Achieving Faster than $O(1/t)$ Convergence in General Convex Federated Learning

Jie Liu *jie9@clemson.edu*
Department of Electrical and Computer Engineering
Clemson University, South Carolina, USA

Zuang Wang *zuangw@clemson.edu*
Department of Electrical and Computer Engineering
Clemson University, South Carolina, USA

Yongqiang Wang* *yongqiw@clemson.edu*
Department of Electrical and Computer Engineering
Clemson University, South Carolina, USA

Reviewed on OpenReview: *https://openreview.net/forum?id=Dae3jVdPod*

Abstract

This paper aims to achieve faster than $O(1/t)$ convergence in federated learning for general convex loss functions. Under the independent and identical distribution (IID) condition, we show that accurate convergence to an optimal solution can be achieved in convex federated learning even when individual clients select stepsizes locally without any coordination. More importantly, this local stepsize strategy allows exploitation of the local geometry of individual clients' loss functions, and is shown to lead to faster convergence than the case where a same universal stepsize is used for all clients. Then, when the distribution is non-IID, we employ the sharing of gradients besides the global model parameter to ensure $o(1/t)$ convergence to an optimal solution in convex federated learning. For both algorithms, we theoretically prove that stepsizes that are much larger than existing counterparts are allowed, which leads to much faster convergence in empirical evaluations. It is worth noting that, beyond providing a general framework for federated learning with drift correction, our second algorithm's achievement of $o(1/t)$ convergence to the exact optimal solution under general convex loss functions has not been previously reported in the federated learning literature—except in certain restricted convex cases with additional constraints. We believe that this is significant because even after incorporating momentum, existing first-order federated learning algorithms can only ensure $O(1/t)$ convergence for general convex loss functions when no additional assumptions on heterogeneity are imposed.

1 Introduction

Federated learning has received intensive attention since it was proposed by McMahan et al. (2017). Nowadays, it has found applications in diverse areas including healthcare (Xu et al., 2021a; Nguyen et al., 2022; Antunes et al., 2022), smart cities (Pandya et al., 2023; Jiang et al., 2020; Ramu et al., 2022), natural language processing (Liu et al., 2021; Lin et al., 2021; Zhu et al., 2020), the Internet of things (Nguyen et al., 2021; Zhang et al., 2022b; Ghimire & Rawat, 2022), among others. In federated learning, the training data sets are located on individual clients which cooperatively learn a common model via periodically sharing their intermediate learning results with a central server (McMahan et al., 2017). Compared to centralized learning where all data are aggregated to a data center, federated learning has many advantages, such as

*Corresponding author

enhanced security (Ma et al., 2020; Mothukuri et al., 2021; Zhang et al., 2022a), better privacy (Yang et al., 2019; Agarwal et al., 2018; Li et al., 2020), and higher communication efficiency (Sattler et al., 2019; Chen et al., 2021; Hamer et al., 2020). To date, many aspects of federated learning have been extensively studied, including stepsize design (see, e.g., Kim et al. (2023); Mukherjee et al. (2023); Pan et al. (2023)), communication efficiency (see, e.g., Nori et al. (2021); Tran et al. (2019); Liu et al. (2022)), optimization mechanism (see, e.g., Luo et al. (2021); Wei et al. (2024); Feng et al. (2021)), among others.

In federated learning, clients perform multiple local training steps before communicating with a central server to reduce the burden of information transmission (McMahan et al., 2017). However, these local training steps move local optimization variables toward the minima of local loss functions and introduce a drift from the optimal solution of the global loss function. Therefore, when the data distribution is non-IID among the clients, local training steps result in slow convergence and learning errors, which is called the "client-drift phenomenon" (Karimireddy et al., 2020; Li et al., 2019; Malinovskiy et al., 2020; Charles & Konečný, 2020; Charles & Konečný, 2021; Pathak & Wainwright, 2020). In fact, under non-IID data, popular federated learning algorithms, such as FedAvg, can only ensure accurate convergence under diminishing stepsizes, which, however, results in slow convergence Mitra et al. (2021b). It is worth noting that by imposing additional assumptions on the loss function (e.g., the interpolation and the strong growth condition used in Ma et al. (2018); Meng et al. (2020); Qin et al. (2022b); Kim et al. (2023)) or introducing additional information sharing (e.g., gradient in Mitra et al. (2021a;b)), accurate convergence can be ensured under a constant stepsize. However, these results only prove $O(1/t)$ convergence for general convex loss functions.

Inspired by the result in Lee & Wright (2019) which proves that $o(1/t)$ convergence rate can be obtained in first-order **centralized** gradient methods by employing large stepsizes, we prove that $o(1/t)$ convergence can be achieved in **general convex** federated learning, in contrast to existing state-of-the-art algorithms—which either guarantee only $O(1/t)$ convergence (Mitra et al., 2021b; Mukherjee et al., 2023; Qin et al., 2022b; Khaled et al., 2020), or rely on **additional assumptions beyond convexity** to establish $o(1/t)$ rates (Jiang et al., 2024; Kovalev et al., 2022).

The main contributions of this paper are summarized as follows:

- Under the IID condition of data distribution (also called strong growth condition in Schmidt & Roux (2013)), we prove that the conventional FedAvg algorithm (called Algorithm 1 in this paper after incorporating local stepsizes) can converge under a stepsize that is much larger than existing counterparts (our theoretically obtained stepsize is at least **two and four times larger** than the ones in Qin et al. (2022b) and Khaled et al. (2020), respectively). More importantly, we prove that our stepsize can lead to an $o(1/t)$ convergence to an accurate optimal solution, faster than the commonly believed $O(1/t)$ convergence. To our knowledge, no $o(1/t)$ convergence results have been reported in the literature for **general** convex federated learning, even after incorporating momentum (see, e.g., Xu et al. (2021b); Liu et al. (2020); Cheng et al. (2023); Yang et al. (2022)).

- Under the same condition, we prove that FedAvg can converge accurately when individual clients select their (constant) stepsizes in an uncoordinated way. This allows individual clients to exploit their local geometry of loss functions and is proven in our numerical experiments to provide a faster convergence compared with the case where a same universal stepsize is used by all clients. To our knowledge, this is the first time that such results are reported for general convex loss functions.

- Under non-IID data, we show that our Algorithm 2 can ensure accurate convergence under constant stepsizes. Compared with existing counterparts, we allow much larger stepsizes (our theoretically obtained stepsize is at least 162/5 and 4 times larger than the ones in Karimireddy et al. (2020) and Mitra et al. (2021b), respectively). More importantly, we prove that under our stpesizes, the algorithm can ensure $o(1/t)$ convergence under a **general convex** loss function, which has not been reported before for first-order federated learning algorithms, even after incorporating momentum (Xu et al., 2021b; Liu et al., 2020; Cheng et al., 2023; Yang et al., 2022). This stands in stark contrast to existing results on convex federated learning, where $o(1/K)$ convergence has only been established under **subclasses of convex functions**—such as gradient difference being uniformly bounded (Jiang et al., 2024), or Hessian difference being uniformly bounded (Kovalev et al., 2022).

- Algorithm 2 introduces a general framework for federated learning with drift correction, unifying and extending a broad class of methods that ensure convergence under non-IID data, including FedLin (Mitra et al., 2021b), FedTrack (Mitra et al., 2021a), Scaffnew (Mishchenko et al., 2022), and SCAFFOLD (Karimireddy et al., 2020). In addition, we develop a novel analytical framework that establishes a key monotonic descent property, enabling us to prove an improved $o(1/t)$ convergence rate under **general convex** objectives—an achievement that, to the best of our knowledge, has previously only been attained for **specific subclasses** of convex functions in federated learning (see, e.g., Jiang et al. (2024); Kovalev et al. (2022)). It is worth noting that extending the monotonic descent property from centralized optimization in Lee & Wright (2019) to federated learning is highly nontrivial, due to the presence of multiple heterogeneous local loss functions arising from non-IID data distributions. To the best of our knowledge, this is the first work to rigorously establish such monotonicity in the context of convex federated learning.

2 Preliminaries

Notations $\mathbb{R}^n$ and $\mathbb{R}^{n\times n}$ denote the set of real n-dimensional vectors and the set of $n \times n$-dimensional matrices, respectively. For $x \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$, $[x]_j$ and $[A]_{ij}$ denote the j^{th} element of the vector x and the $(i,j)^{th}$ element of the matrix A, respectively. For $x, y \in \mathbb{R}^n$, we define $\langle x, y\rangle = \sum_{i=1}^n [x]_i[y]_i$ and $\|x\| = \sqrt{\sum_{j=1}^n [x]_j^2}$. For a matrix $A \in \mathbb{R}^{n\times n}$, we define $\|A\|_2 = \sup_{\|x\|=1, x\in\mathbb{R}^n} \|Ax\|$. $\mathbf{0}_n \in \mathbb{R}^n$ and $\mathbf{1}_n \in \mathbb{R}^n$ are n-dimensional vectors with all elements being 0 and 1, respectively. We use $O(c(t))$ and $o(c(t))$ to represent sequences $d(t)$ satisfying $\limsup_{t\to+\infty} |\frac{d(t)}{c(t)}| < \infty$ and $\lim_{t\to\infty} \frac{d(t)}{c(t)} = 0$, respectively.

2.1 Problem Settings

We consider the following federated learning problem with clients set $\mathcal{S} = \{1, 2, \cdots, N\}$ as follows:

$$\min_{x\in\mathbb{R}^n} f(x) = \frac{1}{N}\sum_{i=1}^N f_i(x), \tag{1}$$

where $f_i : \mathbb{R}^n \to \mathbb{R}$ is the local loss function of client i. The local loss function $f_i(x)$ is dependent on the local training data of client i. We use the following standard assumptions about the loss functions (see Mitra et al. (2021a;b); Qin et al. (2022b); Mukherjee et al. (2023); Acar et al. (2021)).

Assumption 1. *For any $i \in \mathcal{S}$, $f_i(x)$ is L_i-smooth over $\mathbb{R}^n$, i.e., there exists a constant L_i such that $\|\nabla f_i(x) - \nabla f_i(y)\| \le L_i\|x - y\|$ holds for any $i \in \mathcal{S}$ and $x, y \in \mathbb{R}^n$. This implies*

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|$$

where $L = \frac{1}{N}\sum_{i=1}^N L_i$, i.e., $f(x)$ is also L-smooth over $\mathbb{R}^n$.

Assumption 2. *For any $i \in \mathcal{S}$, $f_i(x)$ is convex over $\mathbb{R}^n$. Moreover, the optimal solution set*

$$\mathcal{X}^* = \{x^* \in \mathbb{R}^n | x^* = \arg\min_{x\in\mathbb{R}^n} f(x)\}$$

is not empty, i.e., there exists at least one $x^* \in \mathbb{R}^n$ such that $\nabla f(x^*) = \mathbf{0}_n$ holds.

In existing results for federated learning (see, e.g., Mitra et al. (2021b); Qin et al. (2022b); Khaled et al. (2020); Mukherjee et al. (2023)), the theoretically obtained convergence rates are all on the order of $O(1/t)$ for general convex loss functions, where t is the number of communications between clients and the central server. In this paper, we will show that we can prove a faster $o(1/t)$ convergence rate by using a larger stepsize. To this end, we first introduce the following lemma (see Debnath & Mikusinski (2005) or Lee & Wright (2019)).

Lemma 1. *Let $\{\Delta(t)\}$ be a nonnegative sequence satisfying the following conditions:*

(1) $\{\Delta(t)\}$ is monotonically decreasing;

(2) $\{\Delta(t)\}$ is summable, that is, $\sum_{k=0}^{\infty} \Delta(k) < \infty$.

Then, we have $\Delta(t) = o(1/t)$, i.e., $\lim_{t\to\infty} t\Delta(t) = 0$.

3 Convergence under IID Data

In this section, we consider the case where the data on all clients are IID. In the literature, this is usually formulated as the following assumption (see, e.g., Schmidt & Roux (2013); Qin et al. (2022b); Kim et al. (2023)):

Assumption 3. *There exists a constant $\eta > 0$ such that $\|\nabla f_i(x)\| \leq \eta \|\nabla f(x)\|$ holds for any client $i \in \mathcal{S}$ and $x \in \mathbb{R}^n$.*

This assumption is also sometimes called Strong Growth Condition (Schmidt & Roux, 2013) and has been widely used in machine learning (Ma et al., 2018; Vaswani et al., 2019a;b; Gower et al., 2021; Meng et al., 2020). In fact, Qin et al. (2022b) recently experimentally verified that this condition is satisfied for over-parameterized models. Next, we will prove that the classic federated learning algorithm FedAvg can converge at an $o(1/t)$ rate under judiciously designed stepsizes under Assumption 3. In Section 4, we will consider the more general non-IID case.

3.1 Algorithm Description

For the sake of completeness, we restate FedAvg in McMahan et al. (2017) as Algorithm 1 (with an extension that we allow clients to use heterogeneous local stepsizes). Specifically, in this algorithm, instead of using a universal stepsize α, each client selects its own stepsize α_i without coordination with other clients. As proven in the next subsection and the numerical experimental evaluation, this enables our algorithm to obtain faster convergence than existing counterparts.

Algorithm 1 (FedAvg with local stepsizes)

```
Input: Initial value $\bar{x}(1)$, local training period $\tau$, the stepsize $\alpha_i$ for client $i$
for $t = 1$ to $T$ do
  for $i = 1$ to $N$ do
    Each client $i$ sets $x_{i,0}(t) = \bar{x}(t)$.
    for $k = 0$ to $\tau - 1$ do
      Each client $i$ does local training
          $x_{i,k+1}(t) = x_{i,k}(t) - \alpha_i \nabla f_i(x_{i,k}(t)).$          (2)
    end for
  end for
  Each client $i$ transmits $x_{i,\tau}(t)$ to the central server and receives $\bar{x}(t+1) = \frac{1}{N}\sum_{i=1}^{N} x_{i,\tau}(t)$ from the central server.
end for
```

3.2 Convergence Analysis

Theorem 1. *Under Assumptions 1, 2, and 3, if the stepsize of client i satisfies $\alpha_i = \alpha > 0$ for all $i \in \mathcal{S}$ and*

$$\alpha < \min_{1 \leq i \leq N} \left\{ \frac{1}{L_i \tau}, \frac{8\tau}{L(2\tau + \eta(\tau-1))^2 + 4\eta L \tau(\tau - 1)} \right\}, \tag{3}$$

where $L = \frac{1}{N}\sum_{i=1}^{N} L_i$, then $f(\bar{x}(t))$ converges to $f(x^*)$ with the convergence rate $o(1/t)$, i.e.,

$$\lim_{t\to\infty} t\{f(\bar{x}(t)) - f(x^*)\} = 0.$$

Proof. See Appendix C. □

In fact, we can allow the stepsize α_i of the client $i \in \mathcal{S}$ in Theorem 1 to be larger to achieve faster convergence of Algorithm 1, which is detailed in Theorem 2.

Theorem 2. *Under Assumptions 1, 2, and 3, if the stepsize α_i for client $i \in \mathcal{S}$ in Algorithm 1 satisfies*

$$0 < \alpha_i < \frac{1}{L_i}, \tag{4}$$

we have $\lim_{t\to\infty} f(\bar{x}(t)) = f(x^*)$ and

$$f(\frac{1}{T}\sum_{t=1}^{T}\bar{x}(t)) - f(x^*) \leq \frac{\|\bar{x}(1) - x^*\|^2}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}T}.$$

Proof. See Appendix D. □

The proposed stepsize in Theorem 2 is larger than designed stepsizes for FedAvg in existing theoretical results. For example, Qin et al. (2022b) and Khaled et al. (2020) obtained stepsizes that should satisfy $0 < \alpha \leq \frac{1}{2L}$ and $0 < \alpha \leq \frac{1}{4L}$, respectively. A simple comparison with (4) shows that our stepsize can be **two and four times as large** besides the additional flexibility of allowing different clients to select their local stepsizes to exploit local geometry to speed up convergence. In fact, our numerical experiments in Figure 1 confirm that our stepsize strategy indeed leads to much faster convergence than the ones in Qin et al. (2022b); Khaled et al. (2020); Mukherjee et al. (2023) (see Table 1 for a detailed comparison of stepsizes).

Theorem 2 can also be obtained under a weaker interpolation assumption: $\|\nabla f_i(x^*)\| = 0$ for all $i \in \mathcal{S}$, $x \in \mathbb{R}^n$, and $x^* \in \mathcal{X}^*$, which is also widely investigated in machine learning (see Ma et al. (2018); Vaswani et al. (2019a;b); Gower et al. (2021); Meng et al. (2020)). Compared with Theorem 1, Theorem 2 does not require client i to know information about the global loss function to determine its stepsize. In addition, it allows stepsize that is $\max_{1\leq j\leq N}\{\frac{L_j\tau}{L_i}, \frac{L(2\tau+\eta(\tau-1))^2+4\eta L\tau(\tau-1)}{8L_i\tau}\}$ times larger than that in (3). In fact, our numerical experimental results in Appendix 6.1.2 show that allowing clients to use local stepsizes achieves a faster convergence than the case with a global stepsize. This is intuitive in that utilizing local Lipschitz constants allows the gradient descent steps to exploit the local geometry of loss functions, and hence, enables faster convergence. It is worth noting that although such a phenomenon has been reported in Mukherjee et al. (2023) for one specific example of quadratic functions, we are the first to theoretically establish that local stepsizes can be exploited to achieve faster convergence for a general class of loss functions in federated learning.

