Towards Optimal Communication Complexity in Distributed Non-Convex Optimization

Kumar Kshitij Patel[*]
TTIC
`kkpatel@ttic.edu`

Lingxiao Wang[*]
TTIC
`lingxw@ttic.edu`

Blake Woodworth
SIERRA, INRIA
`blakewoodworth@gmail.com`

Brian Bullins[†]
Purdue University
`bbullins@purdue.edu`

Nati Srebro
TTIC
`nati@ttic.edu`

Abstract

We study the problem of distributed stochastic non-convex optimization with intermittent communication. We consider the full participation setting where M machines work in parallel over R communication rounds and the partial participation setting where M machines are sampled independently every round from some meta-distribution over machines. We propose and analyze a new algorithm that improves existing methods by requiring fewer and lighter variance reduction operations. We also present lower bounds, showing our algorithm is either *optimal* or *almost optimal* in most settings. Numerical experiments demonstrate the superior performance of our algorithm.

1 Introduction

We consider the following distributed optimization problem with M machines:

$$\min_{x\in\mathbb{R}^d} F(x) := \frac{1}{M}\sum_{m=1}^{M} F_m(x), \tag{1.1}$$

where F_m, which denotes the objective on machine m, is a non-convex function for all m, as is the average objective F. We want to solve this problem in the *intermittent communication* (IC) setting [1, 2] where the machines work in parallel and are allowed to make K oracle calls between two communication rounds for R consecutive rounds. The IC setting has been widely studied [3, 4, 5, 6, 7, 8, 9, 10, 11, 2, 12] over the past decade. Many recent works have focused on the problem with non-convex and heterogeneous objectives [13, 14, 15] which are common in cross-device federated learning (FL) [16, 17]. Towards this end, several algorithms [18, 19, 20, 21, 22, 23], all involving *local updates* (à la local-SGD [3, 16]), have been proposed and analyzed under assumptions bounding the heterogeneity of machines' objectives. Although these algorithms demonstrate promising empirical performances, it remains elusive whether these algorithms provably dominate the embarrassingly parallelizable alternatives, i.e., *mini-batch* variants of the optimal sequential algorithms [24, 25, 26] (a.k.a. centralized algorithms).

Until very recently, the situation was similar even in the simpler convex *homogeneous* setting where F_m's are all identical and convex, and Woodworth et al. [2] showed that the optimal algorithm often does not require local updates at all. Even when F_m's are not identical, for high levels of

---

[*] Equal contribution. [†] This work was done while BB was a research assistant professor at TTIC.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

heterogeneity, accelerated mini-batch SGD [27] is optimal [28]. Should we expect something similar in the non-convex setting? Or, can we prove that in some regime *local-update* algorithms improve over the naive centralized baselines?

Method (Reference) (**Oracles used**)	Convergence Rate, i.e. $\mathbb{E}\left\|\nabla F(\widehat{x})\right\|^2 \preceq$
Full Participation Setting	
SCAFFOLD†, MB-SGD† [18] (**Stochastic**)	$\frac{\Delta L}{R} + \left(\frac{\sigma^2 \Delta L}{MKR}\right)^{1/2}$
MB-STORM (Thm. C.2, [26]) (**Stochastic**)	$\frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma \Delta L}{MKR}\right)^{2/3}$
Lower Bound (Centralized) (Theorem 2.1)	$\frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma \Delta L}{MKR}\right)^{2/3}$
STEM [20] (**Stochastic**)	$(\Delta L + \sigma^2 + \zeta^2)\left(\frac{1}{R} + \frac{1}{(MKR)^{2/3}}\right)$
BVR-L-SGD* [22] CE-LSGD (Thm. 3.1) (**Stochastic**)	$\frac{\Delta \tau}{R} + \frac{\Delta L}{\sqrt{KR}} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma \Delta L}{MKR}\right)^{2/3}$
CE-LGD (Thm. 3.1) (**Exact**)	$\frac{\Delta \tau}{R} + \frac{\Delta L}{KR}$
Lower Bound (Theorem 3.2)	$\min\left\{\frac{\Delta \tau}{R}, \frac{\zeta^2}{R}\right\} + \frac{\Delta L}{KR} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma \Delta L}{MKR}\right)^{2/3}$
Partial Participation Setting	
MB-STORM (Thm. D.4) (**Stochastic**)	$\frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma \Delta L}{m\sqrt{KR}}\right)^{2/3} + \frac{\zeta^2}{mR} + \left(\frac{\zeta \Delta L}{mR}\right)^{2/3}$
Lower Bound (Centralized) (Theorem D.2)	$\frac{\Delta L}{R} + \frac{\sigma^2}{mKR} + \left(\frac{\sigma \Delta L}{mKR}\right)^{2/3} + \frac{\zeta^2}{mR} + \left(\frac{\zeta \Delta L}{mR}\right)^{2/3}$
MimeLiteMVR[21] (**Stochastic + Exact**)	$\frac{\Delta \tau}{R} + \frac{\Delta L}{KR} + \frac{\zeta^2 + \sigma^2}{R} + \left(\frac{(\zeta + \sigma)\Delta \tau}{R}\right)^{2/3}$
MimeMVR [21] (**Exact**)	$\frac{\Delta \tau}{R} + \frac{\Delta L}{KR} + \frac{\zeta^2}{mR} + \left(\frac{\zeta \Delta \tau}{\sqrt{mR}}\right)^{2/3}$
CE-LSGD (Thm. 3.3) (**Stochastic**)	$\frac{\Delta \tau}{R} + \frac{\Delta L}{\sqrt{KR}} + \frac{\sigma^2}{mKR} + \left(\frac{\sigma \Delta L}{mKR}\right)^{2/3} + \left(\frac{\sigma \Delta \tau}{m\sqrt{KR}}\right)^{2/3} +$ $\frac{\zeta^2}{mR} + \left(\frac{\zeta \Delta \tau}{mR}\right)^{2/3} + \left(\frac{\zeta \Delta L}{m\sqrt{KR}}\right)^{2/3}$
CE-LGD (Thm. 3.3) (**Exact**)	$\frac{\Delta \tau}{R} + \frac{\Delta L}{KR} + \frac{\zeta^2}{mR} + \left(\frac{\zeta \Delta \tau}{mR}\right)^{2/3}$
Lower Bound (Thm. 3.4)	$\min\left\{\frac{\Delta \tau}{R}, \frac{\zeta^2}{R}\right\} + \frac{\Delta L}{KR} + \frac{\sigma^2}{mKR} + \left(\frac{\sigma \Delta L}{mKR}\right)^{2/3} + \frac{\zeta^2}{mR} +$ $\left(\frac{\zeta \Delta L}{mKR}\right)^{2/3}$

Table 1: Comparison of convergence rate for different algorithms in the intermittent communication setting. ζ and τ are the first and second-order heterogeneity (see Section 2) of the problem. Note that $\tau \leq 2L$ can be much smaller than L. *See Section 3 for a detailed comparison with BVR-L-SGD. We expect the red and blue terms in the bounds to match by improving our bounds (c.f., Section 5). †The variance term is optimal as the algorithms' analyses don't assume mean squared smoothness.

In this paper, we start by noting that in the absence of any heterogeneity assumption (c.f., Section 2), centralized algorithms already have the best worst-case convergence guarantee. Thus, only when the heterogeneity is low can the *local-update* algorithms potentially have an advantage. This was the motivation behind some of the recent works [18, 21, 22]. However, in the absence of any lower bound that explicitly depends on the heterogeneity parameter (such as in [15, 29]), it is not possible to definitively claim such an improvement. To alleviate this, we provide new communication lower bounds which explicitly depends on the heterogeneity parameter. In addition, we develop a novel algorithm which can take advantage of low heterogeneity and is (almost) optimal.

