_i^k\right) \\ z_i^{k+1} = \mathrm{prox}_{\gamma f_i}\left(y_i^{k+1}\right) \\ x_i^{k+1} = 2z_i^{k+1} - y_i^{k+1} \\ x^{k+1} = \mathrm{prox}_{n\gamma(g+l_E)}(x_i^{k+1}). \end{cases}$$

By integrating the error feedback mechanism detailed in Section 3.2, we obtain the EF-Feddr iterative scheme:

$$\begin{cases} y_i^{k+1} = y_i^k + \lambda\left(x^k - z_i^k\right) \\ z_i^{k+1} \approx \mathrm{prox}_{\gamma f_i}\left(y_i^{k+1}\right) \\ x_i^{k+1} = C\left(2z_i^{k+1} - y_i^{k+1} + e_i^k\right) \\ e_i^{k+1} = 2z_i^{k+1} - y_i^{k+1} + e_i^k - x_i^{k+1} \\ x^{k+1} = \mathrm{prox}_{n\gamma(g+l_E)}(x_i^{k+1}), \end{cases} \tag{8}$$

where $\lambda \in (0,2)$ (He et al., 2021) is the relaxation parameter. The variables y_i^{k+1}, z_i^{k+1}, x_i^{k+1} and e_i^{k+1} are updated locally on each client i. The key step involves compression and communication: instead of compressing $2z_i^{k+1} - y_i^{k+1}$ directly, each client compresses the error-compensated vector $2z_i^{k+1} - y_i^{k+1} + e_i^k$. The resulting value x_i^{k+1} is then sent to the server. Furthermore, to compute the server aggregation x^{k+1}, we have the following conclusion.

Proposition 1. For every $k \geq 0$, $x^{k+1} = \mathrm{prox}_{n\gamma(g+l_E)}(x_i^{k+1})$ in Equation 8 is equal to $\mathrm{prox}_{\gamma g}\left(\frac{1}{n}\sum_{i\in S_k} x_i^{k+1}\right)$.

Proof. Let $\bar{x} = \frac{1}{n}\sum_{i\in S_k} x_i^{k+1}$. Actually, the result of $\mathrm{prox}_{n\gamma(g+l_E)}(x_i^{k+1})$ must have blocks equal to some vector z (Mishchenko et al., 2022) such as

$$\begin{aligned} z &= \arg\min_y \left\{ g(y) + \frac{1}{2n\gamma}\sum_{i=1}^n \|y - x_i^{k+1}\|^2 \right\} \\ &= \arg\min_y \left\{ g(y) + \frac{1}{2n\gamma}\sum_{i=1}^n \left(\|y-\bar{x}\|^2 + 2\langle y-\bar{x}, \bar{x} - x_i^{k+1}\rangle \right.\right. \\ &\quad \left.\left. + \|\bar{x} - x_i^{k+1}\|^2 \right) \right\} \\ &= \arg\min_y \left\{ g(y) + \frac{1}{2n\gamma}\left[\sum_{i=1}^n \|y-\bar{x}\|^2 + 2\langle y-\bar{x}, n\bar{x}\rangle \right.\right. \\ &\quad \left.\left. - 2\langle y - \bar{x}, n\bar{x}\rangle \right] \right\} \\ &= \arg\min_y \left\{ g(y) + \frac{1}{2\gamma}\|y-\bar{x}\|^2 \right\} \\ &= \mathrm{prox}_{\gamma g}(\bar{x}) = \mathrm{prox}_{\gamma g}\left(\frac{1}{n}\sum\nolimits_{i\in S_k} x_i^{k+1} \right). \end{aligned}$$

Thus, we have the server aggregation

$$x^{k+1} = \mathrm{prox}_{n\gamma(g+l_E)}(x_i^{k+1}) = \mathrm{prox}_{\gamma g}\left(\frac{1}{n}\sum\nolimits_{i\in S_k} x_i^{k+1}\right).$$

In Algorithm 1, during round k: (1) The clients receive the global model x^k from the server (line 5); (2) A subset of clients S_k is sampled following the sampling scheme described in Section 4. The i-th client performs a relaxation step, where λ is the relaxation parameter, computes the proximal local update to obtain the local model z_i^{k+1}, calculates the compressed local model update x_i^{k+1}, and updates the local compression error accumulator e_i^{k+1} and sends the compressed x_i^{k+1} back to the server (line 6–10); (3) The server receives the compressed x_i^{k+1} from clients $i \in S_k$ and

Initialization Given an initial point $x^0 \in \mathbb{R}^d$, set $z_i^0 = x^0$, $y_i^0 = x^0$, for all $i \in [n]$, the step size $\gamma > 0$, relaxation parameter $\lambda > 0$.

1: **for** $k = 0, 1 \ldots, K-1$ **do**
2: Sample $S_k \subseteq [N]$ with size n uniformly without replacement
3: // Client side:
4: **for** each $i \in S_k$ in parallel **do**
5: receive x^k from the server
6: $y_i^{k+1} = y_i^k + \lambda\left(x^k - z_i^k\right)$
7: $z_i^{k+1} \approx \mathrm{prox}_{\gamma f_i}\left(y_i^{k+1}\right)$
8: $x_i^{k+1} = C\left(2z_i^{k+1} - y_i^{k+1} + e_i^k\right)$
9: $e_i^{k+1} = 2z_i^{k+1} - y_i^{k+1} + e_i^k - x_i^{k+1}$
10: send x_i^{k+1} back to the server
11: **end for**
12: // Server side:
13: server update: $x^{k+1} = \mathrm{prox}_{\gamma g}\left(\frac{1}{n}\sum_{i\in S_k} x_i^{k+1}\right)$
14: broadcast x^{k+1} to each client
15: **end for**

Algorithm 1. EF-Feddr.

performs a global model update using the averaged compressed local model updates (line 13). Particularly, the relaxation strategy, akin to the inertial extrapolation technique (e.g., the heavy ball method), has broadly accelerated iterative algorithms in convex and non-convex optimization, as the cost per iteration stays basically unchanged (He et al., 2021). For any $\gamma > 0$, z_i^{k+1} serves as an approximation of $\mathrm{prox}_{\gamma f_i}\left(y_i^{k+1}\right)$. The evaluation of $\mathrm{prox}_{\gamma f_i}$ can be carried out using several established techniques, such as accelerated GD-type algorithms and local SGD (Parikh et al., 2014; Tran-Dinh et al., 2021). It is worth noting that this algorithm requires $O(d)$ memory and incurs $O(d)$ computational overhead per client per round.

4 Theoretical results

For analyzing the convergence of Algorithm 1, we consider several basic assumptions and auxiliary results. Our analysis is based on the analytical framework outlined in Tran-Dinh et al. (2021). First, we introduce a proper sampling scheme following Tran-Dinh et al. (2021). Let $\mathbf{p}_1, \ldots, \mathbf{p}_n > 0$ such that for all $i \in [N]$, $\mathbb{P}\left(i \in \bar{S}\right) = \mathbf{p}_i \leq 1$. Here, $\bar{S}$ is a proper samping scheme of $[N]$, and each S_k is an i.i.d. realization of $\bar{S}$. Note that $\mathbf{p}_i = \sum_{S\subseteq[N], i\in S}\mathbb{P}\left(\bar{S} = S\right)$. Define $\mathcal{A}_k = \sigma\left(S_0, \ldots, S_k\right)$ as the σ-algebra generated by the sequence $S_0, \ldots, S_k$. This sampling scheme ensures that each client has a significant probability of being updated.

Assumption 1. (L-Smoothness). All local functions $f_i(\cdot)$ are L-smooth, if

$$\forall x, y, \quad \left\|\nabla f_i(x) - \nabla f_i(y)\right\| \leq L\left\|x - y\right\|.$$

Assumption 2. (Boundedness from below). $F(\cdot)$ given in (1) is bounded below, that is, $F^* = \inf_{x\in\mathbb{R}^d} F(x) > -\infty$.

In non-convex FL optimization, Assumptions 1 and 2 are standard. Assumption 2 guarantees that Equation 1 is well-defined and is independent of the choice of algorithms. We first present three useful lemmas that will be instrumental in proving our main theorem.

Lemma 1. Let $\left\{\left(y_i^k, z_i^k, x_i^k, e_i^k, x^k\right)\right\}$ be generated by Algorithm 1, for all $i \in S_k$, $\lambda > 0$, $\beta_1 > 0$ and $\gamma > 0$, we have

$$\left\|x^k - z_i^k\right\|^2 \le \frac{2(\gamma^2L^2+1)}{\lambda^2}\Big[(1+\beta_1)\|z_i^{k+1} - z_i^k\|^2 + 2(1+\frac{1}{\beta_1})\left(\|m_i^{k+1}\|^2 + \|m_i^k\|^2\right)\Big]. \tag{9}$$

Proof. For the relation $z_i^{k+1} \approx \text{prox}_{\gamma f_i}\left(y_i^{k+1}\right)$, where the approximation error satisfies $\|z_i^{k+1} - \text{prox}_{\gamma f_i}\left(y_i^{k+1}\right)\| \le \varepsilon_i^k$ with a given accuracy $\varepsilon_i^k \ge 0$, we introduce auxiliary variables w_i^0 and w_i^{k+1} for $i \in [n]$ to analyze the convergence of Algorithm 1,

$$\begin{aligned} w_i^0 &= \text{prox}_{\gamma f_i}(y_i^0), \\ w_i^{k+1} &= \begin{cases} \text{prox}_{\gamma f_i}(y_i^{k+1}) & \text{if } i \in \mathcal{S}_k \\ w_i^k & \text{if } i \notin \mathcal{S}_k \end{cases}, \\ z_i^k &= w_i^k + m_i^k, \quad \text{where } \|m_i^k\| \le \varepsilon_i^k. \end{aligned} \tag{10}$$

Here, m_i^k denotes the vector of errors associated with the approximations of the proximal operator, and w_i^{k+1} serves as an accurate computation to $\text{prox}_{\gamma f_i}(y_i^{k+1})$. Note that when $i \notin \mathcal{S}_k$, we have $z_i^{k+1} = z_i^k$ and $w_i^{k+1} = w_i^k$, which implies $\|m_i^{k+1}\| = \|z_i^{k+1} - w_i^{k+1}\| = \|m_i^k\| = \|z_i^k - w_i^k\|$. From Equation 10 (Atenas, 2025), we have

$$y_i^k = w_i^k + \gamma\nabla f_i(w_i^k). \tag{11}$$

Then, using the update rule for y_i^{k+1} in Algorithm 1, we get $x^k - z_i^k = \frac{1}{\lambda}\left(y_i^{k+1} - y_i^k\right) = \frac{1}{\lambda}(w_i^{k+1} - w_i^k) + \frac{\gamma}{\lambda}(\nabla f_i(w_i^{k+1}) - \nabla f_i(w_i^k))$. Using Young's inequality $\|a_1 + a_2\|^2 \le (1+\beta)\|a_1\|^2 + (1+\frac{1}{\beta})\|a_2\|^2$, and the L-smoothness of f_i, we bound $\left\|x^k - z_i^k\right\|^2$ for any $\beta_1 > 0$ and $i \in \mathcal{S}_k$ as follows

$$\begin{aligned} \left\|x^k - z_i^k\right\|^2 &= \left\|\frac{1}{\lambda}(w_i^{k+1} - w_i^k) + \frac{\gamma}{\lambda}(\nabla f_i(w_i^{k+1}) - \nabla f_i(w_i^k))\right\|^2 \\ &\le \frac{2}{\lambda^2}\left\|w_i^{k+1} - w_i^k\right\|^2 + \frac{2\gamma^2}{\lambda^2}\left\|(\nabla f_i(w_i^{k+1}) - \nabla f_i(w_i^k))\right\|^2 \\ &\le \frac{2}{\lambda^2}\left\|w_i^{k+1} - w_i^k\right\|^2 + \frac{2\gamma^2L^2}{\lambda^2}\left\|(w_i^{k+1} - w_i^k)\right\|^2 \\ &= \frac{2(\gamma^2L^2+1)}{\lambda^2}\left\|z_i^{k+1} - m_i^{k+1} - z_i^k + m_i^k\right\|^2 \\ &\le \frac{2(\gamma^2L^2+1)}{\lambda^2}\left[(1+\beta_1)\|z_i^{k+1} - z_i^k\|^2 + 2(1 + \frac{1}{\beta_1})\left(\|m_i^{k+1}\|^2 + \|m_i^k\|^2\right)\right], \end{aligned}$$

which proves (9).