Remark 1. *It is worth noting that the $o(1/t)$ convergence rate established in Theorem 1 does not contradict the result in Glasgow et al. (2022), which proves that FedAvg cannot achieve a rate faster than $O(1/t)$ for general convex objectives. The key distinction lies in the fact that Theorem 1 relies on the additional Strong Growth Condition (see Assumption 3). The strong growth condition is a standard assumption of federated learning in the over-parameterized setting (Vaswani et al., 2019a; Qin et al., 2022b;a), i.e., when the model can interpolate the data completely, such that the loss at every data point is minimized simultaneously (usually means zero empirical loss). The strong growth condition posits that the squared norm of any client's local gradient is bounded by a constant multiple of the squared norm of the global gradient, i.e., $\|\nabla f_i(x)\| \leq \eta\|\nabla f(x)\|$. This condition implies that client-level gradients align with the global direction, ensuring that local gradient descent updates do not diverge excessively. That is the reason why we can prove the improved convergence rate $o(1/t)$. In the next section, we introduce a new algorithm that achieves $o(1/t)$ convergence for general convex objectives under non-IID data, without requiring any additional restrictive conditions.*

Remark 2. *The stepsize in Theorem 2 is larger than that in Theorem 1, allowing greater flexibility in practice. However, the convergence rate established in Theorem 1 is $o(1/t)$, which is sharper than the $O(1/t)$ rate in Theorem 2. This reflects a trade-off: the relaxed stepsize condition in Theorem 2 leads to a more conservative theoretical guarantee. Nonetheless, using local stepsizes tailored to individual smoothness constants—as in Theorem 2—yields significantly better empirical convergence performance, as demonstrated in Sections 6.1.1 and 6.1.2.*

4 Convergence under non-IID Data

4.1 Algorithm Description

Under non-IID data, it has been known that except the trivial case where the number of local iterations is one ($\tau = 1$), Algorithm 1 will be subject to errors (Mukherjee et al., 2023; Orvieto et al., 2022; Wang et al., 2020; Karimireddy et al., 2020). Inspired by gradient-tracking-based distributed optimization algorithms (Pu & Nedić, 2021; Nedić et al., 2017), we propose Algorithm 2 to address this issue and ensure accurate convergence under non-IID data.

Algorithm 2

```
Input: Initial values $\bar{x}(1)$, $\nabla f(\bar{x}(1))$, local training period $\tau$, and stepsize $\alpha$;
for $t = 1$ to $T$ do
  for $i = 1$ to $N$ do
    Each client $i$ sets
        $x_{i,0}(t) = \bar{x}(t) \quad \text{and} \quad y_{i,0}(t) = \nabla f(\bar{x}(t)).$    (5)
    for $k = 0$ to $\tau - 1$ do
      Client $i$ does local updating
        $x_{i,k+1}(t) = x_{i,k}(t) - \alpha y_{i,k}(t),$    (6)
        $y_{i,k+1}(t) = y_{i,k}(t) + \nabla f_i(x_{i,k+1}(t)) - \nabla f_i(x_{i,k}(t)).$    (7)
    end for
  end for
  The central server calculates and transmits $\bar{x}(t+1) = \frac{1}{N}\sum_{i=1}^{N} x_{i,\tau}(t)$ to each client. Each client $i$ then transmits $\nabla f_i(\bar{x}(t+1))$ to the central server and receives $\nabla f(\bar{x}(t+1)) = \frac{1}{N}\sum_{i=1}^{N} \nabla f_i(\bar{x}(t+1))$ from the central server.
end for
```

Unlike Algorithm 1 which exchanges only the model parameters $x_{i,k+1}(t)$ between clients and the server, Algorithm 2 requires exchanging an additional variable for the gradient. More specifically, in Algorithm 2, each client uses the global gradient information $\nabla f(\bar{x}(t))$ to initialize its local variable $y_{i,k}(t)$ after each communication round (see (5)). This variable $y_{i,k+1}(t)$, which serves as an estimate of the global gradient, is then used to update the model parameter $x_{i,k+1}(t)$ (see (7)). This is key to eliminating the drift caused by non-IID data.

Our Algorithm 2 provides a general framework for federated learning with drift correction, encompassing a wide range of existing algorithms as special cases. Specifically, by substituting equation (5) into equation (7) and applying mathematical induction, the auxiliary variable $y_{i,k}(t)$ can be expressed as

$$y_{i,k}(t) = \nabla f_i(x_{i,k}(t)) - \nabla f_i(\bar{x}(t)) + \nabla f(\bar{x}(t))$$

for $k = 0, 1, \ldots, \tau$. Substituting this expression into the update rule (6) leads to

$$x_{i,k+1}(t) = x_{i,k}(t) - \alpha\Big(\nabla f_i(x_{i,k}(t)) - \nabla f_i(\bar{x}(t)) + \nabla f(\bar{x}(t))\Big),$$

which recovers the specific update mechanisms used in FedLin (Mitra et al., 2021b) and FedTrack (Mitra et al., 2021a). In addition, as $x_{i,k}(t)$ converges to x^*, it follows from equation (7) that $y_{i,k}(x^*) = \nabla f_i(x^*)$, a key idea leveraged in the "drift-correction" federated learning algorithms Scaffnew (Mishchenko et al., 2022) and SCAFFOLD (Karimireddy et al., 2020). This demonstrates that Algorithm 2 not only generalizes but also unifies prior drift-corrected federated learning methods within a broader and more flexible structure.

Next, we prove that the new framework allows us to obtain $o(1/t)$ convergence in general convex federated learning, which is only established in the literature for special classes of convex functions with restrictions on data heterogeneity (see, e.g., under the bounded gradient difference condition in Jiang et al. (2024) and under the bounded Hessian difference condition in Kovalev et al. (2022)). For the general convex case without any restrictions, existing federated learning algorithms—even those incorporating momentum—only achieve an $O(1/t)$ convergence rate. In addition, the new framework allows using significantly larger step sizes compared to existing drift-corrected federated learning algorithms, as detailed in Section 4.2.

4.2 Convergence Analysis

Theorem 3. *For Algorithm 2, under Assumptions 1 and 2, if the stepsize α of client $i \in \mathcal{S}$ satisfies*

$$0 < \alpha < \min_{1 \le j \le N} \Big\{ \frac{1}{L_j}, \frac{2}{5L\tau - L} \Big\}, \tag{8}$$

where $L = \frac{1}{N}\sum_{i=1}^{N} L_i$, then $f(\bar{x}(t))$ converges to $f(x^*)$ with the convergence rate $o(1/t)$, i.e.,

$$\lim_{t\to\infty} t\{f(\bar{x}(t)) - f(x^*)\} = 0.$$

Proof. See Appendix E. □

In Theorem 3, we establish an $o(1/t)$ convergence rate for federated learning with general convex functions under non-IID data. A key step in this analysis, as shown in Lemma 1, is proving that the sequence $\{f(\bar{x}(t))\}$ is monotonically decreasing, i.e.,

$$f(\bar{x}(t+1)) \le f(\bar{x}(t)).$$

We emphasize that proving this monotonicity under general smooth and convex conditions is highly non-trivial. Our proof of this property, presented in Lemma 2, constitutes a significant technical contribution of this work.

Lemma 2. *If the stepsize α of Algorithm 2 satisfies $0 < \alpha < \frac{2}{5L\tau - L}$, there exists a constant $\gamma > 0$ such that*

$$\gamma\alpha^2 \|\nabla f(\bar{x}(t))\|^2 \le f(\bar{x}(t)) - f(\bar{x}(t+1)).$$

Moreover, the sequence $f(\bar{x}(t))$ is monotonically decreasing.

The proof of Lemma 2 is provided in Appendix F. We provide a brief proof sketch here. Because $\bar{x}(t+1) = \bar{x}(t) - \frac{\alpha}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1} \nabla f_i(x_{i,k}(t))$ (see (37)), the proof begins by applying the L-smoothness property to bound $f(\bar{x}(t+1))$. The deviation $\nabla f_i(x_{i,k}(t)) - \nabla f_i(\bar{x}(t))$ is then bounded using smoothness, together with a carefully derived inequality $\|\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\| \le (2\tau - 1)\|\nabla f(\bar{x}(t))\|$ (see (47)), yielding the refined descent inequality

$$\Big(\alpha\tau - \frac{5L\tau^2 - L\tau}{2}\alpha^2\Big)\big\|\nabla f(\bar{x}(t))\big\|^2 \le f(\bar{x}(t)) - f(\bar{x}(t+1))$$

(see (56)). Finally, under the step-size condition $0 < \alpha < \frac{2}{5L\tau - L}$, this inequality guarantees a strict decrease in $f(\bar{x}(t))$, establishing its monotonic convergence.

Notably, other federated learning algorithms in Mitra et al. (2021a;b), which also follow a gradient-tracking-based framework, only establish an $O(1/t)$ convergence rate under general convex functions in their analyses. In contrast, our work develops a more refined analysis technique—specifically, the nontrivial proof of the monotonically decreasing property, i.e., $f(\bar{x}(t+1)) \le f(\bar{x}(t))$ (see Lemma 2)—which enables us to establish an $o(1/t)$ convergence rate in Theorem 3. Importantly, this analysis framework is not limited to our algorithm and can also be applied to other gradient-tracking-based methods to improve their theoretical guarantees from $O(1/t)$ to $o(1/t)$ under general convex settings. This general methodology, therefore, represents a significant contribution of our work.

Remark 3. *It is worth noting that Lee & Wright (2019) proves that centralized first-order gradient descent algorithms can achieve the convergence rate $o(1/t)$, which is better than the existing well-known convergence rate $O(1/t)$. The key to this improvement lies in establishing the monotonic descent property. This paper* *extends the monotonic descent property from centralized optimization in Lee & Wright (2019) to federated learning, which is highly nontrivial due to the presence of multiple heterogeneous local loss functions arising from non-IID data distributions. To the best of our knowledge, this work provides the first rigorous proof of such monotonicity in convex federated learning and, as a result, achieves the improved $o(1/t)$ convergence rate.*

Remark 4. *To illustrate why Algorithm 2 is well suited for non-IID cases and why its sharing of the additional gradient variable can overcome the need for Assumption 3, we define: $X(k) = [x_1^T(k), x_2^T(k), \cdots, x_N^T(k)]^T$, $\nabla f(X(k)) = [\nabla f_1^T(x_1(k)), \nabla f_2^T(x_2(k)), \cdots, \nabla f_N^T(x_N(k))]^T$, $W(k+1) = \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T$ for $k+1 = \tau t$, and $W(k+1) = \mathbf{I}_N$ for $k+1 \neq \tau t$. It can be verified that Algorithm 2 can be expressed equivalently as:*

$$X(k+1) = W(k+1)\{X(k) - \alpha Y(k)\},$$
$$Y(k+1) = W(k+1)\{Y(k) + \nabla f(X(k+1)) - \nabla f(X(k))\}.$$

To link the notation, the variable $X(k\tau) = [x_1^T(k\tau), x_2^T(k\tau), \cdots, x_N^T(k\tau)]^T$ in the form above corresponds precisely to the variable $\bar{x}(t)$ in the original description of Algorithm 2, satisfying $x_i(k\tau) = \bar{x}(t)$ for any $1 \le i \le N$. We next show that $Y(t) = [y_1^T(t), y_2^T(t), \cdots, y_N^T(t)]^T$ is an estimate of the global gradient. Based on the definition of $W(t)$, taking the network average gives

$$\overline{Y}(t+1) = \overline{Y}(t) + \overline{\nabla f}(X(t+1)) - \overline{\nabla f}(X(t)),$$

where $\overline{\nabla f}(X(t)) = \frac{1}{N}\sum_{i=1}^N \nabla f_i(x_i(t))$ and $\overline{Y}(t) = \frac{1}{N}\sum_{i=1}^N y_i(t)$. Based on the initialization rule (5), i.e., $Y(0) = \nabla f(X(0))$, we can obtain $\overline{Y}(t) = \overline{\nabla f}(X(t))$. Therefore, the average of $Y(t)$ coincides with the true global gradient at every iteration, and the consensus force of $W(t)$ ensures that all local $y_i(t)$ converge to this average, and hence that all local $y_i(t)$ converge to the global gradient. Therefore, at each iteration, the local model parameters (i.e., $X(t)$) are updated based on the global gradient estimate rather than local gradients. This mechanism ensures that the local model parameters converge to the optimal solution of the global loss function, rather than to the optima of the individual local loss functions resulting from non-IID data, effectively overcoming the need for Assumption 3.

Remark 5. *We acknowledge that, compared with the classic federated learning algorithm FedAvg, our Algorithm 2 requires additional communication. However, we argue that this overhead is a necessary cost for effectively addressing non-IID data. To the best of our knowledge, all existing federated learning algorithms that guarantee accurate convergence under non-IID data require sharing additional information beyond the gradients used in FedAvg, in order to correct the drift induced by data heterogeneity. In fact, Karimireddy et al. (2020) provides a counterexample demonstrating that FedAvg (Algorithm 1) cannot ensure exact convergence under non-IID data distributions. Compared with FedAvg, Algorithm 2 requires exchanging an additional variable alongside the gradient. This additional communication is essential for eliminating the drift caused by non-IID data (heterogeneity in the clients' loss functions). Therefore, this increased communication is a necessary price to pay for ensuring exact convergence of federated learning under non-IID client data distributions.*

4.3 Comparison with Existing Results

From Theorem 3, Algorithm 2 allows a much larger stepsize and a better convergence rate compared with existing works. Specifically, the stepsize in Karimireddy et al. (2020) is required to satisfy $0 < \alpha \le \min_{1\le i\le N}\{\frac{1}{81L_i\tau}\}$. In contrast, the stepsize upper bound in Theorem 3 is given by $\min_{1\le i\le N}\{\frac{1}{L_i}, \frac{2}{5L\tau - L}\}$. It can be verified that our permissible stepsize is at least $\frac{162}{5}$ times larger than that in Karimireddy et al. (2020). Similarly, Mitra et al. (2021b) requires the stepsize to satisfy $0 < \alpha \le \min_{1\le i\le N}\{\frac{1}{10L_i\tau}\}$. In contrast, our Theorem 3 permits a stepsize that is at least $\max_{1\le i\le N}\{\frac{20\tau L_i}{5L\tau - L}\} \ge 4$ times larger than that in Mitra et al. (2021b). Table 1 provides a detailed comparison between our proposed stepsize and convergence rate with existing works.

Table 1: Comparison of the proposed stepsizes and obtained convergence rates for Algorithm 1 and Algorithm 2 with existing results. In this table, we represent the total communication round as t, the local training period as τ, and assume that the local loss function $f_i(x)$ satisfies L-smooth property.

Assumption	Algorithm	Stepsize	Convergence Rate	Gradient Setting[1]
IID	Algorithm 1	$1/L$	$O(1/t)$	EG
		$1/L$	$O(1/t)$	SG
	Qin et al. (2022b)	$1/(2L)$	$O(1/t)$	SG
	Khaled et al. (2020)	$1/(4L)$	$O(1/t)$	SG
non-IID	Algorithm 2	$2/(5L\tau - L)$	$o(1/t)$	EG
		$1/(12\tau L)$	$O(1/t)$	SG
	Mitra et al. (2021a)	$1/(18\tau L)$	$O(1/t)$	EG
	Mitra et al. (2021b)	$1/(10\tau L)$	$O(1/t)$	EG
	Allouah et al. (2024)	$1/(16\tau L)$	$O(1/t)$	SG
	Karimireddy et al. (2020)	$1/(81\tau L)$	$O(1/t)$	SG
	Reisizadeh et al. (2020)	$O(1/\sqrt{t})$	$O(1/\sqrt{t})$	SG
	Beikmohammadi et al. (2025)			
	Haddadpour & Mahdavi (2019)			
	Qu et al. (2021); Yu et al. (2019)			
	Cheng et al. (2024); Li & Li (2023)			
	Yang et al. (2021); Zhu et al. (2021)			
	Wang et al. (2020); Yan et al. (2025)			
	Xiang et al. (2024); Huang et al. (2023)			
	Kim et al. (2023)	Adaptive	$O(1/\sqrt{t})$	SG

[1] EG denotes the exact gradient setting; SG denotes the stochastic gradient setting.

5 Convergence Analysis under Stochastic Gradients

We extend our analysis to the more practical setting of stochastic gradients (the mini-batch setting). In this case, the local loss function $f_i(x)$ is determined by

$$f_i(x) = \mathbb{E}_{\xi_i \sim D_i}[f_i(x, \xi_i)], \tag{9}$$

where ξ_i denotes a stochastic data sample drawn from the local distribution D_i of client i. As a result, client i can only access a stochastic estimate $\nabla f_i(x, \xi_i)$ of the true gradient $\nabla f_i(x)$ for any $x \in \mathbb{R}^n$. We use the following standard assumption regarding the stochastic gradient (Karimireddy et al. (2020); Mukherjee et al. (2023); Jhunjhunwala et al. (2023)):

Assumption 4. *The stochastic gradient $\nabla f_i(x, \xi_i)$ is an unbiased estimate of the accurate gradient $\nabla f_i(x)$, with its variance bounded by σ^2. Specifically, we have*

$$\mathbb{E}_{\xi_i \sim D_i}[\nabla f_i(x, \xi_i)] = \nabla f_i(x), \quad \textit{and} \quad \mathbb{E}_{\xi_i \sim D_i}[\|\nabla f_i(x, \xi_i) - \nabla f_i(x)\|^2] \le \sigma^2,$$

for any $x \in \mathbb{R}^n$ and $i \in \mathcal{S}$.