Method (Reference)	Communication Complexity (R)	Oracle Complexity (N)
Full Participation Setting		
SCAFFOLD†, MB-SGD† [18]	$\frac{\Delta L}{\epsilon}$	$\frac{\sigma^2 \Delta L}{\epsilon^2}$
MB-STORM (Theorem C.2) [26]	$\frac{\Delta L}{\epsilon}$	$\frac{\sigma \Delta L}{\epsilon^{3/2}}$
Lower Bound (Centralized) (Theorem 2.1)	$\frac{\Delta L}{\epsilon}$	$\frac{\sigma \Delta L}{\epsilon^{3/2}}$
STEM [20]	$\frac{\Delta L+\sigma^2+\zeta^2}{\epsilon}$	$\frac{(\Delta L)^{3/2}+\sigma^3+\zeta^3}{\epsilon^{3/2}}$
BVR-LSGD* [22] CE-LSGD (Theorem 3.1)	$\frac{\Delta \tau}{\epsilon}$	$\frac{\sigma \Delta L}{\epsilon^{3/2}}$
Lower Bound (Theorem 3.2)	$\min\left\{\frac{\Delta \tau}{\epsilon}, \frac{\zeta^2}{\epsilon}\right\}$	$\frac{\sigma \Delta L}{\epsilon^{3/2}}$
Partial Participation Setting		
MB-STORM (Theorem C.2)	$\frac{\zeta \Delta L}{m\epsilon^{3/2}}$	$\frac{\sigma \Delta L}{\epsilon^{3/2}} \cdot \left(1 + \frac{\sigma}{\zeta}\right)$
Lower Bound (Centralized) (Theorem 2.1)	$\frac{\zeta \Delta L}{m\epsilon^{3/2}}$	$\frac{\sigma \Delta L}{\epsilon^{3/2}}$
MimeMVR [21]	$\frac{\zeta \Delta \tau}{m^{1/2}\epsilon^{3/2}}$	**Uses Exact Oracles**
MimeLiteMVR [21]	$\frac{\zeta^2+\sigma^2}{\epsilon} + \frac{(\zeta+\sigma)\Delta \tau}{\epsilon^{3/2}}$	**Uses Exact Oracles**
CE-LSGD (Theorem 3.3)	$\frac{\zeta \Delta \tau}{m\epsilon^{3/2}}$	$\frac{\zeta \Delta L}{\epsilon^{3/2}} \cdot \frac{L}{\tau} + \frac{\sigma \Delta L}{\epsilon^{3/2}} \cdot \left(1 + \frac{\sigma \tau}{\zeta L}\right)$
Lower Bound (Theorem 3.4)	$\min\left\{\frac{\Delta \tau}{\epsilon}, \frac{\zeta^2}{\epsilon}\right\} + \frac{\zeta^2}{m\epsilon}$	$\frac{\zeta \Delta L}{\epsilon^{3/2}} + \frac{\sigma \Delta L}{\epsilon^{3/2}}$

Table 2: Comparison of optimal communication and oracle complexity required by different algorithms to attain $\mathbb{E}\|\nabla F(\widehat{x})\|_2^2 \leq \epsilon$. ζ and τ are the heterogeneity (see Section 2) of the problem. $\tau \leq 2L$ and can be much smaller than L. The results suppress only numerical constants and assume that $\epsilon^{1/2} \preceq \min\{(\sigma/M) \cdot (\tau/L), \Delta L/\sigma, \Delta\tau/\zeta, \zeta/m\}$, i.e., ϵ is small enough. The first inequality ensures we are in the green regime described in Figure 1 and guarantees that $\Delta L M/\epsilon \preceq \sigma \Delta L/\epsilon^{3/2}$; the second inequality guarantees that $\sigma^2/\epsilon \preceq \sigma\Delta L/\epsilon^{3/2}$; the third inequality guarantees that $\zeta^2/m\epsilon \preceq \zeta\Delta\tau/m\epsilon^{3/2}$; and the fourth inequality guarantees that $\Delta L/\epsilon \leq \zeta\Delta L/m\epsilon^{3/2}$. We expect the red, green, and blue terms in the upper and lower bounds to match by improving our bounds (c.f., Section 5). *Although BVR-L-SGD and CE-LSGD have the same fast convergence rate in the full participation setting, BVR-L-SGD requires each client to compute large batch gradients for many rounds of communications and is thus less communication efficient in practice (see discussion in Section 3). †Note that the oracle complexity is optimal for these algorithms, as they were analyzed under the bounded variance assumption (see Section 2).

We summarize the contributions of our work as follows:

- We provide novel communication complexity lower bounds, under the assumption that F_m's have bounded first-order or second-order heterogeneity (see Section 2). We show that centralized algorithms [24, 25, 26] can never achieve this optimal communication complexity, and most of the existing *local-update* algorithms cannot attain it either.
- We develop a new algorithm **CE-LSGD** that we show to be **min-max optimal when equipped with exact gradient oracles** and near-optimal when provided with stochastic gradient oracles (c.f., Section 2). Our algorithm, like many other *local-update* algorithms, uses variance reduction techniques [24, 26] but requires both fewer and lighter "heavy-batch" operations compared to the existing methods (see discussion in Section 3).
- We also study the partial client participation setting, which is of particular interest in cross-device federated learning (FL) [17] where there is an extremely large number of clients. Not only does CE-LSGD improve over the best-known communication complexity, but it is the only algorithm that *doesn't require exact oracle* queries for variance reduction and still manages to be nearly optimal. Our analysis also provides a convergence guarantee for MB-STORM (a special case of CE-LSGD) in this setting that wasn't known before. Furthermore, if **endowed with exact oracles, CE-LGD is almost min-max optimal even in the partial participation setting**. Thus, our results demonstrate the optimality of local update methods, at least in some regimes. Even in simpler convex settings, we don't know of any local update method (exact or stochastic) known to be min-max optimal in the heterogeneous setting [30, 15]. We summarize our results and the comparison to important baselines in Tables 1 and 2.
- As an auxiliary contribution, we provide a variant of our algorithm which uses stochastic hessian vector product oracles and is thus useful for settings where only a single copy of the model can be stored on the edge device. We also empirically compare our method against centralized and *local-update* algorithms, demonstrating faster convergence and better communication efficiency.

Notation. We use $\mathcal{B}$ to denote the index set and $|\mathcal{B}|$ to denote its cardinality. For $x \in \mathbb{R}^d$, we use $\|x\|$ to denote its ℓ_2-norm. For $A \in \mathbb{R}^{d\times d}$, $\|A\|$ denotes the operator norm. $[n]$ denotes the set $\{1, 2, \ldots, n\}$. We use $\approx, \preceq, \succeq$ to denote equality and inequality up to numerical constants.

2 Our Setting and the Centralized Baselines

In this section, we introduce some definitions and assumptions that will be used in our analysis. Our goal is to find an ϵ-approximate stationary point of F, i.e., a point $x \in \mathbb{R}^d$ such that $\mathbb{E}[\|\nabla F(x)\|^2] \leq \epsilon$, where the expectation is w.r.t. any randomness in the choice of x. We consider client objectives in the class $\mathcal{F}(L)$ of differentiable and L-smooth functions, i.e., for all $G \in \mathcal{F}(L)$, $\|\nabla G(x) - \nabla G(y)\| \leq L\|x - y\|$. We also make assumptions that relate the functions of different clients to one another. These are typically known as assumptions on the *"heterogeneity"* of the problem, and we consider two classes of problems.