We then establish the relationship between $\sum_{i=1}^n \|x^k - z_i^k\|^2$ and the squared norm of the gradient mapping $\|\mathcal{G}_\gamma(x^k)\|^2$.

Lemma 2. Let $\left\{\left(y_i^k, z_i^k, x_i^k, e_i^k, x^k, w_i^k\right)\right\}$ be generated by Algorithm 1 and Equation 10, and the gradient mapping $\mathcal{G}_\gamma$ be defined by (5). Then, for any $\lambda > 0$, $\beta_2 > 0$, and $\gamma > 0$, we have

$$\|\mathcal{G}_\gamma(x^k)\|^2 \le \frac{2(1+\gamma L)^2}{n\gamma^2}\sum_{i=1}^n\Big[(1+\beta_2)\|z_i^k - x^k\|^2 + (1+\frac{1}{\beta_2})\|m_i^k\|^2\Big] + \frac{2}{n\gamma^2}\sum_{i=1}^n\|e_i^{k-1} - e_i^k\|^2. \tag{12}$$

Proof. From the update of x_i^{k+1}, e_i^{k+1} in Algorithm 1 and (11), we have

$$\begin{aligned} \frac{1}{n}\sum_{i=1}^n x_i^k &= \frac{1}{n}\sum_{i=1}^n(2z_i^k - y_i^k + e_i^{k-1} - e_i^k) \\ &= \frac{1}{n}\sum_{i=1}^n(2z_i^k - w_i^k - \gamma\nabla f_i(w_i^k) + e_i^{k-1} - e_i^k). \end{aligned} \tag{13}$$

From the update rule of x^k in Algorithm 1, the definition of $\mathcal{G}_\gamma(x)$, the non-expansive property of $\text{prox}_{\gamma g}$, and the fact $\nabla f(x^k) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(x^k)$, we obtain that

$$\begin{aligned} \|\mathcal{G}_\gamma(x^k)\| &= \frac{1}{\gamma}\|x^k - \text{prox}_{\gamma g}(x^k - \gamma\nabla f(x^k))\| \\ &= \frac{1}{\gamma}\|\,\text{prox}_{\gamma g}(\frac{1}{n}\sum_{i=1}^n x_i^k) - \text{prox}_{\gamma g}(x^k - \gamma\nabla f(x^k))\| \\ &\le \frac{1}{\gamma}\|\frac{1}{n}\sum_{i=1}^n x_i^k - x^k + \gamma\nabla f(x^k)\| \\ &= \frac{1}{n\gamma}\|\sum_{i=1}^n[(2z_i^k - w_i^k - x^k) + \gamma(\nabla f_i(x^k) - \nabla f_i(w_i^k)) \\ &\quad + e_i^{k-1} - e_i^k]\|. \end{aligned}$$

By applying the L-smoothness of f_i and the Young's inequality stated in Lemma 1, for any $\beta_2 > 0$ we deduce that

$$\begin{aligned} &\|\mathcal{G}_\gamma(x^k)\|^2 \\ &\le \frac{1}{n^2\gamma^2}\left[\sum_{i=1}^n\left(\|2z_i^k - w_i^k - x^k\| + \gamma L\|z_i^k - \bar{x}^k\| + \|e_i^{k-1} - e_i^k\|\right)\right]^2 \\ &\le \frac{1}{n\gamma^2}\sum_{i=1}^n\left(\|2z_i^k - w_i^k - x^k\| + \gamma L\|x^k - w_i^k\| + \|e_i^{k-1} - e_i^k\|\right)^2 \\ &\le \frac{1}{n\gamma^2}\sum_{i=1}^n\left[(1+\gamma L)\|z_i^k - x^k\| + (1+\gamma L)\|m_i^k\| + \|e_i^{k-1} - e_i^k\|\right]^2 \\ &\le \frac{1}{n\gamma^2}(1+\gamma L)^2\sum_{i=1}^n\Big[2(1+\beta_2)\|z_i^k - x^k\|^2 + 2(1+\frac{1}{\beta_2})\|m_i^k\|^2 \\ &\quad + \frac{2}{(1+\gamma L)^2}\|e_i^{k-1} - e_i^k\|^2\Big] \\ &\le \frac{2(1+\gamma L)^2}{n\gamma^2}\sum_{i=1}^n\Big[(1+\beta_2)\|z_i^k - x^k\|^2 + (1+\frac{1}{\beta_2})\|m_i^k\|^2 \\ &\quad + \frac{1}{(1+\gamma L)^2}\|e_i^{k-1} - e_i^k\|^2\Big], \end{aligned}$$

which proves (12).

Lemma 3. Let $\left\{\left(y_i^k, z_i^k, x_i^k, e_i^k, x^k\right)\right\}$ be generated by Algorithm 1. Suppose that Assumptions 1 and 2 hold, and we define the Lyapunov function

$$V^k(x^k) = g(x^k) + \frac{1}{n}\sum_{i=1}^{n}\left[f_i(z_i^k) + \left\langle \nabla f_i(z_i^k), x^k - z_i^k\right\rangle + \frac{1}{2\gamma}\left\|x^k - z_i^k\right\|^2\right],$$

then by choosing

$$0 < \gamma < \frac{\sqrt{(1-\frac{\lambda}{4})^2 - \lambda^2\beta_4(4\beta_4+1)} - \frac{\lambda}{4}}{L(2\lambda\beta_4+1)} \quad \text{and}$$

$$0 < \lambda < \frac{\min\{\sqrt{4\beta_4 + \frac{17}{16}} - \frac{1}{4}, 2\}}{4\beta_4+1},$$

and for any $\varepsilon_1, \beta_1, \beta_4 > 0$, we have

$$\mathbb{E}\left[V^{k+1}(x^{k+1})|\mathcal{A}_{k-1}\right] \le V^k(x^k) - \frac{\pi}{2n}\sum_{i=1}^{n}\left\|x^k - z_i^k\right\|^2 + \frac{4\varepsilon_1}{\gamma}\nu^2 + \frac{1}{n}\sum_{i=1}^{n}(\delta_1(\varepsilon_i^k)^2 + \delta_2(\varepsilon_i^{k+1})^2),$$

where

$$\pi = \frac{\mathbf{p}\lambda[2-\lambda(1+L\gamma)-2L^2\gamma^2-4\lambda\beta_4(1+L^2\gamma^2)]}{2\gamma(1+\beta_1)(\gamma^2L^2+1)},$$

$$\delta_1 = \frac{2(1+\gamma L)^2}{\gamma\beta_4\lambda^2} + \frac{[2-\lambda(1+L\gamma)-2L^2\gamma^2-4\lambda\beta_4(1+L^2\gamma^2)]}{\lambda\gamma\beta_1},$$

$$\delta_2 = \delta_1 + \frac{(1+\gamma^2L^2)}{\gamma}.$$

Proof. Given the definition $\bar{x}^k = \frac{1}{n}\sum_{i=1}^{n} x_i^k$, the update rule $x^{k+1} = \mathrm{prox}_{\gamma g}(\bar{x}^{k+1})$ in Algorithm 1 (hence $\frac{\bar{x}^{k+1}-x^{k+1}}{\gamma} \in \partial g(x^{k+1})$), and the convexity of g, we obtain the following inequality

$$g(x^{k+1}) \le g(x^k) - \frac{1}{\gamma}\|x^{k+1}-x^k\|^2 + \frac{1}{\gamma}\langle \bar{x}^{k+1}-x^k, x^{k+1}-x^k\rangle. \quad (14)$$

Combining Equations 10 and 11, we obtain

$$\begin{aligned} z_i^{k+1} + \gamma\nabla f_i(z_i^{k+1}) &= w_i^{k+1} + \gamma\nabla f_i(w_i^{k+1}) + m_i^{k+1} + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1})) \\ &= y_i^{k+1} + m_i^{k+1} + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1})). \end{aligned} \quad (15)$$

Next, using the update rules for x_i^{k+1} and e_i^{k+1} in Algorithm 1, we have

$$\begin{aligned} \bar{x}^{k+1} &= \frac{1}{n}\sum_{i=1}^{n} x_i^{k+1} = \frac{1}{n}\sum_{i=1}^{n}\left(C\left(2z_i^{k+1} - y_i^{k+1} + e_i^k\right)\right) \\ &= \frac{1}{n}\sum_{i=1}^{n}\left(2z_i^{k+1} - y_i^{k+1} + e_i^k - e_i^{k+1}\right). \end{aligned} \quad (16)$$

In order to establish the descent property of the Lyapunov function $V^{k+1}(x^{k+1})$, its second term is expanded and rearranged as follows

$$\begin{aligned}
&\frac{1}{n}\sum_{i=1}^{n}\left[f_i(z_i^{k+1}) + \left\langle\nabla f_i(z_i^{k+1}), x^{k+1} - z_i^{k+1}\right\rangle + \frac{1}{2\gamma}\left\|x^{k+1} - z_i^{k+1}\right\|^2\right] \\
&= \frac{1}{n}\sum_{i=1}^{n}\left[f_i(z_i^{k+1}) + \left\langle\nabla f_i(z_i^{k+1}), x^k - z_i^{k+1} + x^{k+1} - x^k\right\rangle\right] + \frac{1}{2\gamma n}\sum_{i=1}^{n}\left\|x^k - z_i^{k+1} + x^{k+1} - x^k\right\|^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left[f_i(z_i^{k+1}) + \langle\nabla f_i(z_i^{k+1}), x^k - z_i^{k+1}\rangle + \frac{1}{2\gamma}\|x^k - z_i^{k+1}\|^2\right] \\
&\quad + \frac{1}{n\gamma}\sum_{i=1}^{n}\langle x^k - 2z_i^{k+1} + (z_i^{k+1} + \gamma\nabla f_i(z_i^{k+1})), x^{k+1} - x^k\rangle + \frac{1}{2\gamma}\|x^{k+1} - x^k\|^2 \\
&\overset{(15)}{=} \frac{1}{n}\sum_{i=1}^{n}\left[f_i(z_i^{k+1}) + \langle\nabla f_i(z_i^{k+1}), x^k - z_i^{k+1}\rangle + \frac{1}{2\gamma}\|x^k - z_i^{k+1}\|^2\right] \\
&\quad + \frac{1}{n\gamma}\sum_{i=1}^{n}\langle x^k - 2z_i^{k+1} + y_i^{k+1}, x^{k+1} - x^k\rangle + \frac{1}{2\gamma}\|x^{k+1} - x^k\|^2 \\
&\quad + \frac{1}{n\gamma}\sum_{i=1}^{n}\langle m_i^{k+1} + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1})), x^{k+1} - x^k\rangle
\end{aligned} \quad (17)$$

$$\begin{aligned}
&\overset{(16)}{=} \frac{1}{n}\sum_{i=1}^{n}\left[f_i(z_i^{k+1}) + \langle\nabla f_i(z_i^{k+1}), x^k - z_i^{k+1}\rangle + \frac{1}{2\gamma}\|x^k - z_i^{k+1}\|^2\right] \\
&\quad + \frac{1}{\gamma}\langle x^k - \bar{x}^{k+1} + \frac{1}{n}\sum_{i=1}^{n}(e_i^k - e_i^{k+1}), x^{k+1} - x^k\rangle + \frac{1}{2\gamma}\|x^{k+1} - x^k\|^2 + \frac{1}{n\gamma}\sum_{i=1}^{n}\langle m_i^{k+1} + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1})), x^{k+1} - x^k\rangle.
\end{aligned}$$