Theorem 4 establishes the convergence of Algorithm 1 in the stochastic and IID case:

Theorem 4 (Stochastic and IID Case)**.** *Under Assumptions 1, 2, 3, and 4, if the stepsize α_i in Algorithm 1 satisfies $0 < \alpha_i < \frac{1}{L_i}$ for any $i \in \mathcal{S}$, we have*

$$\mathbb{E}\Big[f\Big(\frac{1}{T}\sum_{t=1}^{T}\bar{x}(t)\Big)\Big] - f(x^*) \le \frac{\|\bar{x}(1) - x^*\|^2}{\min_{1\le i\le N}\{2\alpha_i - 2L_i\alpha_i^2\}T} + \frac{2\tau\sum_{i=1}^{N}\alpha_i^2}{\min_{1\le i\le N}\{2\alpha_i - 2L_i\alpha_i^2\}N}\sigma^2.$$

Proof. See Appendix G. □

Theorem 5 establishes the convergence of Algorithm 2 in this stochastic and non-IID setting:

Theorem 5 (Stochastic and Non-IID Case). *Under Assumptions 1, 2, and 4, if the stepsize α of Algorithm 2 satisfies $0 < \alpha \leq \min_{1 \leq j \leq N}\{\frac{1}{L_j}, \frac{1}{12\tau L}\}$, we have*

$$\mathbb{E}\Big[f\Big(\frac{1}{T}\sum_{t=1}^{T}\bar{x}(t)\Big)\Big] - f(x^*) \leq \frac{\|x(1) - x^*\|^2}{\alpha\tau T} + 34\tau\alpha\sigma^2.$$

Proof. See Appendix H. □

6 Experiments

6.1 Evaluation using generated data

6.1.1 Comparison under IID distribution

We use the following regression problem to compare the performance of Algorithm 1 and Algorithm 2 under the proposed stepsizes with existing counterparts[1]:

$$\min_{x\in\mathbb{R}^n} f(x) = \min_{x\in\mathbb{R}^n} \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}\|A_i x - b_i\|^2, \tag{10}$$

where $A_i \in \mathbb{R}^{500\times100}$, $b_i \in \mathbb{R}^{500}$ and $x \in \mathbb{R}^{100}$ for each client $i \in \mathcal{S} = \{1, 2, \cdots, 20\}$. $[A_i]_{jk}$ are generated from $[0, 1]$ randomly for $j \in \{1, 2, \cdots, 500\}$, $k \in \{1, 2, \cdots, 100\}$, and $i \in \mathcal{S}$, and we also set $[A_1]_{j,1} = [A_1]_{j,2}$ for $j \in \{1, 2, \cdots, 500\}$ to obtain a convex but not strongly convex loss function $f_1(x)$. We set $b_i = A_i x_0$ for all $i \in \mathcal{S}$ with $x_0 = 10 \times \mathbf{1}_n$ rather than generating b_i randomly. In this setting, $f_i(x) = \frac{1}{2}\|A_i(x - x_0)\|^2$ and, hence, there exists a constant $\eta = \max_{1\leq j\leq N}\{\frac{\|A_j^T A_j\|_2}{\|(\frac{1}{N}\sum_{i=1}^{N} A_i)^T(\frac{1}{N}\sum_{i=1}^{N} A_i)\|_2}\}$ such that $\|\nabla f_i(x)\| \leq \eta\|\nabla f(x)\|$ holds for all $i \in \mathcal{S}$.

We compare Algorithm 1 and Algorithm 2 under the proposed stepsize strategy with existing counterparts including Qin et al. (2022b); Mukherjee et al. (2023); Mitra et al. (2021b); Khaled et al. (2020). In the evaluation, we use the error

$$e(t) = f(\bar{x}(t)) - f(x^*)$$

to measure the learning accuracy. Moreover, we implement all algorithms using accurate gradients to ensure a fair comparison of them. The corresponding convergence performances with different local training periods $\tau = 2, 3, 4, 5, 6$ are presented in Figure 1.

In Figure 1, the legends 'Algorithm 1 with Universal Stepsizes' and 'Algorithm 1 with Local Stepsizes' denote Algorithm 1 with stepsizes (3) and (4), respectively. Specifically, in the universal stepsize case, we set the universal stepsize α for all clients as $\alpha = \min_{1\leq i\leq N}\{\frac{1}{L_i\tau}, \frac{8\tau}{L(2\tau+\eta(\tau-1))^2+4\eta L\tau(\tau-1)}\}$ according to (3), where $L = \frac{1}{N}\sum_{i=1}^{N} L_i$ is the global Lipschitz constant. In the local stepsize case, we set the stepsize of client i as $\alpha_i = \frac{1}{L_i}$ based on individual Lipschitz constants L_i. From Figure 1, we know that the convergence of Algorithm 1 with the stepsize prescribed in (4) is much faster than other cases, including the case with the universal stepsize (3). Additional experiments with non-IID data are presented in Appendix A.1.

6.1.2 Local Stepsize Strategy Outperforms Universal Stepsize Strategy for Algorithm 1 under $\tau = 1$

We show that better convergence performance[2] of Algorithm 1 can be achieved with local stepsizes $0 < \alpha_i < \frac{1}{L_i}$ than a universal stepsize $0 < \alpha \leq \frac{1}{L}$, where $L = \frac{1}{N}\sum_{i=1}^{N} L_i$. For ease of comparison, we continue to consider the regression problem (10) described in Section 6.1.1, specifying

$$[A_i]_{jk} = i^\rho[B_i]_{jk} \quad \text{and} \quad b_i = A_i x_0$$

[1] Code available at https://anonymous.4open.science/r/o1_t-F814/README.md

[2] Code available at https://anonymous.4open.science/r/o1_t-F814/README.md

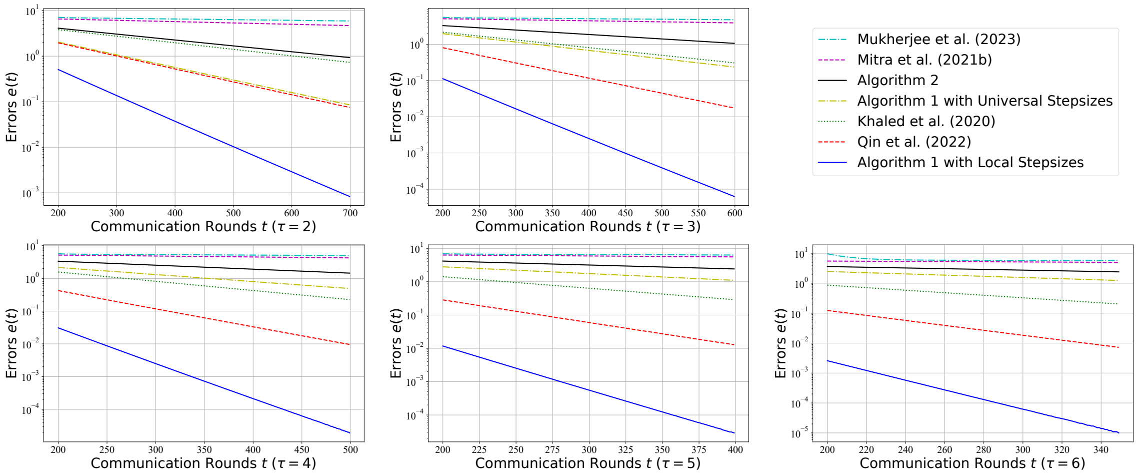


Figure 1: Comparisons of the performance of Algorithm 1 and Algorithm 2 under the proposed stepsize with Qin et al. (2022b); Mukherjee et al. (2023); Mitra et al. (2021b); Khaled et al. (2020) under different local training periods τ.

for $j \in \{1, 2, \cdots, 500\}$, $k \in \{1, 2, \cdots, 100\}$, and $i \in \mathcal{S}$, where $[B_i]_{jk}$ is generated from $[0, 1]$ randomly, ρ measures the heterogeneity in loss functions, and $x_0 = 10 \times \mathbf{1}_n$. It can be seen that a larger parameter ρ leads to more heterogeneity in the local loss functions. Moreover, one can verify that the loss function $f_i(x)$ of client $i \in \mathcal{S}$ satisfies L_i-smooth property with $L_i = i^{\rho} \|B_i^T B_i\|_2$. Then, under $\tau = 1$, we present in Figure 2 the convergence of Algorithm 1 under the local stepsize strategy where $\alpha_i = \frac{1}{L_i}$ of client $i \in \mathcal{S}$ and the universal stepsize strategy where $\alpha = \frac{1}{L}$ for all clients, where L_i is the individual Lipschitz constant of client $i \in \mathcal{S}$ and $L = \frac{1}{N} \sum_{i=1}^{N} L_i$ is the global Lipschitz constant.

In Figure 2, to compare the convergence between the local and the universal stepsize strategies, we plot the learning errors $f(\bar{x}_l(t)) - f(x^*)$ and $f(\bar{x}_g(t)) - f(x^*)$ under different heterogeneity parameters $\rho \in \{1, 1.5, 2, 2.5, 3\}$, where $\bar{x}_l(t)$ and $\bar{x}_g(t)$ are generated under the local and the universal stepsize strategies, respectively. From Figure 2, it is clear that the local stepsize designed based on local Lipschitz constants obtains faster convergence than the case with the universal stepsize designed based on the global Lipschitz constant. Moreover, to quantify the improvement in convergence speed, in Figure 2, we also plot the learning error ratio

$$r(t) = \frac{f(\bar{x}_l(t)) - f(x^*)}{f(\bar{x}_g(t)) - f(x^*)}$$

under different heterogeneity parameters $\rho \in \{1, 1.5, 2, 2.5, 3\}$, respectively. A smaller $r(t)$ ($r(t) < 1$) means more advantage of the convergence speed of the local stepsize strategy over the universal stepsize strategy. Figure 2 shows that a smaller $r(t)$ is obtained under a larger heterogeneity parameter ρ. Thus, it can be concluded that the local stepsize strategy of Algorithm 1 can achieve faster convergence than the global stepsize strategy, especially for large heterogeneity cases.

6.2 Evaluation using CIFAR-10 and CIFAR-100 under non-IID distribution

We also evaluate our algorithms by training a CNN on 10 clients using the benchmark datasets CIFAR-10 and CIFAR-100, respectively[3]. The CNN architecture consists of three convolutional layers with 32, 64, and 128 filters, respectively, each followed by a max-pooling layer. After the final convolutional and pooling layers, the network includes a fully connected layer with 256 units and ReLU activation, a dropout layer with a rate of 0.25 for regularization, and a final dense output layer with 10 units that produces the class

[3] Code available at https://anonymous.4open.science/r/o1_t-F814/README.md

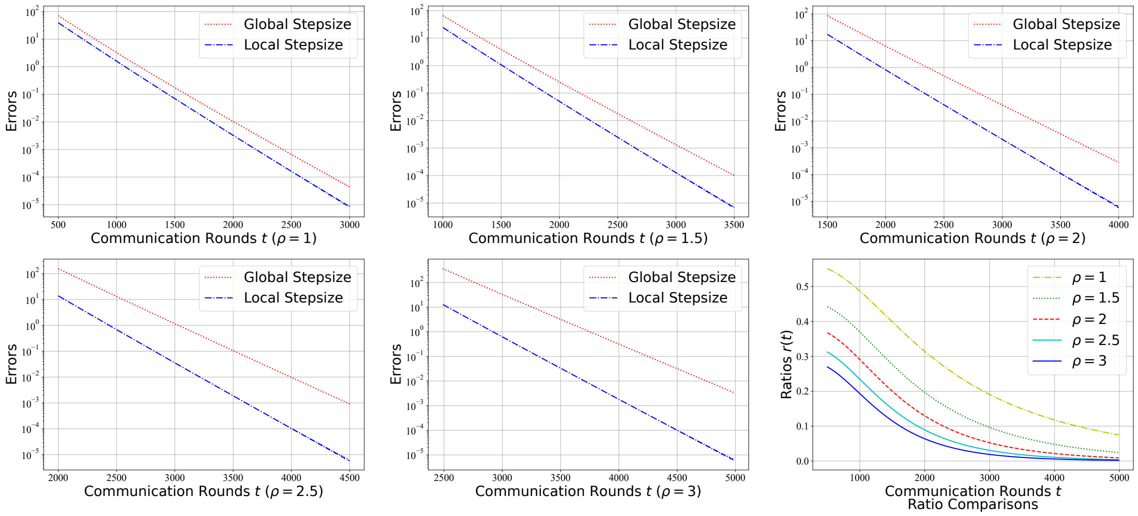


Figure 2: Comparisons of Algorithm 1 with the local stepsize strategy and a universal stepsize strategy under different heterogeneity parameters ρ.

logits. In our experiments, we compare the proposed algorithm against existing federated learning methods specifically designed to address client drift, including SCAFFOLD (Karimireddy et al., 2020), FedLin (Mitra et al., 2021b), and Scaffnew (Mishchenko et al., 2022). Following Hsu et al. (2019) and Kim et al. (2023), we generate heterogeneous data distributions across the 10 agents using a Dirichlet distribution, with the heterogeneity parameter β set to 0.1, 1, and 10, respectively. A higher value of β yields a nearly uniform distribution of data across classes for each client, resulting in approximately IID local datasets. In contrast, a lower β leads to highly skewed distributions, where clients tend to specialize in only a few classes.

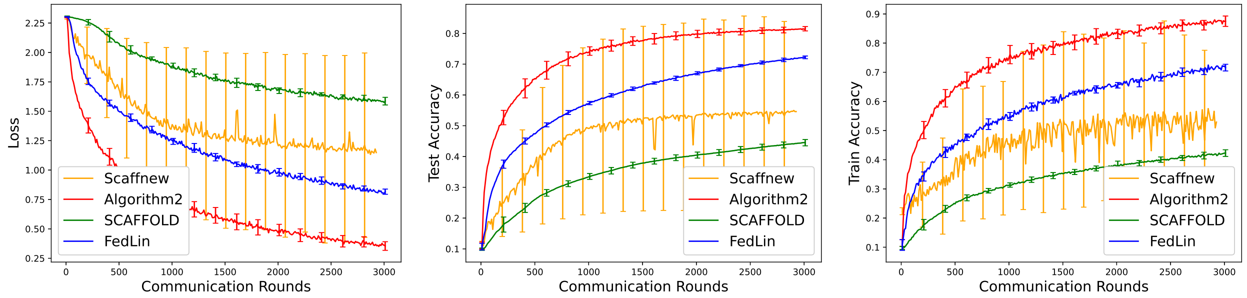
Figures 3 and 4, Figures 5 and 6, and Figures 7 and 8 present results for β set to 1, 0.1 and 10, respectively, corresponding to moderate, high, and low heterogeneity in the data distributions. For all results shown in Figures 3, 5, 7 (CIFAR-10) and Figures 4, 6, 8 (CIFAR-100), the stepsizes for Algorithm 2, SCAFFOLD, FedLin, and Scaffnew are selected according to the guidelines from Theorem 3, Karimireddy et al. (2020), Mitra et al. (2021b), and Mishchenko et al. (2022), respectively, using an estimated smoothness parameter of $L = 2$. For Algorithm 2, SCAFFOLD, and FedLin, the local training period is set to $\tau = 10$. For Scaffnew, the communication probability is set to $\frac{1}{11}$ to ensure that the total number of communicated messages remains consistent across methods. A summary of the experimental setup is given in Table 2. As shown in the figures, our algorithm achieves faster convergence and higher accuracy on both the CIFAR-10 dataset and the CIFAR-100 dataset. Note that the large variance of Scaffnew arises from the additional randomness introduced by its communication mechanism.

Table 2: Experimental Setup in Figures 3-8

	Figure 3	Figure 4	Figure 5	Figure 6	Figure 7	Figure 8
Dataset	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Heterogeneity	$\beta = 1$	$\beta = 1$	$\beta = 0.1$	$\beta = 0.1$	$\beta = 10$	$\beta = 10$
Local Training Period[1]	$\tau = 10$	$\tau = 10$	$\tau = 10$	$\tau = 10$	$\tau = 10$	$\tau = 10$
Number of Agents	10	10	10	10	10	10
Optimizer[2]	See label	See label	See label	See label	See label	See label

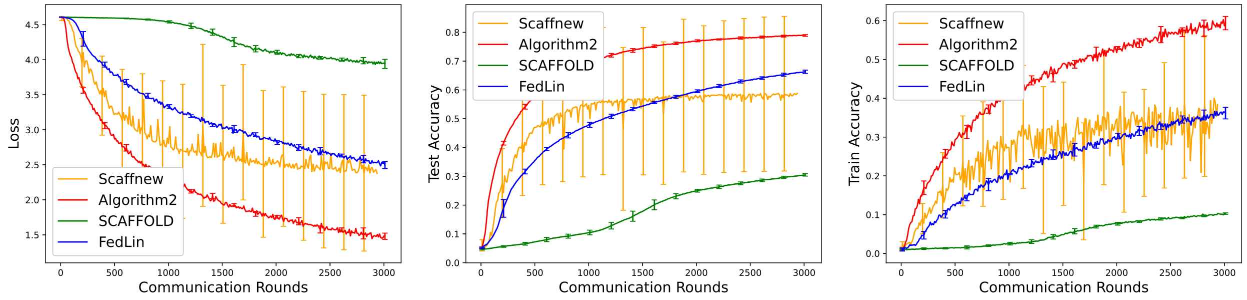
[1] For Scaffnew, the communication probability is set to $\frac{1}{11}$ to ensure that the total number of communicated messages remains consistent across methods.

[2] The stepsizes for Algorithm 2, SCAFFOLD, FedLin, and Scaffnew are selected according to the guidelines from Theorem 3, Karimireddy et al. (2020), Mitra et al. (2021b), and Mishchenko et al. (2022), respectively, using an estimated smoothness parameter of $L = 2$.



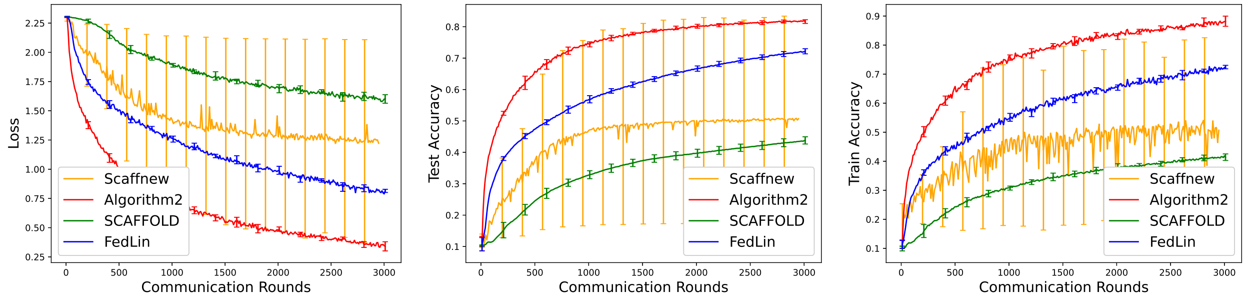
(a) (b) (c)

Figure 3: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—SCAFFOLD, FedLin, and Scaffnew—on the CIFAR-10 dataset. The Dirichlet distribution parameter was set to $\beta = 1$. Each curve represents the average of five independent runs. The test accuracy in Figure 3(b) is top-5 accuracy.