Definition 1. *Assume* $\{F_m \in \mathcal{F}(L)\}_{m=1}^M$ *are first-order* ζ*-heterogeneous, i.e.,* $\sup_{x\in\mathbb{R}^d} \sum_{m=1}^M \|\nabla F_m(x) - \nabla F(x)\|^2/M \leq \zeta^2$. *And for all* $\Delta \geq 0$, $F(0) - \inf_{x\in\mathbb{R}^d} F(x) \leq \Delta$, *i.e., the average objective has bounded sub-optimality at zero. Then we say that* $\{F_m\}_{m\in[M]} \in \boldsymbol{\mathcal{F}_M^1(L, \Delta, \zeta)}$.

Definition 2. *Assume twice-differentiable* $\{F_m \in \mathcal{F}(L)\}_{m=1}^M$ *are second-order* τ*-heterogeneous, i.e.,* $\sup_{m\in[M], x\in\mathbb{R}^d} \|\nabla^2 F_m(x) - \nabla^2 F(x)\| \leq \tau$. *And for all* $\Delta \geq 0$, $F(0) - \inf_{x\in\mathbb{R}^d} F(x) \leq \Delta$, *i.e., the average objective has bounded sub-optimality at zero. Then we say that* $\{F_m\}_{m\in[M]} \in \boldsymbol{\mathcal{F}_M^2(L, \Delta, \tau)}$.

We assume that each machine has access to the following multi-point oracle [31] [Section 5.3, 2].

Definition 3. *Given a function* $G \in \mathcal{F}(L, \Delta)$, $\mathcal{O}_G^{n,L,\sigma} : (\mathbb{R}^d)^n \times \mathcal{Z} \to (\mathbb{R})^n \times (\mathbb{R}^d)^n$ *is a multi-point stochastic first order oracle if for some distribution* $\mathcal{D}$ *on* $\mathcal{Z}$*, and for all* $x_1, \ldots, x_n \in \mathbb{R}^d$*, the oracle samples a random seed* $z \sim \mathcal{D}$ *and returns estimators* $\mathcal{O}_G^{n,L,\sigma}(x_1, \ldots, x_n, z) = (\{f(x_i; z)\}_{i\in[n]}, \{g(x_i; z)\}_{i\in[n]})$ *such that* $\forall i \in [n]$, $\mathbb{E}_{z\sim\mathcal{D}}(f(x_i; z), g(x_i; z)) = (G(x_i), \nabla G(x_i))$ *and* $\mathbb{E}_{z\sim\mathcal{D}} \|g(x_i; z) - \nabla G(x_i)\|^2 \leq \sigma^2$. *Furthermore, the unbiased gradients satisfy* L*-mean smoothness, i.e., for all* $x, y \in \mathbb{R}^d$, $\mathbb{E}_{z\sim\mathcal{D}} [\|g(x; z) - g(y; z)\|] \leq L\|x - y\|$.

As we mentioned before, we want to solve the problem in equation 1.1 in the the intermittent communication (IC) setting, i.e., M machines work in parallel and are allowed to make K oracle calls between two communication rounds for R consecutive rounds (see [1, 2] for detailed definition). Therefore, we consider a generalization of zero-respecting algorithms denoted by $\mathcal{A}_{ZR}$ (see Appendix A) in the IC setting. This class captures various distributed optimization algorithms, including mini-batch SGD, accelerated mini-batch SGD, local SGD, and all the variance-reduction algorithms. Algorithms that are not distributed zero-respecting are those whose iterates have components in directions about which the algorithm has no information, meaning that, in some sense, it is just "wild guessing". We also denote the class of centralized algorithms in $\mathcal{A}_{ZR}$ by $\mathcal{A}_{ZR}^{cent}$ (see Appendix A). These algorithms query the oracles at the same point within each communication round and use the combined MK oracle queries each round to get a *"mini-batch"* estimate of the gradient. Thus, the class $\mathcal{A}_{ZR}^{cent}$ includes algorithms such as mini-batch SGD, mini-batch SARAH [24] and mini-batch STORM [26], but doesn't include local-update algorithms in $\mathcal{A}_{ZR}$ such as local-SGD. Furthermore, these mini-batch algorithms can be naturally implemented in the IC setting.

We first present a lower bound result applicable to centralized algorithms.

Theorem 2.1 (Centralized Lower Bound)**.** *For all $\tau, \Delta, \zeta, \sigma \geq 0$, and $2L \geq \tau$, every algorithm $A \in \mathcal{A}_{ZR}^{cent}$ optimizing a problem in $\mathcal{F}_M^1(L, \Delta, \zeta) \cup \mathcal{F}_M^2(L, \Delta, \tau)$, with access to an oracle $\mathcal{O}_{F_m}^{2,L,\sigma}$ over $R \succeq 1$ communication rounds must output x_R^A such that,*

$$\mathbb{E}\left[\left\|\nabla F(x_R^A)\right\|^2\right] \succeq \frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma \Delta L}{MKR}\right)^{2/3}.$$

The proof of this theorem follows the known oracle complexity lower bounds [32, 31] and is included in Appendix B. This theorem shows that, mini-batch SARAH/STORM which are centralized algorithms, already achieve the optimal communication and oracle complexity (see Table 1) for algorithms in $\mathcal{A}_{ZR}^{cent}$ optimizing problems in $\mathcal{F}_M^2(L, \Delta, \tau)$. In fact most existing non-centralized methods including FEDAVG[16], SCAFFOLD [18] and FEDPAGE [19] do not have any analysis showing improvement over the centralized baselines for problems in $\mathcal{F}_M^2(L, \Delta, \tau)$. These analyses do not improve with smaller heterogeneity τ, even for convex optimization problems. At the same time, the lower bound result holds for all $\tau \leq 2L$, which highlights the limitation of the centralized baselines, showing they **can not** improve with lower heterogeneity. Certain existing *local-update* algorithms such as MIMEMVR [21] and BVR-L-SGD [22] can indeed improve upon centralized algorithms in the low heterogeneity regime. In the next section, we quantify this improvement and show that our algorithm strictly dominates the centralized baselines and almost matches our lower bound for algorithms in $\mathcal{A}_{ZR}$.

3 Our Algorithm and Min-max Optimality

In this section, we present our communication-efficient algorithm abbreviated CE-LSGD and illustrate it in Algorithm 1. Note that for $m \in [M]$, we use the notation $\nabla F_{m,\mathcal{B}^m}(x) := \sum_{l \in \mathcal{B}^m} g(x; z_l \sim \mathcal{D}_m)/|\mathcal{B}^m|$ to denote the stochastic mini-batch gradient obtained by querying $\mathcal{O}_{F_m}^{2,L,\sigma}$ for $|\mathcal{B}^m|$ many times.

At each iteration of Algorithm 1, we need **two rounds** of communication, i.e., two back and forth communications between the server and all clients. Our method uses the extra round of communication, i.e., line 4 to line 9, to update the variance-reduced gradient v_r using the current and previous server models x_r, x_{r-1}, respectively. In the following discussion, we will use the iteration number R and communication complexity of Algorithm 1 interchangeably.

