FedSKETCH: Communication-Efficient and Private Federated Learning via Sketching

Anonymous Author
Anonymous Institution

Abstract

Communication complexity and privacy are the two key challenges in Federated Learning where the goal is to perform a distributed learning through a large volume of devices. In this work, to address jointly both challenges, we introduce `FedSKETCH` and `FedSKETCHGATE` algorithms, which are, respectively, intended to be used for homogeneous and heterogeneous data distribution settings. The key idea is to compress the accumulation of local gradients using count sketch, therefore, the server does not have access to the gradients themselves thus providing privacy. Furthermore, due to the lower dimension of sketches, our method exhibits communication-efficiency property as well. We provide sharp convergence guarantees of our schemes and back up the theory with experimental results.

1 Introduction

Federated Learning (FL) is a recently emerging setting for distributed large scale machine learning problems. In FL, data is distributed across devices [38, 26] and due to privacy concerns, users are only allowed to communicate with the parameter server. Formally, the optimization problem across p distributed devices is defined as follows:

$$\min_{\boldsymbol{x}\in\mathbb{R}^d,\, \sum_{j=1}^{p} q_j=1} f(\boldsymbol{x}) \triangleq \left[\sum_{j=1}^{p} q_j F_j(\boldsymbol{x})\right], \qquad (1)$$

where $F_j(\boldsymbol{x}) = \mathbb{E}_{\xi\in\mathcal{D}_j}\left[L_j\left(\boldsymbol{x},\xi\right)\right]$ is the local cost function at device j, $q_j \triangleq \frac{n_j}{n}$, n_j is the number of data shards at device j and $n = \sum_{j=1}^{p} n_j$ the total number of data samples. ξ is a random variable distributed according to probability distribution $\mathcal{D}_j$, and L_j is a loss function that measures the performance of model $\boldsymbol{x}$ at device j. We note that, while for the homogeneous setting we assume $\mathcal{D}_j$ for $1 \leq j \leq p$ have the same distribution across devices and $L_1 = L_2 = \ldots = L_p$, in the heterogeneous setting these data distributions and loss functions L_j can be different from a device to another. The parameter server orchestrates optimization among devices by aggregating gradient-related information of devices and broadcasts the average of received vectors. Besides, moving data across the devices during the learning a global model can be impractical and could violate the privacy of user devices [7, 39]. There are several challenges that need to be addressed in FL in order to efficiently learn a global model that performs well in average for all devices. First, the *communication-efficiency*, as there could be a million of devices communicating with the server, thus incurring huge communication overhead. One approach to deal with communication cost is the idea of *local SGD with periodic averaging* [57, 49, 55, 51] which instead of taking the average at each iteration, like baseline SGD [6], this average is taken periodically after few local updates, see [35]. Local SGD has been proposed in [38, 26] under the Federated Learning setting and its convergence analysis is studied in [57, 55, 49, 51]. Its convergence analysis is improved in [14, 15, 3, 17, 24, 48] mainly for homogeneous setting. It is further extended to heterogeneous setting, wherein studied under the title of *Federated Learning*, with improved rates in [54, 33, 46, 34, 17, 23]. Additionally, similar Local SGD with adaptive gradient methods can be found in [42, 9]. The second approach to deal with communication cost aims at reducing the size of communicated message per communication round, such as local gradients quantization [1, 4, 50, 52, 53] or sparsification [2, 36, 47, 48].The second challenge is *data heterogeneity*. Since the data in in each device is generated locally in the setting of FL, it may be distributed according to various prob-

Preliminary work. Under review by AISTATS 2021. Do not distribute.

ability distributions and can lead to poor convergence error in practice [31, 34]. In [34, 23, 19, 16] the effect of data heterogeneity is mitigated by exploiting variance reduction or gradient tracking techniques. The last, yet important, issue is *device privacy* [12, 18]. Solving the privacy issue has been widely performed by injecting an additional layer of random noise in order to respect differential-privacy property of the method [39] or using cryptography-based approaches under secure multi-party computation framework [5]. Recent promising approach with a potential to tackle all major issues in FL setting is based on sketching algorithms [8, 10, 25, 29]. For instance, [21] develops a distributed SGD algorithm using sketching along with its convergence analysis in the homogeneous data distribution setting. Focusing on privacy, in [30] derives a single framework in order to tackle these issues jointly and introduces `DiffSketch` algorithm, based on the Count Sketch operator. Yet, [30] does not provide the convergence analysis for the `DiffSketch` in Federated setting, and additionally the estimation error of the `DiffSketch` is relatively higher than the sketching scheme in [21] which may end up in poor convergence. [45] consider using sketching technique in heterogeneous setting from a communication-efficiency perspective. The proposed sketching schemes in [21, 45] are based on a deterministic procedure which requires having access to the exact values of the gradient-related information, thus not meeting the crucial privacy-preserving criteria. Aiming at jointly tackling communication efficiency and data privacy, [21] develop `Sketched-SGD` that leverages sketches of full gradients in a distributed setting while training a global model using SGD [44, 6], and establish a communication complexity of order $\mathcal{O}(\log(d))$ (per round) where d is the dimension of the vector of parameters. Compression methods such as quantized gradients are developed in [1, 36, 47, 19]. Yet, their dependence on the number of devices p makes them harder to be used practice. Recently, [20] jointly exploits variance reduction technique with compression in distributed optimization. In this work, we provide a thorough convergence analysis for Federated Learning using sketching for both homogeneous and heterogeneous settings. Additionally, all of our sketching algorithms including a novel scheme, do not require exact values of the gradient, hence are privacy preserving. The main contributions are summarized as follows:

- We provide a new algorithm – `HEAPRIX` – and theoretically show that it reduces the cost of communication between devices and server, is unbiased and does not require exchanging exact values of gradients, ensuring privacy.
- We develop a general algorithm for communication-efficient and privacy preserving FL based on `HEAPRIX`, namely `FedSKETCH` and `FedSKETCHGATE`, derived under both data distribution settings.
- We establish non asymptotic convergence bounds for convex, Polyak-Łojasiewicz (generalization of strongly-convex) and non-convex functions in Theorem 1 and Theorem 2 for respectively homogeneous and heterogeneous case, and highlight an improvement in the number of iteration to achieve a stationary point.
- We illustrate the benefits of `FedSKETCH` and `FedSKETCHGATE` over baseline methods through a set of experiments. Numerical experiments show the advantages of the `FedSKETCH-HEAPRIX` algorithm that achieves comparable test accuracy as Federated SGD (`FedSGD`) while compressing the information exchanged between devices and server.

Notation: We denote by R and B the number of communication rounds and bits per round per device respectively. The count sketch of any vector $\boldsymbol{x}$ is denoted by $\mathbf{S}(\boldsymbol{x})$. We also denote $[p] = \{1, \ldots, p\}$.

2 Compressions using Count Sketch

A common sketching method to tackle (1), namely `Count Sketch` [8], is described Algorithm 1.

Algorithm 1 `CS` [25]: Count Sketch to compress $\boldsymbol{x} \in \mathbb{R}^d$.

1: **Inputs:** $\boldsymbol{x} \in \mathbb{R}^d, t, k, \mathbf{S}_{m\times t}, h_j(1 \le i \le t), sign_j(1 \le i \le t)$
2: **Compress vector** $\boldsymbol{x} \in \mathbb{R}^d$ **into** $\mathbf{S}(\boldsymbol{x})$**:**
3: **for** $\boldsymbol{x}_i \in \boldsymbol{x}$ **do**
4:   **for** $j = 1, \cdots, t$ **do**
5:     $\mathbf{S}[j][h_j(i)] = \mathbf{S}[j-1][h_{j-1}(i)] + \text{sign}_j(i).\boldsymbol{x}_i$
6:   **end for**
7: **end for**
8: **return** $\mathbf{S}_{m\times t}(\boldsymbol{x})$

Count Sketch is using two sets of functions that encode any input vector $\boldsymbol{x}$ **into a hash table** $\boldsymbol{S}_{m\times t}(\boldsymbol{x})$. Pairwise independent hash functions $\{h_{j,1\le j\le t} : [d] \to m\}$ are used along with another set of pairwise independent sign hash functions

$\{\text{sign}_{j,1\le j\le t} : [d] \to \{+1,-1\}\}$ to map entries of $\boldsymbol{x}$ ($\boldsymbol{x}_i$, $1 \le i \le d$) into t different columns of $\mathbf{S}_{m\times t}$.

2.1 Sketching based Unbiased Compressor

We define an unbiased compressor as follows:

Definition 1 (Unbiased compressor). *A randomized function, $C : \mathbb{R}^d \to \mathbb{R}^d$ is called an unbiased compression operator with $\Delta \ge 1$, if we have*

$$\mathbb{E}\left[C(\boldsymbol{x})\right] = \boldsymbol{x} \quad \textit{and} \quad \mathbb{E}\left[\|C(\boldsymbol{x})\|_2^2\right] \le \Delta \|\boldsymbol{x}\|_2^2 \ .$$

We denote this class of compressors by $C \in \mathbb{U}(\Delta)$.

This definition leads to the following property

$$\mathbb{E}\left[\|\mathrm{C}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2\right] \le (\Delta - 1)\|\boldsymbol{x}\|_2^2 \ .$$

Remark 1. *Note that if $\Delta = 1$ then our algorithm reduces to the case of no compression. This property allows us to control the noise of the compression.*

An instance of such unbiased compressor is `PRIVIX` which obtains an estimate of input $\boldsymbol{x}$ from a count sketch noted $\boldsymbol{S}(\boldsymbol{x})$. In this algorithm, to query the quantity x_i, the $i-th$ element of the vector, we compute the median of t approximated values specified by the indices of $h_j(i)$ for $1 \le j \le t$, see [30] or Algorithm 6 in the appendix. For the purpose of our proof, we state the following crucial properties of the count sketch.

Property 1 ([30]). *For any $\mathbf{x} \in \mathbb{R}^d$:*

Unbiased estimation*: As in [30], we have:*

$$\mathbb{E}_{\mathbf{S}}\left[\mathit{PRIVIX}[\mathbf{S}(\mathbf{x})]\right] = \mathbf{x} \, .$$

Bounded variance: *if $m = \mathcal{O}\left(\frac{e}{\mu^2}\right)$, $t = \mathcal{O}\left(\ln\left(\frac{d}{\delta}\right)\right)$:*

$$\mathbb{E}_{\mathbf{S}}\left[\|\mathit{PRIVIX}[\mathbf{S}(\mathbf{x})] - \mathbf{x}\|_2^2\right] \le \mu^2 d \|\mathbf{x}\|_2^2 \quad w.p.\, 1-\delta \, .$$

Thus, `PRIVIX` $\in \mathbb{U}(1+\mu^2 d)$ with probability $1-\delta$. We note that $\Delta = 1 + \mu^2 d$ implies that if $m \to d$, $\Delta \to 1 + 1 = 2$, which means that the case of no compression is not covered.

Remark 2 (Differentially-privacy property). *As in [30], if the data is normally distributed, PRIVIX provides differential privacy.*

2.2 Sketching based Biased Compressor

We define a biased compressor as follows:

Definition 2 (Biased compressor). *A (randomized) function, $C : \mathbb{R}^d \to \mathbb{R}^d$ is called a compression operator with $\alpha > 0$ and $\Delta \ge 1$, if we have*

$$\mathbb{E}\left[\|\alpha\boldsymbol{x} - C(\boldsymbol{x})\|_2^2\right] \le \left(1 - \frac{1}{\Delta}\right)\|\boldsymbol{x}\|_2^2 \ ,$$

then, any biased compression operator C is indicated by $C \in \mathbb{C}(\Delta, \alpha)$.

The following Lemma links these two definitions:

Lemma 1 ([20]). *We have $\mathbb{U}(\Delta) \subset \mathbb{C}(\Delta, \alpha)$.*

An instance of a biased compression method based on sketching is given in Algorithm 2.

Algorithm 2 `HEAVYMIX`

1: **Inputs:** $\mathbf{S}(\mathbf{g})$; parameter-m
2: **Query the vector $\tilde{\mathbf{g}} \in \mathbb{R}^d$ from $\mathbf{S}(\mathbf{g})$:**
3: Query $\hat{\ell}_2^2 = (1 \pm 0.5)\|\mathbf{g}\|^2$ from sketch $\mathbf{S}(\mathbf{g})$
4: $\forall j$ query $\hat{\mathbf{g}}_j^2 = \hat{\mathbf{g}}_j^2 \pm \frac{1}{2m}\|\mathbf{g}\|^2$ from sketch $\mathbf{S}_{\mathbf{g}}$
5: $H = \{j | \hat{\mathbf{g}}_j \ge \frac{\hat{\ell}_2^2}{m}\}$ and $NH = \{j | \hat{\mathbf{g}}_j < \frac{\hat{\ell}_2^2}{m}\}$
6: $\text{Top}_m = H \cup rand_\ell(NH)$, where $\ell = m - |H|$
7: Get exact values of Top_m
8: **Output:** $\tilde{\mathbf{g}} : \forall j \in \text{Top}_m : \tilde{\mathbf{g}}_i = \mathbf{g}_i$ else $\mathbf{g}_i = 0$

Lemma 2 ([21]). *HEAVYMIX, with sketch size $\Theta\left(m \log\left(\frac{d}{\delta}\right)\right)$ is a biased compressor with $\alpha = 1$ and $\Delta = d/m$ with probability $\ge 1-\delta$. In other words, with probability $1-\delta$, HEAVYMIX $\in C(\frac{d}{m}, 1)$.*

We note that Algorithm 2 is a variation of the sketching algorithm developed in [21] with distinction that `HEAVYMIX` does not require a second round of communication to obtain the exact values of top_m. Additionally, while a sketching algorithm based on `HEAVYMIX` has smaller estimation error compared to `PRIVIX`, it requires having access to the exact values of top_m, therefore not benefiting from differentially privacy similar to `PRIVIX`. In the following we introduce our sketching scheme which enjoys from privacy property as well as smaller estimation error.

2.3 Sketching based Induced Compressor

The following Lemma from [20] shows that we can convert the biased compressor into an unbiased one:

Lemma 3 (Induced Compressor [20]). *For $C_1 \in \mathbb{C}(\Delta_1)$ with $\alpha = 1$, choose $C_2 \in \mathbb{U}(\Delta_2)$ and define the induced compressor with*

$$C(\mathbf{x}) = C_1(\mathbf{x}) + C_2\left(x - C_1(\mathbf{x})\right) \ ,$$

then, $C \in \mathbb{U}(\Delta)$ with $\Delta = \Delta_2 + \frac{1-\Delta_2}{\Delta_1}$.

We note that if $\Delta_2 \geq 1$ and $\Delta_1 \leq 1$, we have $\Delta = \Delta_2 + \frac{1-\Delta_2}{\Delta_1} \leq \Delta_2$. Using this concept of induced compressor, we introduce HEAPRIX in Algorithm 3. Here, the reconstruction of input $\mathbf{x}$ is done using hash table $\mathbf{S}$ and $\mathbf{x}$ similar to PRIVIX and HEAVYMIX.