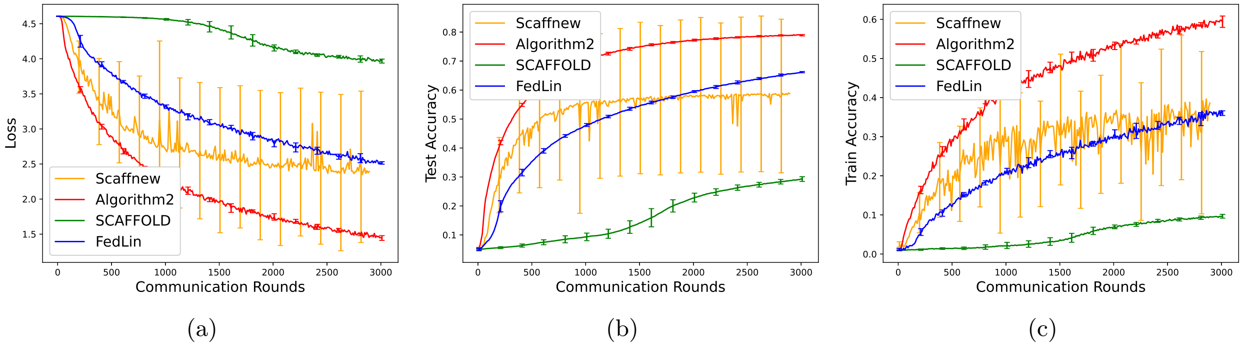
(a) (b) (c)

Figure 4: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—SCAFFOLD, FedLin, and Scaffnew—on the CIFAR-100 dataset. the Dirichlet distribution parameter was set to $\beta = 1$. Each curve represents the average of five independent runs. The test accuracy in Figure 4(b) is top-5 accuracy.



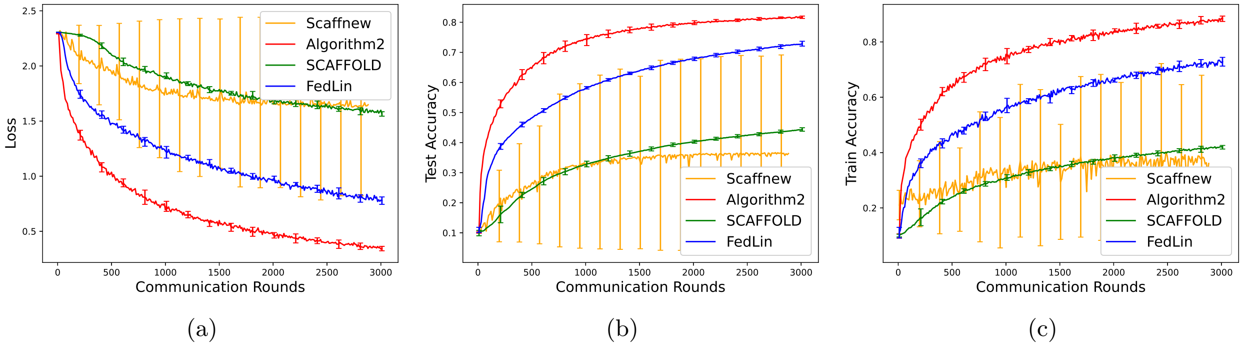
(a) (b) (c)

Figure 5: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—SCAFFOLD, FedLin, and Scaffnew—on the CIFAR-10 dataset. The Dirichlet distribution parameter was set to $\beta = 0.1$. Each curve represents the average of five independent runs. The test accuracy in Figure 5(b) is top-5 accuracy.



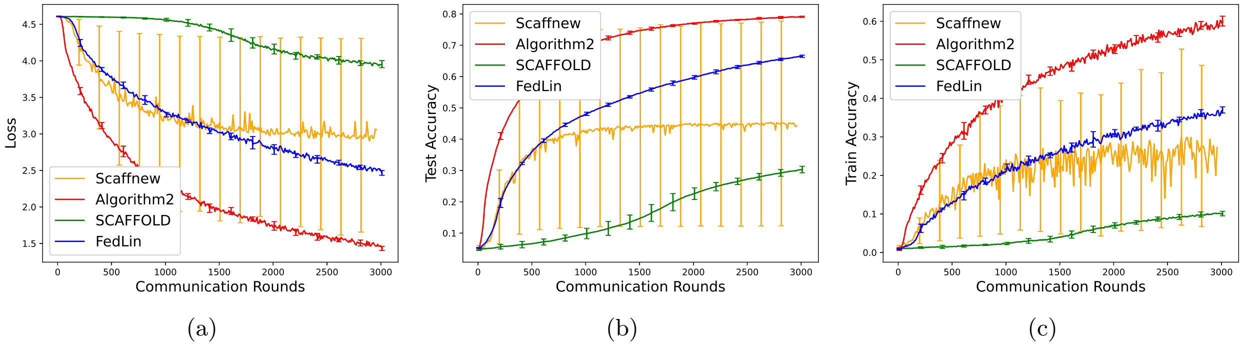
(a) (b) (c)

Figure 6: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—SCAFFOLD, FedLin, and Scaffnew—on the CIFAR-100 dataset. The Dirichlet distribution parameter was set to $\beta = 0.1$. Each curve represents the average of five independent runs. The test accuracy in Figure 6(b) is top-5 accuracy.



(a) (b) (c)

Figure 7: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—SCAFFOLD, FedLin, and Scaffnew—on the CIFAR-10 dataset. the Dirichlet distribution parameter was set to $\beta = 10$. Each curve represents the average of five independent runs. The test accuracy in Figure 7(b) is top-5 accuracy.



(a) (b) (c)

Figure 8: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—SCAFFOLD, FedLin, and Scaffnew—on the CIFAR-100 dataset. The Dirichlet distribution parameter was set to $\beta = 10$. Each curve represents the average of five independent runs. The test accuracy in Figure 8(b) is top-5 accuracy.

7 Conclusion

Enhancing convergence accuracy and speed is key for federated learning. We prove that much larger stepsizes can be used in FedAvg, and hence, much faster convergence can be achieved. In fact, we theoretically show that the proposed stepsize strategy can guarantee $o(1/t)$ convergence to an exact optimal solution for general convex loss functions, under both IID data distribution and non-IID data distribution among local clients. This is significant since existing federated learning results can only theoretically establish $O(1/t)$ convergence under general convex loss functions when no additional restrictions are made, even after incorporating momentum. Moreover, in the IID data distribution setting, we theoretically establish convergence when clients set stepsizes individually using local Lipschitz parameters, and show that such a local stepsize strategy enables exploiting local geometry to expedite convergence. To our knowledge, this is the first time that local stepsizes designed using local Lipschitz parameters is systemtically shown to outperform a universal stepsize designed using the global Lipschitz parameter. Moreover, we propose a general gradient-tracking-based framework that unifies and extends many existing drift-corrected federated learning algorithms. By establishing a key monotonic descent property, our framework broadens the theoretical understanding of gradient tracking and enables an improved $o(1/t)$ convergence rate under non-IID data distributions. This represents a significant advancement, as existing results establish $o(1/t)$ convergence for convex federated learning only under additional restrictions on heterogeneity.

Acknowledgments

The work was supported in part by the National Science Foundation under Grants CCF-2106293, CCF-2215088, CNS-2219487, CCF-2334449, and CNS-2422312.

References

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas Navarro, Matthew Mattina, Paul N Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. *arXiv preprint arXiv:2111.04263*, 2021.

Naman Agarwal, Ananda Theertha Suresh, Felix Xinnan X Yu, Sanjiv Kumar, and Brendan McMahan. cpSGD: Communication-efficient and differentially-private distributed SGD. In *Advances in Neural Information Processing Systems*, volume 31, pp. 7564–7575, 2018.

Youssef Allouah, Sadegh Farhadkhani, Rachid Guerraoui, Nirupam Gupta, Rafael Pinot, Geovani Rizk, and Sasha Voitovych. Byzantine-robust federated learning: Impact of client subsampling and local updates. *arXiv preprint arXiv:2402.12780*, 2024.

Rodolfo Stoffel Antunes, Cristiano André da Costa, Arne Küderle, Imrana Abdullahi Yari, and Björn Eskofier. Federated learning for healthcare: Systematic review and architecture proposal. *ACM Transactions on Intelligent Systems and Technology*, 13(4):1–23, 2022.

Ali Beikmohammadi, Sarit Khirirat, and Sindri Magnússon. On the convergence of federated learning algorithms without data similarity. *IEEE Transactions on Big Data*, 11(2):659–668, 2025.

Zachary Charles and Jakub Konečnỳ. On the outsized importance of learning rates in local update methods. *arXiv preprint arXiv:2007.00878*, 2020.

Zachary Charles and Jakub Konečný. Convergence and accuracy trade-offs in federated learning and meta-learning. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130, pp. 2575–2583. PMLR, 2021.

Mingzhe Chen, Nir Shlezinger, H. Vincent Poor, Yonina C. Eldar, and Shuguang Cui. Communication-efficient federated learning. *Proceedings of the National Academy of Sciences*, 118(17):1–8, 2021.

Ziheng Cheng, Xinmeng Huang, Pengfei Wu, and Kun Yuan. Momentum benefits non-iid federated learning simply and provably. *arXiv preprint arXiv:2306.16504*, 2023.

Ziheng Cheng, Xinmeng Huang, Pengfei Wu, and Kun Yuan. Momentum benefits non-iid federated learning simply and provably. In *The Twelfth International Conference on Learning Representations*, 2024.

Lokenath Debnath and Piotr Mikusinski. *Introduction to Hilbert spaces with applications*. Academic press, 2005.

Chenyuan Feng, Zhongyuan Zhao, Yidong Wang, Tony Q. S. Quek, and Mugen Peng. On the design of federated learning in the mobile edge computing systems. *IEEE Transactions on Communications*, 69(9): 5902–5916, 2021.

Bimal Ghimire and Danda B. Rawat. Recent advances on federated learning for cybersecurity and cybersecurity for federated learning for internet of things. *IEEE Internet of Things Journal*, 9(11):8229–8249, 2022.

Margalit R Glasgow, Honglin Yuan, and Tengyu Ma. Sharp bounds for federated averaging (local SGD) and continuous perspective. In *International Conference on Artificial Intelligence and Statistics*, pp. 9050–9090, 2022.

Robert Gower, Othmane Sebbouh, and Nicolas Loizou. SGD for structured nonconvex functions: Learning rates, minibatching and interpolation. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130, pp. 1315–1323, 2021.

Farzin Haddadpour and Mehrdad Mahdavi. On the convergence of local descent methods in federated learning. *arXiv preprint arXiv:1910.14425*, 2019.

Jenny Hamer, Mehryar Mohri, and Ananda Theertha Suresh. FedBoost: A communication-efficient algorithm for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 3973–3983. PMLR, 2020.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

Minhui Huang, Dewei Zhang, and Kaiyi Ji. Achieving linear speedup in non-IID federated bilevel learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 14039–14059, 2023.

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

Ji Chu Jiang, Burak Kantarci, Sema Oktug, and Tolga Soyata. Federated learning in smart city sensing: Challenges and opportunities. *Sensors*, 20(21):6230, 2020.

Xiaowen Jiang, Anton Rodomanov, and Sebastian U Stich. Stabilized proximal-point methods for federated optimization. *Advances in Neural Information Processing Systems*, 37:99735–99772, 2024.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 5132–5143. PMLR, 2020.

Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local SGD on identical and heterogeneous data. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108, pp. 4519–4529. PMLR, 2020.

Junhyung Lyle Kim, Mohammad Taha Toghani, César A Uribe, and Anastasios Kyrillidis. Adaptive federated learning with auto-tuned clients. *arXiv preprint arXiv:2306.11201*, 2023.

Dmitry Kovalev, Aleksandr Beznosikov, Ekaterina Borodich, Alexander Gasnikov, and Gesualdo Scutari. Optimal gradient sliding and its application to optimal distributed optimization under similarity. *Advances in Neural Information Processing Systems*, 35:33494–33507, 2022.

Ching-Pei Lee and Stephen Wright. First-order algorithms converge faster than $o(1/k)$ on convex problems. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 3754–3762. PMLR, 2019.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019.

Xiaoyun Li and Ping Li. Analysis of error feedback in federated non-convex optimization with biased compression: Fast convergence and partial participation. In *International Conference on Machine Learning*, pp. 19638–19688, 2023.

Zengpeng Li, Vishal Sharma, and Saraju P. Mohanty. Preserving data privacy via federated learning: Challenges and solutions. *IEEE Consumer Electronics Magazine*, 9(3):8–16, 2020.

Bill Yuchen Lin, Chaoyang He, Zihang Zeng, Hulin Wang, Yufen Huang, Christophe Dupuy, Rahul Gupta, Mahdi Soltanolkotabi, Xiang Ren, and Salman Avestimehr. Fednlp: Benchmarking federated learning methods for natural language processing tasks. *arXiv preprint arXiv:2104.08815*, 2021.

Ming Liu, Stella Ho, Mengqi Wang, Longxiang Gao, Yuan Jin, and He Zhang. Federated learning meets natural language processing: A survey. *arXiv preprint arXiv:2107.12603*, 2021.

Shengli Liu, Guanding Yu, Rui Yin, Jiantao Yuan, Lei Shen, and Chonghe Liu. Joint model pruning and device selection for communication-efficient federated edge learning. *IEEE Transactions on Communications*, 70(1):231–244, 2022.

Wei Liu, Li Chen, Yunfei Chen, and Wenyi Zhang. Accelerating federated learning via momentum gradient descent. *IEEE Transactions on Parallel and Distributed Systems*, 31(8):1754–1766, 2020.

Bing Luo, Xiang Li, Shiqiang Wang, Jianwei Huang, and Leandros Tassiulas. Cost-effective federated learning design. In *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, pp. 1–10, 2021.

Chuan Ma, Jun Li, Ming Ding, Howard H. Yang, Feng Shu, Tony Q. S. Quek, and H. Vincent Poor. On safeguarding privacy and security in the framework of federated learning. *IEEE Network*, 34(4):242–248, 2020.

Siyuan Ma, Raef Bassily, and Mikhail Belkin. The power of interpolation: Understanding the effectiveness of SGD in modern over-parametrized learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 3325–3334. PMLR, 2018.

Grigory Malinovskiy, Dmitry Kovalev, Elnur Gasanov, Laurent Condat, and Peter Richtarik. From local SGD to local fixed-point methods for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 6692–6701. PMLR, 2020.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54, pp. 1273–1282. PMLR, 2017.

Si Yi Meng, Sharan Vaswani, Issam Hadj Laradji, Mark Schmidt, and Simon Lacoste-Julien. Fast and furious convergence: Stochastic second order methods under interpolation. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108, pp. 1375–1386. PMLR, 2020.

Konstantin Mishchenko, Grigory Malinovsky, Sebastian Stich, and Peter Richtárik. Proxskip: Yes! local gradient steps provably lead to communication acceleration! finally! In *International Conference on Machine Learning*, pp. 15750–15769. PMLR, 2022.

Aritra Mitra, Rayana Jaafar, George J Pappas, and Hamed Hassani. Federated learning with incrementally aggregated gradients. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pp. 775–782, 2021a.

Aritra Mitra, Rayana Jaafar, George J Pappas, and Hamed Hassani. Linear convergence in federated learning: Tackling client heterogeneity and sparse gradients. *Advances in Neural Information Processing Systems*, 34:14606–14619, 2021b.

Viraaji Mothukuri, Reza M. Parizi, Seyedamin Pouriyeh, Yan Huang, Ali Dehghantanha, and Gautam Srivastava. A survey on security and privacy of federated learning. *Future Generation Computer Systems*, 115:619–640, 2021.

Sohom Mukherjee, Nicolas Loizou, and Sebastian U Stich. Locally adaptive federated learning. *arXiv preprint arXiv:2307.06306*, 2023.

Angelia Nedić, Alex Olshevsky, and Wei Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, 2017.

Dinh C. Nguyen, Ming Ding, Pubudu N. Pathirana, Aruna Seneviratne, Jun Li, and H. Vincent Poor. Federated learning for internet of things: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 23(3):2031–2063, 2021.

Dinh C. Nguyen, Quoc-Viet Pham, Pubudu N. Pathirana, Ming Ding, Aruna Seneviratne, Zihuai Lin, Octavia Dobre, and Won-Joo Hwang. Federated learning for smart healthcare: A survey. *ACM Computing Surveys*, 55(3):1–37, 2022.

Milad Khademi Nori, Sangseok Yun, and Il-Min Kim. Fast federated learning by balancing communication trade-offs. *IEEE Transactions on Communications*, 69(8):5168–5182, 2021.

Antonio Orvieto, Simon Lacoste-Julien, and Nicolas Loizou. Dynamics of sgd with stochastic polyak stepsizes: Truly adaptive variants and convergence to exact solution. In *Advances in Neural Information Processing Systems*, volume 35, pp. 26943–26954, 2022.

Zibin Pan, Shuyi Wang, Chi Li, Haijin Wang, Xiaoying Tang, and Junhua Zhao. Fedmdfg: Federated learning with multi-gradient descent and fair guidance. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(8):9364–9371, 2023.

Sharnil Pandya, Gautam Srivastava, Rutvij Jhaveri, M. Rajasekhara Babu, Sweta Bhattacharya, Praveen Kumar Reddy Maddikunta, Spyridon Mastorakis, Md. Jalil Piran, and Thippa Reddy Gadekallu. Federated learning for smart cities: A comprehensive survey. *Sustainable Energy Technologies and Assessments*, 55:102987, 2023.