At the core of our proposed method is the variance reduction term v_r and the local gradient estimator $v_{r,k}^m$ (lines 9 and 15 in Algorithm 1). The construction of the local gradient estimator is motivated by the variance reduction technique of SARAH [24, 25]. Intuitively, the estimation error between the proposed local gradient estimator $v_{r,k}^m$ and the full gradient $\nabla F(w_{r+1,k}^m)$, i.e., $\mathbb{E}\|v_{r,k}^m - \nabla F(w_{r+1,k}^m)\|$, can be decomposed into two dominating terms: $\mathbb{E}\|v_r - \nabla F(x_r)\|^2$ and $\tau^2 K \sum_{k=1}^K \mathbb{E}\|w_{r+1,k}^m - w_{r+1,k-1}^m\|^2$. The first term is the estimation error between the variance reduction term v_r and the full gradient $\nabla F(x_r)$. Since v_r is updated based on the momentum-based

Algorithm 1 Communication Efficient Local Stochastic Gradient Descent (CE-LSGD)

```
input  Initialization $x_0$, iteration number $R$, step size $\eta$, parameters $b_0$, $b$, $T$ and $\beta \in [0,1]$
 1: Let $x_{-1} = x_0$
 2: for $r = 0, 1, \ldots, R-1$ do
 3:    if $r = 0$ set $\rho = 1$, $Q = 1$, $B = b_0$ else set $\rho = \beta$, $Q = T$, $B = Q$
 4:    Communicate (send) $(x_r, x_{r-1})$ to clients
 5:    on client $m \in [M]$ do
 6:       Sample $\mathcal{B}_r^m \sim \mathcal{D}_m^{\otimes B}$, get $\nabla F_{m,\mathcal{B}_r^m}(x_r)$, $\nabla F_{m,\mathcal{B}_r^m}(x_{r-1})$, where $|\mathcal{B}_r^m| = B$
 7:       Communicate (rec) $\left(\nabla F_{m,\mathcal{B}_r^m}(x_r), \nabla F_{m,\mathcal{B}_r^m}(x_{r-1})\right)$ to the server
 8:    end on client
 9:    $v_r = \frac{1}{M}\sum_{m=1}^M \nabla F_{m,\mathcal{B}_r^m}(x_r) + (1-\rho)\left(v_{r-1} - \frac{1}{M}\sum_{m=1}^M \nabla F_{m,\mathcal{B}_r^m}(x_{r-1})\right)$
10:    Communicate (send) $(x_r, v_r)$ to client $\widetilde{m}_r$, where $\widetilde{m}_r \sim Unif\ ([M])$
11:    on client $\widetilde{m}_r$ do
12:       $w_{r+1,1}^{\widetilde{m}_r} := w_{r+1,0}^{\widetilde{m}_r} := x_r, v_{r,0}^{\widetilde{m}_r} := v_r$
13:       for $k = 1, \ldots, Q$ do
14:          Sample $\mathcal{B}_{r,k}^{\widetilde{m}} \sim \mathcal{D}_{\widetilde{m}}^{\otimes b}$, get $\nabla F_{\widetilde{m},\mathcal{B}_{r,k}^{\widetilde{m}}}(w_{r+1,k}^{\widetilde{m}_r})$, $\nabla F_{\widetilde{m},\mathcal{B}_{r,k}^{\widetilde{m}}}(w_{r+1,k-1}^{\widetilde{m}_r})$, where $|\mathcal{B}_{r,k}^{\widetilde{m}}| = b$
15:          $v_{r,k}^{\widetilde{m}_r} = \nabla F_{\widetilde{m},\mathcal{B}_{r,k}^{\widetilde{m}}}(w_{r+1,k}^{\widetilde{m}_r}) + v_{r,k-1}^{\widetilde{m}_r} - \nabla F_{\widetilde{m},\mathcal{B}_{r,k}^{\widetilde{m}}}(w_{r+1,k-1}^{\widetilde{m}_r})$
16:          $w_{r+1,k+1}^{\widetilde{m}_r} = w_{r+1,k}^{\widetilde{m}_r} - \eta v_{r,k}^{\widetilde{m}_r}$
17:       end for
18:       Communicate (rec) $\left(w_{r+1,Q+1}^{\widetilde{m}_r}\right)$ to the server
19:    end on client
20:    Let $x_{r+1} = w_{r+1,Q+1}^{\widetilde{m}_r}$
21: end for
output  Choose $\widetilde{x}$ uniformly from $\{w_{r,k}^{\widetilde{m}_r}\}_{r\in[R],k\in[Q]}$
```

variance reduction technique [26], this estimation error is dominated by $L^2\mathbb{E}\|x_r - x_{r-1}\|^2$, which approaches zero as the algorithm converges. Similarly, the second term $\mathbb{E}\|w_{r+1,k}^m - w_{r+1,k-1}^m\|^2$ approaches zero as the algorithm converges and the τ factor controls the benefit we can obtain from small heterogeneity. Intuitively, we can make more local updates for smaller values of τ, and the algorithm converges faster. Our method reduces to mini-batch STORM if we choose the number of local updates Q to be one (see Appendix C.1).

As we mentioned before, we are considering the IC setting, and thus we want to implement Algorithm 1 in this setting. To this end, we can choose the input $T = K$ and $b = 1$ (see line 14) in Algorithm 1, and we present the convergence guarantees of our method in the IC setting in the following discussions. Next we present the convergence guarantee of CE-LSGD in the intermittent communication:

Theorem 3.1. *Suppose $\{F_m\}_{m\in[M]} \in \mathcal{F}_M^2(L, \Delta, \tau)$ for $L, \Delta, \tau \geq 0, \tau \leq 2L$ then,*

(a) if each client $m \in [M]$ has a stochastic oracle $\mathcal{O}_{F_m}^{2,L,\sigma}$, and assuming $\frac{\Delta L}{R} \preceq \frac{\sigma^2}{\sqrt{MK}}$, then the output $\widetilde{x}$ of Algorithm 1 using $\beta = \max\left\{\frac{1}{R}, \frac{(\Delta L)^{2/3}(MK)^{1/3}}{\sigma^{4/3}R^{2/3}}\right\}$, $b_0 = KR$, and $\eta = \min\left\{\frac{1}{L}, \frac{1}{K\tau}, \frac{(\beta M)^{1/2}}{LK^{1/2}}\right\}$ satisfies

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \preceq \frac{\Delta\tau}{R} + \frac{\Delta L}{\sqrt{KR}} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma\Delta L}{MKR}\right)^{2/3};$$

(b) if each client $m \in [M]$ has a deterministic oracle $\mathcal{O}_{F_m}^{2,L,0}$, then the output $\widetilde{x}$ of Algorithm 1 using $\beta = 1$ and $\eta = \min\left\{\frac{1}{L}, \frac{1}{K\tau}\right\}$ satisfies

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \preceq \frac{\Delta\tau}{R} + \frac{\Delta L}{KR}.$$

In Appendix C, we derive this result by carefully tuning β, b_0. We show that the convergence rate attained by our algorithm is *almost optimal* by proving the following lower bound result.

Theorem 3.2. *For all $L, \sigma, \tau, \Delta, \zeta \geq 0,\ \tau \leq 2L,\ \zeta \leq \sqrt{\Delta L}$, every algorithm $A \in \mathcal{A}_{zr}$, optimizing a problem in $\mathcal{F}_M^1(L, \Delta, \zeta) \cup \mathcal{F}_M^2(L, \Delta, \tau)$ with $K > 0$ intermittent accesses to two-point first-order oracles $\{\mathcal{O}_{F_m}^{2,L,\sigma}\}_{m \in [M]}$ on all the machines, outputs x_R^A after $R \succeq 1$ rounds such that*

$$\mathbb{E}\left[\left\|\nabla F(x_R^A)\right\|^2\right] \succeq \min\left\{\frac{\zeta^2}{R}, \frac{\Delta\tau}{R}\right\} + \frac{\Delta L}{KR} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma\Delta L}{MKR}\right)^{2/3}.$$

We can make two observations by comparing the upper and lower bounds for problems in $\mathcal{F}_M^2(L, \Delta, \tau)$. First, in the deterministic setting ($\sigma = 0$), our upper bound matches the lower bound; **hence CE-LGD is min-max optimal**. Thus, our result improves over all the existing results in this setting, including MIMEMVR [21]. Second, in the stochastic setting ($\sigma > 0$), our algorithm's upper bound is optimal except for the second term in Theorem 3.1, which has a $\Delta L/(\sqrt{KR})$ factor as opposed to the $\Delta L/(KR)$ term in the lower bound. We discuss this gap further in Section C.2.