Algorithm 3 HEAPRIX

1: **Inputs:** $\boldsymbol{x} \in \mathbb{R}^d, t, m, \mathbf{S}_{m\times t}, h_j(1 \leq i \leq t), sign_j(1 \leq i \leq t)$, parameter-$m$
2: **Approximate** $\mathbf{S}(x)$ **using** HEAVYMIX
3: **Approximate** $\mathbf{S}(x - \texttt{HEAVYMIX}[\mathbf{S}(x)])$ **using** PRIVIX
4: **Output:** $\texttt{HEAVYMIX}[\mathbf{S}(\mathbf{x})] + \texttt{PRIVIX}[\mathbf{S}(\mathbf{x} - \texttt{HEAVYMIX}[\mathbf{S}(\mathbf{x})])]$

Corollary 1. *Based on Lemma 3 and using Algorithm 3, we have $C(x) \in \mathbb{U}(\mu^2 d)$.*

Corollary 1 states that, unlike PRIVIX, HEAPRIX compression noise can be made as small as possible using large size of hash table.

Remark 3. *If $m \to d$, then $C(x) \to x$, meaning that the algorithm convergence can be improved by decreasing the noise of compression m.*

In the following we define two general frameworks for different sketching algorithms for homogeneous and heterogeneous data distributions.

3 Algorithms for Homogeneous and Heterogeneous Settings

We introduce two new algorithms for the iid setting.

3.1 Homogeneous Setting

The proposed algorithms for Federated Learning leverage sketching techniques to reduce communication costs. The main difference between our FedSKETCH and the DiffSketch algorithm in [30] is that we use distinct local and global learning rates. Additionally, unlike [30], we do not add local Gaussian noise to ensure privacy. In FedSKETCH, the number of local updates, between two consecutive communication rounds, at device j is denoted by τ. Unlike [16], server node does not store any global model, instead device j has two models, $\boldsymbol{x}^{(r)}$ and $\boldsymbol{x}_j^{(\ell,r)}$, respectively local and global models. We develop FedSKETCH in Algorithm 4. A variant of this algorithm which uses a different compression scheme, called HEAPRIX is also described in Algorithm 4. We note that for this variant, we need to have an additional communication round between server and worker j to aggregate $\delta_j^{(r)} \triangleq \mathbf{S}_j\left[\texttt{HEAVYMIX}(\mathbf{S}^{(r)})\right]$, see Lines 5 and 12

Remark 4 (Comparison with [16])**.** *An important feature of our algorithm is that due to a lower dimension of the count sketch, the resulting averages ($\mathbf{S}^{(r)}$ and $\tilde{\mathbf{S}}^{(r)}$) received by the server, are also of lower dimension. Therefore, these algorithms exploit bidirectional compression in communication from server to device back and forth. As a result, due to this bidirectional property of communicating sketching for the case of large quantization error shown by $\omega = \theta(\frac{d}{m})$ in [16], our algorithms can outperform FedCOM and FedCOMGATE developed in [16] if bigger hash tables are used and the uplink communication cost is expensive. Furthermore, while in [17] server stores a global model and aggregates the partial gradients from devices which can enable server to extract some information regarding the device's data, in contrast, in our algorithm server does not store global model and only take the average of sketching of gradient and broadcast it. Therefore, sketching-based server-devices communication algorithm such as ours also provides privacy as a by-product.*

We also highlight that these algorithms are applicable to cross-silo and cross-device federated setting.

3.2 Heterogeneous Setting

In this section, we focus on the optimization problem in (1) in the special case of $q_1 = \ldots = q_p = \frac{1}{p}$ with full device participation ($k = p$). We also note that these results can be extended to the scenario where devices are sampled. In the previous section, we discussed Algorithm FedSKETCH, which is originally designed for homogeneous setting. However, in a heterogeneous setting, the aforementioned algorithms may fail to perform well in practice. The main reason is that in Federated learning, devices are using local stochastic descent direction which could be different than global descent direction when the data distribution are non-identical. Therefore, to mitigate the effect of data heterogeneity, we introduce a new algorithm called FedSKETCHGATE described in Algorithm 5. This algorithm leverages the idea of gradient tracking introduced in [16] (with compression) and a special case of $\gamma = 1$ without compression [34]. The main idea is that using an approximation of global gradient, $\mathbf{c}_j^{(r)}$ allows to correct the local gradient direction. For the FedSKETCHGATE with PRIVIX variant, the correction vector $\mathbf{c}_j^{(r)}$ at device j and communication round r is computed in Line 4. While using HEAPRIX compression method, FedSKETCHGATE also updates $\tilde{\mathbf{S}}^{(r)}$ via Line 16. Note that these algorithms are more applicable to cross-silo setting where the number of devices are not extremely large and most of them are available.

Algorithm 4 `FedSKETCH`(R, τ, η, γ): Private Federated Learning with Sketching.

Inputs: $\boldsymbol{x}^{(0)}$: initial model shared by all local devices, global and local learning rates γ and η, respectively
for $r = 0, \ldots, R-1$ **do**
  parallel for device $j \in \mathcal{K}^{(r)}$ **do**:
   if PRIVIX variant:

$$\mathbf{\Phi}^{(r)} \triangleq \texttt{PRIVIX}\left[\mathbf{S}^{(r-1)}\right]$$

   if HEAPRIX variant:

$$\mathbf{\Phi}^{(r)} \triangleq \texttt{HEAVYMIX}\left[\mathbf{S}^{(r-1)}\right] + \texttt{PRIVIX}\left[\mathbf{S}^{(r-1)} - \tilde{\mathbf{S}}^{(r-1)}\right]$$

  Set $\boldsymbol{x}^{(r)} = \boldsymbol{x}^{(r-1)} - \gamma \mathbf{\Phi}^{(r)}$ and $\boldsymbol{x}_j^{(0,r)} = \boldsymbol{x}^{(r)}$
  for $\ell = 0, \ldots, \tau - 1$ **do**
   Sample a mini-batch $\xi_j^{(\ell,r)}$ and compute $\tilde{\mathbf{g}}_j^{(\ell,r)}$
  Update $\boldsymbol{x}_j^{(\ell+1,r)} = \boldsymbol{x}_j^{(\ell,r)} - \eta\, \tilde{\mathbf{g}}_j^{(\ell,r)}$
  end for
  Device j boradcasts $\mathbf{S}_j^{(r)} \triangleq \mathbf{S}_j\left(\boldsymbol{x}_j^{(0,r)} - \boldsymbol{x}_j^{(\tau,r)}\right)$.
  Server **computes** $\mathbf{S}^{(r)} = \frac{1}{k}\sum_{j \in \mathcal{K}} \mathbf{S}_j^{(r)}$.
  Server **broadcasts** $\mathbf{S}^{(r)}$ to devices in randomly drawn devices $\mathcal{K}^{(r)}$.
  if HEAPRIX variant:
   Second round of communication: $\delta_j^{(r)} := \mathbf{S}_j\left[\texttt{HEAVYMIX}(\mathbf{S}^{(r)})\right]$ and broadcasts $\tilde{\mathbf{S}}^{(r)} \triangleq \frac{1}{k}\sum_{j \in \mathcal{K}} \delta_j^{(r)}$ to devices in set $\mathcal{K}^{(r)}$
 end parallel for
end
Output: $\boldsymbol{x}^{(R-1)}$

Algorithm 5 `FedSKETCHGATE`(R, τ, η, γ): Private Federated Learning with Sketching and gradient tracking.

Inputs: $\boldsymbol{x}^{(0)} = \boldsymbol{x}_j^{(0)}$ shared by all local devices, global and local learning rates γ and η.
for $r = 0, \ldots, R-1$ **do**
  parallel for device $j = 1, \ldots, p$ **do**:
   if PRIVIX variant:

$$\mathbf{c}_j^{(r)} = \mathbf{c}_j^{(r-1)} - \frac{\texttt{PRIVIX}\left(\mathbf{S}^{(r-1)}\right) - \texttt{PRIVIX}\left(\mathbf{S}_j^{(r-1)}\right)}{\tau}$$

  where $\mathbf{\Phi}^{(r)} \triangleq \texttt{PRIVIX}(\mathbf{S}^{(r-1)})$
   if HEAPRIX variant:

$$\mathbf{c}_j^{(r)} = \mathbf{c}_j^{(r-1)} - \frac{1}{\tau}\left(\mathbf{\Phi}^{(r)} - \mathbf{\Phi}_j^{(r)}\right)$$

  Set $\boldsymbol{x}^{(r)} = \boldsymbol{x}^{(r-1)} - \gamma \mathbf{\Phi}^{(r)}$ and $\boldsymbol{x}_j^{(0,r)} = \boldsymbol{x}^{(r)}$
  for $\ell = 0, \ldots, \tau - 1$ **do**
   Sample mini-batch $\xi_j^{(\ell,r)}$ and compute $\tilde{\mathbf{g}}_j^{(\ell,r)}$
   $\boldsymbol{x}_j^{(\ell+1,r)} = \boldsymbol{x}_j^{(\ell,r)} - \eta\left(\tilde{\mathbf{g}}_j^{(\ell,r)} - \mathbf{c}_j^{(r)}\right)$
  end for
  Device j broadcasts $\mathbf{S}_j^{(r)} \triangleq \mathbf{S}\left(\boldsymbol{x}_j^{(0,r)} - \boldsymbol{x}_j^{(\tau,r)}\right)$.
  Server **computes** $\mathbf{S}^{(r)} = \frac{1}{p}\sum_{j=1} \mathbf{S}_j^{(r)}$ and **broadcasts** $\mathbf{S}^{(r)}$ to all devices.
  if HEAPRIX variant:
  Device j computes $\mathbf{\Phi}_j^{(r)} \triangleq \texttt{HEAPRIX}[\mathbf{S}_j^{(r)}]$
   Second round of communication to obtain $\delta_j^{(r)} := \mathbf{S}_j\left(\texttt{HEAVYMIX}[\mathbf{S}^{(r)}]\right)$
  Broadcasts $\tilde{\mathbf{S}}^{(r)} \triangleq \frac{1}{p}\sum_{j=1}^{p} \delta_j^{(r)}$ to devices
 end parallel for
end
Output: $\boldsymbol{x}^{(R-1)}$

4 Convergence Analysis

We first state common assumptions needed in the following convergence analysis.

Assumption 1 (Smoothness and Lower Boundedness)**.** *The local objective function $f_j(\cdot)$ of jth device is differentiable for $j \in [p]$ and L-smooth, i.e., $\|\nabla f_j(\boldsymbol{x}) - \nabla f_j(\mathbf{y})\| \leq L\|\boldsymbol{x} - \mathbf{y}\|, \forall\, \boldsymbol{x}, \mathbf{y} \in \mathbb{R}^d$. Moreover, the optimal objective function $f(\cdot)$ is bounded below by $f^* = \min_{\boldsymbol{x}} f(\boldsymbol{x}) > -\infty$.*

Assumption 2 (Polyak-Łojasiewicz)**.** *A function $f(\boldsymbol{x})$ satisfies the Polyak-Łojasiewicz(PL) condition with constant μ if $\frac{1}{2}\|\nabla f(\boldsymbol{x})\|_2^2 \geq \mu\big(f(\boldsymbol{x}) - f(\boldsymbol{x}^*)\big), \forall \boldsymbol{x} \in \mathbb{R}^d$ with $\boldsymbol{x}^*$ is an optimal solution.*

We note that Assumption 1 is a common assumption in the literature of stochastic optimization. Additionally, it is shown in [22] that PL condition implies strong convexity property with same module. Additionally, PL objectives could also be non-convex, hence strong convexity does not imply PL condition necessarily.

4.1 Convergence of `FEDSKETCH`

Now we focus on the homogeneous case where data is distributed i.i.d. among local devices. In this case, the stochastic local gradient of each worker is an unbiased estimator of the global gradient. We will need the following additional common assumption on the stochastic gradients.

Assumption 3 (Bounded Variance)**.** *For all $j \in [m]$, we can sample an independent mini-batch ℓ_j of size $|\Xi_j^{(\ell,r)}| = b$ and compute an unbiased stochastic gradient $\tilde{\mathbf{g}}_j = \nabla f_j(\boldsymbol{w}; \Xi_j), \mathbb{E}_{\xi_j}[\tilde{\mathbf{g}}_j] = \nabla f(\boldsymbol{w}) = \mathbf{g}$ with the variance bounded is bounded by a constant σ^2, i.e., $\mathbb{E}_{\Xi_j}\left[\|\tilde{\mathbf{g}}_j - \mathbf{g}\|^2\right] \leq \sigma^2$.*

Theorem 1. *Assume Assumptions 1-3. Given* $0 < m = O\left(\frac{e}{\mu^2}\right) \leq d$*, and Consider* `FedSKETCH` *in Algorithm 4 with sketch size* $B = O\left(m \log\left(\frac{dR}{\delta}\right)\right)$ *and* $\gamma \geq k$*. In the homogeneous and with probability* $1-\delta$ *we have:*

In the ***non-convex*** *case,* $\{\boldsymbol{w}^{(r)}\}_{r=>0}$ *satisfies* $\frac{1}{R}\sum_{r=0}^{R-1}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2 \leq \epsilon$ *if:*

- `FedSKETCH-PRIVIX`*, for* $\eta = \frac{1}{L\gamma}\sqrt{\frac{k}{R\tau\left(\frac{\mu^2 d}{k}+1\right)}}$*:*

$$R = O\left(1/\epsilon\right) \quad \text{and} \quad \tau = O\left((\mu^2 d+1)/(k\epsilon)\right)$$

- `FedSKETCH-HEAPRIX`*, for* $\eta = \frac{1}{L\gamma}\sqrt{\frac{k}{R\tau\left(\frac{\mu^2 d-1}{k}+1\right)}}$*:*

$$R = O\left(1/\epsilon\right) \quad \text{and} \quad \tau = O\left(\mu^2 d/(k\epsilon)\right)$$

In the ***PL or Strongly convex*** *case,* $\{\boldsymbol{w}^{(r)}\}_{r=>0}$ *satisfies* $\mathbb{E}\Big[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\Big] \leq \epsilon$ *if we set:*

- `FedSKETCH-PRIVIX`*, for* $\eta = \frac{1}{2L\left(\frac{\mu^2 d}{k}+1\right)\tau\gamma}$*:*

$$R = O\left(\left(\mu^2 d/k+1\right)\kappa\log\left(1/\epsilon\right)\right)$$
$$\tau = O\left((\mu^2 d+1)/k\left(\mu^2 d/k+1\right)\epsilon\right)$$

- `FedSKETCH-HEAPRIX`*, for* $\eta = \frac{1}{2L\left(\frac{\mu^2 d-1}{k}+1\right)\tau\gamma}$*:*

$$R = O\left(\left((\mu^2 d-1)/k+1\right)\kappa\log\left(1/\epsilon\right)\right)$$
$$\tau = O\left(\mu^2 d/(k\left((\mu^2 d-1)/k+1\right)\epsilon)\right)$$

In the ***Convex*** *case,* $\{\boldsymbol{w}^{(r)}\}_{r=>0}$ *satisfies* $\mathbb{E}\Big[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\Big] \leq \epsilon$ *if we set:*