Reese Pathak and Martin J Wainwright. Fedsplit: an algorithmic framework for fast federated optimization. In *Advances in Neural Information Processing Systems*, volume 33, pp. 7057–7066, 2020.

Shi Pu and Angelia Nedić. Distributed stochastic gradient tracking methods. *Mathematical Programming*, 187(1):409–457, 2021.

Tiancheng Qin, S. Rasoul Etesami, and Cesar A. Uribe. Decentralized federated learning for over-parameterized models. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pp. 5200–5205, 2022a.

Tiancheng Qin, S Rasoul Etesami, and César A Uribe. Faster convergence of local sgd for over-parameterized models. *arXiv preprint arXiv:2201.12719*, 2022b.

Zhaonan Qu, Kaixiang Lin, Zhaojian Li, and Jiayu Zhou. Federated learning's blessing: Fedavg has linear speedup. In *International Conference on Learning Representations*, pp. 1–47, 2021.

Swarna Priya Ramu, Parimala Boopalan, Quoc-Viet Pham, Praveen Kumar Reddy Maddikunta, Thien Huynh-The, Mamoun Alazab, Thanh Thi Nguyen, and Thippa Reddy Gadekallu. Federated learning enabled digital twins for smart cities: Concepts, recent advances, and future directions. *Sustainable Cities and Society*, 79:103663, 2022.

Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020.

Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, Ali Jadbabaie, and Ramtin Pedarsani. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108, pp. 2021–2031. PMLR, 2020.

Felix Sattler, Simon Wiedemann, Klaus-Robert Müller, and Wojciech Samek. Robust and communication-efficient federated learning from non-iid data. *IEEE Network*, 31(9):3400–3413, 2019.

Mark Schmidt and Nicolas Le Roux. Fast convergence of stochastic gradient descent under a strong growth condition. *arXiv preprint arXiv:1308.6370*, 2013.

Nguyen H. Tran, Wei Bao, Albert Zomaya, Minh N. H. Nguyen, and Choong Seon Hong. Federated learning over wireless networks: Optimization model design and analysis. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 1387–1395, 2019.

Sharan Vaswani, Francis Bach, and Mark Schmidt. Fast and faster convergence of SGD for over-parameterized models and an accelerated perceptron. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89, pp. 1195–1204. PMLR, 16–18 Apr 2019a.

Sharan Vaswani, Aaron Mishkin, Issam Laradji, Mark Schmidt, Gauthier Gidel, and Simon Lacoste-Julien. Painless stochastic gradient: Interpolation, line-search, and convergence rates. In *Advances in Neural Information Processing Systems*, volume 32, pp. 3727–3740, 2019b.

Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. In *Advances in Neural Information Processing Systems*, volume 33, pp. 7611–7623, 2020.

Kang Wei, Jun Li, Ming Ding, Chuan Ma, Yo-Seb Jeon, and H. Vincent Poor. Covert model poisoning against federated learning: Algorithm design and optimization. *IEEE Transactions on Dependable and Secure Computing*, 21(3):1196–1209, 2024.

Ming Xiang, Stratis Ioannidis, Edmund Yeh, Carlee Joe-Wong, and Lili Su. Efficient federated learning against heterogeneous and non-stationary client unavailability. *Advances in Neural Information Processing Systems*, 37:104281–104328, 2024.

Jie Xu, Benjamin S. Glicksberg, Chang Su, Peter Walker, Jiang Bian, and Fei Wang. Federated learning for healthcare informatics. *Journal of Healthcare Informatics Research*, 5:1–19, 2021a.

Jing Xu, Sen Wang, Liwei Wang, and Andrew Chi-Chih Yao. Fedcm: Federated learning with client-level momentum. *arXiv preprint arXiv:2106.10874*, 2021b.

Wenjing Yan, Kai Zhang, Xiaolu Wang, and Xuanyu Cao. Problem-parameter-free federated learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

Haibo Yang, Minghong Fang, and Jia Liu. Achieving linear speedup with partial worker participation in non-iid federated learning. *arXiv preprint arXiv:2101.11203*, 2021.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2):1–19, 2019.

Zhengjie Yang, Wei Bao, Dong Yuan, Nguyen H. Tran, and Albert Y. Zomaya. Federated learning with nesterov accelerated gradient. *IEEE Transactions on Parallel and Distributed Systems*, 33(12):4863–4873, 2022.

Hao Yu, Sen Yang, and Shenghuo Zhu. Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 5693–5700, 2019.

Xiaotong Yuan and Ping Li. On convergence of fedprox: Local dissimilarity invariant bounds, non-smoothness and beyond. *Advances in Neural Information Processing Systems*, 35:10752–10765, 2022.

Kaiyue Zhang, Xuan Song, Chenhan Zhang, and Shui Yu. Challenges and future directions of secure federated learning: a survey. *Frontiers of Computer Science*, 16(5):165817, 2022a.

Tuo Zhang, Lei Gao, Chaoyang He, Mi Zhang, Bhaskar Krishnamachari, and A. Salman Avestimehr. Federated learning for the internet of things: Applications, challenges, and opportunities. *IEEE Internet of Things Magazine*, 5(1):24–29, 2022b.

Xingyu Zhou. On the fenchel duality between strong convexity and lipschitz continuous gradient. *arXiv preprint arXiv:1803.06573*, 2018.

Ligeng Zhu, Hongzhou Lin, Yao Lu, Yujun Lin, and Song Han. Delayed gradient averaging: Tolerate the communication latency for federated learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 29995–30007, 2021.

Xinghua Zhu, Jianzong Wang, Zhenhou Hong, and Jing Xiao. Empirical studies of institutional federated learning for natural language processing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 625–634, 2020.

A Additional Experiments

A.1 Comparison of Algorithm 2 with Existing Works in Least Squares Regression

We consider $[A_i]_{jk}$ and $[b_i]_j$ generated from $[0, 1]$ randomly for $1 \leq j \leq 500$, $1 \leq k \leq 100$, and $i \in \mathcal{S}$. After the initial random generation of data, we purposely set $[A_1]_{j,1} = [A_1]_{j,2}$ for $1 \leq j \leq 500$ to make A_1 not full rank. By doing so, we can obtain a local loss function $f_1(x) = \frac{1}{2}\|A_1x - b_1\|^2$ that is convex but not strongly convex.

We compared the performance of Algorithm 2 under the proposed stepsize (8) in Theorem 3 with those in Mitra et al. (2021a;b). The convergence performances of Algorithm 2 and algorithms in Mitra et al. (2021a;b) under different local training periods $\tau = 2, 3, 4, 5, 6, 7$ are shown as Figure 9. It is clear that the proposed stepsize strategy indeed yields much faster convergence than the compared counterparts.

A.2 Additional Comparison under IID distribution

In Section 6.1.1, Figure 1 compares Algorithm 1 and Algorithm 2 with existing methods Qin et al. (2022b); Mukherjee et al. (2023); Mitra et al. (2021b); Khaled et al. (2020) using universal stepsizes. For further comparison, Figure 10 illustrates the behavior of the same existing methods when equipped with local stepsizes. Both Figures 1 and 10 highlight the superior performance of Algorithm 1 over the existing approaches.

A.3 Additional Evaluation using CIFAR-10 under non-IID distribution

In this subsection, we conduct additional numerical experiments on CIFAR-10 to compare our method with several federated learning baselines: FedAdam (Reddi et al., 2020), FedProx (Yuan & Li, 2022), SCAFFOLD (Karimireddy et al., 2020), FedLin (Mitra et al., 2021b), and Scaffnew (Mishchenko et al., 2022). Figures 11, 12, and 13 present results for β set to 1, 0.1 and 10, respectively, corresponding to moderate, high, and low heterogeneity in the data distributions. For all results shown in Figures 11, 12, and 13, the stepsizes for Algorithm 2, FedAdam, FedProx, SCAFFOLD, FedLin, and Scaffnew are selected according to the guidelines from Theorem 3, Reddi et al. (2020), Yuan & Li (2022), Karimireddy et al. (2020), Mitra et al. (2021b), and

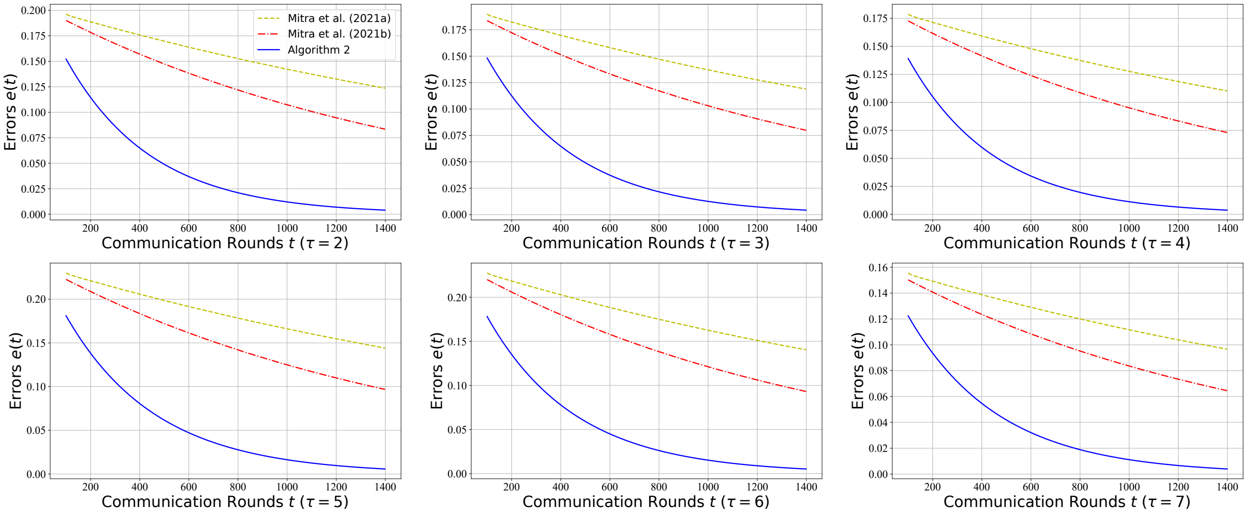


Figure 9: Comparisons of the performance of Algorithm 2 under the proposed stepsize with Mitra et al. (2021a;b) under different local training periods τ

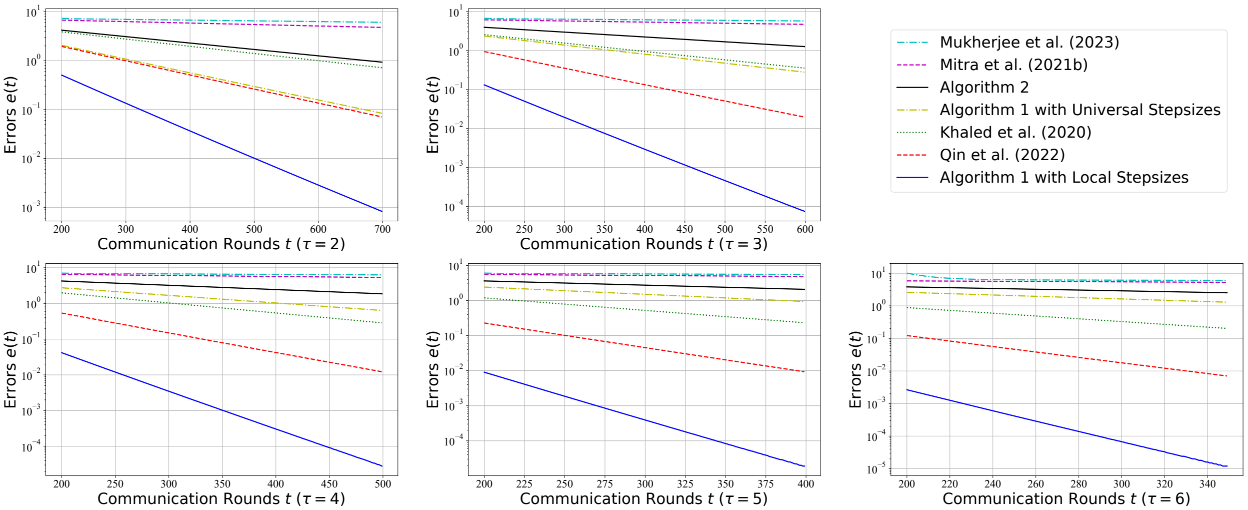


Figure 10: Comparisons of Algorithm 1 and Algorithm 2 with Qin et al. (2022b); Mukherjee et al. (2023); Mitra et al. (2021b); Khaled et al. (2020) with local stepsizes under different local training periods τ.

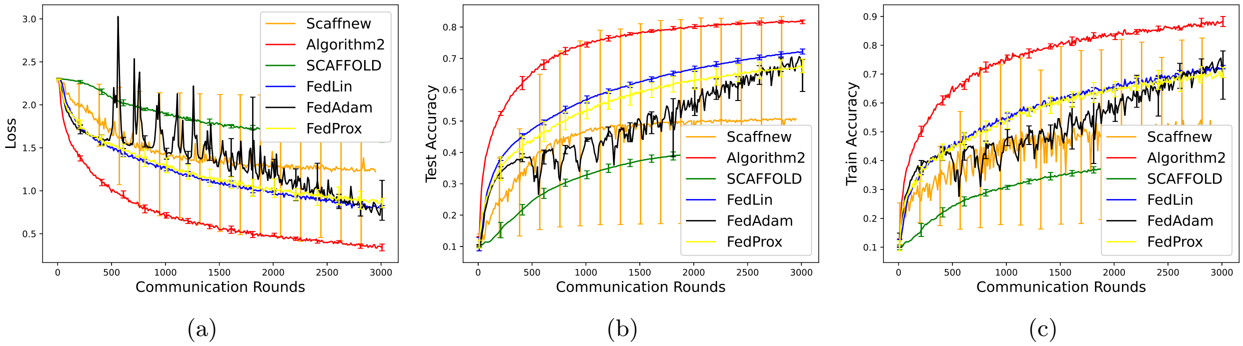
Mishchenko et al. (2022), respectively, using an estimated smoothness parameter of $L = 2$. For Algorithm 2, FedAdam, FedProx, SCAFFOLD, and FedLin, the local training period is set to $\tau = 10$. For Scaffnew, the communication probability is set to $\frac{1}{11}$ to ensure that the total number of communicated messages remains consistent across methods. A summary of the experimental setup is given in Table 3. As shown in the figures, our algorithm achieves faster convergence and higher accuracy than other baseline federated-learning algorithms. Note that the large variance of Scaffnew arises from the additional randomness introduced by its communication mechanism.

Table 3: Experimental Setup in Figures 11, 12, and 13

	Figure 11	Figure 12	Figure 13
Dataset	CIFAR-10	CIFAR-10	CIFAR-10
Heterogeneity	$\beta = 0.1$	$\beta = 1$	$\beta = 10$
Local Training Period[1]	$\tau = 10$	$\tau = 10$	$\tau = 10$
Number of Agents	10	10	10
Optimizer[2]	See label	See label	See label

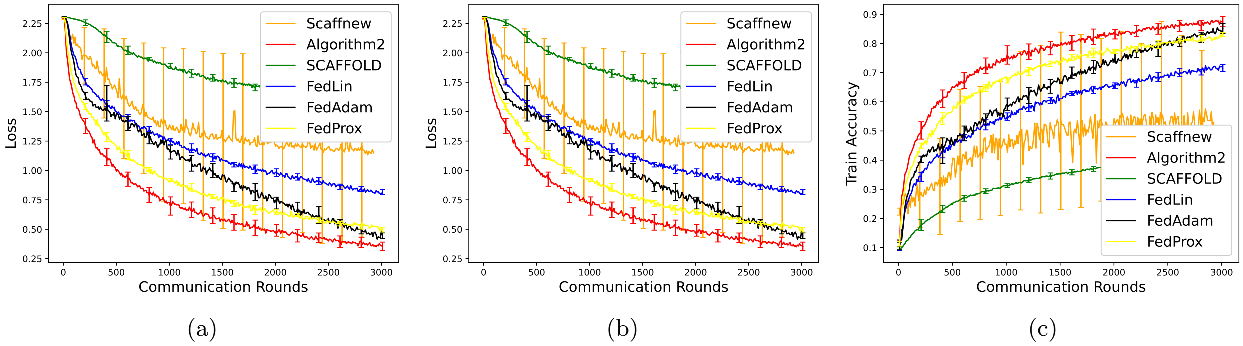
[1] For Scaffnew, the communication probability is set to $\frac{1}{11}$ to ensure that the total number of communicated messages remains consistent across methods.

[2] The stepsizes for Algorithm 2, FedAdam, FedProx, SCAFFOLD, FedLin, and Scaffnew are selected according to the guidelines from Theorem 3, Reddi et al. (2020), Yuan & Li (2022), Karimireddy et al. (2020), Mitra et al. (2021b), and Mishchenko et al. (2022), respectively, using an estimated smoothness parameter of $L = 2$.



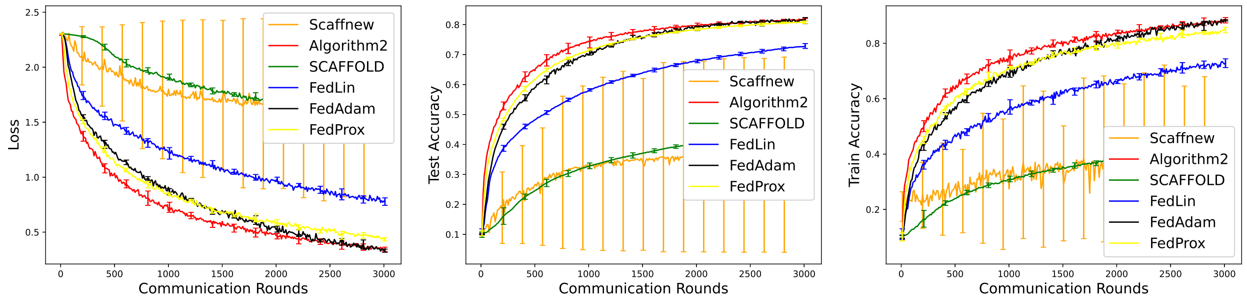
(a) (b) (c)

Figure 11: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—FedAdam, FedProx, SCAFFOLD, FedLin, and Scaffnew—on the CIFAR-10 dataset. The Dirichlet distribution parameter was set to $\beta = 0.1$. Each curve represents the average of five independent runs.



