STL-SGD: Speeding Up Local SGD with Stagewise Communication Period

Shuheng Shen[1], Yifei Cheng[1], Jingchang Liu[2], and Linli Xu*[1]

[1]University of Science and Technology of China
[2]The Hong Kong University of Science and Technology

Abstract

Distributed parallel stochastic gradient descent algorithms are workhorses for large scale machine learning tasks. Among them, local stochastic gradient descent (Local SGD) has attracted significant attention due to its low communication complexity. Previous studies prove that the communication complexity of Local SGD with a fixed or an adaptive communication period is in the order of $O(N^{\frac{3}{2}}T^{\frac{1}{2}})$ and $O(N^{\frac{3}{4}}T^{\frac{3}{4}})$ when the data distributions on clients are identical (IID) or otherwise (Non-IID). In this paper, to accelerate the convergence by reducing the communication complexity, we propose *ST*agewise *L*ocal *SGD* (STL-SGD), which increases the communication period gradually along with decreasing learning rate. We prove that STL-SGD can keep the same convergence rate and linear speedup as mini-batch SGD. In addition, as the benefit of increasing the communication period, when the objective is strongly convex or satisfies the Polyak-Łojasiewicz condition, the communication complexity of STL-SGD is $O(N\log T)$ and $O(N^{\frac{1}{2}}T^{\frac{1}{2}})$ for the IID case and the Non-IID case respectively, achieving significant improvements over Local SGD. Experiments on both convex and non-convex problems demonstrate the superior performance of STL-SGD.

1 Introduction

We consider the task of distributed stochastic optimization, which employs N clients to solve the following empirical risk minimization problem:

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

where $f_i(x) := \frac{1}{|\mathcal{D}_i|}\sum_{\xi\in\mathcal{D}_i} f(x,\xi)$ is the local objective of client i. $\mathcal{D}_i$'s denote the data distributions among clients, which can be possibly different. Specifically, the scenario where $\mathcal{D}_i$'s are identical corresponds to a central problem of traditional distributed optimization. When they are not identical, Formulation (1) captures the federated learning setting [26, 15], where the local data in each mobile client is independent and private, resulting in high variance of the data distributions.

As representatives of distributed stochastic optimization methods, traditional Synchronous SGD (SyncSGD) [6, 7] and Asynchronous SGD (AsyncSGD) [1, 23] achieve linear speedup theoretically with respect to the number of clients. Nevertheless, for both SyncSGD and AsyncSGD, communication needs to be conducted at each iteration and $O(d)$ parameters are communicated each time, incurring significant communication cost which restricts the performance in terms of time speedup.

*Corresponding author: linlixu@ustc.edu.cn

Preprint. Under review.

To address this dilemma, distributed algorithms with low communication cost, either by decreasing the communication frequency [35, 30, 39, 28] or by reducing the communication bits in each round [2, 31, 33], become widely applied for large scale training.

Among them, Local SGD [30] (also called FedAvg [26]), which conducts communication every k iterations, enjoys excellent theoretical and practical performance [25, 30]. In the IID case and the Non-IID case, the communication complexity of Local SGD is respectively proved to be $O(N^{\frac{3}{2}}T^{\frac{1}{2}})$ [35, 30] and $O(N^{\frac{3}{4}}T^{\frac{3}{4}})$ [39, 28], where T is the number of iterations, while the linear speedup is maintained. When the objective satisfies the Polyak-Łojasiewicz condition [16], [9] provides a tighter theoretical analysis which shows that the communication complexity of Local SGD is $O(N^{\frac{1}{3}}T^{\frac{1}{3}})$. In terms of the communication period k, most previous studies of Local SGD choose to fix it through the iterations. In contrast, [34] suggests using an adaptively decreasing k when the learning rate is fixed, and [9] proposes an adaptively increasing k as the iterations go on. Nevertheless, none of them achieve a communication complexity lower than $O(N^{\frac{1}{3}}T^{\frac{1}{3}})$. For strongly convex objectives, if a small fixed learning rate is adopted, Local SGD with fixed communication period is proved to achieve $O(N\log(NT))$ [32] communication complexity. However, the small fixed learning rate results in suboptimal convergence rate $O(\frac{\log T}{NT})$. It remains an open problem as to whether the communication complexity can be further reduced with a varying k when the optimal convergence rate $O(\frac{1}{NT})$ is maintained, to which this paper provides an affirmative solution.

Main Contributions. We propose Stagewise Local SGD (STL-SGD), which adopts a stagewisely increasing communication period , and make the following contributions:

- We first prove that Local SGD achieves $O(\frac{1}{\sqrt{NT}})$ convergence when the objective is general convex. A novel insight from this analysis is that, the convergence rate $O(\frac{1}{\sqrt{NT}})$ can be attained when setting k to be $O(\frac{1}{\eta N})$ and $O(\frac{1}{\sqrt{\eta N}})$ in the IID case and the Non-IID case respectively, where η is the learning rate. This indicates that the communication period is negatively relevant to the learning rate.
- Taking Local SGD as a subalgorithm and tuning its parameters stagewisely, we propose STL-SGD^{sc} for strongly convex problems, which geometrically increases the communication period along with decreasing learning rate. We prove that STL-SGD^{sc} achieves $O(\frac{1}{NT})$ convergence rate with communication complexities $O(N\log T)$ and $O(N^{\frac{1}{2}}T^{\frac{1}{2}})$ for the IID case and the Non-IID case, respectively.
- For non-convex problems, we propose the STL-SGD^{nc} algorithm, which uses Local SGD to optimize a regularized objective $f_{x_s}^{\gamma}(\cdot)$ inexactly at each stage. When the Polyak-Łojasiewicz condition holds, the same communication complexity as in strongly convex problems is achieved. For general non-convex problems, we prove that STL-SGD^{nc} achieves the linear speedup with communication complexities $O(N^{\frac{3}{2}}T^{\frac{1}{2}})$ and $O(N^{\frac{3}{4}}T^{\frac{3}{4}})$ for the IID case and the Non-IID case, respectively.

2 Related Works

Local SGD. When the data distributions on clients are identical, Local SGD is proved to achieve $O(\frac{1}{NT})$ convergence for strongly convex objectives [30] and $O(\frac{1}{\sqrt{NT}})$ convergence for non-convex objectives [35] when the communication period k satisfies $k \leq O(T^{\frac{1}{2}}/N^{\frac{3}{2}})$. As demonstrated in these results, Local SGD achieves a linear speedup with the communication complexity $O(N^{\frac{3}{2}}T^{\frac{1}{2}})$ for both strongly convex and non-convex objectives in the IID case. In addition, [9] justifies that $O(N^{\frac{1}{3}}T^{\frac{1}{3}})$ rounds of communication are sufficient to achieve $O(\frac{1}{NT})$ convergence for objectives which satisfy the Polyak-Łojasiewicz condition. On the other hand, for the Non-IID case, Local SGD is proved with a $O(1/\sqrt{NT})$ convergence rate under a communication complexity of $O(N^{\frac{3}{4}}T^{\frac{3}{4}})$ for non-convex objectives [39, 28]. Meanwhile, for strongly convex objectives, a suboptimal convergence rate of $O(\frac{k^2}{\mu NT})$ [22] is obtained. Beyond that, when a small fixed learning rate is adopted, [32] and [17] prove that the communication complexity of Local SGD is $O(N\log(NT))$ and $O(N^{\frac{1}{2}}T^{\frac{1}{2}})$ for the IID case and the Non-IID case respectively, at the cost of a suboptimal convergence rate $O(\frac{\log T}{NT})$. For general non-convex objectives, [11] proves a lower communication

complexity of $O(N^{\frac{3}{2}}T^{\frac{1}{2}})$ for the Non-IID case under the assumption of bounded gradient diversity. From the practical view, [41] suggests to communicate more frequently in the beginning of the optimization process, and [9] verifies that using a geometrically increasing period does not harm the convergence notably.

Stagewisely Training. For training both strongly convex and non-convex objectives, stagewisely decreasing the learning rate is widely adopted. Epoch-SGD [12] and ASSG [36] use SGD as their subalgorithm and geometrically decrease the learning rate stage by stage. They are proved to achieve the optimal $O(1/T)$ convergence for stochastic strongly convex optimization. For training neural networks, stagewisely decreasing the learning rate [21, 13] is a very important trick. From a theoretical aspect, stagewise SGD is proved with $O(1/\sqrt{T})$ convergence for both general and composite non-convex objectives [3, 4, 5], by adopting SGD to optimize a regularized objective at each stage and decreasing the learning rate linearly stage by stage. Stagewise training is also verified to achieve better testing error than general SGD [40].

Large Batch SGD (LB-SGD). SyncSGD with extremely large batch is proved to achieve a linear speedup with respect to the batch size [32]. Nevertheless, [14] shows that increasing the batch size does not help when the bias dominates the variance. It is also observed from practice that LB-SGD leads to a poor generalization [18, 8, 37]. [38] proposes CR-PSGD which increases the batch size geometrically step by step and proves that CR-PSGD achieves a linear speedup with $O(\log T)$ communication complexity. However, after a large number of iterations, CR-PSGD essentially becomes GD and loses the benefit of SGD.

Local SGD with Varaiance Reduction. Recently, several techniques are proposed to reduce the communication complexity of Local SGD in the Non-IID case. [10] shows that using redundant data among clients yields lower communication complexity. One variant of Local SGD called VRL-SGD [24] incorporates the variance reduction technique and is proved to achieve a $O(N^{\frac{3}{2}}T^{\frac{1}{2}})$ communication complexity for non-convex objectives. SCAFFOLD [17] extends VRL-SGD by involving two separate learning rates, and is proved to achieve $O(\log(NT))$ and $O(N^{\frac{1}{2}}T^{\frac{1}{2}})$ communication complexities for strongly convex objectives and non-convex objectives respectively. As SCAFFOLD adopts a small fixed learning rate, its convergence rate for strongly convex objectives is $O(\frac{\log T}{NT})$, which is suboptimal. Nevertheless, these methods are orthogonal to our study. Combining STL-SGD and variance reduction exceeds the scope of this paper.

For a comprehensive and detailed comparison of STL-SGD and the related works, please refer to Table 3 in the Appendix.

3 Preliminaries

3.1 Notations and Definitions

Throughout the paper, we let $\|\cdot\|$ indicate the ℓ_2 norm of a vector and $\langle\cdot,\cdot\rangle$ indicate the inner product of two vectors. The set $\{1, 2, \cdots, n\}$ is denoted as $[n]$. We use x^* to represent the optimal solution of (1). ∇f represents the gradient of f. $\mathbb{E}$ indicates a full expectation with respect to all the randomness in the algorithm (the stochastic gradients sampled in all iterations and the randomness in return).

The data distributions on different clients may not be identical. To quantify the difference of distributions, we define $\zeta_f^* := \frac{1}{N}\sum_{i=1}^N \|\nabla f_i(x^*)\|^2 = \frac{1}{N}\sum_{i=1}^N \|\nabla f_i(x^*) - \nabla f(x^*)\|^2$, which represents the variance of gradients among clients at x^*. Some literatures assume that the variance of gradients among clients is bounded by a constant ζ^2 [28] or the norm of stochastic gradients is bounded by a constant G^2 [39, 22]. Note that both ζ^2 and G^2 are larger than ζ_f^*. When the data distributions are identical, we have $\|\nabla f_i(x^*)\|^2 = 0$, thus it holds that $\zeta_f^* = 0$.

To state the convergence of algorithms for solving (1), we introduce some definitions, which can be also found in other works [4, 9].

Definition 1 (ρ-weakly convex)**.** *A non-convex function $f(x)$ is ρ-weakly convex ($\rho > 0$) if*

$$f(x) \geq f(y) + \langle \nabla f(y), x - y\rangle - \frac{\rho}{2}\|x - y\|^2, \forall x, y \in R^d.$$

Algorithm 1 Local-SGD(f, x_0, η, T, k)

Initialize: $x_0^i = x_0, \forall i \in [N]$.

for $t = 1, ..., T$ **do**
  Client C_i does:
  Uniformly sample a mini-batch $\xi_{t-1}^i \in \mathcal{D}_i$ and calculate a stochastic gradient $\nabla f_i(x_{t-1}^i, \xi_{t-1}^i)$.
  if t divides k **then**
    Communicate with other clients and update: $x_t^i = \sum_{j=1}^N \frac{1}{N}(x_{t-1}^j - \eta \nabla f(x_{t-1}^j, \xi_{t-1}^j))$.
  else
    Update locally: $x_t^i = x_{t-1}^i - \eta \nabla f_i(x_{t-1}^i, \xi_{t-1}^i)$.
  end if
end for
return $\tilde{x} = \frac{1}{N}\sum_{i=1}^N x_t^i$ for the randomly chosen $t \in \{0, 1, \cdots, T-1\}$.

Definition 2 (μ-Polyak-Łojasiewicz (PL))**.** *A function $f(x)$ satisfies the μ-PL condition ($\mu > 0$) if*

$$2\mu(f(x) - f(x^*)) \leq \|\nabla f(x)\|^2, \forall x \in R^d.$$

3.2 Assumptions

Throughout this paper, we make the following assumptions, all of which are commonly used and basic assumptions [30, 39, 22, 4, 3].

Assumption 1. *$f_i(x)$ is L-smooth in terms of $i \in [N]$ for every $x \in R^d$:*

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|, \forall x, y \in R^d, i \in [N].$$

Assumption 2. *There exists a constant σ such that*

$$\mathbb{E}_{\xi \sim \mathcal{D}_i}\|\nabla f(x, \xi) - \nabla f_i(x)\|^2 \leq \sigma^2, \forall x \in R^d, \forall i \in [N].$$

Assumption 3. *If the objective function is non-convex, we assume it is ρ-weakly convex.*

Remark 1. *Note that if $f(x)$ is L-smooth, it is L-weakly convex. This is because Assumption 1 implies $-\frac{L}{2}\|x - y\|^2 \leq f(x) - f(y) - \langle \nabla f(y), x - y\rangle \leq \frac{L}{2}\|x - y\|^2$ [27]. Therefore, for an L-smooth function, we can immediately get that the weakly-convex parameter ρ satisfies $0 < \rho \leq L$.*

3.3 Review: Synchronous SGD with Periodically Averaging (Local SGD)

To alleviate the high communication cost in SyncSGD, the periodically averaging technique is proposed [30, 39]. Instead of averaging models in all clients at every iteration, Local SGD lets clients update their models locally for k iterations, then one communication is conducted to average the local models to make them consistent. Specifically, the update rule of Local SGD is

$$x_t^i = \begin{cases} \frac{1}{N}\sum_{j=1}^N (x_{t-1}^j - \eta \nabla f(x_{t-1}^j, \xi_{t-1}^j)), & \text{if } t \,\%\, k \;=\; 0, \\ x_{t-1}^i - \eta \nabla f(x_{t-1}^i, \xi_{t-1}^i), & \text{else,} \end{cases}$$

where x_t^i is the local model in client i at iteration t. Therefore, when each client conducts T iterations, the total number of communications is T/k. The complete procedure of Local SGD is summarized in Algorithm 1. Different from previous studies [26, 30, 39], Algorithm 1 returns $\tilde{x} = \frac{1}{N}\sum_{i=1}^N x_t^i$ for a randomly chosen $t \in \{0, 1, \cdots, T-1\}$. In practice, we can determine t at first to avoid redundant iterations.