Here, Equation 15 is used to separate the term y_i^{k+1} from the approximation error m_i^{k+1}, while Equation 16 expresses $2z_i^{k+1} - y_i^{k+1}$ in terms of the average vector $\bar{x}^{k+1}$ and the accumulated compression errors e_i^{k+1} and e_i^k. Then, by combining Equations 14, 17 and using the definition of $V^{k+1}(x^{k+1})$, we obtain that

$$\begin{aligned}
V^{k+1}(x^{k+1}) &\le g(x^k) + \frac{1}{n}\sum_{i=1}^{n}\left[f_i(z_i^{k+1}) + \langle\nabla f_i(z_i^{k+1}), x^k - z_i^{k+1}\rangle + \frac{1}{2\gamma}\|x^k - z_i^{k+1}\|^2\right] \\
&\quad + \frac{1}{n\gamma}\sum_{i=1}^{n}\langle e_i^k - e_i^{k+1}, x^{k+1} - x^k\rangle - \frac{1}{2\gamma}\|x^{k+1} - x^k\|^2 \\
&\quad + \frac{1}{n\gamma}\sum_{i=1}^{n}\langle m_i^{k+1} + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1})), x^{k+1} - x^k\rangle.
\end{aligned} \quad (18)$$

To bound the third term on the right-hand side of Equation 18, we employ the inequality $2\langle a_1, a_2\rangle \le \varepsilon_1\|a_1\|^2 + \frac{1}{\varepsilon_1}\|a_2\|^2$ (for any

$\varepsilon_1 > 0$) as follows

$$\begin{aligned}
&\frac{1}{n\gamma}\sum_{i=1}^{n}\langle e_i^k - e_i^{k+1}, x^{k+1}-x^k\rangle \\
&\le \frac{1}{n\gamma}\sum_{i=1}^{n}\left[\varepsilon_1\left\|e_i^k - e_i^{k+1}\right\|^2 + \frac{1}{\varepsilon_1}\left\|x^{k+1}-x^k\right\|^2\right] \\
&\le \frac{1}{n\gamma}\sum_{i=1}^{n}\left[2\varepsilon_1\left\|e_i^k\right\|^2 + 2\varepsilon_1\left\|e_i^{k+1}\right\|^2\right] \\
&\quad + \frac{1}{\gamma\varepsilon_1}\left\|x^{k+1}-x^k\right\|^2 \\
&\le \frac{2\varepsilon_1}{n\gamma}\sum_{i=1}^{n}\left[\left\|e_i^k\right\|^2 + \left\|e_i^{k+1}\right\|^2\right] \\
&\quad + \frac{1}{\gamma\varepsilon_1}\left\|x^{k+1}-x^k\right\|^2.
\end{aligned} \tag{19}$$

For $i \notin \mathcal{S}_k$, we have $w_i^{k+1} = w_i^k$. Applying Young's inequality stated in Lemma 1 with any $\beta_3 > 0$, we can evaluate the five term on the right-hand side of Equation 18 as follows

$$\begin{aligned}
&\frac{1}{n\gamma}\sum_{i=1}^{n}\langle m_i^{k+1} + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1})), x^{k+1}-x^k\rangle \\
&\le \frac{1}{2n\gamma}\sum_{i=1}^{n}\left[\frac{1}{\beta_3}\|m_i^{k+1} + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1}))\|^2 + \beta_3\|x^{k+1}\right. \\
&\left. - x^k\|^2\right] \\
&\le \frac{1}{n\gamma\beta_3}\sum_{i=1}^{n}\left[\|m_i^{k+1}\|^2 + \gamma^2\sum_{i=1}^{n}\|\nabla f_i(x_i^{k+1}) - \nabla f_i(z_i^{k+1})\|^2\right] \\
&+ \frac{\beta_3}{2\gamma}\|x^{k+1}-x^k\|^2 \\
&\le \frac{(1+\gamma^2L^2)}{n\gamma\beta_3}\left[\sum_{i\notin\mathcal{S}_k}\|m_i^k\|^2 + \sum_{i\in\mathcal{S}_k}\|m_i^{k+1}\|^2\right] + \frac{\beta_3}{2\gamma}\|x^{k+1}-x^k\|^2.
\end{aligned} \tag{20}$$

To streamline the notation, denote

$$\begin{aligned}
\Psi_{k+1} = &-\frac{1}{\gamma}\left(\frac{1}{2} - \frac{1}{\varepsilon_1} - \frac{\beta_3}{2}\right)\left\|x^{k+1}-x^k\right\|^2 \\
&+ \frac{2\varepsilon_1}{n\gamma}\sum_{i=1}^{n}\left[\left\|e_i^k\right\|^2 + \left\|e_i^{k+1}\right\|^2\right] \\
&+ \frac{(1+\gamma^2L^2)}{n\gamma\beta_3}\left[\sum_{i\notin\mathcal{S}_k}\|m_i^k\|^2 + \sum_{i\in\mathcal{S}_k}\|m_i^{k+1}\|^2\right],
\end{aligned} \tag{21}$$

and substituting Equations 19 and 20 into Equations 18, we obtain an expanded expression for V^{k+1}. Differentiating between the active client set $\mathcal{S}_k$ and the inactive set, and employing the L-smoothness of f_i (i.e., $f_i(z_i^{k+1}) \le f_i(z_i^k) + \langle\nabla f_i(z_i^k), z_i^{k+1} - z_i^k\rangle + \frac{L}{2}\|z_i^{k+1} - z_i^k\|^2$), we have

$$\begin{aligned}
V^{k+1}(x^{k+1}) &\le g(x^k) + \frac{1}{n}\sum_{i=1}^{n}\left[f_i(z_i^{k+1}) + \langle\nabla f_i(z_i^{k+1}), x^k - z_i^{k+1}\rangle\right. \\
&\left. + \frac{1}{2\gamma}\|x^k - z_i^{k+1}\|^2\right] + \Psi_{k+1}
\end{aligned}$$

(by the fact that only $i \in S_k$ perform update)

$$\begin{aligned}
&= g(x^k) + \frac{1}{n}\sum_{i\in\mathcal{S}_k} f_i(z_i^{k+1}) + \frac{1}{n}\sum_{i\in\mathcal{S}_k}\langle\nabla f_i(z_i^{k+1}), z_i^k - z_i^{k+1}\rangle \\
&+ \frac{1}{n}\sum_{i\in\mathcal{S}_k}\langle\nabla f_i(z_i^{k+1}), x^k - z_i^k\rangle + \frac{1}{2n\gamma}\sum_{i\in\mathcal{S}_k}\|x^k - z_i^{k+1}\|^2 \\
&+ \frac{1}{n}\sum_{i\notin\mathcal{S}_k} f_i(z_i^k) + \frac{1}{n}\sum_{i\notin\mathcal{S}_k}\langle\nabla f_i(z_i^k), x^k - z_i^k\rangle \\
&+ \frac{1}{2n\gamma}\sum_{i\notin\mathcal{S}_k}\|x^k - z_i^k\|^2 + \Psi_{k+1}
\end{aligned}$$

(by the $L-$ smoothness of f_i)

$$\begin{aligned}
&\le g(x^k) + \frac{1}{n}\sum_{i\in\mathcal{S}_k} f_i(z_i^k) + \frac{L}{2n}\sum_{i\in\mathcal{S}_k}\|z_i^{k+1} - z_i^k\|^2 \\
&+ \frac{1}{n}\sum_{i\in\mathcal{S}_k}\langle\nabla f_i(z_i^{k+1}), x^k - z_i^k\rangle + \frac{1}{2n\gamma}\sum_{i\in\mathcal{S}_k}\|x^k - z_i^{k+1}\|^2 \\
&+ \frac{1}{n}\sum_{i\notin\mathcal{S}_k} f_i(z_i^k) + \frac{1}{n}\sum_{i\notin\mathcal{S}_k}\langle\nabla f_i(z_i^k), x^k - z_i^k\rangle \\
&+ \frac{1}{2n\gamma}\sum_{i\notin\mathcal{S}_k}\|x^k - z_i^k\|^2 + \Psi_{k+1} \\
&= g(x^k) + \frac{1}{n}\sum_{i=1}^{n} f_i(z_i^k) + \frac{1}{n}\sum_{i=1}^{n}\langle\nabla f_i(z_i^k), x^k - z_i^k\rangle \\
&+ \frac{L}{2n}\sum_{i\in\mathcal{S}_k}\|z_i^{k+1} - z_i^k\|^2 \\
&+ \frac{1}{2n\gamma}\sum_{i\in\mathcal{S}_k}\|x^k - z_i^{k+1}\|^2 + \frac{1}{n}\sum_{i\in\mathcal{S}_k}\langle\nabla f_i(z_i^{k+1}) - \nabla f_i(z_i^k), x^k - z_i^k\rangle \\
&+ \frac{1}{2n\gamma}\sum_{i\notin\mathcal{S}_k}\|x^k - z_i^k\|^2 + \Psi_{k+1}.
\end{aligned} \tag{22}$$

Next, applying the square-norm expansion

$$\|x^k - z_i^{k+1}\|^2 = \|x^k - z_i^k\|^2 + 2\langle x^k - z_i^k, z_i^k - z_i^{k+1}\rangle + \|z_i^k - z_i^{k+1}\|^2.$$

For non-updated clients $i \notin \mathcal{S}_k$, the local variable remains unchanged, i.e., $z_i^{k+1} = z_i^k$. Substituting these relations into the original expression gives

$$\begin{aligned}
&\frac{1}{2n\gamma}\sum_{i\in\mathcal{S}_k}\|x^k - z_i^{k+1}\|^2 + \frac{1}{2n\gamma}\sum_{i\notin\mathcal{S}_k}\|x^k - z_i^k\|^2 \\
&= \frac{1}{2n\gamma}\sum_{i=1}^{n}\|x^k - z_i^k\|^2 + \frac{1}{2n\gamma}\sum_{i\in\mathcal{S}_k}\left[2\langle x^k - z_i^k, z_i^k - z_i^{k+1}\rangle\right. \\
&\left. + \|z_i^k - z_i^{k+1}\|^2\right],
\end{aligned}$$