Our construction for Theorem 3.2 uses the non-convex hard instance proposed by Carmon et al. [32] and splits it across different machines to get a communication complexity lower bound. This idea has been used previously to give lower bounds in the heterogeneous setting [33, 15, 34]. We prove the result in Appendix B. From looking at Table 1, we can note that BVR-L-SGD [22] also attains a similar upper bound as our method. In Appendix C.2, we show that with deterministic oracle BVR-L-SGD also attains the min-max optimal rate. This is not surprising, knowing that several variance-reduced algorithms [25, 26, 24] are simultaneously optimal even in the sequential setting. Still, our method requires fewer and lighter variance reduction operations, which leads to better scalability from the algorithmic design perspective. In the next section, we carefully examine the difference between these methods.

3.1 The Perspective of Reducing Communication

So far, we have looked at convergence rates in the intermittent communication model, where K, R is fixed. However, another perspective is reducing the communication complexity to the minimum possible with the minimum required oracle complexity. Both these complexities can be expressed in terms of ϵ using the convergence guarantees we showed, where we want to attain an ϵ-approximate stationary point. This view is often more useful when communication rounds comprise the bulk of the required physical time. This scenario is common in FL, where devices become available intermittently, which delays the synchronous updates. With this in mind, we summarize the communication and oracle complexities attained by both our method and BVR-L-SGD [22] in Figure 1 when optimizing with stochastic oracles. Notice that the figure has three different regimes based on the relative scaling of τ versus ϵ. We focus on the green regime characterized by $\epsilon^{1/2} \in (0, \tau\sigma/(LM)]$. This regime is of most practical interest in deep learning, where modern over-parameterized models often drive ϵ to really small values. And when ϵ is small enough, this regime covers a wide range of values of τ.[3].

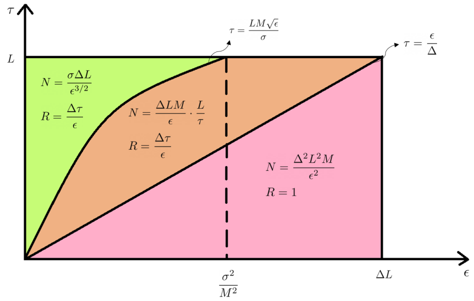

Figure 1: Illustration of the best communication complexity R and oracle complexity N that our method can obtain for different ϵ and τ. Green regime: Our method can obtain the optimal communication and oracle complexity. Orange regime: Our method can obtain the optimal communication using a larger oracle complexity. Red regime: Our method only needs one round of communication using a larger oracle complexity.

In the green regime, both CE-LSGD and BVR-L-SGD require $K = \sigma L/(\tau M \epsilon^{1/2})$ local steps to achieve the optimal communication and oracle complexities. However, BVR-L-SGD requires multiple heavy-batch stochastic gradient computations on each machine with batch size b_{max}. In particular, for BVR-L-SGD, we have $\rho_{\text{BVR}} = b_{max}/K = \sigma\tau/(L\epsilon^{1/2})$, which suggests that for $S = L\Delta/\sigma\sqrt{\epsilon}$ communication rounds, it requires each machine to compute ρ_{BVR} times heavier batch stochastic gradients compared to the other communication rounds. As for CE-LSGD, we have

---

[3] We talk about the other regimes while giving the full statement of Theorem 3.1 in Appendix C

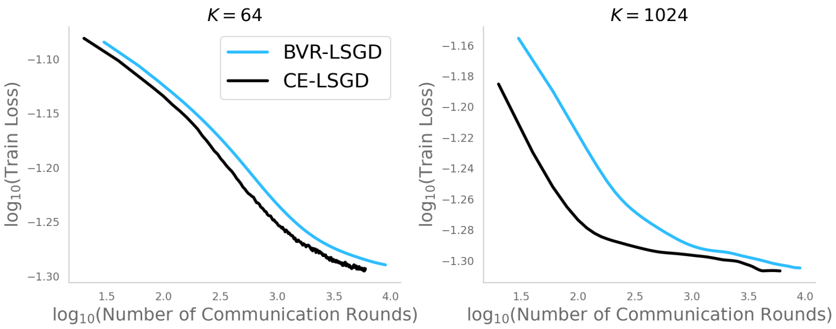

Figure 2: Training loss of CE-LSGD and BVR-L-SGD on CIFAR-10 data-set versus the number of communication rounds in the intermittent communication setting with different local-updates K. We use $M = 10$ machines, and synthetically generate heterogeneous data-sets (see Section 4) with $q = 0.1$. All oracle queries use a mini-batch of size $b = 16$, i.e., each machine has Kb oracle queries between two communication rounds. We note that our method has a faster convergence in all the settings, which highlights its communication efficiency. Fixed step-sizes η for both the methods were tuned in $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5\}$ (to obtain best loss) following [22], our method set the momentum $\beta = 0.3$, $b^{our}_{max} = K$, while $b^{BVR}_{max} = 5000$ according to [22].

$b_0 = \sigma^3/(L\Delta M\epsilon^{1/2})$, which gives us $\rho_{\text{our}} = b_0/K = \sigma^2\tau/(L^2\Delta)$. This suggests that our method only requires each machine to compute ρ_{our} times larger batch stochastic gradient, and that too only once. Furthermore, $\rho_{\text{our}}/\rho_{\text{BVR}} = \sigma\epsilon^{1/2}/(L\Delta) \leq 1$. Thus, the size of our large batch gradient is also smaller than the one for BVR-L-SGD, and **our method has fewer and lighter heavy-batch operations**.

Suppose one implements both these methods in the intermittent communication model, i.e., by breaking the large batch computation across multiple rounds, with local budget $K = \sigma L/(\tau M\epsilon^{1/2})$. In that case, the effective communication complexity of both methods is $\Delta\tau/\epsilon$, and this subtle difference gets washed away. However, in Figure 2, we show that this equivalence up to numerical constants doesn't hold in practice, where our method converges faster than BVR-L-SGD. In Table 2, we summarize the communication and oracle complexities attained by different algorithms in the green regime.

3.2 The Partial Participation Setting

In settings such as cross-device federated learning [17], there are often millions of clients (think of android mobile users), and it is not feasible to consider training on all of the clients synchronously. It is more natural to consider a partial sampling of clients for each communication round. More formally, we can re-state our distributed optimization problem as follows:

$$\min_{x\in\mathbb{R}^d} F(x) := \mathbb{E}_{m\sim\mathcal{P}}\left[F_m(x)\right], \tag{3.1}$$

where $\mathcal{P}$ is a probability distribution on the clients, we assume at each communication round, we can sample M clients independently from $\mathcal{P}$. We also need to modify the IC setting: during each communication round, $S_r \sim \mathcal{P}^m$ clients participate, and each queries their oracle K times. This setting has also been considered in Karimireddy et al. [21]. We consider the problem classes $\mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta)$ and $\mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)$ that are natural generalizations of $\mathcal{F}^1_M(L, \Delta, \zeta)$ and $\mathcal{F}^2_M(L, \Delta, \tau)$ to the partial participation setting as follows, formally defined in Appendix A.