(a) (b) (c)

Figure 12: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—FedAdam, FedProx, SCAFFOLD, FedLin, FedAdam, FedProx, and Scaffnew—on the CIFAR-10 dataset. The Dirichlet distribution parameter was set to $\beta = 1$. Each curve represents the average of five independent runs.



(a)

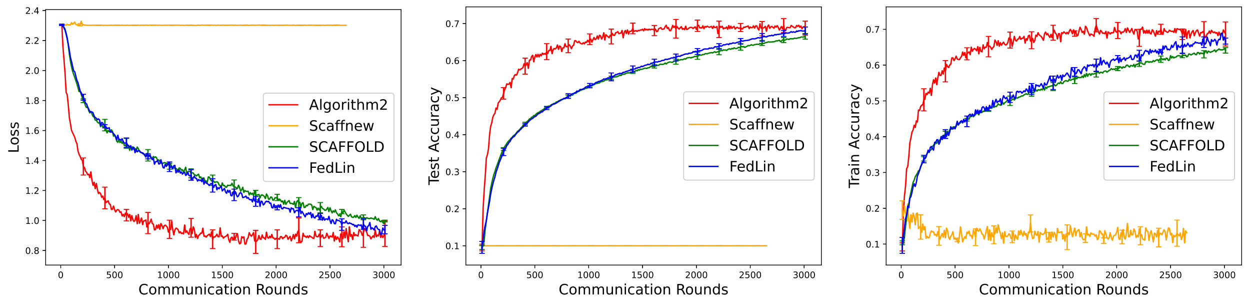
(b)

(c)

Figure 13: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—FedAdam, FedProx, SCAFFOLD, FedLin, FedAdam, FedProx, and Scaffnew—on the CIFAR-10 dataset. The Dirichlet distribution parameter was set to $\beta = 10$. Each curve represents the average of five independent runs.

A.4 Sensitivity of smoothness constants in CIFAR-10 experiments under non-IID settings.

The numerical experiments in Section 6.2 employ an estimated smoothness constant of $L = 2$. To assess the sensitivity of the results to this parameter, we perform additional tests on the CIFAR-10 dataset using estimated values of $L = 1$ (Figure 14) and $L = 1.5$ (Figure 15). Figures 14 and 15 demonstrate that our algorithm maintains faster convergence and higher accuracy even with these varying estimated values.

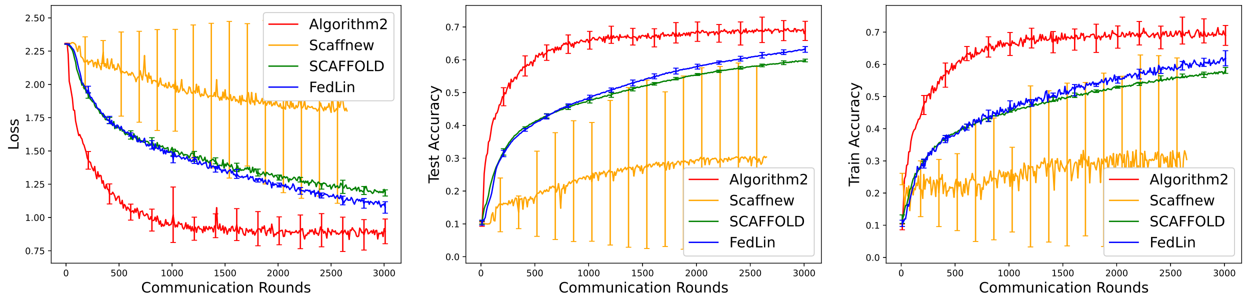


(a)

(b)

(c)

Figure 14: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—SCAFFOLD, FedLin, and Scaffnew—on the CIFAR-10 dataset. The Dirichlet distribution parameter was set to $\beta = 1$ and the smooth constant is estimated as $L = 1$. Each curve represents the average of five independent runs.



(a)

(b)

(c)

Figure 15: Comparison of Algorithm 2 with state-of-the-art federated learning algorithms—SCAFFOLD, FedLin, and Scaffnew—on the CIFAR-10 dataset. The Dirichlet distribution parameter was set to $\beta = 1$ and the smooth constant is estimated as $L = 1.5$. Each curve represents the average of five independent runs.

B Supporting Lemmas for the Proof of Theorem 1

Lemma 3 (Zhou (2018)). *For every L_i-smooth and convex function $f_i(x)$ over $\mathbb{R}^n$, we have*

$$f_i(y) \geq f_i(x) + \langle \nabla f_i(x), y - x \rangle + \frac{1}{2L_i} \|\nabla f_i(x) - \nabla f_i(y)\|^2$$

for any $x, y \in \mathbb{R}^n$ and $i \in \mathcal{S}$.

Lemma 4 (Mitra et al. (2021b)). *Suppose that $f_i(x)$ is L_i-smooth and convex. Then, for any $0 \leq \alpha \leq \frac{1}{L_i}$, we have*

$$\|y - x - \alpha(\nabla f_i(y) - \nabla f_i(x))\| \leq \|y - x\|$$

for any $x, y \in \mathbb{R}^n$.

Lemma 5 (Zhou (2018)). *For the convex and L-smooth function $f(x)$, we have*

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2} \|y - x\|^2$$

for any $x, y \in \mathbb{R}^n$.

C Proof of Theorem 1

Proof. The sequence $\{f(\bar{x}(t)) - f(x^*)\}$ satisfies

$$f(\bar{x}(t)) - f(x^*) \geq 0$$

for any $t \geq 1$. From Lemma 1, to prove Theorem 1, we only need to prove that the nonnegative sequence $\{f(\bar{x}(t)) - f(x^*)\}$ satisfies the summable and monotonically decreasing properties.

- **Summable Property**: Firstly, we establish the summable property.

  From (2) in Algorithm 1, we have

  $$\begin{aligned} &\|x_{i,k+1}(t) - x^*\|^2 \\ =&\|x_{i,k}(t) - x^*\|^2 - 2\alpha\langle \nabla f_i(x_{i,k}(t)), x_{i,k}(t) - x^* \rangle + \alpha^2 \|\nabla f_i(x_{i,k}(t))\|^2. \end{aligned} \tag{11}$$

  From the convexity of $f_i(x)$, we have

  $$-2\alpha\langle \nabla f_i(x_{i,k}(t)), x_{i,k}(t) - x^* \rangle \leq 2\alpha\{f_i(x^*) - f_i(x_{i,k}(t))\}. \tag{12}$$

  Using the strong growth condition (see Assumption 3) yields

  $$\|\nabla f_i(x^*)\| = 0 \tag{13}$$

  for any $i \in \mathcal{S}$ and $x^* \in \mathcal{X}^*$.

  Then, from Lemma 3 and (13), we have

  $$\|\nabla f_i(x_{i,k}(t))\|^2 \leq 2L_i \Big\{ f_i(x_{i,k}(t)) - f_i(x^*) \Big\}. \tag{14}$$

  Combining (11), (12), and (14), we arrive at

  $$\|x_{i,k+1}(t) - x^*\|^2 \leq \|x_{i,k}(t) - x^*\|^2 + (2\alpha - 2L_i\alpha^2) \Big\{ f_i(x^*) - f_i(x_{i,k}(t)) \Big\}. \tag{15}$$

It is worth noting that the following inequality holds

$$\|\bar{x}(t+1) - x^*\|^2 \leq \frac{1}{N}\sum_{i=1}^{N}\|x_{i,\tau}(t) - x^*\|^2.$$

Thus, from Algorithm 1 and (15), we have

$$\begin{aligned}&\|\bar{x}(t+1) - x^*\|^2 \\ \leq&\frac{1}{N}\sum_{i=1}^{N}(2\alpha - 2L_i\alpha^2)\sum_{k=0}^{\tau-1}\Big\{f_i(x^*) - f_i(x_{i,k}(t))\Big\} + \|\bar{x}(t) - x^*\|^2.\end{aligned} \tag{16}$$

Under the stepsize setting (3), we have

$$2\alpha - 2L_i\alpha^2 > 0$$

for any $i \in \mathcal{S}$. Moreover, from Assumption 3, we have $\|\nabla f_i(x^*)\| = 0$ for any $x^* \in \mathcal{X}^*$ and $i \in \mathcal{S}$. Thus, in (16), we have

$$f_i(x^*) - f_i(x_{i,k}(t)) \leq 0$$

for any $i \in \mathcal{S}$ and $k = 0, 1, \cdots, \tau - 1$.

Thus, using (16), we have

$$\|\bar{x}(t+1) - x^*\|^2 \leq \|\bar{x}(t) - x^*\|^2 + \Big\{2\alpha - 2L\alpha^2\Big\}\Big\{f(x^*) - f(\bar{x}(t))\Big\}. \tag{17}$$

From (17), we can obtain

$$f(\bar{x}(t)) - f(x^*) \leq \frac{\|\bar{x}(t) - x^*\|^2 - \|\bar{x}(t+1) - x^*\|^2}{2\alpha - 2L\alpha^2}.$$

Thus, for any $T \geq 1$, we have

$$\sum_{t=1}^{T}\Big\{f(\bar{x}(t)) - f(x^*)\Big\} \leq \frac{\|\bar{x}(1) - x^*\|^2}{2\alpha - 2L\alpha^2}, \tag{18}$$

which establishes the summable property of the sequence $\{f(\bar{x}(t)) - f(x^*)\}$.

- **Monotonically Decreasing**:

  Next, we show that $f(\bar{x}(t))$ is monotonically decreasing.

  From Algorithm 1, we have

$$\begin{aligned}&\|x_{i,k+1}(t) - \bar{x}(t)\| \\ \leq&\|x_{i,k}(t) - \bar{x}(t) - \alpha\Big(\nabla f_i(x_{i,k}(t)) - \nabla f_i(\bar{x}(t))\Big)\| + \alpha\|\nabla f_i(\bar{x}(t))\|.\end{aligned}$$

  Using Lemma 4 and the stepsize setting in (3), we arrive at

$$\|x_{i,k+1}(t) - \bar{x}(t)\| \leq \|x_{i,k}(t) - \bar{x}(t)\| + \alpha\|\nabla f_i(\bar{x}(t))\|. \tag{19}$$

  Using the update rule in Algorithm 1, we obtain

$$\|x_{i,k}(t) - \bar{x}(t)\| \leq k\alpha\|\nabla f_i(\bar{x}(t))\| \tag{20}$$

  for $k = 0, 1, 2, \cdots, \tau - 1$.

It is worth noting that the following inequality always holds:

$$\Big\|\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\| \leq \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\Big\|\nabla f_i(x_{i,k}(t)) - \nabla f_i(\bar{x}(t))\Big\| + \tau\|\nabla f(\bar{x}(t))\|.$$

From Assumption 1 and (3), we have

$$\Big\|\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\| \leq \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}L_i\|x_{i,k}(t) - \bar{x}(t)\| + \tau\|\nabla f(\bar{x}(t))\|.$$

Further using Assumption 3 and the update rule in Algorithm 3, we arrive at

$$\Big\|\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\| \leq \Big\{\tau + \frac{\eta(\tau-1)}{2}\Big\}\|\nabla f(\bar{x}(t))\|. \tag{21}$$

From (2), we have

$$\bar{x}(t+1) = \bar{x}(t) - \frac{\alpha}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t)).$$

The global loss function $f(x)$ is L-smooth with $L = \frac{1}{N}\sum_{i=1}^{N}L_i$. From Lemma 5 and Assumption 1, we have

$$\begin{aligned}
&f(\bar{x}(t+1))\\
\leq& f(\bar{x}(t)) - \langle\nabla f(\bar{x}(t)), \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\alpha\nabla f_i(\bar{x}(t))\rangle + \frac{L}{2}\|\frac{\alpha}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\|^2\\
&- \langle\nabla f(\bar{x}(t)), \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\alpha\nabla f_i(x_{i,k}(t)) - \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\alpha\nabla f_i(\bar{x}(t))\rangle.
\end{aligned}$$

As for the last term of the above inequality, we have

$$\begin{aligned}
&\langle\nabla f(\bar{x}(t)), \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\alpha\nabla f_i(x_{i,k}(t)) - \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\alpha\nabla f_i(\bar{x}(t))\rangle\\
\leq&\|\nabla f(\bar{x}(t))\|\Big\{\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\alpha\|\nabla f_i(x_{i,k}(t)) - \nabla f_i(\bar{x}(t))\|\Big\}\\
\leq&\|\nabla f(\bar{x}(t))\|\Big\{\frac{\alpha}{N}\sum_{k=0}^{\tau-1}\sum_{i=1}^{N}L_i\|x_{i,k}(t) - \bar{x}(t)\|\Big\},
\end{aligned} \tag{22}$$

where the first inequality follows from Cauchy–Schwarz inequality and the second inequality follows from the L-smooth assumption (see Assumption 1). Furthermore, using Assumption 3, (21), and (22), we have

$$\begin{aligned}
&f(\bar{x}(t+1))\\
\leq& f(\bar{x}(t)) - \alpha\tau\|\nabla f(\bar{x}(t))\|^2 + \frac{L}{2}\alpha^2\Big\{\tau + \frac{\eta(\tau-1)}{2}\Big\}^2\|\nabla f(\bar{x}(t))\|^2\\
&+ \alpha\|\nabla f(\bar{x}(t))\|\Big\{\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}L_i\|x_{i,k}(t) - \bar{x}(t)\|\Big\}.
\end{aligned} \tag{23}$$

Taking (20) into (23) yields

$$\begin{aligned}
&f(\bar{x}(t+1))\\
\leq &f(\bar{x}(t)) - \alpha\tau\|\nabla f(\bar{x}(t))\|^2 + \frac{L}{2}\alpha^2\Big\{\tau + \frac{\eta(\tau-1)}{2}\Big\}^2\|\nabla f(\bar{x}(t))\|^2\\
&+ \alpha\|\nabla f(\bar{x}(t))\|\Big\{\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1} L_i k\alpha\|\nabla f_i(\bar{x}(t))\|\Big\}. \qquad (24)
\end{aligned}$$

From Assumption 3 and (24), we arrive at

$$\begin{aligned}
&f(\bar{x}(t+1))\\
\leq &f(\bar{x}(t)) - \alpha\tau\|\nabla f(\bar{x}(t))\|^2 + \frac{L}{2}\alpha^2\Big\{\tau + \frac{\eta(\tau-1)}{2}\Big\}^2\|\nabla f(\bar{x}(t))\|^2\\
&+ \alpha\|\nabla f(\bar{x}(t))\|\Big\{\eta\sum_{k=0}^{\tau-1} k\alpha\|\nabla f(\bar{x}(t))\|\Big\}\Big\{\frac{1}{N}\sum_{i=1}^{N} L_i\Big\}, \qquad (25)
\end{aligned}$$

which further implies

$$\begin{aligned}
&f(\bar{x}(t+1))\\
\leq &f(\bar{x}(t)) - \alpha\tau\|\nabla f(\bar{x}(t))\|^2 + \frac{L}{2}\alpha^2\Big\{\tau + \frac{\eta(\tau-1)}{2}\Big\}^2\|\nabla f(\bar{x}(t))\|^2\\
&+ \frac{\eta L\tau(\tau-1)\alpha^2}{2}\|\nabla f(\bar{x}(t))\|^2.
\end{aligned}$$

Therefore, the stepsize should satisfy

$$-\tau\alpha + \frac{L}{2}\Big\{\tau + \frac{\eta(\tau-1)}{2}\Big\}^2\alpha^2 + \frac{\eta L\tau(\tau-1)}{2}\alpha^2 \leq 0$$

to guarantee the monotonically decreasing property of $f(\bar{x}(t)) - f(x^*)$. Equivalently, stepsize satisfying

$$\alpha \leq \frac{8\tau}{L(2\tau + \eta(\tau-1))^2 + 4\eta L\tau(\tau-1)}$$

guarantees the monotonically decreasing property of $f(\bar{x})$.

□

D Proof of Theorem 2

Proof. From (2) in Algorithm 1, we can obtain

$$\begin{aligned}
&\|x_{i,k+1}(t) - x^*\|^2\\
=&\|x_{i,k}(t) - x^*\|^2 - 2\alpha_i\langle\nabla f_i(x_{i,k}(t)), x_{i,k}(t) - x^*\rangle + \alpha_i^2\|\nabla f_i(x_{i,k}(t))\|^2. \qquad (26)
\end{aligned}$$

From the convexity property of $f_i(x)$, we have

$$-2\alpha_i\langle\nabla f_i(x_{i,k}(t)), x_{i,k}(t) - x^*\rangle \leq 2\alpha_i\{f_i(x^*) - f_i(x_{i,k}(t))\}. \qquad (27)$$

Assumption 3 implies

$$\|\nabla f_i(x^*)\| = 0$$

for any $i \in \mathcal{S}$ and $x^* \in \mathcal{X}^*$. Thus, combining with Lemma 3, we have

$$\|\nabla f_i(x_{i,k}(t))\|^2 \leq 2L_i\Big\{f_i(x_{i,k}(t)) - f_i(x^*)\Big\}. \tag{28}$$

Combining (26), (27), and (28), we can obtain

$$\|x_{i,k+1}(t) - x^*\|^2 \leq \|x_{i,k}(t) - x^*\|^2 + (2\alpha_i - 2L_i\alpha_i^2)\Big\{f_i(x^*) - f_i(x_{i,k}(t))\Big\}. \tag{29}$$

Note that the following inequality always holds:

$$\|\bar{x}(t+1) - x^*\|^2 \leq \frac{1}{N}\sum_{i=1}^{N}\|x_{i,\tau}(t) - x^*\|^2.$$

Hence, Algorithm 1 and (29) imply

$$\|\bar{x}(t+1) - x^*\|^2 \leq \frac{1}{N}\sum_{i=1}^{N}(2\alpha_i - 2L_i\alpha_i^2)\sum_{k=0}^{\tau-1}\Big\{f_i(x^*) - f_i(x_{i,k}(t))\Big\} + \|\bar{x}(t) - x^*\|^2. \tag{30}$$

Under the stepsize setting (4), we have

$$2\alpha_i - 2L_i\alpha_i^2 > 0$$

for any $i \in \mathcal{S}$. Moreover, Assumption 3 ensures

$$f_i(x^*) - f_i(x_{i,k}(t)) \leq 0$$

for any $i \in \mathcal{S}$ and $k = 0, 1, \cdots, \tau - 1$.