Although several studies have analysed the convergence of Local SGD, they assume that the objective $f(x)$ is μ-strongly convex or non-convex. [19] focuses on general convex objectives while they use the full gradient descent. Besides, most of the existing analysis relies on some stronger assumptions, including bounded gradient norm (i.e., $\|\nabla f_i(x, \xi)\|^2 \leq G^2$) [30, 22] or bounded variance of gradients among clients [28]. Here, we give a basic convergence result of Local SGD for the general convex objectives without these assumptions.

Algorithm 2 STL-SGD^{sc}(f, x_1, η_1, T_1, k_1)

1: **for** $s = 1, 2, ..., S$ **do**
2: x_{s+1} = Local-SGD(f, x_s, η_s, T_s, $\max\{\lfloor k_s \rfloor, 1\}$).
3: Set $\eta_{s+1} = \frac{\eta_s}{2}$, $T_{s+1} = 2T_s$ and

$$k_{s+1} = \begin{cases} \sqrt{2}k_s, & \text{Non-IID case,} \\ 2k_s, & \text{IID case.} \end{cases}$$

4: **end for**
5: **return** x_{S+1}.

Theorem 1. *Suppose Assumptions 1 and 2 hold, $f(x)$ is convex and $\eta \leq \frac{1}{6L}$. If we set $k \leq \min\{\frac{1}{6\eta LN}, \frac{1}{9\eta L}\}$ and $k \leq \min\{\frac{\sigma}{\sqrt{6\eta LN(\sigma^2+4\zeta_f^*)}}, \frac{1}{9\eta L}\}$ for the IID case and the Non-IID case respectively, we have*

$$\mathbb{E}f(\tilde{x}) - f(x^*) \leq \frac{3\|x_0 - x^*\|^2}{4\eta T} + \frac{\eta\sigma^2}{N}. \tag{2}$$

Remark 2. *If we set $\eta = \sqrt{\frac{N}{T}}$, we have $\mathbb{E}f(\tilde{x}) - f(x^*) \leq \frac{\|x_0 - x^*\|^2 + \sigma^2}{\sqrt{NT}}$, which is consistent with the result of mini-batch SGD [6].*

4 Local SGD with Stagewise Communication Period

To further reduce the communication complexity, we propose *ST*agewise *L*ocal *SGD* (STL-SGD) in this section with the following features.

- At the beginning, STL-SGD employs Algorithm 1 as a subalgorithm in each stage.
- Instead of using a small fixed learning rate or a gradually decreasing learning rate (e.g. $\frac{\eta_1}{1+\alpha t}$), STL-SGD adopts a stagewisely adaptive scheme. The learning rate is fixed at first, and decreased stage by stage.
- The communication periods are increased stagewisely.

We propose two variants of STL-SGD for strongly convex and non-convex problems, respectively.

4.1 STL-SGD for Strongly Convex Problems

In this subsection, we propose the STL-SGD algorithm for strongly convex problems, which is denoted as STL-SGD^{sc} and summarized in Algorithm 2. At each stage, the learning rate is decreased exponentially. In the meantime, the number of iterations and the communication period are increased exponentially. Specifically, at the s-th stage, we set $\eta_s = \frac{\eta_{s-1}}{2}$ and $T_s = 2T_{s-1}$. The communication period k_s is set as $k_s = 2k_{s-1}$ and $k_s = \sqrt{2}k_{s-1}$ for the IID case and the Non-IID case respectively.

Below, let x_s denote the initial point of the s-th stage. Theorem 2 establishes the convergence rate of STL-SGD^{sc}.

Theorem 2. *Suppose $f(x)$ is μ-strongly convex. Let $\eta_1 \leq \frac{1}{6L}$ and $T_1\eta_1 = \frac{6}{\mu}$. We set $k_1 = \min\{\frac{1}{6\eta_1 LN}, \frac{1}{9\eta_1 L}\}$ and $k_1 = \min\{\frac{\sigma}{\sqrt{6\eta_1 LN(\sigma^2+4\zeta_f)}}, \frac{1}{9\eta_1 L}\}$ for the IID case and the Non-IID case respectively. Under Assumptions 1 and 2, when the number of stages satisfies $S \geq \log(\frac{N(f(x_0)-f(x^*))}{\eta_1\sigma^2}) + 2$, we have the following result for Algorithm 2:*

$$\mathbb{E}f(x_{S+1}) - f(x^*) \leq \frac{9\eta_1\sigma^2}{2^S N}. \tag{3}$$

Defining $T := T_1 + T_2 + \cdots + T_S$, we have

$$\mathbb{E}f(x_{S+1}) - f(x^*) \leq O\left(\frac{1}{NT}\right). \tag{4}$$

Algorithm 3 STL-SGD$^{nc}(f, x_1, \eta_1, T_1, k_1)$

for $s = 1, 2, ..., S$ **do**
  Let $f_{x_s}^{\gamma}(x) = f(x) + \frac{1}{2\gamma}\|x - x_s\|^2$.
  x_{s+1} = Local-SGD($f_{x_s}^{\gamma}$, x_s, η_s, T_s, $\max\{\lfloor k_s \rfloor, 1\}$).
  Option 1: Set $\eta_{s+1} = \frac{\eta_s}{2}$, $T_{s+1} = 2T_s$ and

$$k_{s+1} = \begin{cases} \sqrt{2}k_s, & \text{Non-IID case,} \\ 2k_s, & \text{IID case.} \end{cases}$$

  Option 2: Set $\eta_{s+1} = \frac{\eta_1}{s+1}$, $T_{s+1} = (s+1)T_1$ and

$$k_{s+1} = \begin{cases} \sqrt{s+1}k_1, & \text{Non-IID case,} \\ (s+1)k_1, & \text{IID case.} \end{cases}$$

end for
return x_{S+1}.

Remark 3. *Theorem 2 claims the following properties of STL-SGD^{sc}:*

- ***Linear Speedup.*** *To reach a solution x_{S+1} with $\mathbb{E}f(x_{S+1}) - f(x^*) \leq \epsilon$, the number of iterations is $O(\frac{1}{N\epsilon})$, which indicates a linear speedup.*
- ***Communication Complexity for the Non-IID Case.*** *For the Non-IID case, we set $k_{s+1} = \sqrt{2}k_s$ for Algorithm 2. Therefore, the total communication complexity is $\frac{T_1}{k_1} + \cdots + \frac{T_S}{k_S} = \frac{T_1}{k_1}(1 + 2^{\frac{1}{2}} + \cdots + 2^{\frac{s-1}{2}}) = O(\frac{T_1}{k_1} \cdot (\frac{T}{T_1})^{\frac{1}{2}}) = O(N^{\frac{1}{2}}T^{\frac{1}{2}})$, where the last equality holds because $\frac{T_1^{\frac{1}{2}}}{k_1} = O(\sqrt{T_1\eta_1 N}) = O(N^{\frac{1}{2}})$.*
- ***Communication Complexity for the IID Case.*** *If the data distributions on different clients are identical, we set $k_{s+1} = 2k_s$ for Algorithm 2. Thus, the total communication complexity is $\frac{T_1}{k_1} + \cdots + \frac{T_S}{k_S} = S\frac{T_1}{k_1} = O(N \log T)$.*

4.2 STL-SGD for Non-Convex Problems

In this subsection, we proceed to propose the variant of STL-SGD algorithm for non-convex problems (STL-SGD^{nc}). Different from Algorithm 2, which optimizes a fixed objective during all stages, STL-SGD^{nc} changes the objective once a stage is finished. Specifically, in the s-th stage, the objective is a regularized problem $f_{x_s}^{\gamma} = f(x) + \frac{1}{2\gamma}\|x - x_s\|^2$, where x_s is the initial point of the s-th stage and γ is a constant that satisfies $\gamma < \rho^{-1}$. $f_{x_s}^{\gamma}(x)$ is guaranteed to be convex due to the ρ-weak convexity of $f(x)$. In this way, the theoretical property of Algorithm 1 under convex settings still holds in each stage of STL-SGD^{nc}. Other parameters are set in two different ways (**Option 1** and **Option 2**) for non-convex objectives satisfying the PL condition and otherwise, which are detailed in Algorithm 3.

In **Option 1**, we set η_s, T_s and k_s in the same way as in Algorithm 2. Here we analyse the theoretical property of STL-SGD^{nc} with **Option 1** for non-convex objectives that satisfy the PL condition.

Theorem 3. *Assume $f(x)$ satisfies the PL condition defined in Definition 2 with constant μ. Suppose Assumptions 1, 2 and 3 hold and $f(x)$ is weakly convex with constant $\rho \leq \frac{\mu}{16}$. Let $\eta_1 \leq \frac{1}{12L_\gamma}$, $T_1\eta_1 = \frac{6}{\rho}$. Set $k_1 = \min\{\frac{1}{6\eta_1 L_\gamma N}, \frac{1}{9\eta_1 L_\gamma}\}$ and $k_1 = \min\{\frac{\sigma}{\sqrt{6\eta_1 L_\gamma N(\sigma^2 + 4\zeta_f)}}, \frac{1}{9\eta_1 L_\gamma}\}$ for the IID case and the Non-IID case respectively. When the number of stages satisfies $S \geq \log \frac{N(f(x_0) - f(x^*))}{\eta_1 \sigma^2} + 2$, Algorithm 3 with **Option 1** returns a solution x_{S+1} such that*

$$\mathbb{E}f(x_{S+1}) - f(x^*) \leq O\left(\frac{1}{NT}\right), \tag{5}$$

where $T = T_1 + T_2 + \cdots + T_S$.

Remark 4. *As the result of Theorem 3 is the same as that of Theorem 2, properties stated in Remark 3 all hold here.*

Option 2 is employed for the non-convex objectives which do not satisfy the PL condition. Instead of increasing the communication period geometrically as in **Option 1** of Algorithm 3, we let it increase in a linear manner, i.e., $k_s = sk_1$. Meanwhile, we increase the stage length linearly, that is $T_s = sT_1$, while keeping $T_s\eta_s$ a constant.

Theorem 4. *Suppose Assumptions 1, 2 and 3 hold. Let* $\eta_1 \leq \frac{1}{6L_\gamma}$ *and* $T_1\eta_1 = \frac{3}{\rho}$. *Set* $k_1 = \min\{\frac{1}{6\eta_1 LN}, \frac{1}{9\eta_1 L}\}$ *and* $k_1 = \min\{\frac{\sigma}{\sqrt{6\eta_1 LN(\sigma^2+4\zeta_f)}}, \frac{1}{9\eta_1 L}\}$ *for the IID case and the Non-IID case respectively. Algorithm 3 with* ***Option 2*** *guarantees that*

$$\mathbb{E}\|\nabla f(x_s)\|^2 \leq O\left(\frac{1}{\sqrt{NT}}\right), \tag{6}$$

where s *is randomly sampled from* $\{1, 2, \cdots, S\}$ *with probability* $p_s = \frac{s}{1+2+\cdots+S}$.

Remark 5. *STL-SGD*^{nc} *with* ***Option 2*** *has the following properties:*

- ***Linear Speedup:*** *To achieve* $\mathbb{E}\|\nabla f(x_S)\|^2 \leq \epsilon$, *the total number of iterations when* N *clients are used is* $O(\frac{1}{N\epsilon^2})$, *which shows a linear speedup.*
- ***Communication Complexity for the Non-IID case:*** *Algorithm 3 with* ***Option 2*** *sets* $k_s = \sqrt{s}k_1$. *Thus, the communication complexity is* $\frac{T_1}{k_1} + \frac{T_2}{k_2} + \cdots + \frac{T_S}{k_S} = \frac{T_1}{k_1}(1 + \sqrt{2} + \cdots + \sqrt{S}) = O(\frac{T_1}{k_1}(\frac{T}{T_1})^{\frac{3}{4}}) = O(N^{\frac{3}{4}}T^{\frac{3}{4}})$.
- ***Communication Complexity for the IID case:*** *As* $k_s = sk_1$, *the communication complexity is* $\frac{T_1}{k_1} + \frac{T_2}{k_2} + \cdots + \frac{T_S}{k_S} = \frac{T_1}{k_1}S = O(\frac{T_1}{k_1}(\frac{T}{T_1})^{\frac{1}{2}}) = O\left(N^{\frac{3}{2}}T^{\frac{1}{2}}\right)$.

5 Experiments

We validate the performance of the proposed STL-SGD algorithm with experiments on both convex and non-convex problems. For each type of problems, we conduct experiments for both the IID case and the Non-IID case. Experiments are conducted on a machine with 8 Nvidia Geforce GTX 1080Ti GPUs and 2 Xeon(R) Platinum 8153 CPUs.

To simulate the Non-IID scenarios, we divide the training data among clients and make the distributions of classes very different among them. Similar to the setting in [17], at first, we randomly take $s\%$ i.i.d. data from the training set and divide them equally to each client. For the remaining data, we sort them according to their classes and then assign them to the clients in order. In our experiments, we set $s = 50$ for the convex problems and $s = 0$ for the non-convex problems.

We compare STL-SGD with SyncSGD, LB-SGD, CR-PSGD [38] and Local SGD [30]. We show the comparison of these algorithms in terms of the communication rounds. The investigation regarding convergence is included in the Appendix, which validates that STL-SGD can achieve similar convergence as SyncSGD.