We adapt Algorithm 1 to the partial participation setting in Algorithm 2 by communicating with only M clients at each round and using M_0 clients for the first round to initialize the variance reduction term. We prove the following guarantee for Algorithm 2.

Theorem 3.3. *Suppose for all* m *in support of* $\mathcal{P}$, $F_m \in \mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta) \cap \mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)$ *then,*

(a) if each client m *has a stochastic oracle* $\mathcal{O}^{2,L,\sigma}_{F_m}$, *and assuming that* $\frac{\Delta\tau}{R} + \frac{\Delta L}{\sqrt{KR}} \preceq \frac{\sigma^2}{\sqrt{MK}} + \frac{\zeta^2}{\sqrt{M}}$, *the output* $\widetilde{x}$ *of Algorithm 2 using* $b_0 = K$, $M_0 = MR$, $\beta = \max\left\{\frac{1}{R}, \left(\frac{\Delta(\tau+L/\sqrt{K})\sqrt{M}}{R(\sigma^2/K+\zeta^2)}\right)^{2/3}\right\}$,

and $\eta = \min\left\{\frac{1}{L}, \frac{1}{K\tau}, \frac{1}{\sqrt{KL}}, \frac{\sqrt{\beta M}}{\sqrt{KL}}, \frac{\sqrt{\beta M}}{\tau K}\right\}$ *satisfies*

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \preceq \frac{\Delta\tau}{R}+\frac{\Delta L}{\sqrt{K}R}+\frac{\sigma^2}{MKR}+\left(\frac{\sigma\Delta L}{MKR}\right)^{2/3}+\frac{\zeta^2}{MR}+\left(\frac{\zeta\Delta\tau}{MR}\right)^{2/3}+\left(\frac{\Delta(\sigma\tau+L\zeta)}{M\sqrt{K}R}\right)^{2/3};$$

(b) if each client m *has a deterministic oracle* $\mathcal{O}^{2,L,0}_{F_m}$*, and assuming that* $\frac{\Delta\tau}{R} \preceq \frac{\zeta^2}{\sqrt{M}}$*, then the output* $\widetilde{x}$ *of Algorithm 2 using* $M_0 = MR$*,* $\beta = \max\left\{\frac{1}{R}, \left(\frac{\Delta\tau\sqrt{M}}{\zeta^2 R}\right)^{2/3}\right\}$*, and* $\eta = \min\left\{\frac{1}{L}, \frac{1}{K\tau}, \frac{\sqrt{\beta M}}{\tau K}\right\}$ *satisfies*

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \preceq \frac{\Delta\tau}{R} + \frac{\Delta L}{KR} + \frac{\zeta^2}{MR} + \left(\frac{\zeta\Delta\tau}{MR}\right)^{2/3}.$$

In Tables 1 and 2, we show that with an exact oracle (i.e., $\sigma = 0$), CE-LGD attains a strictly faster convergence rate than the best-known algorithm MIMEMVR [21] that also uses an exact oracle. More specifically, CE-LGD's communication complexity $\zeta\Delta\tau/M\epsilon^{3/2}$, improves over the communication complexity of $\zeta\Delta\tau/\sqrt{M}\epsilon^{3/2}$ for MIME-MVR. We can also recover the guarantee for MB-STORM in the partial participation setting, noting that it is a special case of CE-LSGD (see Appendix C.1). As far as we know, this guarantee isn't known in the literature but straightforwardly follows from our analysis. Furthermore, we prove the following lower bounds showing that the convergence rates of CE-LSGD are *almost optimal*.

Theorem 3.4. *For all* $L, \sigma, \tau, \Delta, \zeta \geq 0,\ \tau \leq 2L,\ \zeta \leq \sqrt{\Delta L}$*, every algorithm* $A \in \mathcal{A}_{zr}$ *optimizing a problem in* $\mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta) \cup \mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)$ *with* $K > 0$ *intermittent accesses to two-point first-order oracles* $\{\mathcal{O}^{2,L,\sigma}_{F_m}\}_{m \in \textit{support}(\mathcal{P})}$ *on all the machines outputs* x^A_R *after* $R \succeq 1$ *rounds such that*

$$\mathbb{E}\left[\left\|\nabla F(x^A_R)\right\|^2\right] \succeq \min\left\{\frac{\Delta\tau}{R}, \frac{\zeta^2}{R}\right\} + \frac{\Delta L}{KR} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma\Delta L}{MKR}\right)^{2/3} + \frac{\zeta^2}{MR} + \left(\frac{\zeta\Delta L}{MKR}\right)^{2/3}.$$

According to Theorem 3.3 and Theorem 3.4, in the deterministic setting (i.e., $\sigma = 0$), the only gap between the rate for CE-LGD and the lower bound is in the last term of CE-LGD's upper bound, i.e., the blue term in Table 1. We **conjecture that CE-LGD is optimal in the partial participation setting, and our lower bound can be improved**. This would also imply a gap between the optimal communication complexity of the full and partial participation settings ($\mathcal{O}(1/\epsilon)$ v/s $\mathcal{O}(1/\epsilon^{3/2})$, see Table 2). All of the known results with our partial participation setting [21] attain at best order $1/\epsilon^{3/2}$ communication complexity, which is consistent with our conjecture. More discussions about the gaps in this setting can be found in Appendix D.1.

4 Simulations

We evaluate the performance of our method by optimizing a two-layer fully connected network for multi-class classification on the CIFAR-10 [35] data-set. Since we are in the heterogeneous setting, we need to artificially generate a data-set. We follow the same data processing procedure as in [22]. We first make sure that all the ten classes in CIFAR-10 have the same number of samples (roughly around 5000), and assign $q \times 100\%$ of class m's samples to client $m \in [10]$ where q is chosen from $\{0.1, 0.35, 0.6, 0.85\}$. For each class m, we evenly split the remaining $(1-q) \times 100\%$ samples to the other 9 clients except client m. Thus, q controls the heterogeneity of our data-set, with small q corresponding to small heterogeneity.

We perform two different experiments. In the first experiment, we directly compare our method, i.e., CE-LSGD, with BVR-L-SGD in the intermittent communication setting (see Figure 2). We observe that while both the methods converge to a similar quality of solution eventually, our method is more communication efficient. In the second experiment, we compare our method with BVR-L-SGD [22] as well as FEDAVG [16], SCAFFOLD [18], MB-SARAH [24] and MB-SGD [5] for the same number of updates/iterations. The last two methods are centralized baselines, and we use the local computation to compute a mini-batch stochastic gradient. We again observe that CE-LSGD and BVR-L-SGD have comparable performance which is better than all the other methods.

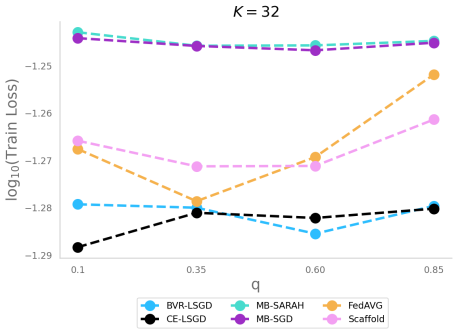

Figure 3: Comparing CE-LSGD to centralized and local-update methods, for fixed $K = 32$ and varying heterogeneity controlled by q on CIFAR-10 [35] data-set. Like Figure 2 we use mini-batch size $b = 16$ for each oracle query. Thus each method makes Kb oracle queries every round per machine. All the methods for different q are tuned separately, following a similar hyper-parameter search as in Figure 2.