- `FedSKETCH-PRIVIX`*, for* $\eta = \frac{1}{2L\left(\frac{\mu^2 d}{k}+1\right)\tau\gamma}$*:*

$$R = O\left(L\left(1+\mu^2 d/k\right)/\epsilon\log\left(1/\epsilon\right)\right)$$
$$\tau = O\left(\left(\mu^2 d+1\right)^2/(k\left(\mu^2 d/k+1\right)^2\epsilon^2)\right)$$

- `FedSKETCH-HEAPRIX`*, for* $\eta = \frac{1}{2L\left(\frac{\mu^2 d-1}{k}+1\right)\tau\gamma}$*:*

$$R = O\left(L\left(1+(\mu^2 d-1)/k\right)/\epsilon\log\left(1/\epsilon\right)\right)$$
$$\tau = O\left(\left(\mu^2 d\right)^2/(k\left((\mu^2 d-1)/k+1\right)^2\epsilon^2)\right)$$

Corollary 2 (Total communication cost)**.** *As a consequence of Remark 6, the total communication cost per-worker becomes*

$$O\left(RB\right) = O\left(Rm\log\left(\frac{dR}{\delta}\right)\right) = O\left(\frac{m}{\epsilon}\log\left(\frac{d}{\epsilon\delta}\right)\right). \tag{2}$$

We note that this result in addition to improving over the communication complexity of federated learning of the state-of-the-art from $O\left(\frac{d}{\epsilon}\right)$ *in [23, 51, 34] to* $O\left(\frac{mk}{\epsilon}\log\left(\frac{dk}{\epsilon\delta}\right)\right)$*, it also implies differential privacy. As a result, total communication cost is*

$$BkR = O\left(\frac{mk}{\epsilon}\log\left(\frac{d}{\epsilon\delta}\right)\right).$$

We note that the state-of-the-art in [23] the total communication cost is $BkR = O\left(\frac{kd}{\epsilon}\frac{p^{2/3}}{k^{2/3}}\right)$ *that we improve, in terms of dependency on d, to*

$$BkR = O\left(\frac{mk}{\epsilon}\log\left(\frac{d}{\epsilon\delta}\right)\right).$$

In comparison to [21], we improve the total communication per worker from $RB = O\left(\frac{m}{\epsilon^2}\log\left(\frac{d}{\epsilon^2\delta}\right)\right)$ *to* $RB = O\left(\frac{m}{\epsilon}\log\left(\frac{d}{\epsilon\delta}\right)\right)$*.*

Remark 5. *Note that most of the available communication-efficient algorithms with quantization or compression only consider communication-efficiency from devices to server. However, Algorithm 4 also improves the communication efficiency from server to devices since it uses lower dimensional sketching size and displays a smaller dimension of the average of sketching communicated from server to devices.*

We note that it is not fair to compare our algorithms with algorithms without compression. However, in the following Corollary we share an interesting observation regarding our algorithm for PL and thus strongly convex objectives in homogeneous setting.

Corollary 3 (Total communication cost for PL or strongly convex)**.** *To achieve the convergence error of* ϵ*, we need to have* $R = O\left(\kappa(\frac{\mu^2 d}{k}+1)\log\frac{1}{\epsilon}\right)$ *and* $\tau = \left(\frac{(\mu^2 d+1)}{(\frac{\mu^2 d}{k}+1)k\epsilon}\right)$*. This leads to the total communication cost per worker of*

$$BR = O\left(m\kappa(\frac{\mu^2 d}{k}+1)\log\left(\frac{\kappa(\frac{\mu^2 d^2}{k}+d)\log\frac{1}{\epsilon}}{\delta}\right)\log\frac{1}{\epsilon}\right).$$

As a consequence, the total communication cost becomes:

$$BkR = O\left(m\kappa(\mu^2 d+k)\log\left(\frac{\kappa(\frac{\mu^2 d^2}{k}+d)\log\frac{1}{\epsilon}}{\delta}\right)\log\frac{1}{\epsilon}\right)$$

We note that the state-of-the-art in [23] the total communication cost is $BkR = O\left(\kappa k d\log\left(\frac{p}{k\epsilon}\right)\right)$ *that we improve, in terms of dependency on d, to*

$$BkR = O\left(m\kappa(\mu^2 d+k)\log\left(\frac{\kappa(\frac{\mu^2 d}{k}+d)\log\frac{1}{\epsilon}}{\delta}\right)\log\frac{1}{\epsilon}\right)$$

leading to an improvement from kd to $k+d$. These results are summarized in Table 1.

4.2 Convergence of FedSKETCHGATE

Assumption 4 (Bounded Local Variance)**.** *For all $j \in [p]$, we can sample an independent mini-batch Ξ_j of size $|\xi_j| = b$ and compute an unbiased stochastic gradient $\tilde{\mathbf{g}}_j = \nabla f_j(\boldsymbol{w}; \Xi_j)$ with $\mathbb{E}_\xi[\tilde{\mathbf{g}}_j] = \nabla f_j(\boldsymbol{w}) = \mathbf{g}_j$. Moreover, the variance of local stochastic gradients is bounded such that $\mathbb{E}_\Xi\left[\|\tilde{\mathbf{g}}_j - \mathbf{g}_j\|^2\right] \leq \sigma^2$.*

Theorem 2. *Assume Assumptions 1 and 4. Given $0 < m = O\left(\frac{e}{\mu^2}\right) \leq d$, and Consider FedSKETCHGATE in Algorithm 5 with sketch size $B = O\left(m \log\left(\frac{dR}{\delta}\right)\right)$ and $\gamma \geq p$. If the local data distributions of all users are identical (homogeneous setting), then with probability $1-\delta$ we have*

*In the **non-convex** case, $\eta = \frac{1}{L\gamma}\sqrt{\frac{p}{R\tau(\mu^2 d)}}$, $\{\boldsymbol{w}^{(r)}\}_{r=>0}$ satisfies $\frac{1}{R}\sum_{r=0}^{R-1}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2 \leq \epsilon$ if:*

- *FedSKETCH-PRIVIX:*

$$R = O\left((\mu^2 d + 1)/\epsilon\right) \quad \text{and} \quad \tau = O\left(1/(p\epsilon)\right)$$

- *FedSKETCH-HEAPRIX:*

$$R = O\left(\mu^2 d/\epsilon\right) \quad \text{and} \quad \tau = O\left(1/(p\epsilon)\right)$$

*In the **PL or Strongly convex** case, $\{\boldsymbol{w}^{(r)}\}_{r=>0}$ satisfies $\mathbb{E}\left[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\right] \leq \epsilon$ if:*

- *FedSKETCH-PRIVIX, for $\eta = \frac{1}{2L(\mu^2 d+1)\tau\gamma}$:*

$$R = O\left((\mu^2 d + 1)\kappa \log(1/\epsilon)\right) \quad \text{and} \quad \tau = O\left(1/(p\epsilon)\right)$$

- *FedSKETCH-HEAPRIX, for $\eta = \frac{1}{2L(\mu^2 d)\tau\gamma}$:*

$$R = O\left((\mu^2 d)\kappa \log(1/\epsilon)\right) \quad \text{and} \quad \tau = O\left(1/(p\epsilon)\right)$$

*In the **Convex** case, $\{\boldsymbol{w}^{(r)}\}_{r=>0}$ satisfies $\mathbb{E}\left[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\right] \leq \epsilon$ if:*

- *FedSKETCH-PRIVIX, for $\eta = \frac{1}{2L(\mu^2 d+1)\tau\gamma}$:*

$$R = O\left(L(\mu^2 d + 1)\epsilon \log(1/\epsilon)\right) \quad \text{and} \quad \tau = O\left(1/(p\epsilon^2)\right)$$

- *FedSKETCH-HEAPRIX, for $\eta = \frac{1}{2L(\mu^2 d)\tau\gamma}$:*

$$R = O\left(L(\mu^2 d)\epsilon \log(1/\epsilon)\right) \quad \text{and} \quad \tau = O\left(1/(p\epsilon^2)\right)$$

4.3 Comparison with Prior Methods

Comparison to [30]. We note that our convergence analysis does not rely on the bounded gradient assumption and it can be seen that we improve both the number of communication rounds R and the size of transmitted vector B per communication round while preserving the privacy property. Additionally, we highlight that, while [30] provides a convergence analysis for convex objectives, our analysis holds for PL (thus strongly convex case), general convex and general non-convex objectives.

Comparison with [45]. Consider two versions of FetchSGD in this reference. First while in our schemes we do not to have access to the exact entries of gradients, since the approaches in [45] is based on top_m queries, both of the proposed algorithms (in [45]) require to have access to the exact value of top_k gradients, hence they do not preserve privacy. Second, both of the convergence results in [45] rely on the uniform bounded gradient assumption which may not be applicable with L-smoothness assumption when data distribution is highly heterogeneous which is the case in Federated Learning (see [24] for more detail). However, our convergence results do not need any bounded gradient assumption. Third, Theorem 1 [45] is based on an Assumption that *Contraction Holds* for the sequence of gradients encountered during the optimization which may not hold necessarily in practice, yet based on this strong assumption their total communication cost (RB) to achieve ϵ error is $BR = O\left(m \max(\frac{1}{\epsilon^2}, \frac{d^2-dm}{m^2\epsilon}) \log\left(\frac{d}{\delta}\max(\frac{1}{\epsilon^2}, \frac{d^2-dm}{m^2\epsilon})\right)\right)$ (Note for the sake of comparison we let the compression ration in [45] to be $\frac{m}{d}$). In contrast, without any extra assumptions, our results in Theorem 2 for PRIVIX and HEAPRIX are respectively $BR = O\left(\frac{m(\mu^2 d+1)}{\epsilon}\log\left(\frac{\mu^2 d^2+d}{\epsilon\delta}\log\left(\frac{1}{\epsilon}\right)\right)\right)$ and $BR = O\left(\frac{m(\mu^2 d)}{\epsilon}\log\left(\frac{\mu^2 d^2}{\epsilon\delta}\log\left(\frac{1}{\epsilon}\right)\right)\right)$ which improves total communication cost in Theorem 1 in [45] in regimes where $\frac{1}{\epsilon} \geq d$ or $d >> m$. Theorem 2 in [45] is based on another assumption of Sliding Window Heavy Hitters, which is similar to gradient diversity assumption in [32, 17] (but it is weaker assumption of contraction in Theorem 1 in [45]), and they showed that the total communication cost is $BR = O\left(\frac{m\max(I^{2/3}, 2-\alpha)}{\epsilon^3\alpha}\log\left(\frac{d\max(I^{2/3}, 2-\alpha)}{\epsilon^3\delta}\right)\right)$ (I is constant comes from the extra assumption over the window of gradients which similar to bounded gradient diversity) which is again worse than ob-

Table 1 Comparison of results with compression and periodic averaging in the homogeneous setting. Here, m is the number of devices, μ is the PL constant, m is the number of bins of hash tables, d is the dimension of the model, κ is the condition number, ϵ is the target accuracy, R is the number of communication rounds, and τ is the number of local updates. UG and PP stand for Unbounded Gradient and Privacy Property respectively.

Reference	PL/Strongly Convex	UG	PP
Ivkin et al. [21]	$R = O\left(\frac{\mu^2 d}{\epsilon}\right),\ \tau = 1,\ B = O\left(m \log\left(\frac{dR}{\delta}\right)\right)$ $pRB = O\left(\frac{p\mu^2 d}{\epsilon} m \log\left(\frac{\mu^2 d^2}{\epsilon\delta}\right)\right)$	✗	✗
Theorem 1	$\boldsymbol{R = O\left(\kappa\left(\frac{\mu^2 d-1}{k}+1\right)\log\left(\frac{1}{\epsilon}\right)\right),\ \tau = O\left(\frac{(\mu^2 d)}{k\left(\frac{\mu^2 d}{k}+1\right)\epsilon}\right), B = O\left(m \log\left(\frac{dR}{\delta}\right)\right)}$ $\boldsymbol{kBR = O\left(m\kappa(\mu^2 d - 1 + k)\log\frac{1}{\epsilon}\log\left(\frac{\kappa(d\frac{\mu^2 d-1}{k}+d)\log\frac{1}{\epsilon}}{\delta}\right)\right)}$	✔	✔

tained result in this paper with weaker assumptions in a regime where $\frac{I^{2/3}}{\epsilon^2} \geq d$. Next, unlike [45] which only focuses on non-convex objectives, in this work we provide the convergence analysis for PL (thus strongly convex case), general convex and general non-convex objectives. Finally, although the algorithm in [45] requires additional memory for the server to store the compression error correction vector, our algorithm does not need such additional storage. On the other hand, we note that unlike [45], our algorithm requires devices to store a local state vector and additionally need a second round of communication for `HEAPRIX`.

5 Numerical Applications

In this section, we provide empirical results on MNIST dataset to demonstrate the effectiveness of our proposed algorithms. The model we use is the LeNet-5 Convolutional Neural Network (CNN) architecture introduced in [27], with 60 000 model parameters in total. We compare Federated SGD (FedSGD), SketchSGD [21], FedSketch-PRIVIX (FS-PRIVIX) and FedSketch-HEAPRIX (FS-HEAPRIX). Note that in Algorithm 4, FS-PRIVIX with global learning rate $\gamma = 1$ is equivalent to the DiffSketch algorithm proposed in [32]. The number of workers is set to 50 and the number of local updates τ is carying for FL methods. For SketchedSGD which is under synchronous distributed learning framework, τ is fixed to 1. We tune the learning rates (both local, i.e. η and global, i.e. γ, if applicable) over the log-scale and report the best results. In each round of local update, we randomly choose half of the local devices to be active, which is the common practice in real-world applications. Numerical results are reported for both *homogeneous* and *heterogeneous* setting. In the former case, each device receives uniformly drawn data samples. In the latter case, each device only receives samples from one or two classes among ten digits in the MNIST dataset.

Homogeneous case. In Figure 3, we provide the training loss and test accuracy for the four algorithms mentioned above, with $\tau = 1$ (since SketchSGD requires single local update per round). We also test different sizes of sketching matrix, $(t, k) = (20, 40)$ and $(50, 100)$. Note that these two choices of sketch size correspond to a $75\times$ and $12\times$ compression ratio, respectively. In general, as one would expect, higher compression ratio leads to worse learning performance. In both cases, FS-HEAPRIX performs the best in terms of both training objective and test accuracy. FS-PRIVIX is better when sketch size is large (i.e. when the estimation from sketches are more accurate), while SketchSGD performs better with small sketch size. Results for multiple local updates $\tau = 5$ are presented Figure 3 ($\tau = 2$ is deffered to the Appendix). We see that FS-HEAPRIX is significantly better than FS-PRIVIX, either with small or large sketching matrix. FS-HEAPRIX yields acceptable extra test error compared to FedSGD, especially when considering the high compression ratio (e.g. $75\times$). However, FS-PRIVIX performs poorly with small sketch size $(20, 40)$, and even diverges with $\tau = 5$. We also observe that the performances of FS-HEAPRIX improve when the number of local updates increases. That is, the proposed method is able to further reduce the communication cost by reducing the number of rounds required for communication. This is also consistent with our theoretical claims established in this paper. For $\tau = 1, 2, 5$, we see that a sketch size of $(50, 100)$ is sufficient to give similar test accuracy as the Federated SGD (FedSGD) algorithm.