5.1 Convex Problems

We consider the binary classification problem with logistic regression, i.e.,

$$\min_{\theta\in R^d} \frac{1}{n}\sum_{i=1}^{n} \log(1 + \exp(-y_i x_i^T\theta)) + \frac{\lambda}{2}\|\theta\|^2, \tag{7}$$

where $(x_i, y_i), i \in [n]$ constitute a set of training examples, and λ is the regularization parameter. It is notable that (7) is a strongly convex problem when $\lambda > 0$, and we set $\lambda = 1/n$. We take two datasets $\mathrm{a9a}$ and MNIST from the libsvm website[2]. $\mathrm{a9a}$ has $32,561$ examples and 123 features. For MNIST, we sample a subset with $11,791$ examples and 784 features from two classes (4 and 9). Experiments are implemented on 32 clients and communication is handled with MPI[3].

[2]https://www.csie.ntu.edu.tw/ cjlin/libsvmtools/datasets/

[3]https://www.open-mpi.org/

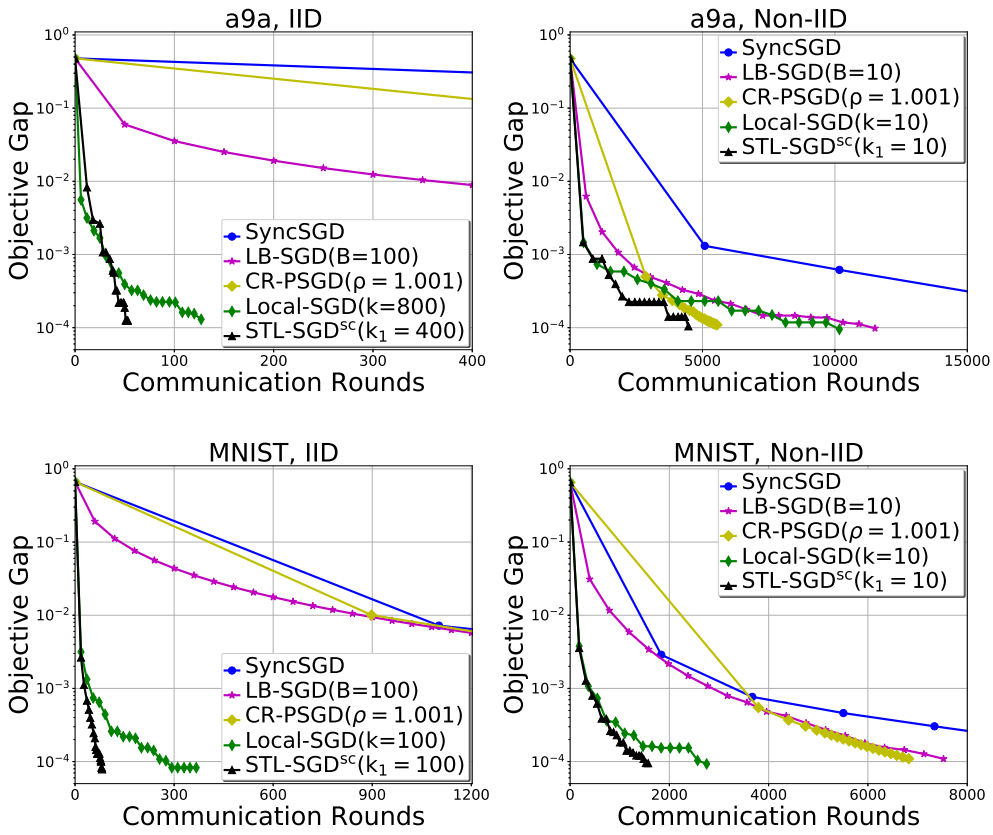

Figure 1: Training objective gap $f(x)-f(x^*)$ w.r.t the communication rounds for logistic regression on a9a and MNIST.

Table 1: Communication rounds to reach 10^{-4} objective gap in convex problems. We also show the speedup of these algorithms compared with SyncSGD.

Algorithms	a9a (IID)	a9a (Non-IID)	MNIST (IID)	MNIST (Non-IID)
SyncSGD	100683 (1×)	90513 (1×)	32664 (1×)	22021 (1×)
LB-SGD	7620 (13.2×)	12221 (7.4×)	7011 (4.7×)	7740 (2.8×)
CR-PSGD	5434 (18.5×)	5772 (15.7×)	6788 (4.8×)	7029 (3.1×)
Local-SGD	184 (547.2×)	10068 (9.0×)	289 (113.0×)	2642 (8.3×)
STL-SGD^{sc}	**61** **(1650.5×)**	**4417** **(20.5×)**	**79** **(413.5×)**	**1518** **(14.5×)**

SyncSGD, LB-SGD and Local SGD are implemented with the decreasing learning rate $\eta_t = \frac{\eta_1}{1+\alpha t}$ as suggested in [30, 22] and we tune α in $\{10^{-2}, 10^{-3}, 10^{-4}\}$ for the best performance. For STL-SGD^{sc}, we set $\eta_1 T_1 = \frac{1}{\lambda}$. The initial learning rate for all algorithms is tuned in $\{N, N/10, N/100\}$. The communication period k and the batch size B for LB-SGD are tuned in $\{100, 200, 400, 800, 1600\}$ for the IID case, and $\{10, 20, 40, 80, 160\}$ for the Non-IID case. The scaling factor of batch size ρ for CR-PSGD is tuned in $\{1.001, 1.01, 1.1\}$. We report the largest k, B and ρ which do not sacrifice the convergence.

Figure 1 shows the objective gap $f(x) - f(x^*)$ with regard to the communication rounds. We can observe that STL-SGD^{sc} converges with the fewest communication rounds for both the IID case and the Non-IID case. Although the initial communication period of STL-SGD^{sc} may need to be set smaller than Local SGD in the the IID case, the total number of communication rounds of STL-SGD^{sc} is still significantly lower, which validates that the communication complexity of

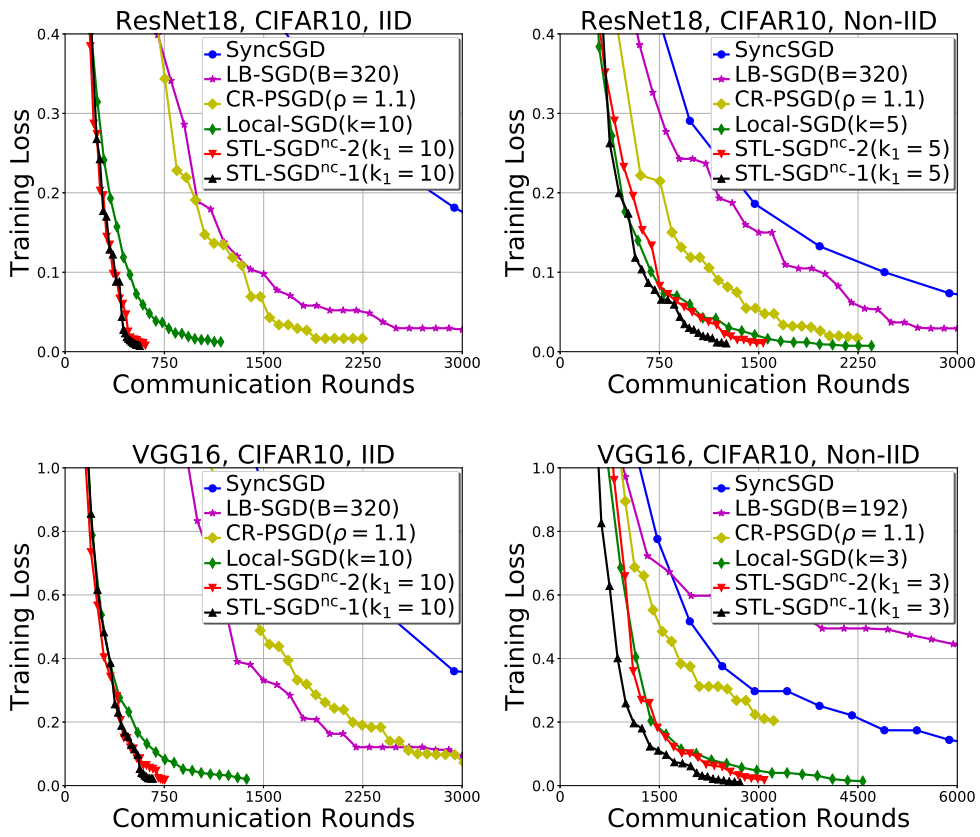

Figure 2: Training loss w.r.t the communication rounds for ResNet18 and VGG16 on CIFAR10.

Table 2: Communication rounds to reach 99% training accuracy in non-convex problems. We run all algorithms for 200 epochs, where an epoch indicates one pass of the dataset. LB-SGD and CR-PSGD can not achieve 99% training accuracy on the VGG16 neural network until the end of training.

Algorithms	ResNet18 (IID)	ResNet18 (Non-IID)	VGG16 (IID)	VGG16 (Non-IID)
SyncSGD	7644 (1×)	5390 (1×)	13622 (1×)	15092 (1×)
LB-SGD	3000 (2.5×)	3180 (1.7×)	– (–)	– (–)
CR-PSGD	1797 (4.3×)	1937 (2.8×)	– (–)	– (–)
Local-SGD	755 (10.1×)	1235 (4.4×)	1245 (10.9×)	3986 (3.8×)
STL-SGD^{nc}-2	**470 (16.3×)**	**1158 (4.7×)**	**696 (19.6×)**	**2732 (5.5×)**
STL-SGD^{nc}-1	**434 (17.6×)**	**954 (5.6×)**	**602 (22.6×)**	**2179 (6.9×)**

STL-SGD^{sc} is much lower than Local SGD. As shown in Table 1, to achieve 10^{-4} objective gap, the communication rounds of STL-SGD^{sc} is almost 1.7-3 times fewer than Local SGD.

5.2 Non-Convex Problems

We train ResNet18 [13] and VGG16 [29] on the CIFAR10 [20] dataset, which includes a training set of 50,000 examples from 10 classes. 8 clients are used in total.

For our proposed algorithm, we denote STL-SGD^{nc} with **Option 1** and **Option 2** as STL-SGD^{nc}-1 and STL-SGD^{nc}-2 respectively. The learning rates of SyncSGD, LB-SGD, CR-PSGD and Local-SGD are all set fixed as suggested in their convergence theory [7, 38, 39]. The initial learning rate for

all algorithms is tuned in $\{N/10, N/100, N/1000\}$. The basic batch size at each client is 64. The first stage length of STL-SGD^{nc} is tuned in $\{20, 40, 60\}$ epochs. The parameter γ in STL-SGD^{nc} is tuned in $\{10^0, 10^2, 10^4\}$. We tune the communication period k in $\{3, 5, 10, 20\}$ and the batch size B for LB-SGD in $\{192, 320, 640, 1280\}$. For ease of implementation, we increase the batch size in CR-PSGD with $B = \rho B$ once an epoch is finished, and ρ is tuned in $\{1.1, 1.2, 1.3\}$. B stops growing when it exceeds 512 as suggested in [38].

The experimental results of training loss regarding communication rounds are presented in Figure 2 and the communication rounds to achieve 99% training accuracy for all algorithms are shown in Table 2. As can be seen, STL-SGD^{nc}-1 and STL-SGD^{nc}-2 converge with much fewer communications than other algorithms. In spite of the same order of communication complexity as Local SGD, the performance of STL-SGD^{nc}-2 is better as the benefit of the negative relevance between the learning rate and the communication period. STL-SGD^{nc}-1 converges with the fewest number of communications, as it uses a geometrically increasing communication period.

6 Conclusion and Future Work

We propose STL-SGD, which adopts a stagewisely increasing communication period to reduce the communication complexity. Two variants of STL-SGD (STL-SGD^{sc} and STL-SGD^{nc}) are provided for strongly convex objectives and non-convex objectives respectively. Theoretically, we prove that: (i) STL-SGD maintains the convergence rate and linear speedup as SyncSGD; (ii) when the objective is strongly convex or satisfies the PL condition, while attaining the optimal convergence rate $O(\frac{1}{NT})$, STL-SGD achieves the state-of-the-art communication complexity; (iii) when the objective is general non-convex, STL-SGD has the same communication complexity as Local SGD, while being more consistent with practical tricks. Experiments on both convex and non-convex problems demonstrate the effectiveness of the proposed algorithm.

Local SGD with variance reduction achieves outstanding communication complexity for the Non-IID case. One interesting idea is to combine the techniques of stagewise training and variance reduction to get better results for the Non-IID case. We will consider it in our future work.

References

[1] Alekh Agarwal and John C Duchi. Distributed delayed stochastic optimization. In *Advances in Neural Information Processing Systems*, pages 873–881, 2011.

[2] 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*, pages 1709–1720, 2017.

[3] Zeyuan Allen-Zhu. How to make the gradients small stochastically: Even faster convex and nonconvex sgd. In *Advances in Neural Information Processing Systems*, pages 1157–1167, 2018.

[4] Zaiyi Chen, Zhuoning Yuan, Jinfeng Yi, Bowen Zhou, Enhong Chen, and Tianbao Yang. Universal stagewise learning for non-convex problems with convergence on averaged solutions. In *International Conference on Learning Representations*, 2019.

[5] Damek Davis and Benjamin Grimmer. Proximally guided stochastic subgradient method for nonsmooth, nonconvex problems. *SIAM Journal on Optimization*, 29(3):1908–1930, 2019.

[6] Ofer Dekel, Ran Gilad-Bachrach, Ohad Shamir, and Lin Xiao. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13(Jan):165–202, 2012.

[7] Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[8] Noah Golmant, Nikita Vemuri, Zhewei Yao, Vladimir Feinberg, Amir Gholami, Kai Rothauge, Michael W. Mahoney, and Joseph Gonzalez. On the computational inefficiency of large batch sizes for stochastic gradient descent, 2018.

[9] Farzin Haddadpour, Mohammad Mahdi Kamani, Mehrdad Mahdavi, and Viveck Cadambe. Local sgd with periodic averaging: Tighter analysis and adaptive synchronization. In *Advances in Neural Information Processing Systems*, pages 11080–11092, 2019.

[10] Farzin Haddadpour, Mohammad Mahdi Kamani, Mehrdad Mahdavi, and Viveck Cadambe. Trading redundancy for communication: Speeding up distributed sgd for non-convex optimization. In *International Conference on Machine Learning*, pages 2545–2554, 2019.