5 Discussion and Open Problems

In this paper, we provide a new communication-efficient local update algorithm CE-LSGD and analyze it in the full and partial client participation settings with intermittent communication. In the deterministic setting, i.e., with access to exact oracles, our algorithm is optimal for the full participation setting and almost optimal for the partial participation setting. Moreover, when equipped with stochastic oracles, our algorithm attains the best-known convergence guarantees to our knowledge in both participation models. Our lower bound results provide a much-needed baseline to measure algorithmic developments in non-convex distributed optimization and help us characterize CE-LGD's optimality.

In Appendix E, we provide an extension of CE-LSGD which uses a stochastic Hessian vector product oracle [12, 36] instead of a multi-point oracle, and recovers similar optimal communication complexity. This is relevant for memory-constrained online settings where it might not be feasible to preserve several copies of a model on the client device for making simultaneous queries for variance reduction algorithms.

Our work leaves several open questions. We believe our lower bound is loose in the deterministic partial participation setting. We expect a $\zeta\Delta\tau/M\epsilon^{3/2}$ term in the lower bound, just like our upper bound in Theorem 3.3 (c.f., the blue terms in Tables 1 and 2). Thus, we conjecture that there is a gap between the optimal communication complexities in the full and partial participation settings, order $1/\epsilon$ versus $1/\epsilon^{3/2}$. We hope to improve our lower bound in the future work.

We expect that CE-LSGD should attain the min-max optimal rate in the stochastic full participation setting. There is a $1/\sqrt{K}$ gap in our optimization term for both participation models, which vanishes in the deterministic setting (see Table 1). As discussed in Section C.2, it is unclear to us how to remove this gap.

There are several gaps w.r.t. the lower bounds in the stochastic partial participation setting (c.f., the blue, green, and red terms in Table 2). We believe some of these can be alleviated by improving the deterministic lower bound, but others seem to imply that our analysis is loose. As we discussed before, one indication that our upper bound is loose is the gap in the rate we obtain for MB-STORM by adapting our analysis for Theorem 3.3 (c.f., the red term in Table 1, section D.1).

Acknowledgements

We thank the anonymous reviewers who helped us improve the writing of the paper. We would also like to thank Mladen Kolar, Boxiang Lyu, Boxin Zhao, and Sebastian Stich for several useful discussions. This research was supported in part by NSF-BSF award 1718970, NSF TRIPOD IDEAL award and the NSF-Simons funded Collaboration on the Theoretical Foundations of Deep Learning.

References

[1] Blake E Woodworth, Jialei Wang, Adam Smith, Brendan McMahan, and Nati Srebro. Graph oracle models, lower bounds, and gaps for parallel stochastic optimization. *Advances in neural information processing systems*, 31, 2018.

[2] Blake E Woodworth, Brian Bullins, Ohad Shamir, and Nathan Srebro. The min-max complexity of distributed stochastic convex optimization with intermittent communication. In *Conference on Learning Theory*, pages 4386–4437. PMLR, 2021.

[3] Martin Zinkevich, Markus Weimer, Lihong Li, and Alex Smola. Parallelized stochastic gradient descent. *Advances in neural information processing systems*, 23, 2010.

[4] Andrew Cotter, Ohad Shamir, Nati Srebro, and Karthik Sridharan. Better mini-batch algorithms via accelerated gradient methods. *Advances in neural information processing systems*, 24, 2011.

[5] Ofer Dekel, Ran Gilad-Bachrach, Ohad Shamir, and Lin Xiao. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13(1), 2012.

[6] Yuchen Zhang, John Duchi, Michael I Jordan, and Martin J Wainwright. Information-theoretic lower bounds for distributed statistical estimation with communication constraints. *Advances in Neural Information Processing Systems*, 26, 2013.

[7] Yuchen Zhang, John Duchi, and Martin Wainwright. Divide and conquer kernel ridge regression: A distributed algorithm with minimax optimal rates. *The Journal of Machine Learning Research*, 16(1):3299–3340, 2015.

[8] Ohad Shamir, Nati Srebro, and Tong Zhang. Communication-efficient distributed optimization using an approximate newton-type method. In *International conference on machine learning*, pages 1000–1008. PMLR, 2014.

[9] Sebastian U Stich. Local sgd converges fast and communicates little. *arXiv preprint arXiv:1805.09767*, 2018.

[10] Aymeric Dieuleveut and Kumar Kshitij Patel. Communication trade-offs for local-sgd with large step size. *Advances in Neural Information Processing Systems*, 32, 2019.

[11] Blake Woodworth, Kumar Kshitij Patel, Sebastian Stich, Zhen Dai, Brian Bullins, Brendan Mcmahan, Ohad Shamir, and Nathan Srebro. Is local sgd better than minibatch sgd? In *International Conference on Machine Learning*, pages 10334–10343. PMLR, 2020.

[12] Brian Bullins, Kshitij Patel, Ohad Shamir, Nathan Srebro, and Blake E Woodworth. A stochastic newton algorithm for distributed convex optimization. *Advances in Neural Information Processing Systems*, 34, 2021.

[13] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized sgd with changing topology and local updates. In *International Conference on Machine Learning*, pages 5381–5393. PMLR, 2020.

[14] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local sgd on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pages 4519–4529. PMLR, 2020.

[15] Blake E Woodworth, Kumar Kshitij Patel, and Nati Srebro. Minibatch vs local sgd for heterogeneous distributed learning. *Advances in Neural Information Processing Systems*, 33:6281–6292, 2020.

[16] H Brendan McMahan, Eider Moore, Daniel Ramage, S Hampson, and B Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data (2016). *arXiv preprint arXiv:1602.05629*, 2016.

[17] 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. corr. *arXiv preprint arXiv:1912.04977*, 2019.

[18] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020.

[19] Haoyu Zhao, Zhize Li, and Peter Richtárik. Fedpage: A fast local stochastic gradient method for communication-efficient federated learning. *arXiv preprint arXiv:2108.04755*, 2021.

[20] Prashant Khanduri, Pranay Sharma, Haibo Yang, Mingyi Hong, Jia Liu, Ketan Rajawat, and Pramod Varshney. Stem: A stochastic two-sided momentum algorithm achieving near-optimal sample and communication complexities for federated learning. *Advances in Neural Information Processing Systems*, 34, 2021.

[21] Sai Praneeth Karimireddy, Martin Jaggi, Satyen Kale, Mehryar Mohri, Sashank J Reddi, Sebastian U Stich, and Ananda Theertha Suresh. Mime: Mimicking centralized stochastic algorithms in federated learning. *arXiv preprint arXiv:2008.03606*, 2020.

[22] Tomoya Murata and Taiji Suzuki. Bias-variance reduced local sgd for less heterogeneous federated learning. *arXiv preprint arXiv:2102.03198*, 2021.

[23] Rudrajit Das, Anish Acharya, Abolfazl Hashemi, Sujay Sanghavi, Inderjit S Dhillon, and Ufuk Topcu. Faster non-convex federated learning via global and local momentum. *arXiv preprint arXiv:2012.04061*, 2020.

[24] Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *International Conference on Machine Learning*, pages 2613–2621. PMLR, 2017.

[25] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. *Advances in Neural Information Processing Systems*, 31, 2018.

[26] Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex sgd. *Advances in neural information processing systems*, 32, 2019.

[27] Saeed Ghadimi and Guanghui Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework. *SIAM Journal on Optimization*, 22(4):1469–1492, 2012.

[28] Blake Woodworth. The minimax complexity of distributed optimization. *arXiv preprint arXiv:2109.00534*, 2021.

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

[30] Jianyu Wang, Rudrajit Das, Gauri Joshi, Satyen Kale, Zheng Xu, and Tong Zhang. On the unreasonable effectiveness of federated averaging with heterogeneous data. *arXiv preprint arXiv:2206.04723*, 2022.