Heterogeneous case. We plot similar sets of results in Figure 4 for non-i.i.d. data distribution

Table 2 Comparison of results with compression and periodic averaging in the heterogeneous setting. Here, p is the number of devices, μ is compression of hash table, d is the dimension of the model, κ is condition number, ϵ is target accuracy, R is the number of communication rounds, and τ is the number of local updates. UG and PP stand for Unbounded Gradient and Privacy Property respectively.

Reference	non-convex	General Convex	UG	PP
Li et al. [30]	–	$R=O\left(\frac{\mu^2 d}{\epsilon^2}\right)$ $\tau=1$ $B=O\left(m\log\left(\frac{\mu^2 d^2}{\epsilon^2\delta}\right)\right)$	✗	✔
Rothchild et al. [45]	$R=O\left(\max(\frac{1}{\epsilon^2},\frac{d^2-md}{m^2\epsilon})\right)$ $\tau=1$ $B=O\left(m\log\left(\frac{d}{\epsilon^2\delta}\right)\right)$ $BR=O\left(\frac{m}{\epsilon^2}\max(\frac{1}{\epsilon^2},\frac{d^2-md}{m^2\epsilon})\log\left(\frac{d}{\delta}\max(\frac{1}{\epsilon^2},\frac{d^2-md}{m^2\epsilon})\right)\right)$	–	✗	✗
Rothchild et al. [45]	$R=O\left(\frac{\max(I^{2/3},2-\alpha)}{\epsilon^3}\right)$ $\tau=1$ $B=O\left(\frac{m}{\alpha}\log\left(\frac{d\max(I^{2/3},2-\alpha)}{\epsilon^3\delta}\right)\right)$ $BR=O\left(\frac{m\max(I^{2/3},2-\alpha)}{\epsilon^3\alpha}\log\left(\frac{d\max(I^{2/3},2-\alpha)}{\epsilon^3\delta}\right)\right)$	–	✗	✗
Theorem 2	$\boldsymbol{R=O\left(\frac{\mu^2 d}{\epsilon}\right)}$ $\boldsymbol{\tau=O\left(\frac{1}{p\epsilon}\right)}$ $\boldsymbol{B=O\left(m\log\left(\frac{\mu^2 d^2}{\epsilon\delta}\right)\right)}$ $\boldsymbol{BR=O\left(\frac{m(\mu^2 d)}{\epsilon}\log\left(\frac{\mu^2 d^2}{\epsilon\delta}\log\left(\frac{1}{\epsilon}\right)\right)\right)}$	$\boldsymbol{R=O\left(\frac{\mu^2 d}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right)}$ $\boldsymbol{\tau=O\left(\frac{1}{p\epsilon^2}\right)}$ $\boldsymbol{B=O\left(m\log\left(\frac{\mu^2 d^2}{\epsilon\delta}\right)\right)}$	✔	✔

(heterogeneous setting). This setting leads to more twists and turns in the training curves. From the first column ($\tau = 1$), we see that SketchSGD performs very poorly in the heterogeneous case, while both our proposed FedSketchGATE methods, see Algorithm 5, achieve similar generalization accuracy as the Federated SGD (FedSGD) algorithm, even with fairly small sketch size (i.e. 75× compression ratio). Note that, the slow convergence of federated SGD in non-i.i.d. data distribution case has also been reported in literature, e.g. [38, 9]. In addition, FS-HEAPRIX is again better than FS-PRIVIX in terms of both training loss and test accuracy. Furthermore, we notice in column 2 and 3 of Figure 4 the advantage of FS-HEAPRIX over FS-PRIVIX with multiple local updates. However, empirically we see that in the heterogeneous setting, more local updates τ tend to undermine the learning performance, especially with small sketch size. Nevertheless, we see that when sketch size is large, i.e. $(50, 100)$, FS-HEAPRIX can still provide comparable test accuracy as FedSGD with $\tau = 5$. Our empirical study demonstrates that our proposed FedSketch (and FedSketchGATE) frameworks are able to perform well in homogeneous (resp. heterogeneous) learning setting, with high compression rate. In particular, FedSketch methods are advantageous over prior SketchedSGD [21] method in both cases. FS-HEAPRIX performs the best among all the tested compressed optimization algorithms, which in many cases achieves similar generalization accuracy as Federated SGD with small sketch size. In general, in any tested case, we achieve 12× compression ratio with little loss in test accuracy.

6 Conclusion

In this paper, we introduced `FedSKETCH` and `FedSKETCHGATE` algorithms for homogeneous and heterogeneous data distribution setting respectively for Federated Learning wherein communication between server and devices is only performed using count sketch. Our algorithms, thus, provide communication-efficiency and privacy. We analyze the convergence error for *non-convex*, *Polyak-Lojasiewicz* and *general convex* objective functions in the scope of Federated Optimization. We provide insightful numerical experiments showcasing the advantages of our `FedSKETCH` and `FedSKETCHGATE` methods over current federated optimization algorithm. The proposed algorithms outperform competing compression method and can achieve comparable test accuracy as Federated SGD, with high compression ratio.

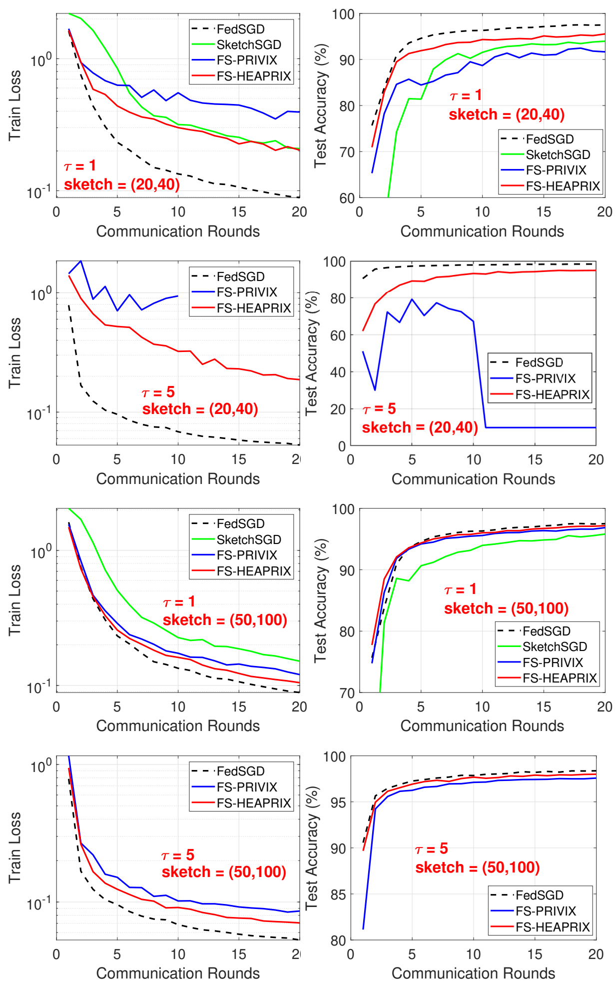

Figure 1 Homogeneous case: Comparison of compressed optimization methods on LeNet CNN.

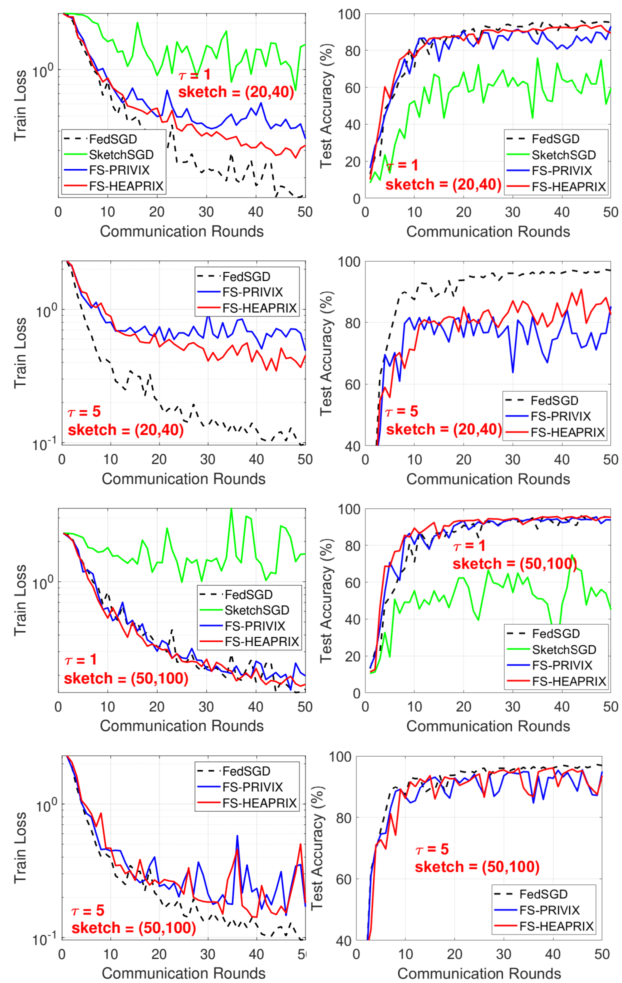

Figure 2 Heterogeneous case: Comparison of compressed optimization algorithms on LeNet CNN.

References

[1] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1709–1720, Long Beach, 2017.

[2] Dan Alistarh, Torsten Hoefler, Mikael Johansson, Nikola Konstantinov, Sarit Khirirat, and Cédric Renggli. The convergence of sparsified gradient methods. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5973–5983, Montréal, Canada, 2018.

[3] Debraj Basu, Deepesh Data, Can Karakus, and Suhas N. Diggavi. Qsparse-local-sgd: Distributed SGD with quantization, sparsification and local computations. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 14668–14679, Vancouver, Canada, 2019.

[4] Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. SIGNSGD: compressed optimisation for non-convex problems. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 559–568, Stockholmsmässan, Stockholm, Sweden, 2018.

[5] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 1175–1191, Dallas, TX, 2017.

[6] Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 161–168, Vancouver, Canada, 2008.

[7] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium, USENIX Security 2019*, pages 267–284, Santa Clara, CA, 2019.

[8] Moses Charikar, Kevin C. Chen, and Martin Farach-Colton. Finding frequent items in data streams. *Theoretical Computer Science*, 312(1):3–15, 2004.

[9] Xiangyi Chen, Xiaoyun Li, and Ping Li. Toward communication efficient adaptive gradient method. In *ACM-IMS Foundations of Data Science Conference (FODS)*, Seattle, WA, 2020.

[10] Graham Cormode and Shan Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. *Journal of Algorithms*, 55(1):58–75, 2005.

[11] Robert Fergus, Barun Singh, Aaron Hertzmann, Sam T. Roweis, and William T. Freeman. Removing camera shake from a single photograph. *ACM Trans. Graph.*, 25(3):787–794, 2006.

[12] Robin C Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.

[13] Yuanhao Gong and Ivo F Sbalzarini. Gradient distribution priors for biomedical image processing. *arXiv preprint arXiv:1408.3300*, 2014.

[14] Farzin Haddadpour, Mohammad Mahdi Kamani, Mehrdad Mahdavi, and Viveck Cadambe. Local sgd with periodic averaging: Tighter analysis and adaptive synchronization. pages 11080–11092, Vancouver, Canada, 2019.

[15] Farzin Haddadpour, Mohammad Mahdi Kamani, Mehrdad Mahdavi, and Viveck R. Cadambe. Trading redundancy for communication: Speeding up distributed SGD for non-convex optimization. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 2545–2554, Long Beach, CA, 2019.

[16] Farzin Haddadpour, Mohammad Mahdi Kamani, Aryan Mokhtari, and Mehrdad Mahdavi. Federated learning with compression: Unified analysis and sharp guarantees. *arXiv preprint arXiv:2007.01154*, 2020.

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

[18] Stephen Hardy, Wilko Henecka, Hamish Ivey-Law, Richard Nock, Giorgio Patrini, Guillaume Smith, and Brian Thorne. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *arXiv preprint arXiv:1711.10677*, 2017.

[19] Samuel Horváth, Dmitry Kovalev, Konstantin Mishchenko, Sebastian Stich, and Peter Richtárik. Stochastic distributed learning with

gradient quantization and variance reduction. *arXiv preprint arXiv:1904.05115*, 2019.

[20] Samuel Horváth and Peter Richtárik. A better alternative to error feedback for communication-efficient distributed learning. *arXiv preprint arXiv:2006.11077*, 2020.

[21] Nikita Ivkin, Daniel Rothchild, Enayat Ullah, Vladimir Braverman, Ion Stoica, and Raman Arora. Communication-efficient distributed SGD with sketching. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 13144–13154, Vancouver, Canada, 2019.

[22] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Proceedings of European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD)*, pages 795–811, Riva del Garda, Italy, 2016.

[23] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J Reddi, Sebastian U Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for on-device federated learning. *arXiv preprint arXiv:1910.06378*, 2019.

[24] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local SGD on identical and heterogeneous data. In *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 4519–4529, Online [Palermo, Sicily, Italy], 2020.

[25] Jon Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397, 2003.

[26] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[27] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[28] Anat Levin, Robert Fergus, Frédo Durand, and William T. Freeman. Image and depth from a conventional camera with a coded aperture. *ACM Trans. Graph.*, 26(3):70, 2007.

[29] Ping Li, Kenneth Ward Church, and Trevor Hastie. One sketch for all: Theory and application of conditional random sampling. In *Advances in Neural Information Processing Systems (NIPS)*, pages 953–960, Vancouver, Canada, 2008.

[30] Tian Li, Zaoxing Liu, Vyas Sekar, and Virginia Smith. Privacy for free: Communication-efficient learning with differential privacy using sketches. *arXiv preprint arXiv:1911.00972*, 2019.

[31] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Process. Mag.*, 37(3):50–60, 2020.

[32] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems (MLSys)*, Austin, TX, 2020.

[33] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, 2020.

[34] Xianfeng Liang, Shuheng Shen, Jingchang Liu, Zhen Pan, Enhong Chen, and Yifei Cheng. Variance reduced local sgd with lower communication complexity. *arXiv preprint arXiv:1912.12844*, 2019.

[35] Tao Lin, Sebastian U. Stich, Kumar Kshitij Patel, and Martin Jaggi. Don't use large mini-batches, use local SGD. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, 2020.

[36] Yujun Lin, Song Han, Huizi Mao, Yu Wang, and Bill Dally. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, Vancouver, Canada, 2018.

[37] Zaoxing Liu, Tian Li, Virginia Smith, and Vyas Sekar. Enhancing the privacy of federated learning with sketching. *arXiv preprint arXiv:1911.01812*, 2019.

[38] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1273–1282, Fort Lauderdale, FL, 2017.

[39] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, Vancouver, Canada, 2018.

[40] Paavo Parmas. Total stochastic gradient algorithms and applications in reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 10225–10235, Montréal, Canada, 2018.