[11] Farzin Haddadpour and Mehrdad Mahdavi. On the convergence of local descent methods in federated learning, 2019.

[12] Elad Hazan and Satyen Kale. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *The Journal of Machine Learning Research*, 15(1):2489–2512, 2014.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[14] Prateek Jain, Sham M. Kakade, Rahul Kidambi, Praneeth Netrapalli, and Aaron Sidford. Parallelizing stochastic gradient descent for least squares regression: mini-batching, averaging, and model misspecification, 2016.

[15] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

[16] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer, 2016.

[17] 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.

[18] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima, 2016.

[19] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. First analysis of local gd on heterogeneous data. *arXiv preprint arXiv:1909.04715*, 2019.

[20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[22] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-{iid} data. In *International Conference on Learning Representations*, 2020.

[23] Xiangru Lian, Yijun Huang, Yuncheng Li, and Ji Liu. Asynchronous parallel stochastic gradient for nonconvex optimization. In *Advances in Neural Information Processing Systems*, pages 2737–2745, 2015.

[24] 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.

[25] Tao Lin, Sebastian U Stich, Kumar Kshitij Patel, and Martin Jaggi. Don't use large mini-batches, use local sgd. *arXiv preprint arXiv:1808.07217*, 2018.

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

[27] Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

[28] Shuheng Shen, Linli Xu, Jingchang Liu, Xianfeng Liang, and Yifei Cheng. Faster distributed deep net training: computation and communication decoupled stochastic gradient descent. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 4582–4589. AAAI Press, 2019.

[29] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[30] Sebastian U. Stich. Local SGD converges fast and communicates little. In *International Conference on Learning Representations*, 2019.

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

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

[33] Hanlin Tang, Chen Yu, Xiangru Lian, Tong Zhang, and Ji Liu. Doublesqueeze: Parallel stochastic gradient descent with double-pass error-compensated compression. In *International Conference on Machine Learning*, pages 6155–6165, 2019.

[34] Jianyu Wang and Gauri Joshi. Adaptive communication strategies to achieve the best error-runtime trade-off in local-update sgd. *arXiv preprint arXiv:1810.08313*, 2018.

[35] 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.

[36] Yi Xu, Qihang Lin, and Tianbao Yang. Stochastic convex optimization: Faster local growth implies faster global convergence. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3821–3830. JMLR. org, 2017.

[37] Dong Yin, Ashwin Pananjady, Max Lam, Dimitris Papailiopoulos, Kannan Ramchandran, and Peter Bartlett. Gradient diversity: a key ingredient for scalable distributed learning, 2017.

[38] Hao Yu and Rong Jin. On the computation and communication complexity of parallel sgd with dynamic batch sizes for stochastic non-convex optimization, 2019.

[39] Hao Yu, Sen Yang, and Shenghuo Zhu. Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5693–5700, 2019.

[40] Zhuoning Yuan, Yan Yan, Rong Jin, and Tianbao Yang. Stagewise training accelerates convergence of testing error over sgd. In *Advances in Neural Information Processing Systems*, pages 2604–2614, 2019.

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

A Comparison to Previous Results

Table 3 summarizes the comparison of Local SGD and its state-of-the-art extensions with the results in this paper. The table shows the convergence rate and the communication complexity of these algorithms when the data distributions are identical or otherwise. Strongly convex objectives, non-convex objectives which satisfy the PL condition and general non-convex objectives are all considered.

For both strongly convex objectives and non-convex objectives which satisfy the PL condition, STL-SGD achieves the state-of-the-art communication complexity while attaining the optimal convergence rate of $O(\frac{1}{NT})$. For general non-convex objectives, STL-SGD keeps the same communication complexity as Local SGD, but SCAFFOLD [17] achieves lower communication complexity when data distributions are not identical. Nevertheless, the variance reduction technique used in SCAFFOLD is orthogonal to our study. It is an interesting direction to combine techniques in STL-SGD and SCAFFOLD for the Non-IID case. Some existing studies make extra assumptions including the bounded gradient and the bounded variance of gradients among clients, while the theoretical analysis in this paper does not depend on these assumptions.

Table 3: A comparison of the results in this paper and previous state-of-the-art results of Local SGD and its variants. Regarding orders of convergence rate and communication complexity, we highlight the dependency on T (the number of iterations), N (the number of clients) and k (communication period). Previous results may depend on some extra assumptions, which include: (1) an upper bound for gradient, (2) an upper bound for the gradient variance among clients and (3) an upper bound for the gradient diversity, which are shown in the last column.

Algorithms	Objectives	Convergence Rate	Communication Complexity	Data Distributions	Extra Assumptions
Local SGD [30]	Strongly Convex	$O(\frac{1}{NT})$	$O(N^{\frac{1}{2}}T^{\frac{1}{2}})$	IID	(1)
Local SGD [32] [3]	Strongly Convex	$O(\frac{\log T}{NT})$	$O(N\log(NT))$	IID	No
STL-SGD	**Strongly Convex**	$\boldsymbol{O(\frac{1}{NT})}$	$\boldsymbol{O(N\log T)}$	**IID**	**No**
Local SGD [22]	Strongly Convex	$O(\frac{k^2}{NT})$	$O(T)$	Non-IID	(1)
Local SGD [17] [3]	Strongly Convex	$O(\frac{\log T}{NT})$	$O(N^{\frac{1}{2}}T^{\frac{1}{2}})$	Non-IID	No
SCAFFOLD [17][3]	Strongly Convex	$O(\frac{\log T}{NT})$	$O(\log(NT))$	Non-IID	No
STL-SGD	**Strongly Convex**	$\boldsymbol{O(\frac{1}{NT})}$	$\boldsymbol{O(N^{\frac{1}{2}}T^{\frac{1}{2}})}$	**Non-IID**	**No**
Local SGD [9] [4]	Non-Convex+PL	$O(\frac{1}{NT})$	$O(N^{\frac{1}{3}}T^{\frac{1}{3}})$	IID	No
STL-SGD	**Non-Convex+PL**	$\boldsymbol{O(\frac{1}{NT})}$	$\boldsymbol{O(N\log T)}$	**IID**	**No**
STL-SGD	**Non-Convex+PL**	$\boldsymbol{O(\frac{1}{NT})}$	$\boldsymbol{O(N^{\frac{1}{2}}T^{\frac{1}{2}})}$	**Non-IID**	**No**
Local SGD [35]	Non-Convex	$O(\frac{1}{\sqrt{NT}})$	$O(N^{\frac{3}{2}}T^{\frac{1}{2}})$	IID	(1)
STL-SGD	**Non-Convex**	$\boldsymbol{O(\frac{1}{\sqrt{NT}})}$	$\boldsymbol{O(N^{\frac{3}{2}}T^{\frac{1}{2}})}$	**IID**	**No**
Local SGD [28]	Non-Convex	$O(\frac{1}{\sqrt{NT}})$	$O(N^{\frac{3}{4}}T^{\frac{3}{4}})$	Non-IID	(2)
Local SGD [11]	Non-Convex	$O(\frac{1}{\sqrt{NT}})$	$O(N^{\frac{3}{2}}T^{\frac{1}{2}})$	Non-IID	(3)
SCAFFOLD [17]	Non-Convex	$O(\frac{1}{\sqrt{NT}})$	$O(N^{\frac{1}{2}}T^{\frac{1}{2}})$	Non-IID	No
STL-SGD	**Non-Convex**	$\boldsymbol{O(\frac{1}{\sqrt{NT}})}$	$\boldsymbol{O(N^{\frac{3}{4}}T^{\frac{3}{4}})}$	**Non-IID**	**No**

[3] Although these studies prove lower communication complexity, a suboptimal $O(\frac{\log T}{NT})$ convergence rate is proved due to the small fixed learning rate.

[4] The adaptive variant of Local SGD proposed in [9] has the same order of communication complexity as Local SGD.

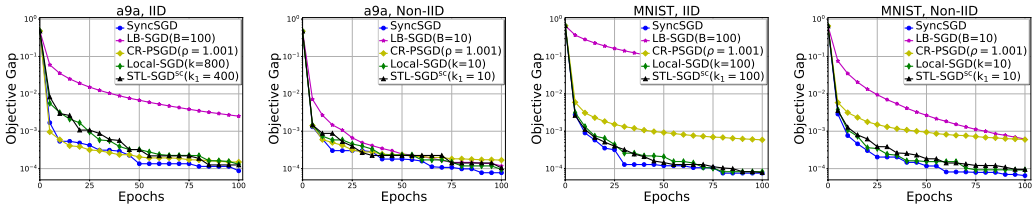

Figure 3: Training objective gap $f(x)-f(x^*)$ w.r.t epochs for logistic regression on a9a and MNIST datasets.

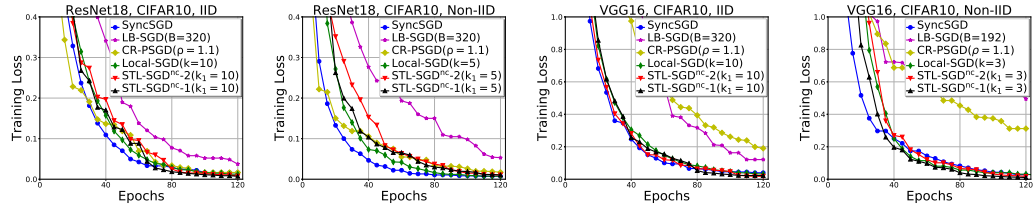

Figure 4: Training loss w.r.t epochs for ResNet18 and VGG16 on CIFAR10 dataset.

B More About Experiments

B.1 Experimental Results for Validating the Convergence Rate

In this subsection, we supplement the experimental results not included in Section 5. The rules for turning the hyper-parameters are presented in Section 5 and we turn all hyper-parameters to make all algorithms to achieve the best convergence speed. We present the experimental results of the training loss with regard to the epochs in this subsection. The results for strongly convex objectives and non-convex objectives are shown in Figure 3 and Figure 4 respectively.

From the theoretical perspective, STL-SGD, CR-PSGD and Local SGD can maintain the same convergence rate with SyncSGD: $O(\frac{1}{NT})$ for strongly convex objectives and $O(\frac{1}{\sqrt{NT}})$ for non-convex objectives. As shown in Figure 3 and Figure 4, when the hyper-parameters are set properly, the convergence speed of the above algorithms is similar. STL-SGD and Local SGD may converge slowly in the beginning, but they match SyncSGD when the number of iterations is relatively large, which is consistent with our theory in Theorem 2 and Theorem 3 that the number of stages can not be too small. Although LB-SGD is theoretically justified to achieve a linear speedup with respect to the batch size, it can not maintain the convergence of mini-batch SGD (or SyncSGD) when the batch size B gets large. The reason could be that the bias dominates the variance as discussed in [14].

C Proofs for Results in Section 3

In this section, we first present some lemmas, then give the proof for Theorem 1.

C.1 Some Basic Lemmas

We bound the norm of the difference between gradients with the Bregman divergence $\mathcal{D}_f(x,y) := f(x) - f(y) - \langle \nabla f(y), x-y\rangle$ for a smooth and convex function.

Lemma 1. *Suppose $f(x)$ is L-smooth and convex. The following inequality holds:*

$$\|\nabla f(x) - \nabla f(y)\|^2 \leq 2L\mathcal{D}_f(x,y).$$

Proof. This Lemma is identical to Theorem 2.1.5 (2.1.10) in [27], which is a basic property of smooth and convex functions. □

For ease of analysis, we define $\hat{x}_t$ as the average of the local models, i.e., $\hat{x}_t = \frac{1}{N}\sum_{i=1}^N x_t^i$. According to the update rule in Algorithm 1, we have

$$\hat{x}_{t+1} = \frac{1}{N}\sum_{i=1}^N x_{t+1}^i = \frac{1}{N}\sum_{i=1}^N (x_t^i - \eta \nabla f(x_t^i, \xi_t^i)) = \hat{x}_t - \eta \frac{1}{N}\sum_{i=1}^N \nabla f(x_t^i, \xi_t^i).$$

We use t_p to denote the last time to communicate, i.e., $t_p = \lfloor t/k \rfloor \cdot k$. Then, we get

$$\hat{x}_t = \hat{x}_{t_p} - \frac{\eta}{N}\sum_{\tau=t_p}^{t-1}\sum_{i=1}^N \nabla f(x_\tau^i, \xi_\tau^i) \quad \text{and} \quad x_t^i = \hat{x}_{t_p} - \eta \sum_{\tau=t_p}^{t-1} \nabla f(x_\tau^i, \xi_\tau^i). \tag{8}$$

As each client updates its model locally and communicates with others periodically, it is important to make sure that the divergence of local models is not very large. We use Lemma 2 to bound the difference between $\hat{x}_t$ and x_t^i to guarantee this.