Inserting the reorganized expression into the expansion of $V^{k+1}(x^{k+1})$ and collecting common terms gives

$$V^{k+1}(x^{k+1}) = V^k(x^k) + \frac{1}{n}\sum_{i\in\mathcal{S}_k}\langle\nabla f_i(z_i^{k+1}) - \nabla f_i(z_i^k), x^k - z_i^k\rangle + \frac{1}{n\gamma}\sum_{i\in\mathcal{S}_k}\langle z_i^{k+1} - z_i^k, z_i^k - x^k\rangle + \frac{1+L\gamma}{2n\gamma}\sum_{i\in\mathcal{S}_k}\|z_i^{k+1} - z_i^k\|^2 + \Psi_{k+1}. \tag{23}$$

Then, from the update rule of y_i^{k+1} in Algorithm 1 together with Equations 10 and 11, we derive an expression for $z_i^k - x^k$:

$$\begin{aligned} z_i^k - x^k &= \frac{1}{\lambda}(y_i^k - y_i^{k+1}) \\ &= \frac{1}{\lambda}(w_i^k - w_i^{k+1}) + \frac{\gamma}{\lambda}(\nabla f_i(w_i^k) - \nabla f_i(w_i^{k+1})) \\ &= \frac{1}{\lambda}(z_i^k - z_i^{k+1}) + \frac{\gamma}{\lambda}(\nabla f_i(z_i^k) - \nabla f_i(z_i^{k+1})) \\ &+ \frac{1}{\lambda}[(m_i^{k+1} + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1}))) \\ &- (m_i^k + \gamma(\nabla f_i(z_i^k) - \nabla f_i(w_i^k)))] \\ &= \frac{1}{\lambda}(z_i^k - z_i^{k+1}) + \frac{\gamma}{\lambda}(\nabla f_i(z_i^k) - \nabla f_i(z_i^{k+1})) + n_i^k, \end{aligned} \tag{24}$$

where n_i^k is a composite error term involving the approximation errors m_i^k, m_i^{k+1} and gradient differences. The subsequent analysis will control the impact of n_i^k via its norm bound. It is defined as

$$n_i^k = \frac{1}{\lambda}[(m_i^{k+1} + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1}))) - (m_i^k + \gamma(\nabla f_i(z_i^k) - \nabla f_i(w_i^k)))],$$

Its squared norm satisfies

$$\begin{aligned} \|n_i^k\|^2 &= \frac{1}{\lambda^2}\|m_i^{k+1} - m_i^k + \gamma(\nabla f_i(z_i^{k+1}) - \nabla f_i(w_i^{k+1})) + \gamma(\nabla f_i(w_i^k) - \nabla f_i(z_i^k))\|^2 \\ &\le \frac{2(1+\gamma L)^2}{\lambda^2}\left[\|m_i^k\|^2 + \|m_i^{k+1}\|^2\right] \end{aligned}$$

By applying the L-smoothness of f_i, the Young's inequality, and Equation 24, we obtain for any $\beta_4 > 0$ that

$$\begin{aligned} V^{k+1}(x^{k+1}) &\le V^k(x^k) + \frac{[\lambda(1+L\gamma) - 2]}{2\lambda\gamma n}\sum_{i\in\mathcal{S}_k}\|z_i^{k+1} - z_i^k\|^2 \\ &+ \frac{\gamma}{\lambda n}\sum_{i\in\mathcal{S}_k}\|\nabla f_i(z_i^{k+1}) - \nabla f_i(z_i^k)\|^2 \\ &+ \frac{1}{\gamma n}\sum_{i\in\mathcal{S}_k}\langle n_i^k, (z_i^{k+1} - z_i^k) + \gamma(\nabla f_i(z_i^k) - \nabla f_i(z_i^{k+1}))\rangle + \Psi_{k+1} \\ &\textit{(by the } L - \textit{smoothness of } f_i) \\ &\le V^k(x^k) + \frac{\gamma L^2}{\lambda n}\sum_{i\in\mathcal{S}_k}\|z_i^{k+1} - z_i^k\|^2 \\ &+ \frac{[\lambda(1+L\gamma) - 2]}{2\lambda\gamma n}\sum_{i\in\mathcal{S}_k}\|z_i^{k+1} - z_i^k\|^2 + \Psi_{k+1} \\ &+ \frac{1}{\gamma n}\sum_{i\in\mathcal{S}_k}\left[\frac{1}{\beta_4}\|n_i^k\|^2 + 2\beta_4\|z_i^k - z_i^{k+1}\|^2 + 2\beta_4\gamma^2\|\nabla f_i(z_i^k) - \nabla f_i(z_i^{k+1})\|^2\right] \\ &\le V^k(x^k) - \frac{[2 - \lambda(1+L\gamma) - 2L^2\gamma^2 - 4\lambda\beta_4(1+L^2\gamma^2)]}{2\lambda\gamma n}\sum_{i\in\mathcal{S}_k}\|z_i^{k+1} - z_i^k\|^2 + \frac{1}{\gamma\beta_4 n}\sum_{i\in\mathcal{S}_k}\|n_i^k\|^2 + \Psi_{k+1} \\ &\le V^k(x^k) - \frac{[2 - \lambda(1+L\gamma) - 2L^2\gamma^2 - 4\lambda\beta_4(1+L^2\gamma^2)]}{2\lambda\gamma n}\sum_{i\in\mathcal{S}_k}\|z_i^{k+1} - z_i^k\|^2 \\ &+ \frac{2(1+\gamma L)^2}{\gamma\beta_4\lambda^2 n}\sum_{i\in\mathcal{S}_k}\left[\|m_i^k\|^2 + \|m_i^{k+1}\|^2\right] + \Psi_{k+1}. \end{aligned} \tag{25}$$

Next, leveraging the L-smoothness of f_i and assuming $\gamma \le \frac{1}{L}$, we demonstrate the boundedness of $V^k(x^k)$

$$\begin{aligned} V^k(x^k) &= g(x^k) + \frac{1}{n}\sum_{i=1}^n\left[f_i(z_i^k) + \left\langle\nabla f_i(z_i^k), x^k - z_i^k\right\rangle + \frac{1}{2\gamma}\left\|x^k - z_i^k\right\|^2\right] \\ &\ge g(x^k) + \frac{1}{n}\sum_{i=1}^n\left[f_i(x^k) - \frac{L}{2}\left\|x^k - z_i^k\right\|^2 + \frac{1}{2\gamma}\left\|x^k - z_i^k\right\|^2\right] \\ &\ge F(x^k) + (\frac{1}{2\gamma} - \frac{L}{2})\frac{1}{n}\sum_{i=1}^n\left\|x^k - z_i^k\right\|^2 \\ &\ge F^*. \end{aligned}$$

From Lemma 1, we have

$$\frac{\lambda^2}{2(1+\beta_1)(\gamma^2L^2+1)}\sum_{i\in\mathcal{S}_k}\left\|x^k - z_i^k\right\|^2 \le \sum_{i\in\mathcal{S}_k}\left[\|z_i^{k+1} - z_i^k\|^2 + \frac{2}{\beta_1}\left(\|m_i^{k+1}\|^2 + \|m_i^k\|^2\right)\right]. \tag{26}$$

According to the sampling scheme, we consider the expectation of $\sum_{i\in\mathcal{S}_k}\|z_i^{k+1} - z_i^k\|^2$ with respect to $\mathcal{S}_k$ conditioned on $\mathcal{A}_{k-1}$. Combined with (26), this yields

$$\begin{aligned} &\mathbb{E}\left[\sum_{i\in\mathcal{S}_k}\left\|z_i^{k+1} - z_i^k\right\|^2 |\mathcal{A}_{k-1}\right] \\ &= \sum_S\mathbb{P}(S_k = S)\sum_{i\in S}\left\|z_i^{k+1} - z_i^k\right\|^2 = \sum_{i=1}^n\mathbf{p}_i\left\|z_i^{k+1} - z_i^k\right\|^2 \\ &\ge \frac{\mathbf{p}\lambda^2}{2(1+\beta_1)(\gamma^2L^2+1)}\sum_{i=1}^n\left\|x^k - z_i^k\right\|^2 - \frac{2\mathbf{p}}{\beta_1}\sum_{i=1}^n\left(\|m_i^{k+1}\|^2 + \|m_i^k\|^2\right), \end{aligned} \tag{27}$$

where $\mathbf{p} = \min\mathbf{p}_i \in (0,1], i \in [n]$. By taking the conditional expectation of Equation 25 with respect to $\mathcal{S}_k$ conditioned on $\mathcal{A}_{k-1}$, and combining it with Equations 10, 21, 27 under the setting $\beta_3 = 1$, we derive the following

$$
\begin{aligned}
&\mathbb{E}\left[V^{k+1}(x^{k+1})|\mathcal{A}_{k-1}\right] \\
&\overset{(21)}{\leq} V^k(x^k) + \frac{2(1+\gamma L)^2}{\gamma\beta_4\lambda^2 n}\sum_{i=1}^{n}\mathbf{p}_i\left[\|m_i^k\|^2 + \|m_i^{k+1}\|^2\right] \\
&\quad - \frac{[2-\lambda(1+L\gamma)-2L^2\gamma^2-4\lambda\beta_4(1+L^2\gamma^2)]}{2\lambda\gamma n} \\
&\mathbb{E}\left[\sum\nolimits_{i\in S_k}\left\|z_i^{k+1}-z_i^k\right\|^2|\mathcal{A}_{k-1}\right] \\
&\quad + \frac{2\varepsilon_1}{n\gamma}\mathbb{E}\left[\sum_{i=1}^{n}\left\|e_i^k\right\|^2 + \sum_{i=1}^{n}\left\|e_i^{k+1}\right\|^2\right] + \frac{(1+\gamma^2L^2)}{n\gamma} \\
&\sum_{i=1}^{n}\left[(1-\mathbf{p}_i)\|m_i^k\|^2 + \mathbf{p}_i\|m_i^{k+1}\|^2\right] \\
&\qquad \text{(by the definition of absolute compressor)} \\
&\overset{(27)}{\leq} V^k(x^k) + \frac{2(1+\gamma L)^2}{\gamma\beta_4\lambda^2 n}\sum_{i=1}^{n}\mathbf{p}_i\left[\|m_i^k\|^2 + \|m_i^{k+1}\|^2\right] \\
&\quad - \frac{\mathbf{p}\lambda[2-\lambda(1+L\gamma)-2L^2\gamma^2-4\lambda\beta_4(1+L^2\gamma^2)]}{4\gamma n(1+\beta_1)(\gamma^2L^2+1)} \\
&\quad \sum_{i=1}^{n}\left\|x^k - z_i^k\right\|^2 \\
&\quad + \frac{\mathbf{p}[2-\lambda(1+L\gamma)-2L^2\gamma^2-4\lambda\beta_4(1+L^2\gamma^2)]}{\lambda\gamma\beta_1 n} \\
&\quad \sum_{i=1}^{n}\left(\|m_i^{k+1}\|^2 + \|m_i^k\|^2\right) \\
&\quad + \frac{4\varepsilon_1}{\gamma}\nu^2 + \frac{(1+\gamma^2L^2)}{n\gamma}\sum_{i=1}^{n}\left[(1-\mathbf{p}_i)\|m_i^k\|^2 + \mathbf{p}_i\|m_i^{k+1}\|^2\right] \\
&\overset{(10)}{\leq} V^k(x^k) - \frac{\pi}{2n}\sum_{i=1}^{n}\left\|x^k - z_i^k\right\|^2 + \frac{4\varepsilon_1}{\gamma}\nu^2 + \frac{1}{n}\sum_{i=1}^{n}(\delta_1(\varepsilon_i^k)^2 \\
&\quad + \delta_2(\varepsilon_i^{k+1})^2).
\end{aligned}
$$