[41] Constantin Philippenko and Aymeric Dieuleveut. Artemis: tight convergence guarantees for bidirectional compression in federated learning. *arXiv preprint arXiv:2006.14591*, 2020.

[42] 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.

[43] Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, Ali Jadbabaie, and Ramtin Pedarsani. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. In *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2021–2031, Online [Palermo, Sicily, Italy], 2020.

[44] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407, 1951.

[45] Daniel Rothchild, Ashwinee Panda, Enayat Ullah, Nikita Ivkin, Ion Stoica, Vladimir Braverman, Joseph Gonzalez, and Raman Arora. FetchSGD: Communication-efficient federated learning with sketching. *arXiv preprint arXiv:2007.07682*, 2020.

[46] Anit Kumar Sahu, Tian Li, Maziar Sanjabi, Manzil Zaheer, Ameet Talwalkar, and Virginia Smith. On the convergence of federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 2018.

[47] Sebastian U Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified sgd with memory. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4447–4458, Montréal, Canada, 2018.

[48] Sebastian U Stich and Sai Praneeth Karimireddy. The error-feedback framework: Better rates for sgd with delayed gradients and compressed communication. *arXiv preprint arXiv:1909.05350*, 2019.

[49] Sebastian Urban Stich. Local sgd converges fast and communicates little. In *Proceedings of the 7th International Conference on Learning Representations (ICLR)*, New Orleans, LA, 2019.

[50] Hanlin Tang, Shaoduo Gan, Ce Zhang, Tong Zhang, and Ji Liu. Communication compression for decentralized training. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 7652–7662, Montréal, Canada, 2018.

[51] Jianyu Wang and Gauri Joshi. Cooperative sgd: A unified framework for the design and analysis of communication-efficient sgd algorithms. *arXiv preprint arXiv:1808.07576*, 2018.

[52] Wei Wen, Cong Xu, Feng Yan, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Terngrad: Ternary gradients to reduce communication in distributed deep learning. In *Advances in neural information processing systems (NIPS)*, pages 1509–1519, Long Beach, CA, 2017.

[53] Jiaxiang Wu, Weidong Huang, Junzhou Huang, and Tong Zhang. Error compensated quantized sgd and its applications to large-scale distributed optimization. *arXiv preprint arXiv:1806.08054*, 2018.

[54] Hao Yu, Rong Jin, and Sen Yang. On the linear speedup analysis of communication efficient momentum SGD for distributed nonconvex optimization. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 7184–7193, Long Beach, CA, 2019.

[55] 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 *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI)*, pages 5693–5700, Honolulu, HI, 2019.

[56] Jian Zhang, Christopher De Sa, Ioannis Mitliagkas, and Christopher Ré. Parallel sgd: When does averaging help? *arXiv preprint arXiv:1606.07365*, 2016.

[57] Fan Zhou and Guojing Cong. On the convergence properties of a k-step averaging stochastic gradient descent algorithm for nonconvex optimization. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3219–3227, Stockholm, Sweden, 2018.

Appendix

Notation. Here we indicate the count sketch of the vector $\boldsymbol{x}$ with $\mathbf{S}(\boldsymbol{x})$ and with abuse of notation we indicate the expectation over the randomness of count sketch with $\mathbb{E}_{\mathbf{S}}[.]$. We illustrate the random subset of the devices selected by server with $\mathcal{K}$ with size $|\mathcal{K}| = k \leq p$, and we represent the expectation over the device sampling with $\mathbb{E}_{\mathcal{K}}[.]$.

We will use the following fact (which is also used in [33, 17]) in proving results.

Fact 3 ([33, 17])**.** *Let $\{x_i\}_{i=1}^p$ denote any fixed deterministic sequence. We sample a multiset $\mathcal{P}$ (with size K) uniformly at random where x_j is sampled with probability q_j for $1 \leq j \leq p$ with replacement. Let $\mathcal{P} = \{i_1, \ldots, i_K\} \subset [p]$ (some i_js may have the same value). Then*

$$\mathbb{E}_{\mathcal{P}}\left[\sum_{i\in\mathcal{P}} x_i\right] = \mathbb{E}_{\mathcal{P}}\left[\sum_{k=1}^{K} x_{i_k}\right] = K\mathbb{E}_{\mathcal{P}}\left[x_{i_k}\right] = K\left[\sum_{j=1}^{p} q_j x_j\right] \tag{3}$$

A Various known algorithms

Algorithm 6 `PRIVIX` [30]: Unbiased compressor based on sketching.

1: **Inputs:** $\boldsymbol{x} \in \mathbb{R}^d, t, m, \mathbf{S}_{m\times t}, h_j(1 \leq i \leq t), sign_j(1 \leq i \leq t)$
2: **Query** $\tilde{\boldsymbol{x}} \in \mathbb{R}^d$ **from** $\mathbf{S}(\boldsymbol{x})$**:**
3: **for** $i = 1, \ldots, d$ **do**
4: $\quad\tilde{\boldsymbol{x}}[i] = \text{Median}\{\text{sign}_j(i).\mathbf{S}[j][h_j(i)] : 1 \leq j \leq t\}$
5: **end for**
6: **Output:** $\tilde{\boldsymbol{x}}$

B Comparison with [41].

The reference [41] considers two-way compression from parameter server to devices and vice versa. They provide the convergence rate of $R = O\left(\frac{\omega^{\text{Up}}\omega^{\text{Down}}}{\epsilon^2}\right)$ for strongly-objective functions where ω^{Up} and ω^{Down} are uplink and downlink's compression noise (specializing to our case for the sake of comparison $\omega^{\text{Up}} = \omega^{\text{Down}} = \theta(d)$) for general heterogeneous data distribution. In contrast, while as pointed out in Remark 4 that our algorithms are using bidirectional compression due to use of sketching for communication, our convergence rate for strongly-convex objective is $R = O(\kappa\mu^2 d \log\left(\frac{1}{\epsilon}\right))$ with probability $1 - \delta$.

Corollary 4. *Based on Lemma 3 and using Algorithm 3, we have $C(x) \in \mathbb{U}(\mu^2 d)$. This shows that unlike `PRIVIX` the compression noise can be made as small as possible using large size of hash table.*

Proof. The proof simply follows from Lemma 3 and Algorithm 3 by setting $\Delta_1 = \mu^2 d$ and $\Delta_2 = 1 + \mu^2 d$ we obtain $\Delta = \Delta_2 + \frac{1-\Delta_2}{\Delta_1} = \mu^2 d$. □

C Results for the Homogeneous Setting

In this section, we study the convergence properties of our `FedSKETCH` method presented in Algorithm 4. Before stating the proofs for `FedSKETCH` in the homogeneous setting, we first mention the following intermediate lemmas.

Lemma 4. *Using unbiased compression and under Assumption 3, we have the following bound:*

$$\mathbb{E}_{\mathcal{K}}\left[\mathbb{E}_{\mathbf{S},\xi^{(r)}}\left[\|\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\|^2\right]\right] = \mathbb{E}_{\xi^{(r)}}\mathbb{E}_{\mathbf{S}}\left[\|\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\|^2\right] \leq \tau(\frac{\omega}{k}+1)\sum_{j=1}^{m} q_j\left[\sum_{c=0}^{\tau-1}\|\mathbf{g}_j^{(c,r)}\|^2 + \sigma^2\right] \tag{4}$$

Proof.

$$
\begin{aligned}
&\mathbb{E}_{\xi^{(r)}|\boldsymbol{w}^{(r)}}\mathbb{E}_{\mathcal{K}}\left[\mathbb{E}_{\mathbf{S}}\left[\|\frac{1}{k}\sum_{j\in\mathcal{K}}\mathbf{S}\left(\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\right)\|^2\right]\right]\\
=&\mathbb{E}_{\xi^{(r)}}\left[\mathbb{E}_{\mathcal{K}}\left[\mathbb{E}_{\mathbf{S}}\left[\|\frac{1}{k}\sum_{j\in\mathcal{K}}\underbrace{\mathbf{S}\left(\overbrace{\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}}^{\tilde{\mathbf{g}}_j^{(r)}}\right)}_{\tilde{\mathbf{g}}_{\mathbf{S}_j}^{(r)}}\|^2\right]\right]\right]\\
\overset{①}{=}&\mathbb{E}_{\xi^{(r)}}\left[\mathbb{E}_{\mathcal{K}}\left[\left[\|\frac{1}{k}\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_{\mathbf{S}_j}^{(r)}-\frac{1}{k}\sum_{j\in\mathcal{K}}\mathbb{E}_{\mathbf{S}}\left[\tilde{\mathbf{g}}_{\mathbf{S}_j}^{(r)}\right]\|^2\right]+\|\mathbb{E}_{\mathbf{S}}\left[\frac{1}{k}\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_{\mathbf{S},j}^{(r)}\right]\|^2\right]\right]\\
\overset{②}{=}&\mathbb{E}_{\xi^{(r)}}\left[\mathbb{E}_{\mathcal{K}}\left[\mathbb{E}_{\mathbf{S}}\left[\|\frac{1}{k}\left[\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_{\mathbf{S}_j}^{(r)}-\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_j^{(r)}\right]\|^2\right]+\|\frac{1}{k}\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_j^{(r)}\|^2\right]\right]\\
=&\mathbb{E}_{\xi^{(r)}}\left[\mathbb{E}_{\mathcal{K}}\left[\left[\mathrm{Var}_{\mathbf{S}}\left[\frac{1}{k}\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_{\mathbf{S}_j}^{(r)}\right]\right]+\|\frac{1}{k}\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_j^{(r)}\|^2\right]\right]\\
=&\mathbb{E}_{\xi^{(r)}}\left[\mathbb{E}_{\mathcal{K}}\left[\frac{1}{k^2}\sum_{j\in\mathcal{K}}\mathrm{Var}_{\mathbf{S}_j}\left[\tilde{\mathbf{g}}_{\mathbf{S}_j}^{(r)}\right]+\|\frac{1}{k}\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_j^{(r)}\|^2\right]\right]\\
\leq&\mathbb{E}_{\xi^{(r)}}\left[\mathbb{E}_{\mathcal{K}}\left[\frac{1}{k^2}\sum_{j\in\mathcal{K}}\omega\left\|\tilde{\mathbf{g}}_j^{(r)}\right\|^2+\|\frac{1}{k}\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_j^{(r)}\|^2\right]\right]\\
=&\left[\mathbb{E}_{\xi}\left[\frac{1}{k}\sum_{j\in\mathcal{K}}\omega\left\|\tilde{\mathbf{g}}_j^{(r)}\right\|^2+\mathbb{E}_{\mathcal{K}}\mathbb{E}_{\xi^{(r)}}\|\frac{1}{k}\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_j^{(r)}\|^2\right]\right]\\
=&\left[\mathbb{E}_{\xi}\left[\frac{\omega}{k}\sum_{j=1}^{p}q_j\left\|\tilde{\mathbf{g}}_j^{(r)}\right\|^2+\mathbb{E}_{\mathcal{K}}\left[\mathrm{Var}\left(\frac{1}{k}\sum_{j\in\mathcal{K}}\tilde{\mathbf{g}}_j^{(r)}\right)+\|\frac{1}{k}\sum_{j\in\mathcal{K}}\mathbf{g}_j^{(r)}\|^2\right]\right]\right]\\
=&\frac{\omega}{k}\sum_{j=1}^{p}q_j\mathbb{E}_{\xi}\left\|\tilde{\mathbf{g}}_j^{(r)}\right\|^2+\mathbb{E}_{\mathcal{K}}\left[\frac{1}{k^2}\sum_{j\in\mathcal{K}}\mathrm{Var}\left(\tilde{\mathbf{g}}_j^{(r)}\right)+\|\frac{1}{k}\sum_{j\in\mathcal{K}}\mathbf{g}_j^{(r)}\|^2\right]\\
\leq&\frac{\omega}{k}\sum_{j=1}^{p}q_j\mathbb{E}_{\xi}\left\|\tilde{\mathbf{g}}_j^{(r)}\right\|^2+\mathbb{E}_{\mathcal{K}}\left[\frac{1}{k^2}\sum_{j\in\mathcal{K}}\tau\sigma^2+\frac{1}{k}\sum_{j\in\mathcal{K}}\|\mathbf{g}_j^{(r)}\|^2\right]\\
=&\frac{\omega}{k}\sum_{j=1}^{p}q_j\left[\mathrm{Var}\left(\tilde{\mathbf{g}}_j^{(r)}\right)+\left\|\mathbf{g}_j^{(r)}\right\|^2\right]+\left[\frac{\tau\sigma^2}{k}+\sum_{j=1}^{p}q_j\|\mathbf{g}_j^{(r)}\|^2\right]\\
\leq&\frac{\omega}{k}\sum_{j=1}^{p}q_j\left[\tau\sigma^2+\left\|\mathbf{g}_j^{(r)}\right\|^2\right]+\left[\frac{\tau\sigma^2}{k}+\sum_{j=1}^{p}q_j\|\mathbf{g}_j^{(r)}\|^2\right]\\
=&(\omega+1)\frac{\tau\sigma^2}{k}+(\frac{\omega}{k}+1)\left[\sum_{j=1}^{p}q_j\|\mathbf{g}_j^{(r)}\|^2\right] \qquad (5)
\end{aligned}
$$

where ① holds due to $\mathbb{E}\left[\|\mathbf{x}\|^2\right] = \mathrm{Var}[\mathbf{x}] + \|\mathbb{E}[\mathbf{x}]\|^2$, ② is due to $\mathbb{E}_{\mathbf{S}}\left[\frac{1}{p}\sum_{j=1}^{p}\tilde{\mathbf{g}}_{\mathbf{S}j}^{(r)}\right] = \frac{1}{p}\sum_{j=1}^{m}\tilde{\mathbf{g}}_{j}^{(r)}$.

Next we show that from Assumptions 4, we have

$$\mathbb{E}_{\xi^{(r)}}\left[\left[\|\tilde{\mathbf{g}}_j^{(r)} - \mathbf{g}_j^{(r)}\|^2\right]\right] \leq \tau\sigma^2 \tag{6}$$

To do so, note that

$$\begin{aligned}
\mathrm{Var}\left(\tilde{\mathbf{g}}_j^{(r)}\right) &= \mathbb{E}_{\xi^{(r)}}\left[\left\|\tilde{\mathbf{g}}_j^{(r)} - \mathbf{g}_j^{(r)}\right\|^2\right] \\
&\overset{①}{=} \mathbb{E}_{\xi^{(r)}}\left[\left\|\sum_{c=0}^{\tau-1}\left[\tilde{\mathbf{g}}_j^{(c,r)} - \mathbf{g}_j^{(c,r)}\right]\right\|^2\right] \\
&= \mathrm{Var}\left(\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\right) \\
&\overset{②}{=} \sum_{c=0}^{\tau-1}\mathrm{Var}\left(\tilde{\mathbf{g}}_j^{(c,r)}\right) \\
&= \sum_{c=0}^{\tau-1}\mathbb{E}\left[\left\|\tilde{\mathbf{g}}_j^{(c,r)} - \mathbf{g}_j^{(c,r)}\right\|^2\right] \\
&\overset{③}{\leq} \tau\sigma^2
\end{aligned} \tag{7}$$

where in ① we use the definition of $\tilde{\mathbf{g}}_j^{(r)}$ and $\mathbf{g}_j^{(r)}$, in ② we use the fact that mini-batches are chosen in i.i.d. manner at each local machine, and ③ immediately follows from Assumptions 3.