Lemma 2. *Under Assumptions 1 and 2, for any $x \in R^d$, Algorithm 1 ensures that*

$$\frac{1}{N}\sum_{i=1}^N \sum_{t=0}^{T-1} \mathbb{E}\|\hat{x}_t - x_t^i\|^2 \le \frac{k-1}{1-2k^2\eta^2L^2}\left(T\eta^2\sigma^2 + 8k\eta^2 L \sum_{t=0}^{T-1} \mathbb{E}\mathcal{D}_f(\hat{x}_\tau, x) + 4Tk\eta^2\zeta_f^x\right). \tag{9}$$

Proof. According to (8), we have

$$\begin{aligned}
\|\hat{x}_t - x_t^i\|^2 &= \left\| \hat{x}_{t_p} - \frac{\eta}{N}\sum_{\tau=t_p}^{t-1}\sum_{j=1}^N \nabla f(x_\tau^j, \xi_\tau^j) - \left(\hat{x}_{t_p} - \eta\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i)\right)\right\|^2 \\
&= \eta^2 \left\| \sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i) - \frac{1}{N}\sum_{j=1}^N\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^j, \xi_\tau^j)\right\|^2.
\end{aligned}$$

Since $\frac{1}{N}\sum_{i=1}^N \left\|A_i - \frac{1}{N}\sum_{j=1}^N A_j\right\|^2 = \frac{1}{N}\sum_{i=1}^N \|A_i\|^2 - \left\|\frac{1}{N}\sum_{i=1}^N A_i\right\|^2$, we have

$$\begin{aligned}
\frac{1}{N}\sum_{i=1}^N \mathbb{E}\left\|\hat{x}_t - x_t^i\right\|^2 &= \eta^2\left(\frac{1}{N}\sum_{i=1}^N \mathbb{E}\left\|\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i)\right\|^2 - \mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^N\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i)\right\|^2\right) \\
&\le \frac{\eta^2}{N}\sum_{i=1}^N \mathbb{E}\left\|\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i)\right\|^2.
\end{aligned} \tag{10}$$

Next, we bound $\mathbb{E}\left\|\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i)\right\|^2$:

$$\begin{aligned}
\mathbb{E}\left\|\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i)\right\|^2 &= \mathbb{E}\left\|\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i) - \sum_{\tau=t_p}^{t-1}\nabla f_i(x_\tau^i) + \sum_{\tau=t_p}^{t-1}\nabla f_i(x_\tau^i)\right\|^2 \\
&\overset{(a)}{=} \mathbb{E}\left\|\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i) - \sum_{\tau=t_p}^{t-1}\nabla f_i(x_\tau^i)\right\|^2 + \mathbb{E}\left\|\sum_{\tau=t_p}^{t-1}\nabla f_i(x_\tau^i)\right\|^2 \\
&\overset{(b)}{=} \sum_{\tau=t_p}^{t-1}\mathbb{E}\left\|\nabla f(x_\tau^i, \xi_\tau^i) - \nabla f_i(x_\tau^i)\right\|^2 + \mathbb{E}\left\|\sum_{\tau=t_p}^{t-1}\nabla f_i(x_\tau^i)\right\|^2 \\
&\overset{(c)}{\le} \sum_{\tau=t_p}^{t-1}\mathbb{E}\left\|\nabla f(x_\tau^i, \xi_\tau^i) - \nabla f_i(x_\tau^i)\right\|^2 + (t-t_p)\sum_{\tau=t_p}^{t-1}\mathbb{E}\left\|\nabla f_i(x_\tau^i)\right\|^2 \\
&\overset{(d)}{\le} (t-t_p)\sigma^2 + (t-t_p)\sum_{\tau=t_p}^{t-1}\mathbb{E}\left\|\nabla f_i(x_\tau^i)\right\|^2,
\end{aligned} \tag{11}$$

where (a) and (b) hold because $\mathbb{E}\nabla f(x_\tau^i, \xi_\tau^i) = \nabla f_i(x_\tau^i)$ and ξ_τ^i's are independent; (c) follows from Cauchy's inequality; (d) is due to Assumption 2. We then bound $\mathbb{E}\left\|\nabla f_i(x_\tau^i)\right\|^2$:

$$\begin{aligned}
\mathbb{E}\left\|\nabla f_i(x_\tau^i)\right\|^2 &= \mathbb{E}\left\|\nabla f_i(x_\tau^i) - \nabla f_i(\hat{x}_\tau) + \nabla f_i(\hat{x}_\tau)\right\|^2 \\
&\overset{(a)}{\leq} 2\mathbb{E}\|\nabla f_i(x_\tau^i) - \nabla f_i(\hat{x}_\tau)\|^2 + 2\mathbb{E}\|\nabla f_i(\hat{x}_\tau)\|^2 \\
&\overset{(b)}{\leq} 2L^2\mathbb{E}\|x_\tau^i - \hat{x}_\tau\|^2 + 2\mathbb{E}\|\nabla f_i(\hat{x}_\tau) - \nabla f_i(x) + \nabla f_i(x)\|^2 \\
&\overset{(c)}{\leq} 2L^2\mathbb{E}\|x_\tau^i - \hat{x}_\tau\|^2 + 4\mathbb{E}\|\nabla f_i(\hat{x}_\tau) - \nabla f_i(x)\|^2 + 4\mathbb{E}\|\nabla f_i(x)\|^2 \\
&\overset{(d)}{\leq} 2L^2\mathbb{E}\|x_\tau^i - \hat{x}_\tau\|^2 + 8L\mathbb{E}\mathcal{D}_{f_i}(\hat{x}_\tau, x) + 4\mathbb{E}\|\nabla f_i(x)\|^2, \qquad (12)
\end{aligned}$$

where (a) and (c) come from $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$, (b) holds because of Assumption 1, (d) follows from Lemma 1. Substituting (12) into (11) and based on $t - t_p \leq k - 1$, we have

$$\begin{aligned}
&\mathbb{E}\left\|\sum_{\tau=t_p}^{t-1}\nabla f(x_\tau^i, \xi_\tau^i)\right\|^2 \\
\leq\ & (k-1)\sigma^2 + (k-1)\sum_{\tau=t_p}^{t-1}\left(2L^2\mathbb{E}\|x_\tau^i - \hat{x}_\tau\|^2 + 8L\mathbb{E}\mathcal{D}_{f_i}(\hat{x}_\tau, x) + 4\mathbb{E}\|\nabla f_i(x)\|^2\right). \qquad (13)
\end{aligned}$$

Substituting (13) into (10) and according to the definition of ζ_f^x, we get

$$\begin{aligned}
\frac{1}{N}\sum_{i=1}^N \mathbb{E}\left\|\hat{x}_t - x_t^i\right\|^2 \leq\ & \eta^2(k-1)\sigma^2 + \frac{2(k-1)\eta^2L^2}{N}\sum_{i=1}^N\sum_{\tau=t_p}^{t-1}\mathbb{E}\|x_\tau^i - \hat{x}_\tau\|^2 \\
&+8(k-1)\eta^2L\sum_{\tau=t_p}^{t-1}\mathbb{E}\mathcal{D}_f(\hat{x}_\tau, x) + 4(k-1)\eta^2\sum_{\tau=t_p}^{t-1}\zeta_f^x.
\end{aligned}$$

Summing up this inequality from $t = 0$ to $T - 1$, we have

$$\begin{aligned}
&\frac{1}{N}\sum_{i=1}^N\sum_{t=0}^{T-1}\mathbb{E}\left\|\hat{x}_t - x_t^i\right\|^2 \\
\leq\ & (k-1)\left(T\eta^2\sigma^2 + \frac{2\eta^2L^2}{N}\sum_{i=1}^N\sum_{t=0}^{T-1}\sum_{\tau=t_p}^{t-1}\|x_\tau^i - \hat{x}_\tau\|^2\right. \\
&\left.+8\eta^2L\sum_{t=0}^{T-1}\sum_{\tau=t_p}^{t-1}\mathbb{E}\mathcal{D}_f(\hat{x}_\tau, x) + 4\eta^2\sum_{t=0}^{T-1}\sum_{\tau=t_p}^{t-1}\zeta_f^x\right) \\
\leq\ & (k-1)\left(T\eta^2\sigma^2 + \frac{2k\eta^2L^2}{N}\sum_{i=1}^N\sum_{t=0}^{T-1}\|x_\tau^i - \hat{x}_\tau\|^2 + 8k\eta^2L\sum_{t=0}^{T-1}\mathbb{E}\mathcal{D}_f(\hat{x}_\tau, x) + 4Tk\eta^2\zeta_f^x\right), \qquad (14)
\end{aligned}$$

where the second inequality comes from a simple counting argument: $\sum_{t=0}^T\sum_{\tau=t_p}^{t-1}A_\tau \leq \sum_{t=0}^T\sum_{\tau=t-k}^{t-1}A_\tau \leq k\sum_{t=0}^T A_t, A_t \geq 0$. Rearranging (14), we get

$$\frac{1}{N}\sum_{i=1}^N\sum_{t=0}^{T-1}\mathbb{E}\|\hat{x}_t - x_t^i\|^2 \leq \frac{k-1}{1-2k^2\eta^2L^2}\left(T\eta^2\sigma^2 + 8k\eta^2L\sum_{t=0}^{T-1}\mathbb{E}\mathcal{D}_f(\hat{x}_\tau, x) + 4Tk\eta^2\zeta_f^x\right).$$

□

Below, we use Lemma 3 to bound the average of stochastic gradients.

Lemma 3. *Under Assumptions 1 and 2, we have*

$$\mathbb{E}\left\|\sum_{i=1}^N\frac{1}{N}\nabla f(x_t^i, \xi_t^i)\right\|^2 \leq \frac{\sigma^2}{N} + \frac{3L^2}{N}\sum_{i=1}^N\mathbb{E}\left\|x_t^i - \hat{x}_t\right\|^2 + \frac{3}{2}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2. \qquad (15)$$

Proof. Since ξ_t^i's are independent, we have

$$\begin{aligned}
\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f(x_t^i,\xi_t^i)\right\|^2 &= \mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_t^i,\xi_t^i)-\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_t^i)+\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_t^i)\right\|^2 \\
&= \frac{1}{N^2}\mathbb{E}\left\|\sum_{i=1}^{N}\nabla f(x_t^i,\xi_t^i)-\sum_{i=1}^{N}\nabla f_i(x_t^i)\right\|^2+\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_t^i)\right\|^2 \\
&= \frac{1}{N^2}\sum_{i=1}^{N}\mathbb{E}\left\|\nabla f(x_t^i,\xi_t^i)-\nabla f_i(x_t^i)\right\|^2+\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_t^i)\right\|^2 \\
&\le \frac{\sigma^2}{N}+\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_t^i)\right\|^2, \qquad (16)
\end{aligned}$$

where the last inequality comes from Assumption 2. According to Young's Inequality and Cauchy's Inequality, we have

$$\begin{aligned}
\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_t^i)\right\|^2 &= \mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_t^i)-\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(\hat{x}_t)+\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(\hat{x}_t)\right\|^2 \\
&\le 3\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\left(\nabla f_i(x_t^i)-\nabla f_i(\hat{x}_t)\right)\right\|^2+\frac{3}{2}\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(\hat{x}_t)\right\|^2 \\
&= \frac{3}{N^2}\mathbb{E}\left\|\sum_{i=1}^{N}\left(\nabla f_i(x_t^i)-\nabla f_i(\hat{x}_t)\right)\right\|^2+\frac{3}{2}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2 \\
&\le \frac{3}{N}\sum_{i=1}^{N}\mathbb{E}\left\|\nabla f_i(x_t^i)-\nabla f_i(\hat{x}_t)\right\|^2+\frac{3}{2}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2 \\
&\le \frac{3L^2}{N}\sum_{i=1}^{N}\mathbb{E}\left\|x_t^i-\hat{x}_t\right\|^2+\frac{3}{2}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2, \qquad (17)
\end{aligned}$$

where the last inequality holds since $f_i(x)$ is L-smooth. Substituting (17) into (16), we complete the proof. □

Next, we bounded $f(\hat{x}_t)-f(x)$ for any $x\in R^d$ with Lemma 4.

Lemma 4. *Suppose Assumptions 1 and 2 hold and $f(x)$ is convex. When Algorithm 1 runs with a fixed learning rate η, for any $x\in R^d$, we have*

$$\begin{aligned}
&2\eta\sum_{t=0}^{T-1}\mathbb{E}\left(f(\hat{x}_t)-f(x)\right)-\frac{3\eta^2}{2}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2 \\
&-\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}8k\eta^2L\sum_{t=0}^{T-1}\mathbb{E}\mathcal{D}_f(\hat{x}_t,x) \\
\le\ &\|\hat{x}_0-x^*\|^2+\frac{T\eta^2\sigma^2}{N}+\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}(T\eta^2\sigma^2+4Tk\eta^2\zeta_f^x). \qquad (18)
\end{aligned}$$

Proof. Based on the update rule of Algorithm 1, we obtain

$$\begin{aligned}
\mathbb{E}\|\hat{x}_{t+1}-x\|^2 &= \mathbb{E}\|\hat{x}_t-x\|^2-2\eta\mathbb{E}\langle\hat{x}_t-x,\frac{1}{N}\sum_{i=1}^{N}\nabla f(x_t^i,\xi_t^i)\rangle+\eta^2\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f(x_t^i,\xi_t^i)\right\|^2. \\
&= \mathbb{E}\|\hat{x}_t-x\|^2-2\eta\mathbb{E}\langle\hat{x}_t-x,\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_t^i)\rangle+\eta^2\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f(x_t^i,\xi_t^i)\right\|^2 \qquad (19)
\end{aligned}$$

Since $f_i(x)$ is convex and L-smooth, we have

$$\begin{aligned}
& -\langle \hat{x}_t - x, \frac{1}{N}\sum_{i=1}^{N} \nabla f_i(x_t^i)\rangle \\
= \;& \langle x - \hat{x}_t, \frac{1}{N}\sum_{i=1}^{N} \nabla f_i(x_t^i)\rangle \\
= \;& \frac{1}{N}\sum_{i=1}^{N} \left(\langle x - x_t^i, \nabla f_i(x_t^i)\rangle + \langle x_t^i - \hat{x}_t, \nabla f_i(x_t^i)\rangle\right) \\
\leq \;& \frac{1}{N}\sum_{i=1}^{N} \left(\left(f_i(x) - f_i(x_t^i)\right) + \left(f_i(x_t^i) - f_i(\hat{x}_t) + \frac{L}{2}\|x_t^i - \hat{x}_t\|^2\right)\right) \\
= \;& \frac{1}{N}\sum_{i=1}^{N} \left(f_i(x) - f_i(\hat{x}_t) + \frac{L}{2}\|x_t^i - \hat{x}_t\|^2\right). \\
= \;& f(x) - f(\hat{x}_t) + \frac{L}{2N}\sum_{i=1}^{N} \|x_t^i - \hat{x}_t\|^2 \qquad (20)
\end{aligned}$$

Substituting (20) into (19) yields

$$\mathbb{E}\|\hat{x}_{t+1} - x\|^2 \leq \mathbb{E}\|\hat{x}_t - x\|^2 + 2\eta\left(f(x) - f(\hat{x}_t) + \frac{L}{2N}\sum_{i=1}^{N}\|x_t^i - \hat{x}_t\|^2\right) + \eta^2\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f(x_t^i, \xi_t^i)\right\|^2. \qquad (21)$$

According to (15) in Lemma 3, we have

$$\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f(x_t^i, \xi_t^i)\right\|^2 \leq \frac{\sigma^2}{N} + \frac{3L^2}{N}\sum_{i=1}^{N}\mathbb{E}\left\|x_t^i - \hat{x}_t\right\|^2 + \frac{3}{2}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2 \qquad (22)$$