Substituting the above inequality into (30) yields

$$\|\bar{x}(t+1) - x^*\|^2 \leq \|\bar{x}(t) - x^*\|^2 + \min_{1\leq i\leq N}\Big\{2\alpha_i - 2L_i\alpha_i^2\Big\}\Big\{f(x^*) - f(\bar{x}(t))\Big\}. \tag{31}$$

From (31), we can obtain

$$f(\bar{x}(t)) - f(x^*) \leq \frac{\|\bar{x}(t) - x^*\|^2 - \|\bar{x}(t+1) - x^*\|^2}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}}.$$

Thus, for any $T \geq 1$, we have

$$\sum_{t=1}^{T}\Big\{f(\bar{x}(t)) - f(x^*)\Big\} \leq \frac{\|\bar{x}(1) - x^*\|^2}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}}. \tag{32}$$

Since $f(\bar{x}(t)) - f(x^*) \geq 0$ holds for any t, we have

$$\lim_{t\to\infty} f(\bar{x}(t)) = f(x^*).$$

In addition, from (32), for any $T \geq 1$, we can obtain

$$f(\frac{1}{T}\sum_{t=1}^{T}\bar{x}(t)) - f(x^*) \leq \frac{1}{T}\sum_{t=1}^{T}\Big\{f(\bar{x}(t)) - f(x^*)\Big\} \leq \frac{\|\bar{x}(1) - x^*\|^2}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}T},$$

which completes the proof.

□

E Proof Theorem 3

Proof. From Algorithm 2, the updated rules (6) and (7) can be equivalently expressed as

$$x_{i,k+1}(t) = x_{i,k}(t) - \alpha(\nabla f(\bar{x}(t)) - \nabla f_i(\bar{x}(t)) + \nabla f_i(x_{i,k}(t))). \tag{33}$$

The relation in (33), we further implies

$$\|x_{i,k+1}(t) - \bar{x}(t)\| \leq \Big\|x_{i,k}(t) - \bar{x}(t) + \alpha\Big(\nabla f_i(x_{i,k}(t)) - \nabla f_i(\bar{x}(t))\Big)\Big\| + \alpha(t)\|\nabla f(\bar{x}(t))\|. \tag{34}$$

From Lemma 4 and (34), if the stepsize satisfies $0 \leq \alpha \leq \frac{1}{L_i}$, we have

$$\|x_{i,k+1}(t) - \bar{x}(t)\| \leq \|x_{i,k}(t) - \bar{x}(t)\| + \alpha\|\nabla f(\bar{x}(t))\|. \tag{35}$$

Using induction, we obtain

$$\|x_{i,k}(t) - \bar{x}(t)\| \leq k\alpha\|\nabla f(\bar{x}(t))\|. \tag{36}$$

Using the update rule in Algorithm 2, we can obtain

$$x_{i,\tau}(t) = \bar{x}(t) - \alpha\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t)) - \tau\alpha\Big(\nabla f(\bar{x}(t)) - \nabla f_i(\bar{x}(t))\Big).$$

Therefore, the average parameter $\bar{x}(t+1)$ satisfies

$$\bar{x}(t+1) = \frac{1}{N}\sum_{i=1}^{N} x_{i,\tau}(t) = \bar{x}(t) - \frac{\alpha}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t)), \tag{37}$$

which further implies

$$\begin{aligned}&\|\bar{x}(t+1) - x^*\|^2 - \|\bar{x}(t) - x^*\|^2\\ =& -2\Big\langle \frac{\alpha}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t)), \bar{x}(t) - x^*\Big\rangle + \Big\|\frac{\alpha}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\|^2.\end{aligned} \tag{38}$$

For the first term on the right hand side of (38), we have

$$\begin{aligned}&-\Big\langle \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t)), \bar{x}(t) - x^*\Big\rangle\\ =&\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\langle x^* - x_{i,k}(t), \nabla f_i(x_{i,k}(t))\rangle + \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\langle x_{i,k}(t) - \bar{x}(t), \nabla f_i(\bar{x}(t))\rangle\\ &+ \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\langle x_{i,k}(t) - \bar{x}(t), \nabla f_i(x_{i,k}(t)) - \nabla f_i(\bar{x}(t))\rangle.\end{aligned} \tag{39}$$

Furthermore, using Assumption 1 and the convexity property of $f_i(x)$, we can obtain

$$\begin{aligned}&-\Big\langle \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t)), \bar{x}(t) - x^*\Big\rangle\\ \leq&\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\Big\{f_i(x^*) - f_i(x_{i,k}(t))\Big\} + \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\Big\{f_i(x_{i,k}(t)) - f_i(\bar{x}(t))\Big\}\\ &+ \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}L_i\|x_{i,k}(t) - \bar{x}(t)\|^2.\end{aligned} \tag{40}$$

Combining (36) and (40), we arrive at

$$\begin{aligned}&-\Big\langle \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t)), \bar{x}(t)-x^*\Big\rangle\\ \le&\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\Big\{f_i(x^*)-f_i(\bar{x}(t))\Big\}+\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}L_i k^2\alpha^2\|\nabla f(\bar{x}(t))\|^2. \end{aligned} \tag{41}$$

Plugging the stepsize condition $0<\alpha L_i \le 1$ into (41) yields

$$-\Big\langle \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t)), \bar{x}(t)-x^*\Big\rangle \le \tau\Big\{f(x^*)-f(\bar{x}(t))\Big\}+\alpha\|\nabla f(\bar{x}(t))\|^2\sum_{k=0}^{\tau-1}k^2. \tag{42}$$

Applying the relation $\sum_{k=1}^{n}k^2=\frac{n(n+1)(2n+1)}{6}$ to the second term on the right hand side of (42) yields

$$-\Big\langle \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t)), \bar{x}(t)-x^*\Big\rangle \le \tau\Big\{f(x^*)-f(\bar{x}(t))\Big\}+A_1\alpha\|\nabla f(\bar{x}(t))\|^2, \tag{43}$$

where $A_1=\frac{\tau(\tau-1)(2\tau-1)}{6}$.

For the second term on the right hand side of (38), we have

$$\Big\|\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\| \le \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\Big\|\nabla f_i(x_{i,k}(t))-\nabla f_i(\bar{x}(t))\Big\|+\tau\|\nabla f(\bar{x}(t))\|. \tag{44}$$

Using the smoothness condition in Assumption 1, we can further obtain

$$\Big\|\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\| \le \frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}L_i\|x_{i,k}(t)-\bar{x}(t)\|+\tau\|\nabla f(\bar{x}(t))\|. \tag{45}$$

Combining (36) and (45), we can obtain

$$\Big\|\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\| \le \sum_{k=0}^{\tau-1}L_i k\alpha\|\nabla f(\bar{x}(t))\|+\tau\|\nabla f(\bar{x}(t))\|. \tag{46}$$

Applying the stepsize condition $0<\tau\alpha L_i\le 1$ to (46) yields

$$\Big\|\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\| \le A_2\|\nabla f(\bar{x}(t))\|, \tag{47}$$

where $A_2=2\tau-1$.

Then, combining (38), (43), and (47), we can obtain

$$\|\bar{x}(t+1)-x^*\|^2-\|\bar{x}(t)-x^*\|^2\le 2\tau\alpha\Big\{f(x^*)-f(\bar{x}(t))\Big\}+\Big\{2A_1+A_2^2\Big\}\alpha^2\|\nabla f(\bar{x}(t))\|^2. \tag{48}$$

Using the relation established in Lemma 2, we can obtain the following inequality from (48):

$$\begin{aligned}&\|\bar{x}(t+1)-x^*\|^2-\|\bar{x}(t)-x^*\|^2\\ \le&2\tau\alpha\Big\{f(x^*)-f(\bar{x}(t))\Big\}+\frac{2A_1+A_2^2}{\gamma}\Big\{f(\bar{x}(t))-f(\bar{x}(t+1))\Big\}.\end{aligned} \tag{49}$$

Rearranging terms yields

$$2\tau\alpha\Big\{f(\bar{x}(t))-f(x^*)\Big\}\leq\|\bar{x}(t)-x^*\|^2-\|\bar{x}(t+1)-x^*\|^2+\frac{2A_1+A_2^2}{\gamma}\Big\{f(\bar{x}(t))-f(\bar{x}(t+1))\Big\}. \tag{50}$$

Thus, for any $T>0$, summarizing (50) from $t=1$ to $t=T$ leads to

$$\sum_{t=1}^{T}\Big\{f(\bar{x}(t))-f(x^*)\Big\}\leq\frac{1}{2\tau\alpha}\|\bar{x}(1)-x^*\|^2+\frac{2A_1+A_2^2}{2\tau\alpha\gamma}\Big\{f(\bar{x}(1))-f(x^*)\Big\}. \tag{51}$$

Using (51), Lemma 2, and Lemma 1, we can conclude that $f(\bar{x}(t))$ converges to $f(x^*)$ with the convergence rate $o(1/t)$, which completes the proof. □

F Proof of Lemma 2

Under Assumption 1, we know that $f(x)$ is L-smooth. Thus, from Lemma 5, we have

$$\begin{aligned}&f(\bar{x}(t+1))\\ \leq&f(\bar{x}(t))-\alpha\Big\langle\nabla f(\bar{x}(t)),\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\rangle+\frac{L}{2}\Big\|\frac{\alpha}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\nabla f_i(x_{i,k}(t))\Big\|^2.\end{aligned} \tag{52}$$

Substituting (47) into (52) leads to

$$\begin{aligned}f(\bar{x}(t+1))\leq&f(\bar{x}(t))-\alpha\tau\|\nabla f(\bar{x}(t))\|^2+2L\tau^2\alpha^2\|\nabla f(\bar{x}(t))\|^2\\&+\alpha\|\nabla f(\bar{x}(t))\|\Big\{\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}\|\nabla f_i(x_{i,k}(t))-\nabla f_i(\bar{x}(t))\|\Big\}.\end{aligned} \tag{53}$$

Using the smoothness condition in Assumption 1, we can have the following relationship for (53):

$$\begin{aligned}f(\bar{x}(t+1))\leq&f(\bar{x}(t))-\alpha\tau\|\nabla f(\bar{x}(t))\|^2+2L\tau^2\alpha^2\|\nabla f(\bar{x}(t))\|^2\\&+\alpha\|\nabla f(\bar{x}(t))\|\Big\{\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{\tau-1}L_i\|x_{i,k}(t)-\bar{x}(t)\|\Big\}.\end{aligned} \tag{54}$$

Plugging (36) into (54) yields

$$f(\bar{x}(t+1))\leq f(\bar{x}(t))-\alpha\tau\|\nabla f(\bar{x}(t))\|^2+\frac{5\tau^2-\tau}{2}L\alpha^2\|\nabla f(\bar{x}(t))\|^2. \tag{55}$$

Rearranging like terms leads to

$$\Big\{\alpha\tau-\frac{5L\tau^2-L\tau}{2}\alpha^2\Big\}\|\nabla f(\bar{x}(t))\|^2\leq f(\bar{x}(t))-f(\bar{x}(t+1)). \tag{56}$$

Since the stepsize α satisfies $0<\alpha<\frac{2}{5L\tau-L}$, there exist $\gamma>0$ such that

$$\alpha\tau-\frac{5\tau^2-\tau}{2}L\alpha^2\geq\gamma\alpha^2.$$

Thus, we have

$$\gamma\alpha^2\|\nabla f(\bar{x}(t))\|^2\leq f(\bar{x}(t))-f(\bar{x}(t+1)),$$

implying that the sequence $f(\bar{x}(t))$ is monotonically decreasing, which completes the proof.

G Proof of Theorem 4

Proof. We use $g_i(x)$ to represent the unbiased estimate of the gradient $\nabla f_i(x)$. From (2) in Algorithm 1, we can obtain

$$\begin{aligned}&\|x_{i,k+1}(t)-x^*\|^2\\ \le&\|x_{i,k}(t)-x^*\|^2-2\alpha_i\langle\nabla f_i(x_{i,k}(t)),x_{i,k}(t)-x^*\rangle+2\alpha_i^2\|\nabla f_i(x_{i,k}(t))\|^2\\ &-2\alpha_i\langle g_i(x_{i,k}(t))-\nabla f_i(x_{i,k}(t)),x_{i,k}(t)-x^*\rangle+2\alpha_i^2\|g_i(x_{i,k}(t))-\nabla f_i(x_{i,k}(t))\|^2.\end{aligned}\tag{57}$$

Using the convexity of $f_i(x)$, we arrive at

$$-2\alpha_i\langle\nabla f_i(x_{i,k}(t)),x_{i,k}(t)-x^*\rangle\le 2\alpha_i\{f_i(x^*)-f_i(x_{i,k}(t))\}.\tag{58}$$

Assumption 3 implies

$$\|\nabla f_i(x^*)\|=0$$

for any $i\in\mathcal{S}$ and $x^*\in\mathcal{X}^*$. Thus, combining the preceding relation with Lemma 3, we can obtain

$$\|\nabla f_i(x_{i,k}(t))\|^2\le 2L_i\{f_i(x_{i,k}(t))-f_i(x^*)\}.\tag{59}$$

Combining (57), (58), and (59), we arrive at

$$\begin{aligned}&\|x_{i,k+1}(t)-x^*\|^2\\ \le&\|x_{i,k}(t)-x^*\|^2+(2\alpha_i-2L_i\alpha_i^2)\{f_i(x^*)-f_i(x_{i,k}(t))\}\\ &-2\alpha_i\langle g_i(x_{i,k}(t))-\nabla f_i(x_{i,k}(t)),x_{i,k}(t)-x^*\rangle+2\alpha_i^2\|g_i(x_{i,k}(t))-\nabla f_i(x_{i,k}(t))\|^2.\end{aligned}\tag{60}$$

Using (60) and the property of the stochastic gradient, we have

$$\mathbb{E}[\|x_{i,k+1}(t)-x^*\|^2]\le\mathbb{E}[\|x_{i,k}(t)-x^*\|^2]+(2\alpha_i-2L_i\alpha_i^2)\{f_i(x^*)-\mathbb{E}[f_i(x_{i,k}(t))]\}+2\alpha_i^2\sigma^2.\tag{61}$$

Note that the following inequality always holds:

$$\mathbb{E}[\|\bar{x}(t+1)-x^*\|^2]\le\frac{1}{N}\sum_{i=1}^N\mathbb{E}[\|x_{i,\tau}(t)-x^*\|^2].$$

Using the update rule in Algorithm 1 and (60), we arrive at

$$\begin{aligned}&\mathbb{E}[\|\bar{x}(t+1)-x^*\|^2]\\ \le&\frac{1}{N}\sum_{i=1}^N(2\alpha_i-2L_i\alpha_i^2)\sum_{k=0}^{\tau-1}\Big\{f_i(x^*)-\mathbb{E}[f_i(x_{i,k}(t))]\Big\}+\mathbb{E}[\|\bar{x}(t)-x^*\|^2]+\frac{1}{N}\sum_{i=1}^N2\tau\alpha_i^2\sigma^2.\end{aligned}\tag{62}$$

Under the stepsize setting (4), we can obtain $2\alpha_i-2L_i\alpha_i^2>0$ for any $i\in\mathcal{S}$.

Moreover, Assumption 3 ensures $f_i(x^*)-\mathbb{E}[f_i(x_{i,k}(t))]\le 0$ for any $i\in\mathcal{S}$ and $k=0,1,\cdots,\tau-1$.

Thus, from (62), we have

$$\begin{aligned}\mathbb{E}[\|\bar{x}(t+1)-x^*\|^2]\le&\min_{1\le i\le N}\Big\{2\alpha_i-2L_i\alpha_i^2\Big\}\Big\{f(x^*)-\mathbb{E}[f(\bar{x}(t))]\Big\}\\ &+\frac{1}{N}\sum_{i=1}^N2\tau\alpha_i^2\sigma^2+\mathbb{E}[\|\bar{x}(t)-x^*\|^2],\end{aligned}\tag{63}$$

which further implies

$$\mathbb{E}[f(\bar{x}(t))] - f(x^*) \leq \frac{\mathbb{E}[\|\bar{x}(t) - x^*\|^2] - \mathbb{E}[\|\bar{x}(t+1) - x^*\|^2]}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}} + \frac{\frac{1}{N}\sum_{i=1}^N 2\tau\alpha_i^2}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}}\sigma^2.$$

Thus, for any $T \geq 1$, we have

$$\sum_{t=1}^{T}\Big\{\mathbb{E}[f(\bar{x}(t))] - f(x^*)\Big\} \leq \frac{\|\bar{x}(1) - x^*\|^2}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}} + \frac{\frac{1}{N}\sum_{i=1}^N 2\tau\alpha_i^2 T}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}}\sigma^2. \tag{64}$$

From (64), for any $T \geq 1$, we obtain

$$\mathbb{E}\Big[f\Big(\frac{1}{T}\sum_{t=1}^{T}\bar{x}(t)\Big)\Big] - f(x^*) \leq \frac{\|\bar{x}(1) - x^*\|^2}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}T} + \frac{\frac{1}{N}\sum_{i=1}^N 2\tau\alpha_i^2}{\min_{1\leq i\leq N}\{2\alpha_i - 2L_i\alpha_i^2\}}\sigma^2,$$

which completes the proof.