To guarantee the descent property, let

$$
\pi = \frac{\mathbf{p}\lambda[2-\lambda(1+L\gamma)-2L^2\gamma^2-4\lambda\beta_4(1+L^2\gamma^2)]}{2\gamma(1+\beta_1)(\gamma^2L^2+1)} > 0.
$$

Then, we have

$$
0 < \lambda < \frac{\min\{\sqrt{4\beta_4+\frac{17}{16}}-\frac{1}{4},2\}}{4\beta_4+1} \quad \text{and} \quad 0 < \gamma < \frac{\sqrt{(1-\frac{\lambda}{4})^2-\lambda^2\beta_4(4\beta_4+1)}-\frac{\lambda}{4}}{L(2\lambda\beta_4+1)}.
$$

Theorem 1. Let $\left\{\left(y_i^k, z_i^k, x_i^k, e_i^k, x^k\right)\right\}$ be generated by Algorithm 1. Suppose that Assumptions 1 and 2 hold, for $0 < \gamma < \frac{\sqrt{(1-\frac{\lambda}{4})^2-\lambda^2\beta_4(4\beta_4+1)}-\frac{\lambda}{4}}{L(2\lambda\beta_4+1)}$ and $0 < \lambda < \frac{\min\{\sqrt{4\beta_4+\frac{17}{16}}-\frac{1}{4},2\}}{4\beta_4+1}$, we have

$$
\begin{aligned}
\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\mathcal{G}_\gamma(x^k)\|^2\right] &\leq \frac{M_1}{K}(F(x^0)-F^*) \\
&\quad + \frac{1}{nK}\sum_{k=0}^{K-1}\sum_{i=1}^{n}\left[M_2(\varepsilon_i^k)^2 + M_3(\varepsilon_i^{k+1})^2\right] \\
&\quad + \frac{M_4}{K}\nu^2,
\end{aligned} \tag{28}
$$

where

$$
\begin{aligned}
M_1 &= \frac{4(1+\beta_2)(1+\gamma L)^2}{\pi\gamma^2}, \quad M_2 = \frac{(2\delta_1\beta_2+\pi)}{\beta_2}M_1 \\
M_3 &= \delta_2 M_1, \quad M_4 = \frac{4\varepsilon_1 K}{\gamma}M_1 + \frac{4K}{n\gamma^2},
\end{aligned}
$$

with $\varepsilon_1, \beta_2 > 0$, and π, δ_1, δ_2 defined in Lemma 3.

Proof. First, it follows from Lemma 3 that

$$
\begin{aligned}
\sum_{i=1}^{n}\left\|x^k - z_i^k\right\|^2 &\leq \frac{2n}{\pi}\Bigg[V^k(x^k) - \mathbb{E}\left[V^{k+1}(x^{k+1})|\mathcal{A}_{k-1}\right] \\
&\quad + \frac{4\varepsilon_1}{\gamma}\nu^2 + \frac{1}{n}\sum_{i=1}^{n}(\delta_1(\varepsilon_i^k)^2 + \delta_2(\varepsilon_i^{k+1})^2)\Bigg].
\end{aligned}
$$

Combining the derived estimates and Lemma 2, we obtain

$$
\begin{aligned}
\|\mathcal{G}_\gamma(x^k)\|^2 &\leq \frac{2(1+\gamma L)^2}{n\gamma^2}\sum_{i=1}^{n}\Bigg[(1+\beta_2)\|z_i^k - x^k\|^2 \\
&\quad + (1+\frac{1}{\beta_2})\|m_i^k\|^2\Bigg] + \frac{2}{n\gamma^2}\sum_{i=1}^{n}\|e_i^{k-1}-e_i^k\|^2 \\
&\leq \frac{4(1+\beta_2)(1+\gamma L)^2}{\pi\gamma^2}\Bigg[V^k(x^k) \\
&\quad - \mathbb{E}\left[V^{k+1}(x^{k+1})|\mathcal{A}_{k-1}\right]\Bigg] \\
&\quad + \frac{4(1+\beta_2)(1+\gamma L)^2}{n\pi\gamma^2}\sum_{i=1}^{n}(\delta_1(\varepsilon_i^k)^2 + \delta_2(\varepsilon_i^{k+1})^2) \\
&\quad + \frac{2(1+\beta_2)(1+\gamma L)^2}{n\gamma^2\beta_2}(\varepsilon_i^k)^2 + \frac{2}{n\gamma^2}\sum_{i=1}^{n}\|e_i^{k-1}-e_i^k\|^2 \\
&\quad + \frac{16(1+\beta_2)(1+\gamma L)^2\varepsilon_1}{\pi\gamma^3}\nu^2.
\end{aligned} \tag{29}
$$

Taking the total expectation of $\|\mathcal{G}_\gamma(x^k)\|^2$ with respect to $\mathcal{A}_k$, and by using the update of e_i^k and the definition of the absolute compressor, we obtain the following result

$$
\begin{aligned}
\mathbb{E}\left[\|\mathcal{G}_\gamma(x^k)\|^2\right] &\leq M_1(\mathbb{E}\left[V^k(x^k)\right] - \mathbb{E}\left[V^{k+1}(x^{k+1})\right]) \\
&\quad + \frac{M_2}{n}\sum_{i=1}^{n}(\varepsilon_i^k)^2 + \frac{M_3}{n}\sum_{i=1}^{n}(\varepsilon_i^{k+1})^2 + \frac{M_4}{K}\nu^2,
\end{aligned}
$$

where

$$M_1 = \frac{4(1+\beta_2)(1+\gamma L)^2}{\pi\gamma^2}, \quad M_2 = \frac{2(1+\beta_2)(1+\gamma L)^2(4\delta_1\beta_2 + 2\pi)}{\gamma^2\beta_2\pi}$$
$$M_3 = \frac{4(1+\beta_2)(1+\gamma L)^2\delta_2}{\pi\gamma^2}, \quad M_4 = \frac{16(1+\beta_2)(1+\gamma L)^2\varepsilon_1 K}{\pi\gamma^3} + \frac{4K}{n\gamma^2},$$

is four constants. Summing the inequality over k from 0 to $K-1$, and then scaling the resultant sum by $\frac{1}{K}$, we derive

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\mathcal{G}_\gamma(x^k)\|^2\right] \le \frac{M_1}{K}\left(\mathbb{E}\left[V^0(x^0)\right] - \mathbb{E}\left[V^K(x^K)\right]\right) + \frac{1}{K}\sum_{k=0}^{K-1}\left[\frac{M_2}{n}\sum_{i=1}^{n}(\varepsilon_i^k)^2 + \frac{M_3}{n}\sum_{i=1}^{n}(\varepsilon_i^{k+1})^2 + \frac{M_4}{K}\nu^2\right]. \tag{30}$$

With the initial condition $z_i^0 = x^0$, we obtain $V^0(x^0) = g(x^0) + \frac{1}{n}\sum_{i=1}^{n} f_i(z_i^0) = F(x^0)$. Together with the lower bound $\mathbb{E}\left[V^{k+1}(x^{k+1})\right] \ge F^*$, this implies that Equation 30 simplifies to

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\mathcal{G}_\gamma(x^k)\|^2\right] \le \frac{M_1}{K}(F(x^0) - F^*) + \frac{1}{nK}\sum_{k=0}^{K-1}\sum_{i=1}^{n}\left[M_2(\varepsilon_i^k)^2 + M_3(\varepsilon_i^{k+1})^2\right] + \frac{M_4}{K}\nu^2, \tag{31}$$

which proves Equation 28.

Corollary 1. Suppose that Assumptions 1 and 2 hold, EF-Feddr (Algorithm 1) will find a ε-stationary point x such that $\mathbb{E}\left\|\mathcal{G}_\gamma(x^k)\right\| \le \varepsilon$ in the following number of iterations

$$K \ge \frac{M_1\left[F(x^0) - F^*\right] + (M_2 + M_3)M + M_4\nu^2}{\varepsilon^2},$$

where $M > 0$ is a constant, and M_1, M_2, M_3, M_4 are defined in Theorem 1. Consequently, the communication complexity is $K = O(\frac{1}{\varepsilon^2})$.

Proof. As described in Tran-Dinh et al. (2021), the choice of accuracies ε_i^k is constrained such that for a given constant $M > 0$, $\frac{1}{n}\sum_{k=0}^{K-1}\sum_{i=1}^{n}(\varepsilon_i^k)^2 \le M$. Therefore,

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\|\mathcal{G}_\gamma(x^k)\|^2\right] \le \frac{M_1(F(x^0) - F^*) + (M_2 + M_3)M + M_4\nu^2}{K}. \tag{32}$$

Consequently, to guarantee $\mathbb{E}\left\|\mathcal{G}_\gamma(x^k)\right\| \le \varepsilon$, we have

$$K \ge \frac{M_1\left[F(x^0) - F^*\right] + (M_2 + M_3)M + M_4\nu^2}{\varepsilon^2}.$$

Therefore, we can take $K = \left\lfloor \frac{M_1\left[F(x^0)-F^*\right]+(M_2+M_3)M+M_4\nu^2}{\varepsilon^2} \right\rfloor = O\left(\frac{1}{\varepsilon^2}\right)$ as its lower bound.

5 Experiments

In the experiments, we evaluate EF-Feddr against Eco-FedSplit (Khirirat et al., 2022), Eco-FedProx (Khirirat et al., 2022), and FedDR (Tran-Dinh et al., 2021). In all compression-based baselines, the compression operator C denotes Top-k sparsification. For a fair comparison, we implement Eco-FedSplit, Eco-FedProx, and EF-Feddr on top of the FedDR framework. All experiments are conducted in TensorFlow (Abadi et al., 2016) on a cluster equipped with NVIDIA Tesla P100 (16 GB) GPUs. We next describe the datasets and models used in our study.

5.1 Non-IID datasets

We evaluate on both synthetic and real-world datasets: synthetic-(l, s), FEMNIST, and Shakespeare. Following prior studies (Caldas et al., 2018; Tran-Dinh et al., 2021), we generate synthetic-(l, s) with $(l, s) = \{(0, 0), (1, 1)\}$, where l controls the number of differing local models and s controls the degree of local data heterogeneity; larger l and s imply stronger non-IID heterogeneity. FEMNIST extends MNIST to 62 classes with over 800k samples; we use an 80%/20% train/test split and partition by writer, which naturally induces client-level heterogeneity. Shakespeare is a character-level language modeling corpus; we partition by user/play, so each client holds a distinct subset of texts (plays/scenes), yielding non-uniform label distributions across clients. In this context, the degree of non-IID-ness within each client's dataset is quantified by the number of classes present. Specifically, the Shakespeare dataset's non-IID-ness is delineated by the allocation of various plays' texts among clients. Each client is allocated a distinct subset of the corpus, which may include a varying number of plays and scenes. This results in a non-uniform distribution of text, where certain clients predominantly receive data from specific plays, whereas others obtain a more diverse range of content. Analogously, the FEMNIST dataset establishes non-IID-ness through the distribution of handwriting samples across different writers. Each client's dataset comprises samples from a subset of writers, thereby leading to variability in handwriting styles and features among clients. The datasets and model configurations used in our experiments are summarized in Table 2, which outlines their key statistical characteristics.