Combining (21) and (22), we get

$$\begin{aligned}
\mathbb{E}\|\hat{x}_{t+1} - x\|^2 \;\leq\; & \mathbb{E}\|\hat{x}_t - x\|^2 + 2\eta\left(f(x) - f(\hat{x}_t) + \frac{L}{2N}\sum_{i=1}^{N}\|x_t^i - \hat{x}_t\|^2\right) \\
& + \eta^2\left(\frac{\sigma^2}{N} + \frac{3L^2}{N}\sum_{i=1}^{N}\mathbb{E}\left\|x_t^i - \hat{x}_t\right\|^2 + \frac{3}{2}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2\right). \\
= \;& \mathbb{E}\|\hat{x}_t - x\|^2 - 2\eta\mathbb{E}\left(f(\hat{x}_t) - f(x)\right) + \frac{3\eta^2}{2}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2 \\
& + \frac{\eta L + 3\eta^2 L^2}{N}\sum_{i=1}^{N}\|\hat{x}_t - x_t^i\|^2 + \frac{\eta^2\sigma^2}{N}. \qquad (23)
\end{aligned}$$

Summing up this inequality from $t = 0$ to $T - 1$, we have

$$\begin{aligned}
\mathbb{E}\|\hat{x}_T - x\|^2 \;\leq\; & \|\hat{x}_0 - x\|^2 - 2\eta\sum_{t=0}^{T-1}\mathbb{E}\left(f(\hat{x}_t) - f(x)\right) + \frac{3\eta^2}{2}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2 \\
& + \frac{\eta L + 3\eta^2 L^2}{N}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\|\hat{x}_t - x_t^i\|^2 + \frac{T\eta^2\sigma^2}{N}. \qquad (24)
\end{aligned}$$

Substituting (9) in Lemma 2 into (24), it holds that

$$\begin{aligned}
&2\eta\sum_{t=0}^{T-1}\mathbb{E}\left(f(\hat{x}_t)-f(x)\right)\\
\leq\ & \|\hat{x}_0-x\|^2-\mathbb{E}\|\hat{x}_T-x\|^2+\frac{3\eta^2}{2}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2\\
&+\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}\left(T\eta^2\sigma^2+8k\eta^2L\sum_{t=0}^{T-1}\mathbb{E}\mathcal{D}_f(\hat{x}_t,x)+4Tk\eta^2\zeta_f^x\right)+\frac{T\eta^2\sigma^2}{N}.\\
\leq\ & \|\hat{x}_0-x\|^2+\frac{3\eta^2}{2}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2\\
&+\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}\left(T\eta^2\sigma^2+8k\eta^2L\sum_{t=0}^{T-1}\mathbb{E}\mathcal{D}_f(\hat{x}_t,x)+4Tk\eta^2\zeta_f^x\right)+\frac{T\eta^2\sigma^2}{N}
\end{aligned}\tag{25}$$

Rearranging (25), we get

$$\begin{aligned}
&2\eta\sum_{t=0}^{T-1}\mathbb{E}\left(f(\hat{x}_t)-f(x)\right)-\frac{3\eta^2}{2}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2\\
&-\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}8k\eta^2L\sum_{t=0}^{T-1}\mathbb{E}\mathcal{D}_f(\hat{x}_t,x)\\
\leq\ & \|\hat{x}_0-x\|^2+\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}(T\eta^2\sigma^2+4Tk\eta^2\zeta_f^x)+\frac{T\eta^2\sigma^2}{N}.
\end{aligned}\tag{26}$$

□

C.2 Proof of Theorem 1

Proof. Applying (18) in Lemma 4 with $x=x^*$, it holds that

$$\begin{aligned}
&2\eta\sum_{t=0}^{T-1}\mathbb{E}\left(f(\hat{x}_t)-f(x^*)\right)-\frac{3\eta^2}{2}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla f(\hat{x}_t)\right\|^2\\
&-\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}8k\eta^2L\sum_{t=0}^{T-1}\mathbb{E}\mathcal{D}_f(\hat{x}_t,x^*)\\
\leq\ & \|\hat{x}_0-x^*\|^2+\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}(T\eta^2\sigma^2+4Tk\eta^2\zeta_f^*)+\frac{T\eta^2\sigma^2}{N}.
\end{aligned}\tag{27}$$

As $f_i(x), i\in[N]$ are L-smooth, it is easy to verify that $f(x)$ is L-smooth. According to Lemma 1, we have

$$\begin{aligned}
\|\nabla f(\hat{x}_t)\|^2 &= \|\nabla f(\hat{x}_t)-\nabla f(x^*)\|^2\\
&\leq 2L\mathcal{D}_f(\hat{x}_t,x^*)\\
&= 2L\left(f(\hat{x}_t)-f(x^*)\right).
\end{aligned}\tag{28}$$

Substituting (28) into the left hand side of (27) yields

$$\begin{aligned}
&\left(2\eta-3\eta^2L-\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}8k\eta^2L\right)\sum_{t=0}^{T-1}\mathbb{E}\left(f(\hat{x}_t)-f(x^*)\right)\\
\leq\ & \|\hat{x}_0-x^*\|^2+\frac{(\eta L+3\eta^2L^2)(k-1)}{1-2k^2\eta^2L^2}(T\eta^2\sigma^2+4Tk\eta^2\zeta_f^*)+\frac{T\eta^2\sigma^2}{N}.
\end{aligned}\tag{29}$$

Setting the learning rate η so that $\eta\leq\frac{1}{6L}$ and $\eta k\leq\frac{1}{9L}$, we have

$$\frac{\eta L+3\eta^2L^2}{1-2k^2\eta^2L^2}\leq\frac{\eta L+\frac{\eta L}{2}}{1-\frac{2}{81}}\leq\frac{7\eta L}{4},\tag{30}$$

and

$$\begin{aligned}
2\eta - 3\eta^2 L - \frac{(\eta L + 3\eta^2 L^2)8(k-1)k\eta^2 L}{1-2k^2\eta^2 L^2} &\geq 2\eta - 3\eta^2 L - \frac{(\eta L + 3\eta^2 L^2)8k^2\eta^2 L}{1-2k^2\eta^2 L^2} \\
&\geq 2\eta - \frac{\eta}{2} - \frac{(\eta + \frac{\eta}{2})8k^2\eta^2 L^2}{1-\frac{2}{81}} \\
&\geq 2\eta - \frac{\eta}{2} - \frac{81}{79} \times \frac{3}{2} \times \frac{8}{81}\eta \\
&\geq \frac{4}{3}\eta. \qquad (31)
\end{aligned}$$

Substituting (30) and (31) into (29), we get

$$\frac{4\eta}{3}\sum_{t=0}^{T-1}\mathbb{E}\left(f(\hat{x}_t) - f(x^*)\right) \leq \|\hat{x}_0 - x^*\|^2 + \frac{7}{4}T\eta^3 L(k-1)(\sigma^2 + 4k\zeta_f^*) + \frac{T\eta^2\sigma^2}{N}.$$

Dividing by $\frac{4\eta T}{3}$ on both sides of the above inequality yields

$$\begin{aligned}
\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left(f(\hat{x}_t) - f(x^*)\right) &\leq \frac{3\|\hat{x}_0 - x^*\|^2}{4\eta T} + \frac{21}{16}\eta^2 L(k-1)(\sigma^2 + 4k\zeta_f^*) + \frac{3\eta\sigma^2}{4N}. \\
&\leq \frac{3\|\hat{x}_0 - x^*\|^2}{4\eta T} + \frac{3}{2}\eta^2 L(k-1)(\sigma^2 + 4k\zeta_f^*) + \frac{3\eta\sigma^2}{4N}.
\end{aligned}$$

Recall that we let $\tilde{x} = \hat{x}_t$ for randomly chosen t from $\{0, 1, \cdots, T-1\}$. Taking the expectation with regard to t, we get

$$\mathbb{E}f(\tilde{x}) - f(x^*) \leq \frac{3\|\hat{x}_0 - x^*\|^2}{4\eta T} + \frac{3}{2}\eta^2 L(k-1)(\sigma^2 + 4k\zeta_f^*) + \frac{3\eta\sigma^2}{4N}. \qquad (32)$$

Under the result of (32), we set k as

$$k = \begin{cases} \min\{\frac{1}{6\eta LN}, \frac{1}{9\eta L}\} & \zeta_f^* = 0, \\ \min\{\frac{\sigma}{\sqrt{6\eta LN(\sigma^2 + 4\zeta_f)}}, \frac{1}{9\eta L}\} & else. \end{cases} \qquad (33)$$

For the IID case, i.e., $\zeta_f^* = 0$, based on the setting of k in (33), we have

$$\begin{aligned}
\frac{3}{2}\eta^2 L(k-1)(\sigma^2 + 4k\zeta_f^*) &\leq \frac{3}{2}\eta^2 Lk\sigma^2 \\
&\leq \frac{3}{2}\eta^2 L\frac{1}{6\eta LN}\sigma^2 \\
&= \frac{\eta\sigma^2}{4N}. \qquad (34)
\end{aligned}$$

For the Non-IID case, we get

$$\begin{aligned}
\frac{3}{2}\eta^2 L(k-1)(\sigma^2 + 4k\zeta_f^*) &\leq \frac{3}{2}\eta^2 Lk^2(\sigma^2 + 4\zeta_f^*) \\
&\leq \frac{3}{2}\eta^2 L\frac{\sigma^2}{6\eta LN(\sigma^2 + 4\zeta_f^*)}(\sigma^2 + 4\zeta_f^*) \\
&= \frac{\eta\sigma^2}{4N}. \qquad (35)
\end{aligned}$$

Substituting (34) and (35) into (32) yields

$$\mathbb{E}f(\tilde{x}) - f(x^*) \leq \frac{3\|\hat{x}_0 - x^*\|^2}{4\eta T} + \frac{\eta\sigma^2}{N}, \qquad (36)$$

which completes the proof. □

D Proofs for Results in Section 4.1

Proof of Theorem 2

Proof. Based on the parameter settings in Algorithm 2, we have

$$\eta_s T_s = \frac{\eta_1}{2^{s-1}} \cdot 2^{s-1} T_1 = \eta_1 T_1 = \frac{6}{\mu} \tag{37}$$

and

$$\begin{aligned} k_s &= \begin{cases} (\sqrt{2})^{s-1} k_1 \\ 2^{s-1} k_1 \end{cases} \leq \begin{cases} (\sqrt{2})^{s-1} \min\{\frac{\sigma}{\sqrt{6\eta_1 LN(\sigma^2+4\zeta_f)}}, \frac{1}{9\eta_1 L}\} \\ 2^{s-1} \min\{\frac{1}{6\eta_1 LN}, \frac{1}{9\eta_1 L}\} \end{cases} \\ &= \begin{cases} \min\{\frac{\sigma}{\sqrt{6\eta_s LN(\sigma^2+4\zeta_f)}}, \frac{1}{9(\sqrt{2})^{s-1}\eta_s L}\} \\ \min\{\frac{1}{6\eta_s LN}, \frac{1}{9\eta_s L}\} \end{cases} \\ &\leq \begin{cases} \min\{\frac{\sigma}{\sqrt{6\eta_s LN(\sigma^2+4\zeta_f)}}, \frac{1}{9\eta_s L}\}, & \text{Non-IID case,} \\ \min\{\frac{1}{6\eta_s LN}, \frac{1}{9\eta_s L}\}, & \text{IID case.} \end{cases} \end{aligned} \tag{38}$$

Thus, according to (37), (38) and (2) in Theorem 1, we get

$$\mathbb{E}f(x_{s+1}) - f(x^*) \leq \frac{3\mathbb{E}\|x_s - x^*\|^2}{4\eta_s T_s} + \frac{\eta_s \sigma^2}{N} = \frac{\mu \mathbb{E}\|x_s - x^*\|^2}{8} + \frac{\eta_1 \sigma^2}{2^{s-1}N}. \tag{39}$$

Since the objective $f(x)$ is μ-strongly convex, we have

$$\frac{\mu \mathbb{E}\|x_s - x^*\|^2}{8} \leq \frac{\mathbb{E}f(x_s) - f(x^*)}{4}. \tag{40}$$

Substituting (40) into (39) yields

$$\mathbb{E}f(x_{s+1}) - f(x^*) \leq \frac{\mathbb{E}f(x_s) - f(x^*)}{4} + \frac{\eta_1 \sigma^2}{2^{s-1}N}. \tag{41}$$

Subtracting $\frac{\eta_1 \sigma^2}{2^{s-2}}$ on both sides of (41), we get

$$\mathbb{E}f(x_{s+1}) - f(x^*) - \frac{8\eta_1 \sigma^2}{2^{s+1}N} \leq \frac{1}{4}(\mathbb{E}f(x_s) - f(x^*) - \frac{8\eta_1 \sigma^2}{2^s N}).$$

Based on the property of geometric progression, we have

$$\mathbb{E}f(x_S) - f(x^*) - \frac{8\eta_1 \sigma^2}{2^S N} \leq \frac{1}{4^{S-1}}(\mathbb{E}f(x_1) - f(x^*) - \frac{4\eta_1 \sigma^2}{N}). \tag{42}$$

Setting $S \geq \log(\frac{N(f(x_1)-f(x^*))}{\eta_1 \sigma^2}) + 2$ gives

$$f(x_1) - f(x^*) \leq \frac{2^{S-2}\eta_1 \sigma^2}{N}. \tag{43}$$

By substituting (43) into (42) and rearranging the result further, we obtain

$$\begin{aligned} \mathbb{E}f(x_S) - f(x^*) &\leq \frac{8\eta_1 \sigma^2}{2^S N} + \frac{1}{4^{S-1}}(\mathbb{E}f(x_1) - f(x^*) - \frac{4\eta_1 \sigma^2}{N}) \\ &\leq \frac{8\eta_1 \sigma^2}{2^S N} + \frac{\mathbb{E}f(x_1) - f(x^*)}{4^{S-1}} \\ &\leq \frac{8\eta_1 \sigma^2}{2^S N} + \frac{\eta_1 \sigma^2}{2^S N} \\ &= \frac{9\eta_1 \sigma^2}{2^S N}. \end{aligned} \tag{44}$$

Since $T_s = 2^{s-1} T_1$, we have

$$\begin{aligned} T &= T_1 + T_2 + \cdots + T_S \\ &= T_1(1 + 2 + \cdots + 2^{S-1}) \\ &= T_1(2^S - 1). \end{aligned}$$

Thus, it holds that

$$S = \log\left(\frac{T}{T_1} + 1\right).$$

Replacing S with $\log(\frac{T}{T_1} + 1)$ in (44) and combining $\eta_1 T_1 = \frac{6}{\mu}$, we have

$$\begin{aligned}
\mathbb{E}f(x_S) - f(x^*) &\leq \frac{9\eta_1\sigma^2}{(\frac{T}{T_1}+1)N} \\
&= \frac{9\eta_1 T_1\sigma^2}{(T+T_1)N} \\
&= \frac{54\sigma^2}{\mu(T+T_1)N} \\
&= O\left(\frac{1}{NT}\right).
\end{aligned}$$

□

E Proofs for Results in Section 4.2

E.1 Proof for result of STL-SGD^{nc} with Option 1

We will first analyse the convergence of Local-SGD for a single stage in Lemma 5. Then we extend the result to S stages in Theorem 3.