□

H Proof of Theorem 5

We use $g_i(x)$ to represent the unbiased estimate of the gradient $\nabla f_i(x)$. We need the following Lemma 6 to prove Corollary 5.

Lemma 6. *If the stepsize satisfies* $0 < \alpha < \min_{\{1\leq j\leq N\}}\{\frac{1}{L_j}\}$*, we have*

$$\mathbb{E}[\|x_{i,h}(t) - \bar{x}(t)\|^2] \leq 12\tau^2 L\alpha^2\mathbb{E}[f(\bar{x}(t)) - f(x^*)] + 27\tau\alpha^2\sigma^2$$

for $0 \leq h < \tau$ *and* $i \in \mathcal{S}$*.*

Proof. From Algorithm 2, we can obtain

$$x_{i,k+1}(t) = x_{i,k}(t) - \alpha\Big\{\frac{1}{N}\sum_{j=1}^{N} g_j(\bar{x}(t)) - g_i(\bar{x}(t)) + g_i(x_{i,k}(t)))\Big\}.$$

Thus, we have

$$\begin{aligned}
&x_{i,k+1}(t) - \bar{x}(t)\\
=&x_{i,k}(t) - \bar{x}(t) - \alpha\Big\{\nabla f(\bar{x}(t)) - \nabla f_i(\bar{x}(t)) + \nabla f_i(x_{i,k}(t))\Big\}\\
&- \alpha\Big\{\frac{1}{N}\sum_{j=1}^{N} g_j(\bar{x}(t)) - \frac{1}{N}\sum_{j=1}^{N}\nabla f_j(\bar{x}(t)) + \nabla f_i(\bar{x}(t)) - g_i(\bar{x}(t)) + g_i(x_{i,k}(t)) - \nabla f_i(x_{i,k}(t))\Big\}.
\end{aligned}$$

From the property of stochastic gradients, we have

$$\begin{aligned}
&\mathbb{E}\Big[\|x_{i,k+1}(t) - \bar{x}(t)\|^2\Big]\\
=&\mathbb{E}\Big[\Big\|x_{i,k}(t) - \bar{x}(t) - \alpha\Big(\nabla f(\bar{x}(t)) - \nabla f_i(\bar{x}(t)) + \nabla f_i(x_{i,k}(t))\Big)\Big\|^2\Big]\\
&+ \alpha^2\mathbb{E}\Big[\Big\|\frac{1}{N}\sum_{j=1}^{N} g_j(\bar{x}(t)) - \frac{1}{N}\sum_{j=1}^{N}\nabla f_j(\bar{x}(t)) + \nabla f_i(\bar{x}(t)) - g_i(\bar{x}(t)) + g_j(x_{i,k}(t)) - \nabla f_i(x_{i,k}(t))\Big\|^2\Big].
\end{aligned} \tag{65}$$

For the first term of the right hand side of (65), we can obtain

$$\begin{aligned}
&\mathbb{E}\Big[\Big\|x_{i,k}(t) - \bar{x}(t) - \alpha\Big(\nabla f(\bar{x}(t)) - \nabla f_i(\bar{x}(t)) + \nabla f_i(x_{i,k}(t))\Big)\Big\|^2\Big]\\
\leq&\Big(1 + \frac{1}{\tau}\Big)\mathbb{E}\Big[\Big\|x_{i,k}(t) - \bar{x}(t) - \alpha(\nabla f_i(x_{i,k}(t)) - \nabla f_i(\bar{x}(t)))\Big\|^2\Big] + (1+\tau)\alpha^2\mathbb{E}[\|\nabla f(\bar{x}(t))\|^2].
\end{aligned}$$

From Lemma 4, we can obtain

$$\begin{aligned}&\mathbb{E}\Big[\Big\|x_{i,k}(t)-\bar{x}(t)-\alpha\Big(\nabla f(\bar{x}(t))-\nabla f_i(\bar{x}(t))+\nabla f_i(x_{i,k}(t))\Big)\Big\|^2\Big]\\ \leq&\Big(1+\frac{1}{\tau}\Big)\mathbb{E}[\|x_{i,k}(t)-\bar{x}(t)\|^2]+(1+\tau)\alpha^2\mathbb{E}[\|\nabla f(\bar{x}(t))\|^2],\end{aligned} \tag{66}$$

if the stepsize satisfies $0<\alpha L_i\leq 1$ for any $1\leq i\leq N$.

For the second term of the right hand side of (65), we have

$$\begin{aligned}&\mathbb{E}\Big[\Big\|\frac{1}{N}\sum_{j=1}^N g_j(\bar{x}(t))-\frac{1}{N}\sum_{j=1}^N\nabla f_j(\bar{x}(t))+\nabla f_i(\bar{x}(t))-g_i(\bar{x}(t))+g_i(x_{i,k}(t))-\nabla f_i(x_{i,k}(t))\Big\|^2\Big]\\ \leq&\frac{3}{N}\sum_{j=1}^N\mathbb{E}[\|g_j(\bar{x}(t))-\nabla f_j(\bar{x}(t))\|^2]+3\mathbb{E}[\|\nabla f_i(\bar{x}(t))-g_i(\bar{x}(t))\|^2]+3\mathbb{E}[\|g_i(x_{i,k}(t))-\nabla f_i(x_{i,k}(t))\|^2]\end{aligned}$$

for any $i\in\mathcal{S}$.

Using the properties of stochastic gradients, we have

$$\mathbb{E}\Big[\Big\|\frac{1}{N}\sum_{j=1}^N g_j(\bar{x}(t))-\frac{1}{N}\sum_{j=1}^N\nabla f_j(\bar{x}(t))+\nabla f_i(\bar{x}(t))-g_i(\bar{x}(t))+g_i(x_{i,k}(t))-\nabla f_i(x_{i,k}(t))\Big\|^2\Big]\leq 9\sigma^2 \tag{67}$$

for any $i\in\mathcal{S}$.

Combining (65), (66), and (67), we arrive at

$$\mathbb{E}[\|x_{i,k+1}(t)-\bar{x}(t)\|^2]\leq\Big(1+\frac{1}{\tau}\Big)\mathbb{E}[\|x_{i,k}(t)-\bar{x}(t)\|^2]+(1+\tau)\alpha^2\mathbb{E}[\|\nabla f(\bar{x}(t))\|^2]+9\alpha^2\sigma^2.$$

Using induction, we obtain the following relation holding for any $0\leq k<\tau$:

$$\mathbb{E}\Big[\|x_{i,k}(t)-\bar{x}(t)\|^2\Big]\leq\Big\{(1+\tau)\alpha^2\mathbb{E}[\|\nabla f(\bar{x}(t))\|^2]+9\alpha^2\sigma^2\Big\}\sum_{h=0}^{\tau-1}\Big(1+\frac{1}{\tau}\Big)^h,$$

which further implies

$$\mathbb{E}\Big[\|x_{i,k}(t)-\bar{x}(t)\|^2\Big]\leq\Big\{(1+\tau)\alpha^2\mathbb{E}[\|\nabla f(\bar{x}(t))\|^2]+9\alpha^2\sigma^2\Big\}\frac{(1+\frac{1}{\tau})^\tau-1}{(1+\frac{1}{\tau})-1}.$$

Using the relation $\|\nabla f(\bar{x}(t))\|^2\leq 2L(f(\bar{x}(t))-f(x^*))$ from Assumption 1 and the convex property of $f_i(x)$, we can obtain

$$\mathbb{E}\Big[\|x_{i,k}(t)-\bar{x}(t)\|^2\Big]\leq 12\tau^2L\alpha^2\mathbb{E}[f(\bar{x}(t))-f(x^*)]+27\tau\alpha^2\sigma^2,$$

which completes the proof. □

Next we proceed to prove Corollary 5. From the update rule in Algorithm 2, we have

$$\bar{x}(t+1)=\bar{x}(t)-\frac{\alpha}{N}\sum_{j=1}^N\sum_{h=0}^{\tau-1}g_j(x_{j,h}(t)),$$

which further implies

$$\begin{aligned}&\|\bar{x}(t+1)-x^*\|^2-\|\bar{x}(t)-x^*\|^2\\ =&-2\alpha\Big\langle\frac{1}{N}\sum_{j=1}^N\sum_{h=0}^{\tau-1}g_j(x_{j,h}(t)),x_i(k\tau)-x^*\Big\rangle+\alpha^2\Big\|\frac{1}{N}\sum_{j=1}^N\sum_{h=0}^{\tau-1}g_j(x_{j,h}(t))\Big\|^2.\end{aligned} \tag{68}$$

For the term $-2\alpha\Big\langle \frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1} g_j(x_{j,h}(t)), x_i(k\tau)-x^*\Big\rangle$ in (68), we have

$$\begin{aligned}
&-2\alpha\mathbb{E}\Big[\Big\langle \frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1} g_j(x_{j,h}(t)), x_i(k\tau)-x^*\Big\rangle\Big]\\
=&\frac{2\alpha}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1}\mathbb{E}\Big[\langle x^*-x_{j,h}(t), \nabla f_j(x_{j,h}(t))\rangle\Big]+\frac{2\alpha}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1}\mathbb{E}\Big[\langle x_{j,h}(t)-\bar{x}(t), \nabla f_j(x_{j,h}(t))\rangle\Big].
\end{aligned}$$

Using the convexity of $f_i(x)$ and Assumption 1, we arrive at

$$\begin{aligned}
&-2\alpha\mathbb{E}\Big[\Big\langle \frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1} g_j(x_{j,h}(t)), x_i(k\tau)-x^*\Big\rangle\Big]\\
\le&\frac{2\alpha}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1}\mathbb{E}\Big[f_j(x^*)-f_j(x_{j,h}(t))\Big]+\frac{2\alpha}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1}\mathbb{E}\Big[f_j(x_{j,h}(t))-f_j(\bar{x}(t))+\frac{L}{2}\|x_{j,h}(t)-\bar{x}(t)\|^2\Big],
\end{aligned}$$

which further implies

$$\begin{aligned}
&-2\alpha\mathbb{E}\Big[\Big\langle \frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1} g_j(x_{j,h}(t)), x_i(k\tau)-x^*\Big\rangle\Big]\\
\le&2\alpha\tau\mathbb{E}\Big[f(x^*)-f(\bar{x}(t))\Big]+\frac{\alpha L}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1}\mathbb{E}\Big[\|x_{j,h}(t)-\bar{x}(t)\|^2\Big].
\end{aligned}\tag{69}$$

From Lemma 6, we have

$$\mathbb{E}[\|x_{j,h}(t)-\bar{x}(t)\|^2]\le 12\tau^2L\alpha^2\mathbb{E}[f(\bar{x}(t))-f(x^*)]+27\tau\alpha^2\sigma^2\tag{70}$$

for $1\le h<\tau$.

Combining (69) and (70), yields

$$\begin{aligned}
&-2\alpha\mathbb{E}\Big[\Big\langle \frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1} g_j(x_{j,h}(t)), \bar{x}(t)-x^*\Big\rangle\Big]\\
\le&2\alpha\tau\mathbb{E}[f(x^*)-f(\bar{x}(t))]+12\tau^3L^2\alpha^3\mathbb{E}[f(\bar{x}(t))-f(x^*)]+27\tau^2L\alpha^3\sigma^2.
\end{aligned}$$

When the stepsize satisfies $0<6\tau\alpha L\le 1$, we have

$$\begin{aligned}
&-2\alpha\mathbb{E}\Big[\langle \frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1} g_j(x_{j,h}(t)), \bar{x}(t)-x^*\rangle\Big]\\
\le&2\alpha\tau\mathbb{E}[f(x^*)-f(\bar{x}(t))]+2\tau^2L\alpha^2\mathbb{E}[f(\bar{x}(t))-f(x^*)]+9\tau^2\alpha^2\sigma^2.
\end{aligned}\tag{71}$$

For the term $\alpha^2\|\frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1} g_j(x_{j,h}(t))\|^2$ in (68), we have

$$\begin{aligned}
&\alpha^2\Big\|\frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1} g_j(x_{j,h}(t))\Big\|^2\\
\le&2\alpha^2\Big\|\frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1}\Big\{g_j(x_{j,h}(t))-g_j(\bar{x}(t))\Big\}\Big\|^2+2\alpha^2\Big\|\frac{1}{N}\sum_{j=1}^{N}\sum_{h=0}^{\tau-1} g_j(\bar{x}(t))\Big\|^2.
\end{aligned}\tag{72}$$

Using the smoothness conditon in Assumption 1 and the inequality $\| \sum_{i=1}^{k} a_i\|^2 \leq k \sum_{i=1}^{k} \|a_i\|^2$, we have

$$
\begin{aligned}
&\alpha^2 \Big\| \frac{1}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} \Big\{ g_j(x_{j,h}(t)) - g_j(\bar{x}(t)) \Big\} \Big\|^2 \\
\leq & \frac{3\tau L^2 \alpha^2}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} \|x_{j,h}(t) - \bar{x}(t)\|^2 + \frac{3\tau\alpha^2}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} \|g_j(x_{j,h}(t)) - \nabla f_j(x_{j,h}(t))\|^2 \\
&+ \frac{3\tau\alpha^2}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} \|\nabla f_j(\bar{x}(t)) - g_j(\bar{x}(t))\|^2. \qquad (73)
\end{aligned}
$$

From (73) and the property of stochastic gradient, we have

$$
2\alpha^2 \mathbb{E}\Big[\Big\| \frac{1}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} \{ g_j(x_{j,h}(t)) - g_j(\bar{x}(t)) \} \Big\|^2 \Big] \leq \frac{6\tau\alpha^2 L^2}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} \mathbb{E}[\|x_{j,h}(t) - \bar{x}(t)\|^2] + 12\alpha^2\tau^2\sigma^2. \qquad (74)
$$

Plugging the inequality in Lemma 6 into (74) leads to

$$
\begin{aligned}
&2\alpha^2 \mathbb{E}\Big[\Big\| \frac{1}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} \{ g_j(x_{j,h}(t)) - g_j(\bar{x}(t)) \} \Big\|^2 \Big] \\
\leq & 72\tau^4 L^3 \alpha^4 \mathbb{E}[f(\bar{x}(t)) - f(x^*)] + 162\tau^3 L^2 \alpha^4 \sigma^2 + 12\alpha^2\tau^2\sigma^2. \qquad (75)
\end{aligned}
$$

For the term $\| \frac{1}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} g_j(x_j(k\tau))\|^2$ in (72), we have

$$
2\alpha^2 \Big\| \frac{1}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} g_j(\bar{x}(t)) \Big\|^2 \leq \frac{4\alpha^2\tau^2}{N} \sum_{j=1}^{N} \|g_j(\bar{x}(t)) - \nabla f_j(\bar{x}(t))\|^2 + 4\alpha^2\tau^2 \|\nabla f(\bar{x}(t))\|^2.
$$

Using Lemma 6 and the property of stochastic gradients, we have

$$
2\alpha^2 \mathbb{E}\Big[\Big\| \frac{1}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} g_j(\bar{x}(t)) \Big\|^2 \Big] \leq 4\alpha^2\tau^2\sigma^2 + 8\alpha^2\tau^2 L \mathbb{E}[f(\bar{x}(t)) - f(x^*)]. \qquad (76)
$$

Combining (72), (75), and (76), we have

$$
\alpha^2 \mathbb{E}\Big[\Big\| \frac{1}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} g_j(x_{j,h}(t)) \Big\|^2 \Big] \leq (72\tau^4 L^3 \alpha^4 + 8\alpha^2\tau^2 L) \mathbb{E}[f(\bar{x}(t)) - f(x^*)] + 162\tau^3 L^2 \alpha^4 \sigma^2 + 16\alpha^2\tau^2\sigma^2.
$$

When the stepsize satisfies $0 < 6\tau\alpha L \leq 1$, we have

$$
\alpha^2 \mathbb{E}\Big[\Big\| \frac{1}{N} \sum_{j=1}^{N} \sum_{h=0}^{\tau-1} g_j(x_{j,h}(t)) \Big\|^2 \Big] \leq 10\tau^2 L \alpha^2 \mathbb{E}[f(\bar{x}(t)) - f(x^*)] + 25\alpha^2\tau^2\sigma^2. \qquad (77)
$$

Combining (68), (71), and (77), we have

$$
\mathbb{E}[\|\bar{x}(t+1) - x^*\|^2] - \mathbb{E}[\|\bar{x}(t) - x^*\|^2] \leq (2\alpha\tau - 12\tau^2 L\alpha^2) \mathbb{E}[f(x^*) - f(\bar{x}(t))] + 34\tau^2\alpha^2\sigma^2. \qquad (78)
$$

When the stepsize satisfies $0 < \alpha \leq \frac{1}{12\tau L}$, we have $\alpha\tau - 12\tau^2 L\alpha^2 \geq 0$. Plugging the preceding inequality into (78) yileds

$$
\frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[f(\bar{x}(t)) - f(x^*)] \leq \frac{\|x_i(1) - x^*\|^2}{\alpha\tau T} + \frac{34\tau^2\alpha^2}{\alpha\tau}\sigma^2.
$$

Moreover, using the stepsize condition $0 < \alpha \leq \min_{1 \leq j \leq N}\{\frac{1}{L_j}, \frac{1}{12\tau L}\}$ and the convexity of $f(x)$, we can obtain

$$\mathbb{E}\Big[f\Big(\frac{1}{T}\sum_{t=1}^{T}\bar{x}(t)\Big)\Big] - f(x^*) \leq \frac{\|x(1) - x^*\|^2}{\alpha\tau T} + 34\tau\alpha\sigma^2$$

for any $i \in \mathcal{S}$, which completes the proof.