Replacing $\mathbb{E}_{\xi^{(r)}}\left[\|\tilde{\mathbf{g}}_j^{(r)} - \mathbf{g}_j^{(r)}\|^2\right]$ in (5) by its upper bound in (6) implies that

$$\mathbb{E}_{\xi^{(r)}|\boldsymbol{w}^{(r)}}\mathbb{E}_{\mathbf{S},\mathcal{K}}\Big[\|\frac{1}{k}\sum_{j\in\mathcal{K}}\mathbf{S}\left(\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\right)\|^2\Big] \leq (\omega+1)\frac{\tau\sigma^2}{k} + (\frac{\omega}{k}+1)\sum_{j=1}^{p}q_j\|\mathbf{g}_j^{(r)}\|^2 \tag{8}$$

Further note that we have

$$\left\|\mathbf{g}_j^{(r)}\right\|^2 = \|\sum_{c=0}^{\tau-1}\mathbf{g}_j^{(c,r)}\|^2 \leq \tau\sum_{c=0}^{\tau-1}\|\mathbf{g}_j^{(c,r)}\|^2 \tag{9}$$

where the last inequality is due to $\left\|\sum_{j=1}^{n}\mathbf{a}_i\right\|^2 \leq n\sum_{j=1}^{n}\|\mathbf{a}_i\|^2$, which together with (8) leads to the following bound:

$$\mathbb{E}_{\xi^{(r)}|\boldsymbol{w}^{(r)}}\mathbb{E}_{\mathbf{S}}\Big[\|\frac{1}{k}\sum_{j\in\mathcal{K}}\mathbf{S}\left(\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\right)\|^2\Big] \leq (\omega+1)\frac{\tau\sigma^2}{k} + \tau(\frac{\omega}{k}+1)\sum_{j=1}^{p}q_j\|\mathbf{g}_j^{(c,r)}\|^2, \tag{10}$$

and the proof is complete. □

Lemma 5. *Under Assumption 1, and according to the* `FedCOM` *algorithm the expected inner product between stochastic gradient and full batch gradient can be bounded with:*

$$-\mathbb{E}_{\xi,\mathbf{S},\mathcal{K}}\left[\left\langle\nabla f(\boldsymbol{w}^{(r)}), \tilde{\mathbf{g}}^{(r)}\right\rangle\right] \leq \frac{1}{2}\eta\frac{1}{m}\sum_{j=1}^{m}\sum_{c=0}^{\tau-1}\left[-\|\nabla f(\boldsymbol{w}^{(r)})\|_2^2 - \|\nabla f(\boldsymbol{w}_j^{(c,r)})\|_2^2 + L^2\|\boldsymbol{w}^{(r)} - \boldsymbol{w}_j^{(c,r)}\|_2^2\right] \tag{11}$$

Proof. We have:

$$
\begin{aligned}
&-\mathbb{E}_{\{\xi_1^{(t)},\ldots,\xi_m^{(t)}|\boldsymbol{w}_1^{(t)},\ldots,\boldsymbol{w}_m^{(t)}\}}\mathbb{E}_{\mathbf{S},\mathcal{K}}\left[\left\langle\nabla f(\boldsymbol{w}^{(r)}),\tilde{\mathbf{g}}_{\mathbf{S},\mathcal{K}}^{(r)}\right\rangle\right]\\
&=-\mathbb{E}_{\{\xi_1^{(t)},\ldots,\xi_m^{(t)}|\boldsymbol{w}_1^{(t)},\ldots,\boldsymbol{w}_m^{(t)}\}}\left[\left\langle\nabla f(\boldsymbol{w}^{(r)}),\eta\sum_{j\in\mathcal{K}}q_j\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\right\rangle\right]\\
&=-\left\langle\nabla f(\boldsymbol{w}^{(r)}),\eta\sum_{j=1}^{m}q_j\sum_{c=0}^{\tau-1}\mathbb{E}_{\xi,\mathbf{S}}\left[\tilde{\mathbf{g}}_{j,\mathbf{S}}^{(c,r)}\right]\right\rangle\\
&=-\eta\sum_{c=0}^{\tau-1}\sum_{j=1}^{m}q_j\left\langle\nabla f(\boldsymbol{w}^{(r)}),\mathbf{g}_j^{(c,r)}\right\rangle\\
&\overset{①}{=}\frac{1}{2}\eta\sum_{c=0}^{\tau-1}\sum_{j=1}^{m}q_j\left[-\|\nabla f(\boldsymbol{w}^{(r)})\|_2^2-\|\nabla f(\boldsymbol{w}_j^{(c,r)})\|_2^2+\|\nabla f(\boldsymbol{w}^{(r)})-\nabla f(\boldsymbol{w}_j^{(c,r)})\|_2^2\right]\\
&\overset{②}{\leq}\frac{1}{2}\eta\sum_{c=0}^{\tau-1}\sum_{j=1}^{m}q_j\left[-\|\nabla f(\boldsymbol{w}^{(r)})\|_2^2-\|\nabla f(\boldsymbol{w}_j^{(c,r)})\|_2^2+L^2\|\boldsymbol{w}^{(r)}-\boldsymbol{w}_j^{(c,r)}\|_2^2\right]
\end{aligned}
\tag{12}
$$

where ① is due to $2\langle\mathbf{a},\mathbf{b}\rangle=\|\mathbf{a}\|^2+\|\mathbf{b}\|^2-\|\mathbf{a}-\mathbf{b}\|^2$, and ② follows from Assumption 1. □

The following lemma bounds the distance of local solutions from global solution at rth communication round.

Lemma 6. *Under Assumptions 3 we have:*

$$
\mathbb{E}\left[\|\boldsymbol{w}^{(r)}-\boldsymbol{w}_j^{(c,r)}\|_2^2\right]\leq\eta^2\tau\sum_{c=0}^{\tau-1}\left\|\mathbf{g}_j^{(c,r)}\right\|_2^2+\eta^2\tau\sigma^2
$$

Proof. Note that

$$
\begin{aligned}
\mathbb{E}\left[\left\|\boldsymbol{w}^{(r)}-\boldsymbol{w}_j^{(c,r)}\right\|_2^2\right]&=\mathbb{E}\left[\left\|\boldsymbol{w}^{(r)}-\left(\boldsymbol{w}^{(r)}-\eta\sum_{k=0}^{c}\tilde{\mathbf{g}}_j^{(k,r)}\right)\right\|_2^2\right]\\
&=\mathbb{E}\left[\left\|\eta\sum_{k=0}^{c}\tilde{\mathbf{g}}_j^{(k,r)}\right\|_2^2\right]\\
&\overset{①}{=}\mathbb{E}\left[\left\|\eta\sum_{k=0}^{c}\left(\tilde{\mathbf{g}}_j^{(k,r)}-\mathbf{g}_j^{(k,r)}\right)\right\|_2^2\right]+\left[\left\|\eta\sum_{k=0}^{c}\mathbf{g}_j^{(k,r)}\right\|_2^2\right]\\
&\overset{②}{=}\eta^2\sum_{k=0}^{c}\mathbb{E}\left[\left\|\left(\tilde{\mathbf{g}}_j^{(k,r)}-\mathbf{g}_j^{(k,r)}\right)\right\|_2^2\right]+(c+1)\,\eta^2\sum_{k=0}^{c}\left[\left\|\mathbf{g}_j^{(k,r)}\right\|_2^2\right]\\
&\leq\eta^2\sum_{k=0}^{\tau-1}\mathbb{E}\left[\left\|\left(\tilde{\mathbf{g}}_j^{(k,r)}-\mathbf{g}_j^{(k,r)}\right)\right\|_2^2\right]+\tau\eta^2\sum_{k=0}^{\tau-1}\left[\left\|\mathbf{g}_j^{(k,r)}\right\|_2^2\right]\\
&\overset{③}{\leq}\eta^2\sum_{k=0}^{\tau-1}\sigma^2+\tau\eta^2\sum_{k=0}^{\tau-1}\left[\left\|\mathbf{g}_j^{(k,r)}\right\|_2^2\right]\\
&=\eta^2\tau\sigma^2+\eta^2\sum_{k=0}^{\tau-1}\tau\left\|\mathbf{g}_j^{(k,r)}\right\|_2^2
\end{aligned}
\tag{13}
$$

where ① comes from $\mathbb{E}\left[\mathbf{x}^2\right]=\mathrm{Var}\left[\mathbf{x}\right]+\left[\mathbb{E}\left[\mathbf{x}\right]\right]^2$ and ② holds because $\mathrm{Var}\left(\sum_{j=1}^{n}\mathbf{x}_j\right)=\sum_{j=1}^{n}\mathrm{Var}\left(\mathbf{x}_j\right)$ for i.i.d. vectors $\mathbf{x}_i$ (and i.i.d. assumption comes from i.i.d. sampling), and finally ③ follows from Assumption 3. □

C.1 Main result for the non-convex setting

Now we are ready to present our result for the homogeneous setting. We first state and prove the result for the general non-convex objectives.

Theorem 4 (non-convex)**.** *For* `FedSKETCH`(τ, η, γ)*, for all* $0 \leq t \leq R\tau - 1$*, under Assumptions 1 to 3, if the learning rate satisfies*

$$1 \geq \tau^2 L^2 \eta^2 + \left(\frac{\omega}{k} + 1\right) \eta\gamma L\tau \tag{14}$$

and all local model parameters are initialized at the same point $\boldsymbol{w}^{(0)}$*, then the average-squared gradient after* τ *iterations is bounded as follows:*

$$\frac{1}{R}\sum_{r=0}^{R-1}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2 \leq \frac{2\left(f(\boldsymbol{w}^{(0)}) - f(\boldsymbol{w}^{(*)})\right)}{\eta\gamma\tau R} + \frac{L\eta\gamma(\omega+1)}{k}\sigma^2 + L^2\eta^2\tau\sigma^2 \tag{15}$$

where $\boldsymbol{w}^{(*)}$ *is the global optimal solution with function value* $f(\boldsymbol{w}^{(*)})$*.*

Proof. Before proceeding to the proof of Theorem 4, we would like to highlight that

$$\boldsymbol{w}^{(r)} - \ \boldsymbol{w}_j^{(\tau,r)} = \eta \sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)}. \tag{16}$$

From the updating rule of Algorithm 4 we have

$$\boldsymbol{w}^{(r+1)} = \boldsymbol{w}^{(r)} - \gamma\eta\left(\frac{1}{k}\sum_{j\in\mathcal{K}}\mathbf{S}\Big(\sum_{c=0,r}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\Big)\right) = \boldsymbol{w}^{(r)} - \gamma\left[\frac{\eta}{k}\sum_{j\in\mathcal{K}}\mathbf{S}\left(\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\right)\right]$$

In what follows, we use the following notation to denote the stochastic gradient used to update the global model at rth communication round

$$\tilde{\mathbf{g}}_{\mathbf{S},\mathcal{K}}^{(r)} \triangleq \frac{\eta}{p}\sum_{j=1}^{p}\mathbf{S}\left(\frac{\boldsymbol{w}^{(r)} - \ \boldsymbol{w}_j^{(\tau,r)}}{\eta}\right) = \frac{1}{k}\sum_{j\in\mathcal{K}}\mathbf{S}\left(\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\right).$$

and notice that $\boldsymbol{w}^{(r)} = \boldsymbol{w}^{(r-1)} - \gamma\tilde{\mathbf{g}}^{(r)}$.

Then using the unbiased estimation property of sketching we have:

$$\mathbb{E}_{\mathbf{S}}\left[\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\right] = \frac{1}{k}\sum_{j\in\mathcal{K}}\left[-\eta\mathbb{E}_{\mathbf{S}}\left[\mathbf{S}\left(\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\right)\right]\right] = \frac{1}{k}\sum_{j\in\mathcal{K}}\left[-\eta\left(\sum_{c=0}^{\tau-1}\tilde{\mathbf{g}}_j^{(c,r)}\right)\right] \triangleq \tilde{\mathbf{g}}_{\mathbf{S},\mathcal{K}}^{(r)}$$

From the L-smoothness gradient assumption on global objective, by using $\tilde{\mathbf{g}}^{(r)}$ in inequality (16) we have:

$$f(\boldsymbol{w}^{(r+1)}) - f(\boldsymbol{w}^{(r)}) \leq -\gamma\big\langle\nabla f(\boldsymbol{w}^{(r)}), \tilde{\mathbf{g}}^{(r)}\big\rangle + \frac{\gamma^2 L}{2}\|\tilde{\mathbf{g}}^{(r)}\|^2 \tag{17}$$

By taking expectation on both sides of above inequality over sampling, we get:

$$\begin{aligned}\mathbb{E}\left[\mathbb{E}_{\mathbf{S}}\left[f(\boldsymbol{w}^{(r+1)}) - f(\boldsymbol{w}^{(r)})\right]\right] &\leq -\gamma\mathbb{E}\left[\mathbb{E}_{\mathbf{S}}\left[\big\langle\nabla f(\boldsymbol{w}^{(r)}), \tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\big\rangle\right]\right] + \frac{\gamma^2 L}{2}\mathbb{E}\left[\mathbb{E}_{\mathbf{S}}\|\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\|^2\right] \\ &\stackrel{(a)}{=} -\gamma\underbrace{\mathbb{E}\left[\left[\big\langle\nabla f(\boldsymbol{w}^{(r)}), \tilde{\mathbf{g}}^{(r)}\big\rangle\right]\right]}_{\mathrm{(I)}} + \frac{\gamma^2 L}{2}\underbrace{\mathbb{E}\left[\mathbb{E}_{\mathbf{S}}\left[\|\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\|^2\right]\right]}_{\mathrm{(II)}}\end{aligned} \tag{18}$$