Lemma 5. *Suppose Assumptions 1, 2 and 3 hold. Let $\gamma^{-1} = 2\rho$, $\eta_s \leq \frac{1}{12L_\gamma}$ and $k_s\eta_s \leq \frac{1}{9L_\gamma}$, where $L_\gamma = L + \gamma^{-1}$. We have the following result for stage s of Algorithm 3 with **Option 1**:*

$$\begin{aligned}
&\mathbb{E}f(x_{s+1}) - f(x^*) \\
\leq\ & \left(\frac{3}{4\eta_s T_s} + \frac{1127\rho}{632}\right)\|x_s - x^*\|^2 + \frac{3\eta_s\sigma^2}{4N} + \frac{3}{2}\eta_s^2 L_\gamma(k_s-1)(\sigma^2 + 4k_s\zeta_f^*). \qquad (45)
\end{aligned}$$

Proof. We let the objectives in all stages be convex by setting $\gamma^{-1} > \rho$, where ρ is the weakly convex parameter in Assumption 3. Recall that $f(x)$ is L-smooth. Denoting $L_\gamma = L + \frac{1}{\gamma}$, we have

$$\begin{aligned}
\|\nabla f_{x_s}^\gamma(x) - \nabla f_{x_s}^\gamma(y)\| &= \left\|\nabla f(x) - \nabla f(y) + \frac{1}{\gamma}(x-y)\right\| \\
&\leq \|\nabla f(x) - \nabla f(y)\| + \frac{1}{\gamma}\|(x-y)\| \\
&\leq \left(L + \frac{1}{\gamma}\right)\|x-y\| \\
&= L_\gamma\|x-y\|, \qquad (46)
\end{aligned}$$

where the first inequality comes from the Triangle Inequality. Thus, $f_{x_s}^\gamma(x)$ is L_γ-smooth. Based on Assumption 2, we further have

$$\mathbb{E}_{\xi\sim\mathcal{D}_i}\|\nabla f_{x_s}^\gamma(x,\xi) - \nabla f_{x_s,i}^\gamma(x)\|^2 = \mathbb{E}_{\xi\sim\mathcal{D}_i}\|\nabla f(x,\xi) - \nabla f_i(x)\|^2 \leq \sigma^2. \qquad (47)$$

As we set $\gamma^{-1} > \rho$, $f^{\gamma}_{x_s}$ is $(\gamma^{-1}-\rho)$-strongly convex, thus we have

$$\begin{aligned}
& -\langle \hat{x}_t - x, \frac{1}{N}\sum_{i=1}^{N} \nabla f^{\gamma}_{x_s,i}(x^i_t)\rangle \\
= \;& \langle x - \hat{x}_t, \frac{1}{N}\sum_{i=1}^{N} \nabla f^{\gamma}_{x_s,i}(x^i_t)\rangle \\
= \;& \frac{1}{N}\sum_{i=1}^{N} \left(\langle x - x^i_t, \nabla f^{\gamma}_{x_s,i}(x^i_t)\rangle + \langle x^i_t - \hat{x}_t, \nabla f^{\gamma}_{x_s,i}(x^i_t)\rangle\right) \\
\leq \;& \frac{1}{N}\sum_{i=1}^{N} \Bigg(\left(f^{\gamma}_{x_s,i}(x) - f^{\gamma}_{x_s,i}(x^i_t) - \frac{\gamma^{-1}-\rho}{2}\|x^i_t - x\|^2 \right) \\
& + \left(f^{\gamma}_{x_s,i}(x^i_t) - f^{\gamma}_{x_s,i}(\hat{x}_t) + \frac{L}{2}\|x^i_t - \hat{x}_t\|^2 \right) \Bigg) \\
= \;& \frac{1}{N}\sum_{i=1}^{N} \left(f^{\gamma}_{x_s,i}(x) - f^{\gamma}_{x_s,i}(\hat{x}_t) + \frac{L}{2}\|x^i_t - \hat{x}_t\|^2 - \frac{\gamma^{-1}-\rho}{2}\|x^i_t - x\|^2 \right). \\
\leq \;& f^{\gamma}_{x_s}(x) - f^{\gamma}_{x_s}(\hat{x}_t) + \frac{L}{2N}\sum_{i=1}^{N}\|x^i_t - \hat{x}_t\|^2 - \frac{\gamma^{-1}-\rho}{2}\|\hat{x}_t - x\|^2, \qquad (48)
\end{aligned}$$

where the last inequality holds because the function $g(x) = \|x\|^2$ is convex. Respectively replacing (20) with (48), L with L_γ and x with x^*, going through the proof process in Lemma 4 again, we get

$$\begin{aligned}
& 2\eta_s \sum_{t=0}^{T_s-1} \mathbb{E}\left(f^{\gamma}_{x_s}(\hat{x}_t) - f^{\gamma}_{x_s}(x^*)\right) - \frac{3\eta_s^2}{2}\sum_{t=0}^{T_s-1}\mathbb{E}\left\|\nabla f^{\gamma}_{x_s}(\hat{x}_t)\right\|^2 \\
& - \frac{(\eta_s L_\gamma + 3\eta_s^2 L_\gamma^2)(k_s-1)}{1-2k_s^2\eta_s^2 L_\gamma^2} 8k_s\eta_s^2 L_\gamma \sum_{t=0}^{T_s-1}\mathbb{E}\mathcal{D}_{f^{\gamma}_{x_s}}(\hat{x}_t, x^*) \\
\leq \;& \|\hat{x}_0 - x^*\|^2 - \eta_s(\gamma^{-1}-\rho)\sum_{t=0}^{T_s-1}\|\hat{x}_t - x^*\|^2 \\
& + \frac{(\eta_s L_\gamma + 3\eta_s^2 L_\gamma^2)(k_s-1)}{1-2k_s^2\eta_s^2 L_\gamma^2}(T_s\eta_s^2\sigma^2 + 4T_s k_s\eta_s^2\zeta^*_{f^{\gamma}_{x_s}}) + \frac{T_s\eta_s^2\sigma^2}{N}, \qquad (49)
\end{aligned}$$

where $\mathcal{D}_{f^{\gamma}_{x_s}}(\hat{x}_t, x^*) = f^{\gamma}_{x_s}(\hat{x}_t) - f^{\gamma}_{x_s}(x^*) - \langle \nabla f^{\gamma}_{x_s}(x^*), \hat{x}_t - x^*\rangle$ and $\zeta^*_{f^{\gamma}_{x_s}} = \frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(x^*) + \frac{x^*-x_s}{\gamma}\|^2$. We bound $\|\nabla f^{\gamma}_{x_s}(\hat{x}_t)\|^2$ as

$$\begin{aligned}
\|\nabla f^{\gamma}_{x_s}(\hat{x}_t)\|^2 &= \|\nabla f^{\gamma}_{x_s}(\hat{x}_t) - \nabla f^{\gamma}_{x_s}(x^*) + \nabla f^{\gamma}_{x_s}(x^*)\|^2 \\
&\leq 2\|\nabla f^{\gamma}_{x_s}(\hat{x}_t) - \nabla f^{\gamma}_{x_s}(x^*)\|^2 + 2\|\nabla f^{\gamma}_{x_s}(x^*)\|^2 \\
&\leq 4L_\gamma \mathcal{D}_{f^{\gamma}_{x_s}}(\hat{x}_t, x^*) + \frac{2}{\gamma^2}\|x^* - x_s\|^2, \qquad (50)
\end{aligned}$$

where the last inequality comes from Lemma 1. As $\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x^*) = \nabla f(x^*) = 0$, we have

$$\begin{aligned}
\zeta^*_{f^{\gamma}_{x_s}} &= \frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(x^*) + \frac{x^* - x_s}{\gamma}\|^2 \\
&= \frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(x^*)\|^2 + \|\frac{x^* - x_s}{\gamma}\|^2 \\
&= \zeta^*_f + \frac{1}{\gamma^2}\|x^* - x_s\|^2 \qquad (51)
\end{aligned}$$

and

$$\begin{aligned}\mathcal{D}_{f_{x_s}^{\gamma}}(\hat{x}_t, x^*) &= f_{x_s}^{\gamma}(\hat{x}_t) - f_{x_s}^{\gamma}(x^*) + \frac{1}{\gamma}\langle x^* - x_s, x^* - \hat{x}_t\rangle \\ &\overset{(a)}{=} f_{x_s}^{\gamma}(\hat{x}_t) - f_{x_s}^{\gamma}(x^*) + \frac{1}{2\gamma}\left(\|x^* - x_s\|^2 - \|x_s - \hat{x}_t\|^2 + \|x^* - \hat{x}_t\|^2\right), \quad (52)\end{aligned}$$

where (a) is based on the fact that $\langle x-y, x-z\rangle = \frac{1}{2}\|x-y\|^2 - \frac{1}{2}\|y-z\|^2 + \frac{1}{2}\|x-z\|^2$. Substituting (50), (51), (52) into (49) and taking the expectation regarding t, we get

$$\begin{aligned}&T_s\left(2\eta_s - 6\eta_s^2 L_\gamma - 8A_\gamma k_s \eta_s^2 L_\gamma\right)\left(f_{x_s}^{\gamma}(x_{s+1}) - f_{x_s}^{\gamma}(x^*)\right) \\ &- \left(\frac{3\eta_s^2 T_s}{\gamma^2} + \frac{3\eta_s^2 L_\gamma T_s}{\gamma} + \frac{4A_\gamma k_s \eta_s^2 L_\gamma T_s}{\gamma}\right)\|x^* - x_s\|^2 \\ \leq\ & (1 + \frac{4A_\gamma k_s \eta_s^2 T_s}{\gamma^2})\|x_s - x^*\|^2 + \left(\frac{4A_\gamma k_s \eta_s^2 L_\gamma}{\gamma} + \frac{3\eta_s^2 L_\gamma}{\gamma} - \eta_s(\gamma^{-1} - \rho)\right)\sum_{t=0}^{T_s-1}\|\hat{x}_t - x^*\|^2 \\ &+ A_\gamma T_s \eta_s^2(\sigma^2 + 4k_s\zeta_f^*) + \frac{T_s\eta_s^2\sigma^2}{N}, \quad (53)\end{aligned}$$

where $A_\gamma = \frac{(\eta_s L_\gamma + 3\eta_s^2 L_\gamma^2)(k_s-1)}{1 - 2k_s^2\eta_s^2 L_\gamma^2}$. Setting $\gamma = \frac{1}{2\rho}$, $\eta_s \leq \frac{1}{12L_\gamma}$ and $\eta_s k_s \leq \frac{1}{9L_\gamma}$, we have

$$A_\gamma k_s \eta_s^2 = \frac{(\eta_s L_\gamma + 3\eta_s^2 L_\gamma^2)(k_s-1)}{1 - 2k_s^2\eta_s^2 L_\gamma^2} k_s \eta_s^2 \leq \frac{(\eta_s + \frac{\eta_s}{4})k_s^2\eta_s^2 L_\gamma^2}{(1 - \frac{2}{81})L_\gamma} \leq \frac{\frac{5\eta_s}{4}}{\frac{79}{81}L_\gamma}\frac{1}{81} = \frac{5\eta_s}{316L_\gamma}, \quad (54)$$

$$2\eta_s - 6\eta_s^2 L_\gamma - 8A_\gamma k_s\eta_s^2 L_\gamma \geq 2\eta_s - \frac{\eta_s}{2} - \frac{10}{79}\eta_s \geq \frac{4}{3}\eta_s \quad (55)$$

and

$$\frac{4A_\gamma k_s\eta_s^2 L_\gamma}{\gamma} + \frac{3\eta_s^2 L_\gamma}{\gamma} - \eta_s(\gamma^{-1} - \rho) \leq \frac{10\eta_s\rho}{79} + \frac{\eta_s\rho}{2} - \eta_s\rho \leq 0. \quad (56)$$

Substituting (54), (55) and (56) into (53) yields

$$\begin{aligned}&\frac{4\eta_s T_s}{3}\left(f_{x_s}^{\gamma}(x_{s+1}) - f_{x_s}^{\gamma}(x^*)\right) \\ \leq\ & \left(1 + \frac{20T_s\eta_s\rho^2}{79L_\gamma} + 12\eta_s^2 T_s\rho^2 + 6\eta_s^2 L_\gamma T_s\rho + \frac{10T_s\eta_s\rho}{79}\right)\|x_s - x^*\|^2 \\ &+ \frac{3}{2}T_s\eta_s^3 L_\gamma(k_s-1)(\sigma^2 + 4k_s\zeta_f^*) + \frac{T_s\eta_s^2\sigma^2}{N}. \quad (57)\end{aligned}$$

By the definition of $f_{x_s}^{\gamma}(x)$ and $\gamma^{-1} = 2\rho$, we have

$$\begin{aligned}f_{x_s}^{\gamma}(x_{s+1}) - f_{x_s}^{\gamma}(x^*) &= f(x_{s+1}) - f(x^*) + \rho\|x_{s+1} - x_s\|^2 - \rho\|x^* - x_s\|^2 \\ &\geq f(x_{s+1}) - f(x^*) - \rho\|x^* - x_s\|^2. \quad (58)\end{aligned}$$