5.2 Models and hyper-parameters selection

We use a fully connected network with a 60-32-10 architecture and train it for 200 communication rounds with a learning rate of 0.01 on all synthetic datasets. At each round, 10 out of 30 clients are sampled. To evaluate the algorithm's performance with an increased number of clients, we further extended the Synthetic-(1,1) setup from the original 30 clients to 90 clients while preserving the statistical characteristics defined by the (l, s) parameters. The data generation process maintained the same non-IID partition pattern and per-client data distribution profile as the original setup. The client sampling ratio was kept constant at 1/3 (that is, selecting 30 out of 90 clients per round). Eco-FedSplit applies error-compensated compression to FedSplit, and Eco-FedProx does so to FedProx. To study an image classification problem on FEMNIST, we employ artificial neural networks (ANN) consisting of two fully connected layers. The first layer has 128 neurons followed by a ReLU activation function, and the second layer has 62 neurons followed by a softmax activation function for classification. In this experiment, we sample 50 clients out of 200 to perform updates at each communication round for all the above-mentioned algorithms. The model used for FEMNIST is trained for 200 communication rounds in total with an optimal learning rate of 0.003. Consistent with prior research (Li et al., 2020), our approach to character-level prediction in the Shakespeare dataset utilizes a recurrent neural network (RNN) architecture. Specifically, we deploy a two-layer stacked LSTM classifier, each layer comprising 256 hidden units. Each input sequence is structured to include 80 characters, which are initially embedded into an eight-dimensional space prior to LSTM processing. The model subsequently generates a 62-class softmax distribution over the character vocabulary for each training instance. The training regimen involves a total of 50 communication rounds. An optimal learning rate of 0.08 is determined for the four operator-splitting-based federated learning algorithms employed in this study. Parameters for each algorithm such as $\alpha \in (0, 2)$ and $\eta \in [1, 1,000]$ for FedDR, $\mu \in [0.001, 1]$ for Eco-FedProx, and $\lambda \in (0, 2)$ and $\gamma \in [1, 1,000]$ for EF-Feddr are tuned from a large range of values. For each dataset, we pick the most suitable parameters for each algorithm.

TABLE 2 Dataset and model characteristics for federated training.

Dataset	Client participation	Samples	Model	Parameters
Synthetic-(0, 0)	1/3	75,349	ANN	2,282
Synthetic-(1, 1)	1/3	75,349	ANN	2,282
FEMNIST	1/4	18,345	CNN	214,370
Shakespeare	10/143	517,106	LSTM	817,872

5.3 Comparison of methods

Figures 1–3 report training loss/accuracy and test accuracy vs. communication rounds and communication cost on the synthetic datasets; Figure 4 shows the same on FEMNIST. A key observation is that expanding the total number of clients does not substantially degrade the performance of EF-Feddr. Experimental results under the scaled setting (Figure 3) confirm this: the algorithm maintains nearly identical convergence speed and final accuracy compared to the original 30-client scenario (Figure 2). Across heterogeneous settings, EF-Feddr consistently outperforms the baselines. On FEMNIST, EF-Feddr reaches 80.5% test accuracy at round 50, whereas Eco-FedSplit attains 74.5% only at round 200. Within 200 rounds, EF-Feddr improves accuracy by 12.97% and 7.93% over Eco-FedSplit and Eco-FedProx, respectively. On synthetic-$(0, 0)$, EF-Feddr exceeds the

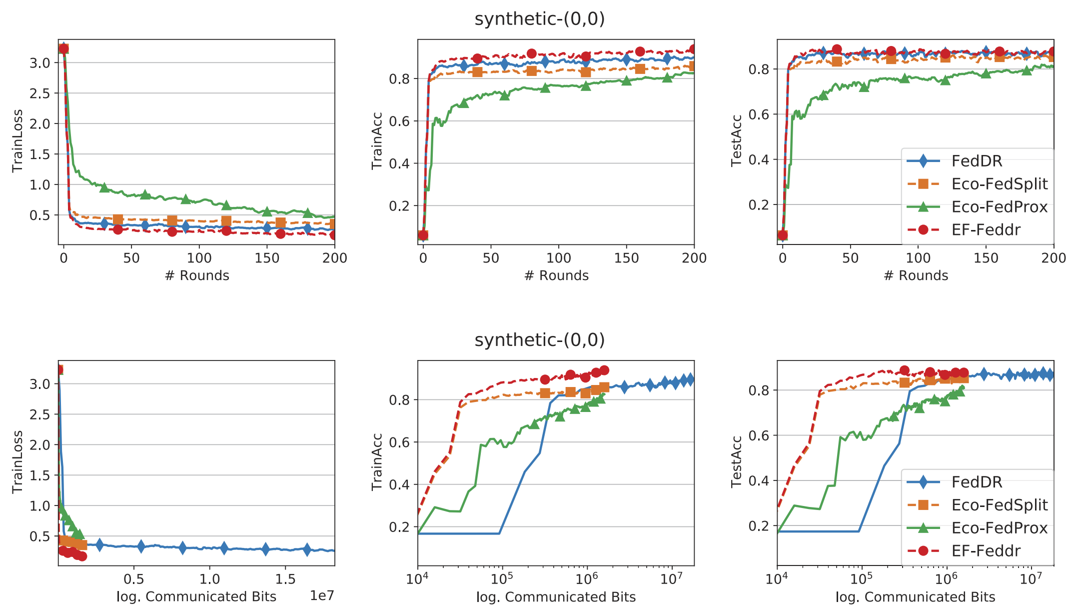

FIGURE 1
Convergence performance of different methods on the synthetic-$(0, 0)$ dataset with Top-k and participation rate $p = 0.3$.

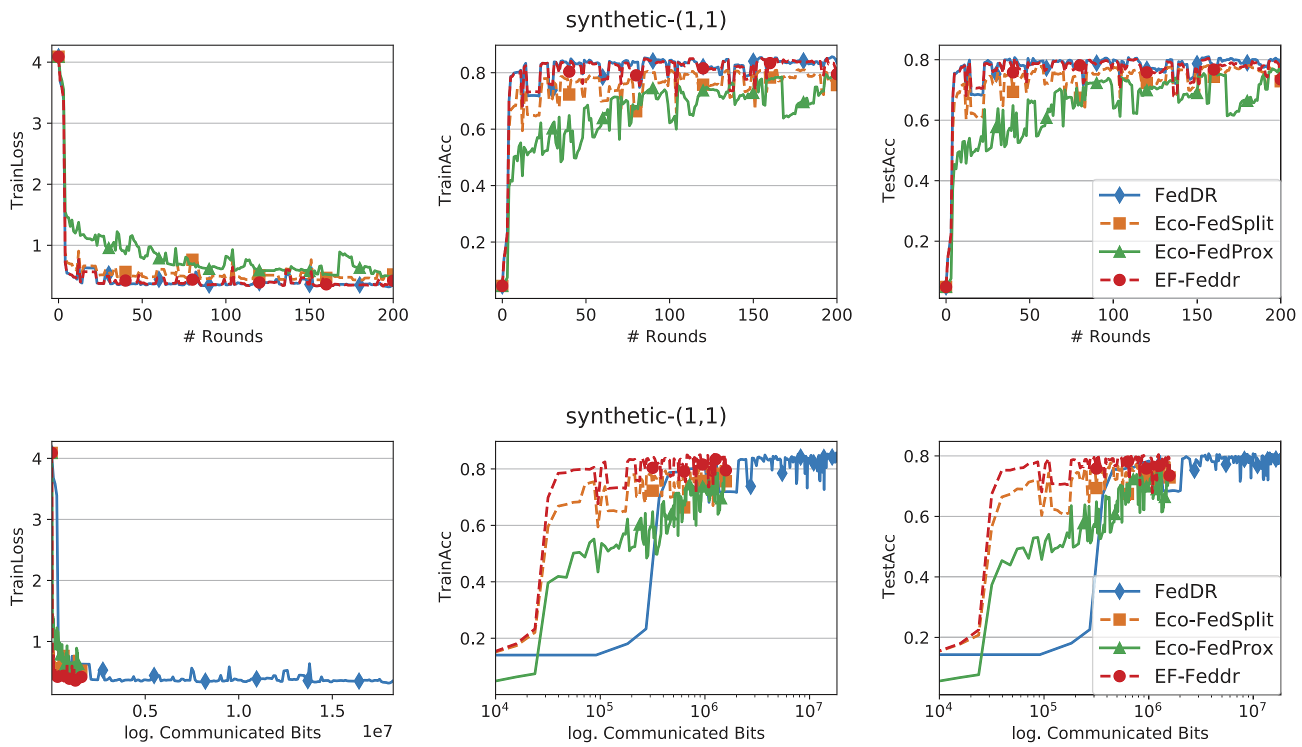

FIGURE 2
Convergence performance of different methods on the synthetic-(1, 1) dataset with Top-k, participation rate $p = 0.3$, and $N = 30$ total clients.

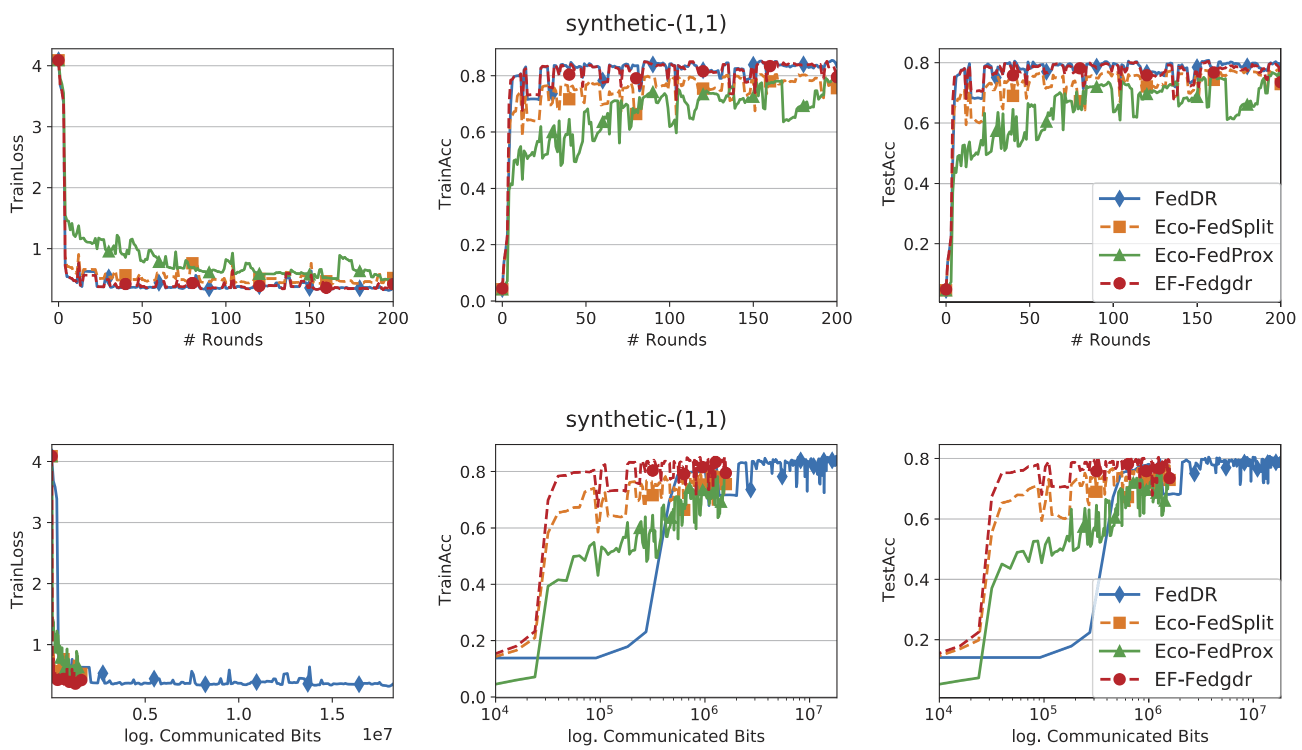

FIGURE 3
Convergence performance of different methods on the synthetic-(1, 1) dataset with Top-k, participation rate $p = 0.3$, and $N = 90$ total clients.