We proceed to use Lemma 4, Lemma 5, and Lemma 6, to bound terms (I) and (II) in right hand side of (18), which gives

$$\begin{aligned}
&\mathbb{E}\left[\mathbb{E}_{\mathbf{S}}\left[f(\boldsymbol{w}^{(r+1)})-f(\boldsymbol{w}^{(r)})\right]\right]\\
\leq&\gamma\frac{1}{2}\eta\sum_{j=1}^{p}q_j\sum_{c=0}^{\tau-1}\left[-\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2-\left\|\mathbf{g}_j^{(c,r)}\right\|_2^2+L^2\eta^2\sum_{c=0}^{\tau-1}\left[\tau\left\|\mathbf{g}_j^{(c,r)}\right\|_2^2+\sigma^2\right]\right]\\
&+\frac{\gamma^2L(\frac{\omega}{k}+1)}{2}\left[\eta^2\tau\sum_{j=1}^{p}q_j\sum_{c=0}^{\tau-1}\|\mathbf{g}_j^{(c,r)}\|^2\right]+\frac{\gamma^2\eta^2L(\omega+1)}{2}\frac{\tau\sigma^2}{k}\\
\overset{①}{\leq}&\frac{\gamma\eta}{2}\sum_{j=1}^{p}q_j\sum_{c=0}^{\tau-1}\left[-\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2-\left\|\mathbf{g}_j^{(c,r)}\right\|_2^2+\tau L^2\eta^2\left[\tau\left\|\mathbf{g}_j^{(c,r)}\right\|_2^2+\sigma^2\right]\right]\\
&+\frac{\gamma^2L(\frac{\omega}{k}+1)}{2}\left[\eta^2\tau\sum_{j=1}^{p}q_j\sum_{c=0}^{\tau-1}\|\mathbf{g}_j^{(c,r)}\|^2\right]+\frac{\gamma^2\eta^2L(\omega+1)}{2}\frac{\tau\sigma^2}{k}\\
=&-\eta\gamma\frac{\tau}{2}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2\\
&-\left(1-\tau L^2\eta^2\tau-(\frac{\omega}{k}+1)\eta\gamma L\tau\right)\frac{\eta\gamma}{2}\sum_{j=1}^{p}q_j\sum_{c=0}^{\tau-1}\|\mathbf{g}_j^{(c,r)}\|^2+\frac{L\tau\gamma\eta^2}{2k}\left(kL\tau\eta+\gamma(\omega+1)\right)\sigma^2\\
\overset{②}{\leq}&-\eta\gamma\frac{\tau}{2}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2+\frac{L\tau\gamma\eta^2}{2k}\left(kL\tau\eta+\gamma(\omega+1)\right)\sigma^2
\end{aligned}\tag{19}$$

where in ① we incorporate outer summation $\sum_{c=0}^{\tau-1}$, and ② follows from condition

$$1\geq\tau L^2\eta^2\tau+(\frac{\omega}{k}+1)\eta\gamma L\tau.$$

Summing up for all R communication rounds and rearranging the terms gives:

$$\frac{1}{R}\sum_{r=0}^{R-1}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2\leq\frac{2\left(f(\boldsymbol{w}^{(0)})-f(\boldsymbol{w}^{(*)})\right)}{\eta\gamma\tau R}+\frac{L\eta\gamma(\omega+1)}{k}\sigma^2+L^2\eta^2\tau\sigma^2$$

From above inequality, is it easy to see that in order to achieve a linear speed up, we need to have $\eta\gamma=O\left(\frac{\sqrt{k}}{\sqrt{R\tau}}\right)$. ☐

Corollary 5 (Linear speed up)**.** *In (15) for the choice of $\eta\gamma=O\left(\frac{1}{L}\sqrt{\frac{k}{R\tau(\omega+1)}}\right)$, and $\gamma\geq k$ the convergence rate reduces to:*

$$\frac{1}{R}\sum_{r=0}^{R-1}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2\leq O\left(\frac{L\sqrt{(\omega+1)}\left(f(\boldsymbol{w}^{(0)})-f(\boldsymbol{w}^{*})\right)}{\sqrt{kR\tau}}+\frac{\left(\sqrt{(\omega+1)}\right)\sigma^2}{\sqrt{kR\tau}}+\frac{k\sigma^2}{R\gamma^2}\right).\tag{20}$$

Note that according to (20), if we pick a fixed constant value for γ, in order to achieve an ϵ-accurate solution, $R=O\left(\frac{1}{\epsilon}\right)$ communication rounds and $\tau=O\left(\frac{\omega+1}{k\epsilon}\right)$ local updates are necessary. We also highlight that (20) also allows us to choose $R=O\left(\frac{\omega+1}{\epsilon}\right)$ and $\tau=O\left(\frac{1}{k\epsilon}\right)$ to get the same convergence rate.

Remark 6. *Condition in (14) can be rewritten as*

$$\begin{aligned}
\eta&\leq\frac{-\gamma L\tau\left(\frac{\omega}{k}+1\right)+\sqrt{\gamma^2\left(L\tau\left(\frac{\omega}{k}+1\right)\right)^2+4L^2\tau^2}}{2L^2\tau^2}\\
&=\frac{-\gamma L\tau\left(\frac{\omega}{k}+1\right)+L\tau\sqrt{\left(\frac{\omega}{k}+1\right)^2\gamma^2+4}}{2L^2\tau^2}\\
&=\frac{\sqrt{\left(\frac{\omega}{k}+1\right)^2\gamma^2+4}-\left(\frac{\omega}{k}+1\right)\gamma}{2L\tau}
\end{aligned}\tag{21}$$

So based on (21), if we set $\eta = O\left(\frac{1}{L\gamma}\sqrt{\frac{p}{R\tau(\omega+1)}}\right)$*, it implies that:*

$$R \geq \frac{\tau k}{\left(\omega+1\right)\gamma^2\left(\sqrt{\left(\frac{\omega}{k}+1\right)^2\gamma^2+4}-\left(\frac{\omega}{k}+1\right)\gamma\right)^2} \tag{22}$$

We note that $\gamma^2\left(\sqrt{\left(\frac{\omega}{k}+1\right)^2\gamma^2+4}-\left(\frac{\omega}{k}+1\right)\gamma\right)^2 = \Theta(1) \leq 5$ *therefore even for* $\gamma \geq m$ *we need to have*

$$R \geq \frac{\tau k}{5\left(\omega+1\right)} = O\left(\frac{\tau k}{\left(\omega+1\right)}\right) \tag{23}$$

Therefore, for the choice of $\tau = O\left(\frac{\omega+1}{k\epsilon}\right)$*, due to condition in (23), we need to have* $R = O\left(\frac{1}{\epsilon}\right)$*. Similarly, we can have* $R = O\left(\frac{\omega+1}{\epsilon}\right)$ *and* $\tau = O\left(\frac{1}{k\epsilon}\right)$*.*

Corollary 6 (Special case, $\gamma = 1$)**.** *By letting* $\gamma = 1$*,* $\omega = 0$ *and* $k = p$ *the convergence rate in (15) reduces to*

$$\frac{1}{R}\sum_{r=0}^{R-1}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2 \leq \frac{2\left(f(\boldsymbol{w}^{(0)})-f(\boldsymbol{w}^{(*)})\right)}{\eta R\tau} + \frac{L\eta}{p}\sigma^2 + L^2\eta^2\tau\sigma^2$$

which matches the rate obtained in [51]. In this case the communication complexity and the number of local updates become

$$R = O\left(\frac{p}{\epsilon}\right), \quad \tau = O\left(\frac{1}{\epsilon}\right).$$

This simply implies that in this special case the convergence rate of our algorithm reduces to the rate obtained in [51], which indicates the tightness of our analysis.

C.2 Main result for the PL/Strongly convex setting

We now turn to stating the convergence rate for the homogeneous setting under PL condition which naturally leads to the same rate for strongly convex functions.

Theorem 5 (PL or strongly convex)**.** *For* `FedSKETCH`(τ,η,γ)*, for all* $0 \leq t \leq R\tau - 1$*, under Assumptions 1 to 3 and 2,if the learning rate satisfies*

$$1 \geq \tau^2 L^2\eta^2 + \left(\frac{\omega}{k}+1\right)\eta\gamma L\tau$$

and if the all the models are initialized with $\boldsymbol{w}^{(0)}$ *we obtain:*

$$\mathbb{E}\Big[f(\boldsymbol{w}^{(R)})-f(\boldsymbol{w}^{(*)})\Big] \leq \left(1-\eta\gamma\mu\tau\right)^R\left(f(\boldsymbol{w}^{(0)})-f(\boldsymbol{w}^{(*)})\right) + \frac{1}{\mu}\left[\frac{1}{2}L^2\tau\eta^2\sigma^2 + \left(1+\omega\right)\frac{\gamma\eta L\sigma^2}{2k}\right]$$

Proof. From (19) under condition:

$$1 \geq \tau L^2\eta^2\tau + (\frac{\omega}{k}+1)\eta\gamma L\tau$$

we obtain:

$$\begin{aligned}\mathbb{E}\Big[f(\boldsymbol{w}^{(r+1)})-f(\boldsymbol{w}^{(r)})\Big] &\leq -\eta\gamma\frac{\tau}{2}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2 + \frac{L\tau\gamma\eta^2}{2k}\left(kL\tau\eta+\gamma(\omega+1)\right)\sigma^2 \\ &\leq -\eta\mu\gamma\tau\left(f(\boldsymbol{w}^{(r)})-f(\boldsymbol{w}^{(r)})\right) + \frac{L\tau\gamma\eta^2}{2k}\left(kL\tau\eta+\gamma(\omega+1)\right)\sigma^2 \end{aligned} \tag{24}$$

which leads to the following bound:

$$\mathbb{E}\Big[f(\boldsymbol{w}^{(r+1)}) - f(\boldsymbol{w}^{(*)})\Big] \leq (1-\eta\mu\gamma\tau)\left[f(\boldsymbol{w}^{(r)}) - f(\boldsymbol{w}^{(*)})\right] + \frac{L\tau\gamma\eta^2}{2k}\left(kL\tau\eta + (\omega+1)\gamma\right)\sigma^2$$

By setting $\Delta = 1-\eta\mu\gamma\tau$ we obtain the following bound:

$$\begin{aligned}
&\mathbb{E}\Big[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\Big] \\
\leq &\Delta^R\Big[f(\boldsymbol{w}^{(0)}) - f(\boldsymbol{w}^{(*)})\Big] + \frac{1-\Delta^R}{1-\Delta}\frac{L\tau\gamma\eta^2}{2k}\left(kL\tau\eta + (\omega+1)\gamma\right)\sigma^2 \\
\leq &\Delta^R\Big[f(\boldsymbol{w}^{(0)}) - f(\boldsymbol{w}^{(*)})\Big] + \frac{1}{1-\Delta}\frac{L\tau\gamma\eta^2}{2k}\left(kL\tau\eta + (\omega+1)\gamma\right)\sigma^2 \\
= &(1-\eta\mu\gamma\tau)^R\Big[f(\boldsymbol{w}^{(0)}) - f(\boldsymbol{w}^{(*)})\Big] + \frac{1}{\eta\mu\gamma\tau}\frac{L\tau\gamma\eta^2}{2k}\left(kL\tau\eta + (\omega+1)\gamma\right)\sigma^2
\end{aligned} \tag{25}$$

□

Corollary 7. *If we let* $\eta\gamma\mu\tau \leq \frac{1}{2}$, $\eta = \frac{1}{2L\left(\frac{\omega}{k}+1\right)\tau\gamma}$ *and* $\kappa = \frac{L}{\mu}$ *the convergence error in Theorem 5, with* $\gamma \geq k$ *results in:*

$$\begin{aligned}
&\mathbb{E}\Big[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\Big] \\
\leq &e^{-\eta\gamma\mu\tau R}\left(f(\boldsymbol{w}^{(0)}) - f(\boldsymbol{w}^{(*)})\right) + \frac{1}{\mu}\left[\frac{1}{2}\tau L^2\eta^2\sigma^2 + (1+\omega)\frac{\gamma\eta L\sigma^2}{2k}\right] \\
\leq &e^{-\frac{R}{2\left(\frac{\omega}{k}+1\right)\kappa}}\left(f(\boldsymbol{w}^{(0)}) - f(\boldsymbol{w}^{(*)})\right) + \frac{1}{\mu}\left[\frac{1}{2}L^2\frac{\tau\sigma^2}{L^2\left(\frac{\omega}{k}+1\right)^2\gamma^2\tau^2} + \frac{(1+\omega)L\sigma^2}{2\left(\frac{\omega}{k}+1\right)L\tau k}\right] \\
= &O\left(e^{-\frac{R}{2\left(\frac{\omega}{k}+1\right)\kappa}}\left(f(\boldsymbol{w}^{(0)}) - f(\boldsymbol{w}^{(*)})\right) + \frac{\sigma^2}{\left(\frac{\omega}{k}+1\right)^2\gamma^2\mu\tau} + \frac{(\omega+1)\sigma^2}{\mu\left(\frac{\omega}{k}+1\right)\tau k}\right) \\
= &O\left(e^{-\frac{R}{2\left(\frac{\omega}{k}+1\right)\kappa}}\left(f(\boldsymbol{w}^{(0)}) - f(\boldsymbol{w}^{(*)})\right) + \frac{\sigma^2}{\gamma^2\mu\tau} + \frac{(\omega+1)\sigma^2}{\mu\left(\frac{\omega}{k}+1\right)\tau k}\right)
\end{aligned} \tag{26}$$

which indicates that to achieve an error of ϵ*, we need to have* $R = O\left(\left(\frac{\omega}{k}+1\right)\kappa\log\left(\frac{1}{\epsilon}\right)\right)$ *and* $\tau = \frac{(\omega+1)}{k\left(\frac{\omega}{k}+1\right)\epsilon}$*. Additionally, we note that if* $\gamma \to \infty$*, yet* $R = O\left(\left(\frac{\omega}{k}+1\right)\kappa\log\left(\frac{1}{\epsilon}\right)\right)$ *and* $\tau = \frac{(\omega+1)}{k\left(\frac{\omega}{k}+1\right)\epsilon}$ *will be necessary.*

C.3 Main result for the general convex setting

Theorem 6 (Convex)**.** *For a general convex function $f(\boldsymbol{w})$ with optimal solution $\boldsymbol{w}^{(*)}$, using* `FedSKETCH`(τ,η,γ) *to optimize $\tilde{f}(\boldsymbol{w},\phi)=f(\boldsymbol{w})+\frac{\phi}{2}\|\boldsymbol{w}\|^2$, for all $0\leq t\leq R\tau-1$, under Assumptions 1 to 3, if the learning rate satisfies*

$$1\geq\tau^2L^2\eta^2+\left(\frac{\omega}{k}+1\right)\eta\gamma L\tau$$

and if the all the models initiate with $\boldsymbol{w}^{(0)}$, with $\phi=\frac{1}{\sqrt{k\tau}}$ and $\eta=\frac{1}{2L\gamma\tau\left(1+\frac{\omega}{k}\right)}$ we obtain:

$$\begin{aligned}\mathbb{E}\Big[f(\boldsymbol{w}^{(R)})-f(\boldsymbol{w}^{(*)})\Big]&\leq e^{-\frac{R}{2L\left(1+\frac{\omega}{k}\right)\sqrt{m\tau}}}\left(f(\boldsymbol{w}^{(0)})-f(\boldsymbol{w}^{(*)})\right)\\&\quad+\left[\frac{\sqrt{k}\sigma^2}{8\sqrt{\tau}\gamma^2\left(1+\frac{\omega}{k}\right)^2}+\frac{\left(\omega+1\right)\sigma^2}{4\left(\frac{\omega}{k}+1\right)\sqrt{k\tau}}\right]+\frac{1}{2\sqrt{k\tau}}\left\|\boldsymbol{w}^{(*)}\right\|^2\end{aligned}\tag{27}$$

We note that above theorem implies that to achieve a convergence error of ϵ we need to have $R=O\left(L\left(1+\frac{\omega}{k}\right)\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right)$ and $\tau=O\left(\frac{(\omega+1)^2}{k\left(\frac{\omega}{k}+1\right)^2\epsilon}\right)$.