Substituting (58) into (57) and rearranging the result further, we get

$$\begin{aligned}&\frac{4\eta_s T_s}{3}\left(f(x_{s+1}) - f(x^*)\right) \\ \leq\ & \left(1 + \frac{20T_s\eta_s\rho^2}{79L_\gamma} + 12\eta_s^2 T_s\rho^2 + 6\eta_s^2 L_\gamma T_s\rho + \frac{10T_s\eta_s\rho}{79} + \frac{4\eta_s T_s\rho}{3}\right)\|x_s - x^*\|^2 \\ &+ \frac{3}{2}T_s\eta_s^3 L_\gamma(k_s-1)(\sigma^2 + 4k_s\zeta_f^*) + \frac{T_s\eta_s^2\sigma^2}{N}.\end{aligned}$$

Dividing by $\frac{4\eta_s T_s}{3}$ on both sides of the above inequality yields

$$\begin{aligned}f(x_{s+1}) - f(x^*) \leq\ & \left(\frac{3}{4\eta_s T_s} + \frac{15\rho^2}{79L_\gamma} + 9\eta_s\rho^2 + \frac{9\eta_s L_\gamma\rho}{2} + \frac{15\rho}{158} + \rho\right)\|x_s - x^*\|^2 \\ &+ \frac{3}{2}\eta_s^2 L_\gamma(k_s-1)(\sigma^2 + 4k_s\zeta_f^*) + \frac{3\eta_s\sigma^2}{4N}.\end{aligned}$$

As $L \geq \rho$, we have $L_\gamma = L + \frac{1}{\gamma} \geq 3\rho$, $\eta_s \leq \frac{1}{12L_\gamma} \leq \frac{1}{36\rho}$ and

$$f(x_{s+1}) - f(x^*) \leq \left(\frac{3}{4\eta_s T_s} + \frac{1127\rho}{632}\right)\|x_s - x^*\|^2 + \frac{3}{2}\eta_s^2 L_\gamma(k_s - 1)(\sigma^2 + 4k_s\zeta_f^*) + \frac{3\eta_s\sigma^2}{4N}.$$

□

Proof of Theorem 3

Proof. Since $f(x)$ satisfies the PL condition with parameter μ, we have

$$\frac{\mu}{2}\|x - x^*\|^2 \leq f(x) - f(x^*). \tag{59}$$

Combining (59) with the result of Lemma 5, we have

$$\begin{aligned} & f(x_{s+1}) - f(x^*) \\ \leq & \left(\frac{3}{4\eta_s T_s} + \frac{1127\rho}{632}\right)\|x_s - x^*\|^2 + \frac{3}{2}\eta_s^2 L_\gamma(k_s - 1)(\sigma^2 + 4k_s\zeta_f^*) + \frac{3\eta_s\sigma^2}{4N} \\ \leq & \left(\frac{3}{4\eta_s T_s} + \frac{1127\rho}{632}\right)\frac{2}{\mu}\left(f(x_s) - f(x^*)\right) + \frac{3}{2}\eta_s^2 L_\gamma(k_s - 1)(\sigma^2 + 4k_s\zeta_f^*) + \frac{3\eta_s\sigma^2}{4N}. \end{aligned} \tag{60}$$

According to the parameter settings in **Option 1** of Algorithm 3, we have

$$\eta_s T_s = \frac{\eta_1}{2^{s-1}} \cdot 2^{s-1} T_1 = \eta_1 T_1 = \frac{6}{\rho} \tag{61}$$

and

$$\begin{aligned} k_s &= \begin{cases} (\sqrt{2})^{s-1} k_1 \\ 2^{s-1} k_1 \end{cases} \leq \begin{cases} (\sqrt{2})^{s-1} \min\{\frac{\sigma}{\sqrt{6\eta_1 L_\gamma N(\sigma^2 + 4\zeta_f)}}, \frac{1}{9\eta_1 L_\gamma}\} \\ 2^{s-1} \min\{\frac{1}{6\eta_1 L_\gamma N}, \frac{1}{9\eta_1 L_\gamma}\} \end{cases} \\ &= \begin{cases} \min\{\frac{\sigma}{\sqrt{6\eta_s L_\gamma N(\sigma^2 + 4\zeta_f)}}, \frac{1}{9(\sqrt{2})^{s-1}\eta_s L_\gamma}\} \\ \min\{\frac{1}{6\eta_s L_\gamma N}, \frac{1}{9\eta_s L_\gamma}\} \end{cases} \\ &\leq \begin{cases} \min\{\frac{\sigma}{\sqrt{6\eta_s L_\gamma N(\sigma^2 + 4\zeta_f)}}, \frac{1}{9\eta_s L_\gamma}\}, & \text{Non-IID case,} \\ \min\{\frac{1}{6\eta_s L_\gamma N}, \frac{1}{9\eta_s L_\gamma}\}, & \text{IID case.} \end{cases} \end{aligned} \tag{62}$$

Similar to the proof of (34) and (35), we have

$$\frac{3}{2}\eta_s^2 L_\gamma(k_s - 1)(\sigma^2 + 4k_s\zeta_f^*) \leq \frac{\eta_s\sigma^2}{4N}. \tag{63}$$

Substituting (61) and (63) into (60), according to $\mu \geq 16\rho$, we have

$$\begin{aligned} f(x_{s+1}) - f(x^*) &\leq \left(\frac{3}{4\eta_s T_s} + \frac{1127\rho}{632}\right)\frac{2}{\mu}\left(f(x_s) - f(x^*)\right) + \frac{\eta_s\sigma^2}{N}. \\ &= \left(\frac{\rho}{8} + \frac{1127\rho}{632}\right)\frac{2}{\mu}\left(f(x_s) - f(x^*)\right) + \frac{\eta_1\sigma^2}{2^{s-1}N} \\ &\leq \frac{1}{4}\left(f(x_s) - f(x^*)\right) + \frac{\eta_1\sigma^2}{2^{s-1}N}. \end{aligned} \tag{64}$$

Note that the formula of (64) is the same as (41). Thus, the rest of the proof is a duplicate to that of Theorem 2. □

E.2 Proof for result of STL-SGD^{nc} with Option 2

Proof of Theorem 4

Proof. For convenience of analysis, we let x_s^* denote the optimal solution of the objective used in the s-th stage $f_{x_s}^\gamma(x)$. According to (46) and (47), we have that $f_{x_s}^\gamma$ is L_γ-smooth and the variance of its stochastic gradients is bounded by σ^2. We set $\eta_1 \leq \frac{1}{6L_\gamma}$, $k_1 = \min\{\frac{1}{6\eta_1 L_\gamma N}, \frac{1}{9\eta_1 L_\gamma}\}$ when $\zeta_f^* = 0$ and $k_1 = \min\{\frac{\sigma}{\sqrt{6\eta_1 L_\gamma N(\sigma^2+4\zeta_f^*)}}, \frac{1}{9\eta_1 L_\gamma}\}$ when $\zeta_f^* \neq 0$. As $\eta_s = \eta_1/s$ and $k_s = \begin{cases} sk_1, & \text{IID case} \\ \sqrt{s}k_1, & \text{else} \end{cases}$, we have

$$\eta_s \leq \frac{1}{6L_\gamma} \tag{65}$$

and

$$k_s \leq \begin{cases} \min\{\frac{1}{6\eta_s L_\gamma N}, \frac{1}{9\eta_s L_\gamma}\}, & \text{IID case,} \\ \min\{\frac{\sigma}{\sqrt{6\eta_s L_\gamma N(\sigma^2+4\zeta_f^*)}}, \frac{1}{9\eta_s L_\gamma}\}, & \text{else.} \end{cases} \tag{66}$$

By setting $\gamma^{-1} > \rho$, we can ensure that $f_{x_s}^\gamma$ is strongly convex. Based on these settings, we apply Theorem 1 in each call of Local-SGD in STL-SGD^{nc}:

$$f_{x_s}^\gamma(x_{s+1}) - f_{x_s}^\gamma(x_s^*) \leq \frac{3\|x_s - x_s^*\|^2}{4\eta_s T_s} + \frac{\eta_s \sigma^2}{N}. \tag{67}$$

Under the definition $f_{x_s}^\gamma(x_{s+1}) = f(x_{s+1}) + \frac{1}{2\gamma}\|x_{s+1} - x_s\|^2$, and the strong convexity $f_{x_s}^\gamma(x_s) - f_{x_s}^\gamma(x_s^*) \geq \frac{\gamma^{-1}-\rho}{2}\|x_s - x_s^*\|^2$, we have

$$f(x_{s+1}) + \frac{1}{2\gamma}\|x_{s+1} - x_s\|^2 + \frac{\gamma^{-1}-\rho}{2}\|x_s - x_s^*\|^2 - f(x_s) \leq \frac{3\|x_s - x_s^*\|^2}{4\eta_s T_s} + \frac{\eta_s \sigma^2}{N}. \tag{68}$$

Setting $\gamma^{-1} = 2\rho$ and rearranging (68) yields

$$\rho\|x_{s+1} - x_s\|^2 + \frac{\rho}{2}\|x_s - x_s^*\|^2 \leq f(x_s) - f(x_{s+1}) + \frac{3\|x_s - x_s^*\|^2}{4\eta_s T_s} + \frac{\eta_s \sigma^2}{N}. \tag{69}$$

As $\eta_s = \eta_1/s$, $T_s = sT_1$ and $\eta_1 T_1 = \frac{3}{\rho}$, we have

$$\rho\|x_{s+1} - x_s\|^2 + \frac{\rho}{4}\|x_s - x_s^*\|^2 \leq f(x_s) - f(x_{s+1}) + \frac{\eta_1 \sigma^2}{sN}. \tag{70}$$

According to the L_γ-smoothness of $f_{x_s}^\gamma(x)$, we have

$$\|\nabla f(x_s)\|^2 = \|\nabla f_{x_s}^\gamma(x_s)\|^2 = \|\nabla f_{x_s}^\gamma(x_s) - \nabla f_{x_s}^\gamma(x_s^*)\|^2 \leq L_\gamma^2\|x_s - x_s^*\|^2. \tag{71}$$

Combining (70) and (71) yields

$$\frac{\rho}{4L_\gamma^2}\|\nabla f(x_s)\|^2 \leq \frac{\rho}{4}\|x_s - x_s^*\|^2 \leq f(x_s) - f(x_{s+1}) + \frac{\eta_1 \sigma^2}{sN}. \tag{72}$$

Define $w_s = s$ and $\Delta_s = f(x_s) - f(x_{s+1})$. Multiplying both sides by w_s, we have

$$\frac{\rho w_s}{4L_\gamma^2}\|\nabla f(x_s)\|^2 \leq w_s\Delta_s + \frac{w_s \eta_1 \sigma^2}{sN}. \tag{73}$$

After telescoping (72) for $s = 1, 2, \cdots, S$, we get

$$\sum_{s=1}^S w_s\|\nabla f(x_s)\|^2 \leq \frac{4L_\gamma^2}{\rho}\left(\sum_{s=1}^S w_s\Delta_s + \sum_{s=1}^S \frac{w_s\eta_1\sigma^2}{sN}\right). \tag{74}$$

Taking the expectation w.r.t $s \in \{1, 2, \cdots, S\}$ with probability $p_s = \frac{s}{1+2+\cdots+S}$, we have

$$\mathbb{E}\|\nabla f(x_s)\|^2 \leq \frac{4L_\gamma^2}{\rho}\left(\frac{\sum_{s=1}^S w_s\Delta_s}{\sum_{s=1}^S w_s} + \frac{\sum_{s=1}^S \frac{w_s\eta_1\sigma^2}{sN}}{\sum_{s=1}^S w_s}\right). \tag{75}$$

Based on the definition of w_s and Δ_s, setting $w_0 = 0$, we have

$$\begin{aligned}\sum_{s=1}^{S} w_s \Delta_s &= \sum_{s=1}^{S} w_s \left(f(x_s) - f(x_{s+1})\right) = \sum_{s=1}^{S} f(x_s) - S f(x_{S+1}) \\ &\leq S(f(\bar{x}) - f(x_{S+1})) \leq w_S(f(\bar{x}) - f(x^*)), \end{aligned} \tag{76}$$

where $\bar{x} = arg\max_{x_i, i\in[S]} f(x_i)$. Substituting (76) into (75), we get

$$\begin{aligned}\mathbb{E}\|\nabla f(x_s)\|^2 &\leq \frac{4L_\gamma^2}{\rho}\left(\frac{w_S(f(\bar{x}) - f(x^*))}{\sum_{s=1}^{S} w_s} + \frac{\sum_{s=1}^{S} \frac{w_s \eta_1 \sigma^2}{sN}}{\sum_{s=1}^{S} w_s}\right) \\ &= \frac{8L_\gamma^2}{\rho}\left(\frac{f(\bar{x}) - f(x^*)}{S+1} + \frac{\eta_1 \sigma^2}{(S+1)N}\right). \end{aligned} \tag{77}$$

As $T_s = sT_1$, we have

$$T = T_1 + T_2 + \cdots + T_S = T_1(1 + 2 + \cdots + S) = T_1 \frac{S(S+1)}{2} \leq T_1 \frac{(S+1)^2}{2}. \tag{78}$$

Substituting $S + 1 \geq \sqrt{\frac{2T}{T_1}}$ into (77), we get

$$\begin{aligned}\mathbb{E}\|\nabla f(x_s)\|^2 &\leq \frac{8L_\gamma^2}{\rho}\left(\frac{(f(\bar{x}) - f(x^*))}{\sqrt{\frac{2T}{T_1}}} + \frac{\eta_1 \sigma^2}{\sqrt{\frac{2T}{T_1}} N}\right) \\ &= O\left(\frac{(f(\bar{x}) - f(x^*))\sqrt{T_1}}{\sqrt{T}} + \frac{\sqrt{T_1}\eta_1 \sigma^2}{N\sqrt{T}}\right) \\ &= O\left(\frac{f(\bar{x}) - f(x^*)}{\sqrt{T\eta_1}} + \frac{\sqrt{\eta_1}\sigma^2}{N\sqrt{T}}\right), \end{aligned} \tag{79}$$

where the last equality holds since $\eta_1 T_1 = 3/\rho$. We use η_1^N to denote the learning rate when using N clients. Setting $\eta_1^N = N\eta_1^1$ yields

$$\mathbb{E}\|\nabla f(x_s)\|^2 \leq O\left(\frac{1}{\sqrt{NT}}\right), \tag{80}$$

which completes the proof.

□