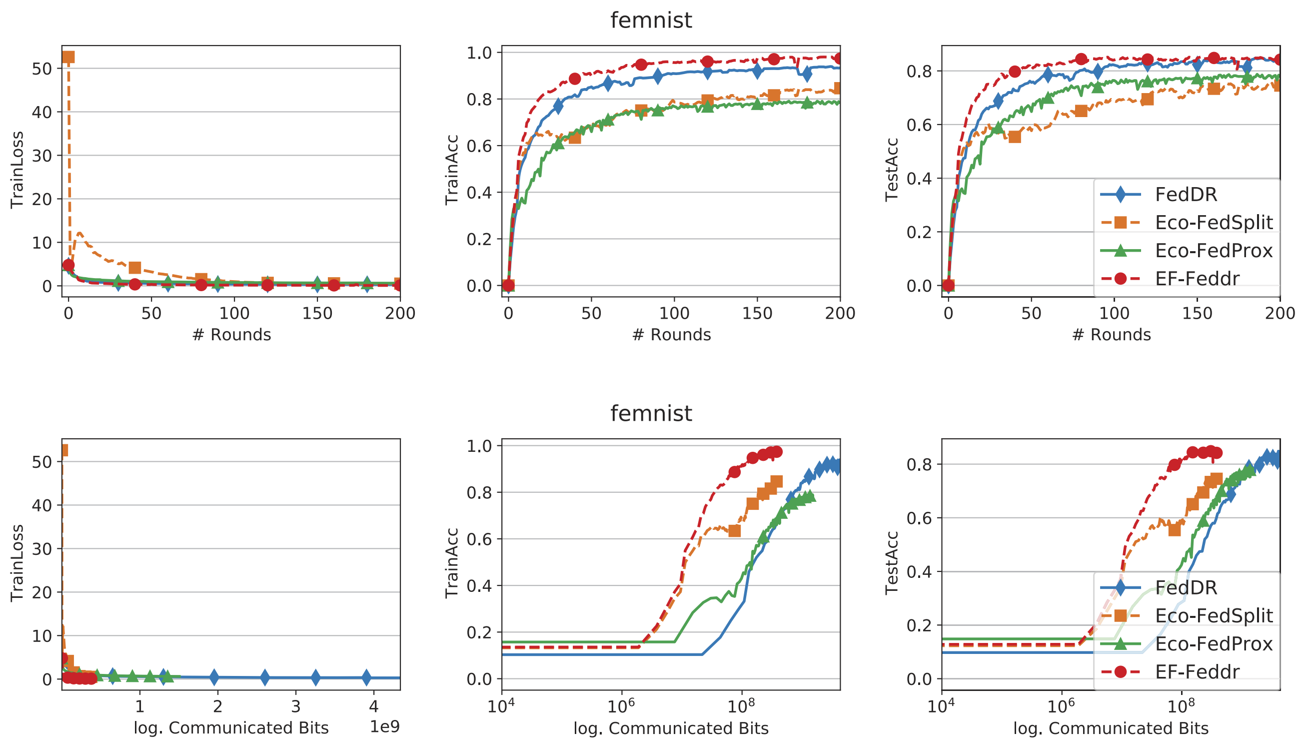

FIGURE 4
Convergence performance of different methods on the FEMNIST dataset with Top-k and participation rate $p = 0.3$.

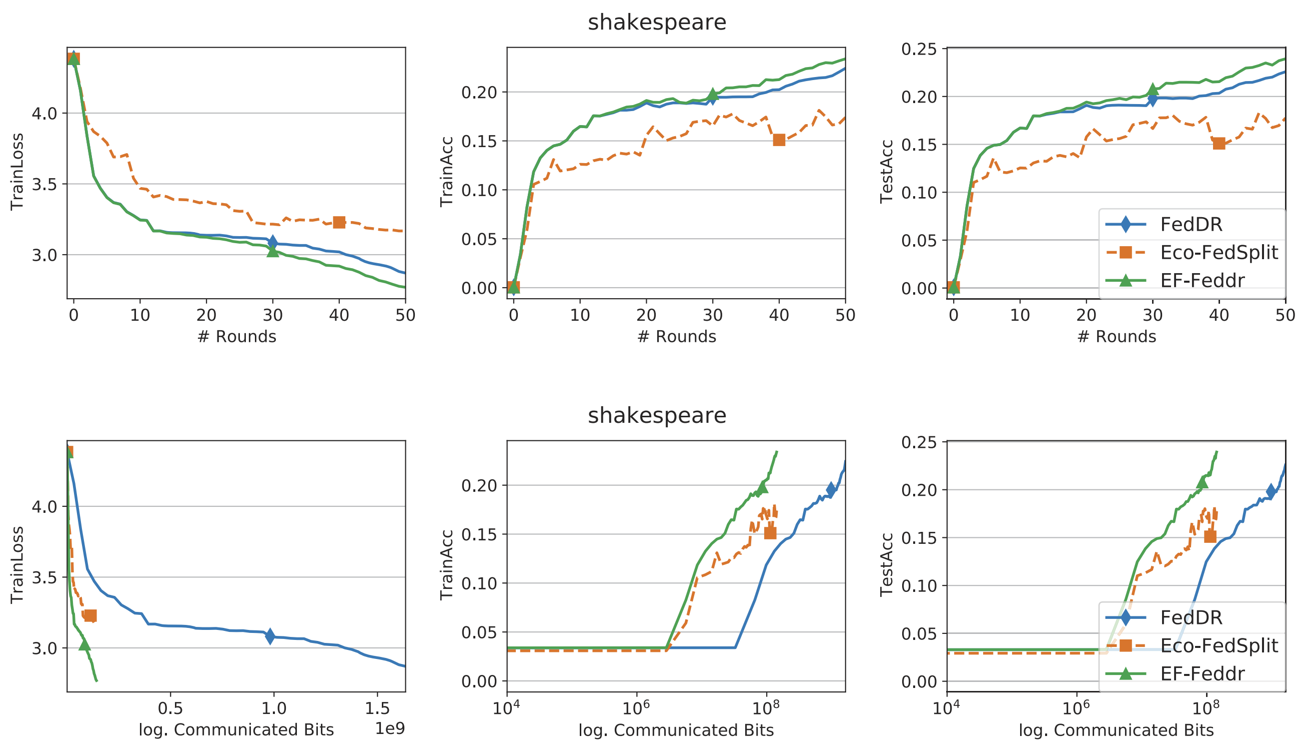

FIGURE 5
Convergence performance of different methods on the Shakespeare dataset with Top-k and participation rate $p = 0.3$.

TABLE 3 Efficiency comparison on synthetic-(1, 1) and femnist datasets.

Method	Synthetic-(1, 1) (60% accuracy)			FEMNIST (70% accuracy)		
	Rounds	Time (min)	Comm (MB)	Rounds	Time (min)	Comm (GB)
Eco-FedProx	29	6.58	0.21	61	19.72	0.24
Eco-FedSplit	12	2.85	0.09	98	48.72	0.172
FedDR	5	1.18	0.44	33	15.95	0.67
EF-Feddr	4	0.96	0.03	17	8.29	0.059

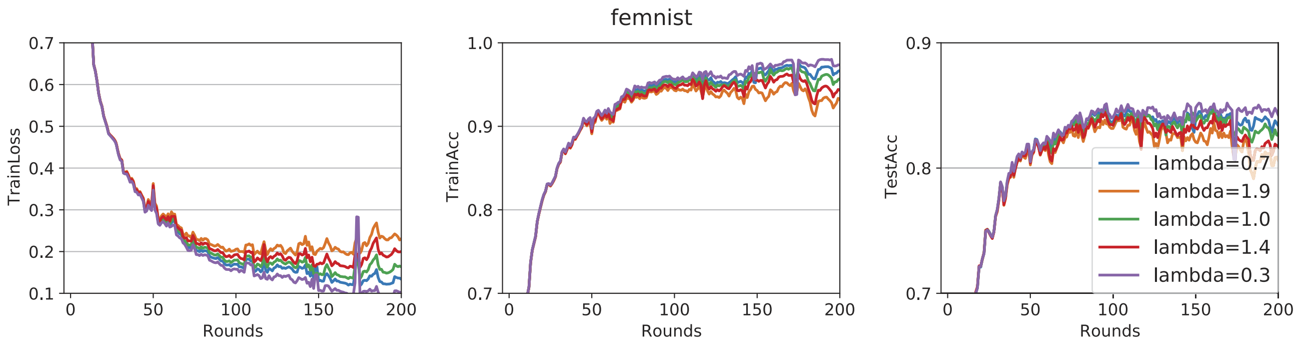

FIGURE 6
EF-Feddr on FEMNIST with relaxation parameter λ analysis.

two baselines by 3.88% and 8.40%; on synthetic-(1, 1), by 7.20% and 3.29%. On Shakespeare, Figure 5 shows EF-Feddr also surpasses two Douglas–Rachford splitting-based FL algorithms: Eco-FedSplit and FedDR. As shown in Table 3, EF-Feddr requires 18.64%–85.41% less runtime and 48.03%–93.18% less communication than baseline methods to achieve the same target test accuracy of 60% on synthetic and 70% on FEMNIST. Specifically, on FEMNIST, it meets this target in only 17 communication rounds (8.29 min), significantly outperforming competitors like Eco-FedSplit. These substantial reductions in overhead are consistently observed across the synthetic datasets. Additionally, EF-Feddr achieves a substantial reduction in communication costs without compromising performance relative to the uncompressed FedDR.

5.4 Effect of the relaxation parameter

Figure 6 examines the effect of the relaxation parameter λ over 200 iterations. Empirically, the best convergence is observed at $\lambda = 0.3$. Consistent with prior findings on FL adaptations of Douglas–Rachford splitting, choosing $0 < \lambda < 1$ often leads to faster convergence than the classical (unrelaxed) variant.

6 Discussion

This study presents EF-Feddr, a communication-efficient federated learning algorithm that combines error-compensated compression with Douglas–Rachford splitting. The method's robustness is demonstrated across controlled synthetic and real-world benchmarks, yet we recognize that extreme heterogeneity, such as single-class clients, remains a challenging frontier. Furthermore, while our experiments simulate realistic constraints (partial participation, compression), fully asynchronous updates and dynamic network conditions warrant further study in real deployments.

Recent advances in behavior-based threat hunting (Bhardwaj et al., 2022), IoT firmware security assessment (Bhardwaj et al., 2023), and energy-efficient proactive fault tolerance in cloud environments (Talwar et al., 2021) provide complementary perspectives for building reliable and secure federated systems. While this study focuses on optimization efficiency under non-IID and communication constraints, these studies collectively point toward an integrated "Optimization + System + Security" paradigm for future research. Specifically, they motivate investigations into client behavior profiling for attack detection, trusted execution at the edge, and proactive fault-tolerant scheduling, all of which are essential for deploying robust and efficient federated learning in real-world, dynamic environments. Furthermore, to strengthen the generalizability of our findings, future studies will also include evaluations on a wider variety of datasets, encompassing diverse domains, scales, and heterogeneity patterns, thereby providing a more comprehensive assessment of the algorithm's practical applicability.