Proof. Since $\tilde{f}(\boldsymbol{w}^{(r)},\phi)=f(\boldsymbol{w}^{(r)})+\frac{\phi}{2}\left\|\boldsymbol{w}^{(r)}\right\|^2$ is ϕ-PL, according to Theorem 5, we have:

$$\begin{aligned}&\tilde{f}(\boldsymbol{w}^{(R)},\phi)-\tilde{f}(\boldsymbol{w}^{(*)},\phi)\\&=f(\boldsymbol{w}^{(r)})+\frac{\phi}{2}\left\|\boldsymbol{w}^{(r)}\right\|^2-\left(f(\boldsymbol{w}^{(*)})+\frac{\phi}{2}\left\|\boldsymbol{w}^{(*)}\right\|^2\right)\\&\leq(1-\eta\gamma\phi\tau)^R\left(f(\boldsymbol{w}^{(0)})-f(\boldsymbol{w}^{(*)})\right)+\frac{1}{\phi}\left[\frac{1}{2}L^2\tau\eta^2\sigma^2+(1+\omega)\frac{\gamma\eta L\sigma^2}{2k}\right]\end{aligned}\tag{28}$$

Next rearranging (28) and replacing μ with ϕ leads to the following error bound:

$$\begin{aligned}&f(\boldsymbol{w}^{(R)})-f^*\\&\leq(1-\eta\gamma\phi\tau)^R\left(f(\boldsymbol{w}^{(0)})-f(\boldsymbol{w}^{(*)})\right)+\frac{1}{\phi}\left[\frac{1}{2}L^2\tau\eta^2\sigma^2+(1+\omega)\frac{\gamma\eta L\sigma^2}{2k}\right]\\&\quad+\frac{\phi}{2}\left(\|\boldsymbol{w}^*\|^2-\left\|\boldsymbol{w}^{(r)}\right\|^2\right)\\&\leq e^{-(\eta\gamma\phi\tau)R}\left(f(\boldsymbol{w}^{(0)})-f(\boldsymbol{w}^{(*)})\right)+\frac{1}{\phi}\left[\frac{1}{2}L^2\tau\eta^2\sigma^2+(1+\omega)\frac{\gamma\eta L\sigma^2}{2k}\right]+\frac{\phi}{2}\left\|\boldsymbol{w}^{(*)}\right\|^2\end{aligned}$$

Next, if we set $\phi=\frac{1}{\sqrt{k\tau}}$ and $\eta=\frac{1}{2\left(1+\frac{\omega}{k}\right)L\gamma\tau}$, we obtain that

$$\begin{aligned}&f(\boldsymbol{w}^{(R)})-f^*\\&\leq e^{-\frac{R}{2\left(1+\frac{\omega}{k}\right)L\sqrt{m\tau}}}\left(f(\boldsymbol{w}^{(0)})-f(\boldsymbol{w}^{(*)})\right)+\sqrt{k\tau}\left[\frac{\sigma^2}{8\tau\gamma^2\left(1+\frac{\omega}{k}\right)^2}+\frac{\left(\omega+1\right)\sigma^2}{4\left(\frac{\omega}{k}+1\right)\tau k}\right]+\frac{1}{2\sqrt{k\tau}}\left\|\boldsymbol{w}^{(*)}\right\|^2,\end{aligned}$$

thus the proof is complete. □

D Proof of Main Theorems

The proof of Theorem 1 follows directly from the results in [16]. For the sake of the completeness we review an assumptions from this reference for the quantization with their notation.

Assumption 5 ([16]). *The output of the compression operator $Q(\mathbf{x})$ is an unbiased estimator of its input $\mathbf{x}$, and its variance grows with the squared of the squared of ℓ_2-norm of its argument, i.e., $\mathbb{E}[Q(\mathbf{x})] = \mathbf{x}$ and $\mathbb{E}\left[\|Q(\mathbf{x}) - \mathbf{x}\|^2\right] \leq \omega \|\mathbf{x}\|^2$.*

D.1 Proof of Theorem 1

Based on Assumption 5 we have:

Theorem 7 ([16]). *Consider* `FedCOM` *in [16]. Suppose that the conditions in Assumptions 1, 3 and 5 hold. If the local data distributions of all users are identical (homogeneous setting), then we have*

- ***non-convex:*** *By choosing stepsizes as $\eta = \frac{1}{L\gamma}\sqrt{\frac{p}{R\tau\left(\frac{\omega}{p}+1\right)}}$ and $\gamma \geq p$, the sequence of iterates satisfies $\frac{1}{R}\sum_{r=0}^{R-1}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2 \leq \epsilon$ if we set $R = O\left(\frac{1}{\epsilon}\right)$ and $\tau = O\left(\frac{\frac{\omega}{p}+1}{p\epsilon}\right)$.*
- ***Strongly convex or PL:*** *By choosing stepsizes as $\eta = \frac{1}{2L\left(\frac{\omega}{p}+1\right)\tau\gamma}$ and $\gamma \geq m$, we obtain that the iterates satisfy $\mathbb{E}\left[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\right] \leq \epsilon$ if we set $R = O\left(\left(\frac{\omega}{p}+1\right)\kappa\log\left(\frac{1}{\epsilon}\right)\right)$ and $\tau = O\left(\frac{1}{p\epsilon}\right)$.*
- ***Convex:*** *By choosing stepsizes as $\eta = \frac{1}{2L\left(\frac{\omega}{p}+1\right)\tau\gamma}$ and $\gamma \geq p$, we obtain that the iterates satisfy $\mathbb{E}\left[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\right] \leq \epsilon$ if we set $R = O\left(\frac{L\left(1+\frac{\omega}{p}\right)}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right)$ and $\tau = O\left(\frac{1}{p\epsilon^2}\right)$.*

Proof. Since the sketching `PRIVIX` and `HEAPRIX`, satisfy Assumption 5 with $\omega = \mu^2 d$ and $\omega = \mu^2 d - 1$ respectively with probability $1-\delta$. Therefore, all the results in Theorem 1, conclude from Theorem 7 with probability $1-\delta$ and plugging $\omega = \mu^2 d$ and $\omega = \mu^2 d - 1$ respectively into the corresponding convergence bounds. □

D.2 Proof of Theorem 2

For the heterogeneous setting, the results in [16] requires the following extra assumption that naturally holds for the sketching:

Assumption 6 ([16]). *The compression scheme Q for the heterogeneous data distribution setting satisfies the following condition $\mathbb{E}_Q[\|\frac{1}{m}\sum_{j=1}^m Q(\boldsymbol{x}_j)\|^2 - \|Q(\frac{1}{m}\sum_{j=1}^m \boldsymbol{x}_j)\|^2] \leq G_q$.*

We note that since sketching is a linear compressor, in the case of our algorithms for heterogeneous setting we have $G_q = 0$.

Next, we restate the Theorem in [16] here as follows:

Theorem 8. *Consider* `FedCOMGATE` *in [16]. If Assumptions 1, 4, 5 and 6 hold, then even for the case the local data distribution of users are different (heterogeneous setting) we have*

- ***non-convex:*** *By choosing stepsizes as* $\eta = \frac{1}{L\gamma}\sqrt{\frac{p}{R\tau(\omega+1)}}$ *and* $\gamma \geq p$*, we obtain that the iterates satisfy* $\frac{1}{R}\sum_{r=0}^{R-1}\left\|\nabla f(\boldsymbol{w}^{(r)})\right\|_2^2 \leq \epsilon$ *if we set* $R = O\left(\frac{\omega+1}{\epsilon}\right)$ *and* $\tau = O\left(\frac{1}{p\epsilon}\right)$.

- ***Strongly convex or PL:*** *By choosing stepsizes as* $\eta = \frac{1}{2L(\frac{\omega}{p}+1)\tau\gamma}$ *and* $\gamma \geq \sqrt{p\tau}$*, we obtain that the iterates satisfy* $\mathbb{E}\Big[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\Big] \leq \epsilon$ *if we set* $R = O\left((\omega+1)\,\kappa\log\left(\frac{1}{\epsilon}\right)\right)$ *and* $\tau = O\left(\frac{1}{p\epsilon}\right)$.

- ***Convex:*** *By choosing stepsizes as* $\eta = \frac{1}{2L(\omega+1)\tau\gamma}$ *and* $\gamma \geq \sqrt{p\tau}$*, we obtain that the iterates satisfy* $\mathbb{E}\Big[f(\boldsymbol{w}^{(R)}) - f(\boldsymbol{w}^{(*)})\Big] \leq \epsilon$ *if we set* $R = O\left(\frac{L(1+\omega)}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right)$ *and* $\tau = O\left(\frac{1}{p\epsilon^2}\right)$.

Proof. Since the sketching methods `PRIVIX` and `HEAPRIX`, satisfy the Assumption 5 with $\omega = \mu^2 d$ and $\omega = \mu^2 d - 1$ respectively with probability $1-\delta$, we conclude the proofs of Theorem 2 using Theorem 8 with probability $1-\delta$ and plugging $\omega = \mu^2 d$ and $\omega = \mu^2 d - 1$ respectively into the convergence bounds. $\square$

E Additional Plots for the Numerical Experiments

E.1 Homogeneous setting

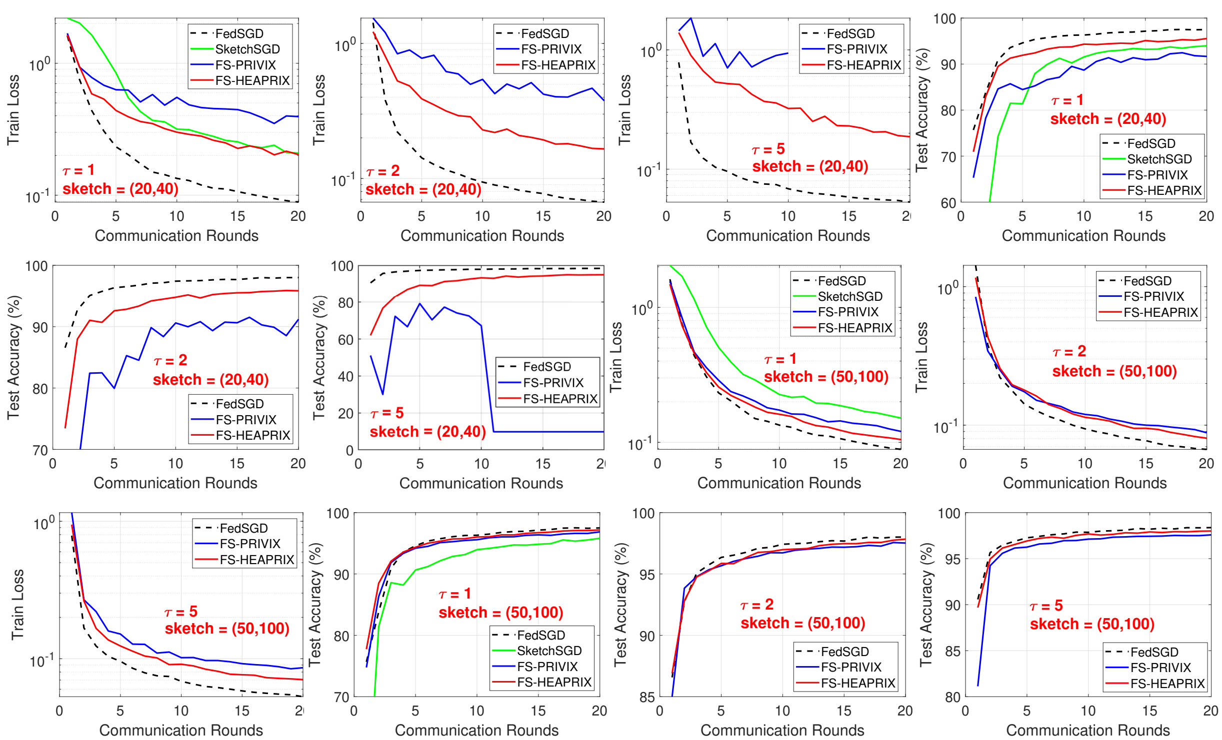

Figure 3 Homogeneous case: Comparison of compressed optimization methods on LeNet CNN architecture.

E.2 Heterogeneous setting

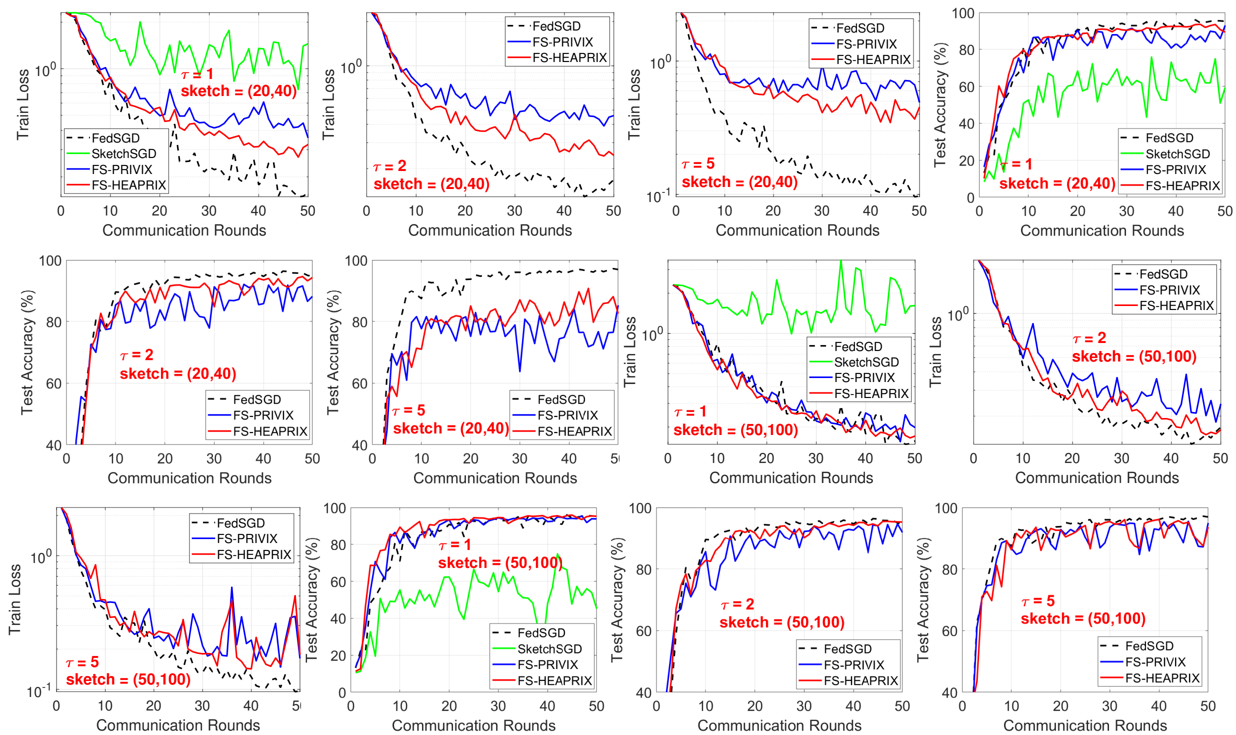

Figure 4 Heterogeneous case: Comparison of compressed optimization algorithms on LeNet CNN architecture.