7 Conclusion

In this study, we introduced EF-Feddr, a communication-efficient algorithm for non-convex federated learning that leverages the Douglas–Rachford splitting method, error feedback compression, and a relaxation strategy. EF-Feddr improves communication efficiency while preserving solution accuracy. Both theoretical analysis and empirical experiments demonstrated that EF-Feddr substantially reduces the number of bits transmitted from clients to the server compared with uncompressed FedDR. In terms of solution accuracy, EF-Feddr performs comparably to the uncompressed FedDR. Building on the Douglas–Rachford envelope, we established convergence guarantees and analyzed the communication complexity of EF-Feddr under mild assumptions. Extensive experiments further confirmed that our method significantly outperforms existing state-of-the-art approaches in non-IID settings.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: arXiv preprint arXiv:1812.01097.

Author contributions

JX: Validation, Conceptualization, Methodology, Formal analysis, Data curation, Writing – original draft, Software. CW: Visualization, Investigation, Supervision, Resources, Funding acquisition, Project administration, Writing – review & editing.

Funding

The author(s) declared that financial support was not received for this work and/or its publication.

Conflict of interest

The author(s) declared that this work was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declared that generative AI was not used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). "$TensorFlow$: a system for $Large − Scale$ machine learning," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)* (Savannah, GA), 265–283.

Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. (2017). "QSGD: communication-efficient SGD via gradient quantization and encoding," in *Advances in Neural Information Processing Systems 30*.

Atenas, F. (2025). Understanding the Douglas-Rachford splitting method through the lenses of moreau-type envelopes. *Comput. Optim. Appl.* 90, 881–910. doi: 10.1007/s10589-024-00646-9

Bao, H., Chen, P., Sun, Y., and Li, Z. (2025). EFSKIP: a new error feedback with linear speedup for compressed federated learning with arbitrary data heterogeneity. *Proc. AAAI Conf. Artif. Intell.* 39, 15489–15497. doi: 10.1609/aaai.v39i15.33700

Bernstein, J., Wang, Y.-X., Azizzadenesheli, K., and Anandkumar, A. (2018). "SIGNSGD: compressed optimisation for non-convex problems," in *International Conference on Machine Learning* (Stockholm: PMLR), 560–569.

Bhardwaj, A., Kaushik, K., Alomari, A., Alsirhani, A., Alshahrani, M. M., Bharany, S., et al. (2022). BTH: behavior-based structured threat hunting framework to analyze and detect advanced adversaries. *Electronics* 11:2992. doi: 10.3390/electronics11192992

Bhardwaj, A., Kaushik, K., Bharany, S., and Kim, S. (2023). Forensic analysis and security assessment of IOT camera firmware for smart homes. *Egypt. Inf. J.* 24:100409. doi: 10.1016/j.eij.2023.100409

Caldas, S., Duddu, S. M. K., Wu, P., Li, T., Konečnỳ, J., McMahan, H. B., et al. (2018). Leaf: a benchmark for federated settings. *arXiv [preprint]*. arXiv:1812.01097. doi: 10.4885/arXiv.1812.01097

Ezequiel, C. E. J., Gjoreski, M., and Langheinrich, M. (2022). Federated learning for privacy-aware human mobility modeling. *Front. Artif. Intell.* 5:867046. doi: 10.3389/frai.2022.867046

Godavarthi, D., Jaswanth, V., Mohanty, S., Dinesh, P., Venkata Charan Sathvik, R., Moreira, F., et al. (2025). Federated quantum-inspired anomaly detection using collaborative neural clients. *Front. Artif Intell.* 8:1648609. doi: 10.3389/frai.2025.1648609

Goel, C., Anita, X., and Anbarasi, J. L. (2025). Federated knee injury diagnosis using few shot learning. *Front. Artif. Intell.* 8:1589358. doi: 10.3389/frai.2025.1589358

He, S., Dong, Q.-L., Tian, H., and Li, X.-H. (2021). On the optimal relaxation parameters of Krasnosel'ski-Mann iteration. *Optimization* 70, 1959–1986. doi: 10.1080/02331934.2020.1767101

Islam, F., Mahmood, A., Mukhtiar, N., Wijethilake, K. E., and Sheng, Q. Z. (2024). "Fairequityfl-a fair and equitable client selection in federated learning for heterogeneous IOV networks," in *International Conference on Advanced Data Mining and Applications* (Cham: Springer), 254–269. doi: 10.1007/978-981-96-0814-0_17

Jhunjhunwala, D., Sharma, P., Nagarkatti, A., and Joshi, G. (2022). "Fedvarp: tackling the variance due to partial client participation in federated learning," in *Uncertainty in Artificial Intelligence* (Eindhoven: PMLR), 906–916.

Kant, S., da Silva, J. M. B., Fodor, G., Göransson, B., Bengtsson, M., and Fischione, C. (2022). Federated learning using three-operator ADMM. *IEEE J. Sel. Topics Signal Processing* 17, 205–221. doi: 10.1109/JSTSP.2022.3221681

Karimireddy, S. P., Rebjock, Q., Stich, S., and Jaggi, M. (2019). "Error feedback fixes signsgd and other gradient compression schemes," in *International Conference on Machine Learning* (Long Beach, CA: PMLR), 3252–3261.

Khirirat, S., Johansson, M., and Alistarh, D. (2018). "Gradient compression for communication-limited convex optimization," in *2018 IEEE Conference on Decision and Control (CDC)* (Miami, FL: IEEE), 166–171. doi: 10.1109/CDC.2018.8619625

Khirirat, S., Magnússon, S., and Johansson, M. (2022). "Eco-fedsplit: federated learning with error-compensated compression," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (Singapore: IEEE), 5952–5956. doi: 10.1109/ICASSP43922.2022.9747809

Konecný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. (2016). Federated learning: strategies for improving communication efficiency. *arXiv [preprint]*. arXiv:1610.05492. doi: 10.48550/arXiv.1610.05492

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V., et al. (2020). Federated optimization in heterogeneous networks. *Proc. Mach. Learn. Syst*. 2, 429–450. doi: 10.48550/arXiv.1812.06127

Li, X., and Li, P. (2023). "Analysis of error feedback in federated non-convex optimization with biased compression: fast convergence and partial participation," in *International Conference on Machine Learning* (Honolulu, HI: PMLR), 19638–19688.

Liu, J., Xu, L., Shen, S., and Ling, Q. (2019). An accelerated variance reducing stochastic method with Douglas-Rachford splitting. *Mach. Learn*. 108, 859–878. doi: 10.1007/s10994-019-05785-3

Liu, Y., Zhou, Y., and Lin, R. (2024). The proximal operator of the piece-wise exponential function. *IEEE Signal Process. Lett*. 31, 894–898. doi: 10.1109/LSP.2024.3370493

Long, Z., Chen, Y., Dou, H., Zhang, Y., and Chen, Y. (2024). Fedsq: sparse-quantized federated learning for communication efficiency. *IEEE Trans. Consum. Electron*. 70, 4050–4061. doi: 10.1109/TCE.2024.3352432

Malekmohammadi, S., Shaloudegi, K., Hu, Z., and Yu, Y. (2021). An operator splitting view of federated learning. *arXiv [preprint]*. arXiv:2108.05974. doi: 10.48550/arXiv.2108.05974

McMahan, B., Moore, E., Ramage, D., Hampson, S., and Arcas, B. A. (2017). "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics* (Fort Lauderdale, FL: PMLR), 1273–1282.

Mishchenko, K., Khaled, A., and Richtárik, P. (2022). "Proximal and federated random reshuffling," in *International Conference on Machine Learning* (Baltimore, MA: PMLR), 15718–15749.

Parikh, N., Boyd, S., et al. (2014). Proximal algorithms. *Found. Trends Optim*. 1, 127–239. doi: 10.1561/2400000003

Pathak, R., and Wainwright, M. J. (2020). Fedsplit: an algorithmic framework for fast federated optimization. *Adv. Neural Inf. Process. Syst*. 33, 7057–7066. doi: 10.48550/arXiv.2005.05238

Reisizadeh, A., Mokhtari, A., Hassani, H., Jadbabaie, A., and Pedarsani, R. (2020). "FEDPAQ: a communication-efficient federated learning method with periodic averaging and quantization," in *International Conference on Artificial Intelligence and Statistics* (PMLR), 2021–2031.

Richtárik, P., Sokolov, I., and Fatkhullin, I. (2021). Ef21: a new, simpler, theoretically better, and practically faster error feedback. *Adv. Neural Inf. Process. Syst*. 34, 4384–4396. doi: 10.48550/arXiv.2106.05203

Sahu, A., Dutta, A., Abdelmoniem, M., Banerjee, A., Canini, T., Kalnis, M., et al. (2021). Rethinking gradient sparsification as total error minimization. *Adv. Neural Inf. Process. Syst*. 34, 8133–8146. doi: 10.48550/arXiv.2108.00951

Saifullah, S., Mercier, D., Lucieri, A., Dengel, A., and Ahmed, S. (2024). The privacy-explainability trade-off: unraveling the impacts of differential privacy and federated learning on attribution methods. *Front. Artif. Intell*. 7:1236947. doi: 10.3389/frai.2024.1236947

Seide, F., Fu, H., Droppo, J., Li, G., and Yu, D. (2014). "1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs," in *Interspeech, Vol. 2014* (Singapore), 1058–1062. doi: 10.21437/Interspeech.2014-274

Sun, W., Wang, A., Gao, Z., and Zhou, Y. (2024). "A communication-concerned federated learning framework based on clustering selection," in *International Conference on Advanced Data Mining and Applications* (Cham: Springer), 285–300. doi: 10.1007/978-981-96-0814-0_19

Talwar, B., Arora, A., and Bharany, S. (2021). "An energy efficient agent aware proactive fault tolerance for preventing deterioration of virtual machines within cloud environment," in *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)* (*ICRITO*) (Noida: IEEE), 1–7. doi: 10.1109/ICRITO51393.2021.9596453

Tang, Z., Wang, Y., and Chang, T.-H. (2024). z-signfedavg: a unified stochastic sign-based compression for federated learning. *Proc. AAAI Conf. Artif. Intell*. 38, 15301–15309. doi: 10.1609/aaai.v38i14.29454

Tran-Dinh, Q., Pham, N. H., Phan, D. T., and Nguyen, L. M. (2021). Feddr-randomized Douglas-Rachford splitting algorithms for nonconvex federated composite optimization. *Adv. Neural Inf. Process. Syst*. 34, 30326–30338. doi: 10.48550/arXiv.2103.0345

Valdeira, P., Xavier, J., Soares, C., and Chi, Y. (2025). Communication-efficient vertical federated learning via compressed error feedback. *IEEE Trans. Signal Process*. 73, 1065–1080. doi: 10.1109/TSP.2025.3540655

Wang, H., Marella, S., and Anderson, J. (2022). "FEDADMM: a federated primal-dual algorithm allowing partial participation," in *2022 IEEE 61st Conference on Decision and Control (CDC)* (Cancún: IEEE), 287–294. doi: 10.1109/CDC51059.2022.9992745

Zhou, X., Chang, L., and Cao, J. (2023). Communication-efficient nonconvex federated learning with error feedback for uplink and downlink. *IEEE Trans. Neural Netw. Learn. Syst*. 36, 1003–1014. doi: 10.1109/TNNLS.2023.3333804