[31] Yossi Arjevani, Yair Carmon, John C Duchi, Dylan J Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *arXiv preprint arXiv:1912.02365*, 2019.

[32] Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points i. *Mathematical Programming*, 184(1):71–120, 2020.

[33] Yossi Arjevani and Ohad Shamir. Communication complexity of distributed convex learning and optimization. *Advances in neural information processing systems*, 28, 2015.

[34] Xinwei Zhang, Mingyi Hong, Sairaj Dhople, Wotao Yin, and Yang Liu. Fedpd: A federated learning framework with optimal rates and adaptivity to non-iid data. *arXiv preprint arXiv:2005.11418*, 2020.

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

[36] Yossi Arjevani, Yair Carmon, John C Duchi, Dylan J Foster, Ayush Sekhari, and Karthik Sridharan. Second-order information in non-convex stochastic optimization: Power and limitations. In *Conference on Learning Theory*, pages 242–299. PMLR, 2020.

[37] Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2018.

[38] Arkadi Nemirovski. Efficient methods in convex programming. *Lecture notes*, 1994.

[39] 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.

[40] Christopher J Shallue, Jaehoon Lee, Joseph Antognini, Jascha Sohl-Dickstein, Roy Frostig, and George E Dahl. Measuring the effects of data parallelism on neural network training. *arXiv preprint arXiv:1811.03600*, 2018.

Checklist

1. For all authors...
    (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
    (b) Did you describe the limitations of your work? [Yes]
    (c) Did you discuss any potential negative societal impacts of your work? [N/A]
    (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
    (a) Did you state the full set of assumptions of all theoretical results? [Yes]
    (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
    (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
    (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
    (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
    (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] In the appendix
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
    (a) If your work uses existing assets, did you cite the creators? [Yes]
    (b) Did you mention the license of the assets? [N/A]
    (c) Did you include any new assets either in the supplemental material or as a URL? [No]
    (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
    (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
    (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
    (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
    (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Additional Definitions

In this section, we present more detailed definitions of our oracle and algorithm classes.

Oracle Class

The oracle framework is a very common abstraction in optimization literature [37, 38, 1], and an oracle call can be seen as a unit of information and/or computation. This is especially useful when providing lower-bound results. Stochastic optimization often considers single-point oracles that return unbiased function and gradient estimators. However, several studies ([31], c.f., Section 5.3, [2]) have shown separations between algorithms that can query the oracle and obtain unbiased gradients just once for each random seed versus multiple times for the same seed. The latter kind is a multi-point stochastic oracle, formally defined as follows.

Definition 4. *Given a function $G \in \mathcal{F}(L)$, $\mathcal{O}_G^{n,L,\sigma} : (\mathbb{R}^d)^n \times \mathcal{Z} \to (\mathbb{R})^n \times (\mathbb{R}^d)^n$ is a multi-point stochastic first order oracle if for some distribution $\mathcal{D}$ on $\mathcal{Z}$, and for all $x_1, \dots, x_n \in \mathbb{R}^d$, the oracle samples a random seed $z \sim \mathcal{D}$ and returns estimators*

$$\mathcal{O}_G^{n,L,\sigma}(x_1, \dots, x_n, z) = \left(\{f(x_i; z)\}_{i \in [n]}, \{g(x_i; z)\}_{i \in [n]}\right),$$

such that $\forall i \in [n]$,

$$\mathbb{E}_{z \sim \mathcal{D}}\left[(f(x_i; z), g(x_i; z))\right] = (G(x_i), \nabla G(x_i)) \text{ and } \mathbb{E}_{z \sim \mathcal{D}}\left[\|g(x_i; z) - \nabla G(x_i)\|^2\right] \leq \sigma^2.$$

Furthermore, the unbiased gradients satisfy L-mean smoothness, i.e., for all $x, y \in \mathbb{R}^d$,

$$\mathbb{E}_{z \sim \mathcal{D}}\left[\|g(x; z) - g(y; z)\|\right] \leq L \|x - y\|.$$

In this paper, we assume each client $m \in [M]$ has access to a two-point stochastic oracle $\boldsymbol{\mathcal{O}}_{\boldsymbol{F_m}}^{\boldsymbol{2,L,\sigma}}$, which is sufficient to implement popular variance-reduced algorithms. All the random seeds are sampled independently across machines and time steps.

Remark. *For empirical risk minimization (ERM), querying multiple times at the same seed is easy. The $z \sim \mathcal{D}$ corresponds to sampling a data point, and one could just use the same data point multiple times. So even though the multi-point oracle is more powerful, in machine learning applications, it is equally practical. [31] prove all their results for an even stronger oracle, called an active oracle (see section 5.2 in their paper), which better exploits the finite sum structure of ERM problems, but we don't consider active oracles in this paper.*

Remark. *The mean(-squared) smoothness property is necessary to obtain a $\mathcal{O}(1/\epsilon^{3/2})$ oracle complexity in the serial optimization ($M = 1$) setting [31] for obtaining an ϵ-stationary point. Usually, different constants are used to demarcate the $\bar{L}$-mean-smoothness from L-smoothness because one is a property of the oracle while the other of the objective [31]. We do not make this demarcation here to make the presentation simpler. In the setting of stochastic optimization, where each machine's objective is defined as $F_m(\cdot) := \mathbb{E}_{z \sim \mathcal{D}_m}[f(\cdot; z)]$ using L-smooth functions $f(\cdot; z)$, these constants are the same, i.e., $\bar{L} = L$. Even though our results apply more broadly, we have the distributed stochastic optimization problem in our minds throughout the paper.*

Remark. *[31] show that if a first-order oracle only satisfies the bounded variance assumption but not the mean(-squared) smoothness assumption, $\Omega(1/\epsilon^2)$ queries must be made to such an oracle to obtain an ϵ-stationary point. Distributed algorithms such as* FedAvg, Scaffold *and* MB-SGD *only assume this weaker oracle, which explains the worse oracle complexity these algorithms attain (c.f., Table 2).*

Algorithm Class

We consider the problem of finding an approximate stationary point in the intermittent communication setting, where M machines work in parallel and are allowed to make K oracle calls during each communication for R consecutive rounds. We refer the reader to [1] for a formal description of this setting in the graph oracle framework. Intermittent communication is motivated by the sizeable gap between the wall-clock time $\mathcal{C}$ required for a single synchronous communication and the time required per unit of computation $\mathcal{T}$, say a single oracle call [16, 17]. For an efficient implementation, typically, we want our local computation budget K to be comparable to $\mathcal{C}/\mathcal{T}$, i.e., we want to increase our computation load per communication to match the time required for a single communication round. We consider a generalization of zero respecting algorithms [32] denoted by $\mathcal{A}_{ZR}$ in the intermittent communication (IC) setting defined as follows.

Definition 5 (Distributed Zero-respecting Algorithms)**.** *Consider M machines in the IC setting, each of which is endowed with an oracle $\mathcal{O}_m : \mathcal{I} \times \mathcal{Z} \to \mathcal{V}$ and a distribution $\mathcal{D}_m$ on $\mathcal{Z}$. Let $I^m_{r,k}$ denote the input to the k^{th} oracle call, leading up to the r^{th} communication round on machine m. An optimization algorithm initialized at 0 is distributed zero-respecting if:*

1. *for all $r \in [R], k \in [K], m \in [M]$, $I^m_{r,k}$ is in*

$$\left\{\bigcup_{l\in[k-1]} \mathrm{supp}\mathcal{O}_{F_m}(I^m_{r,l}; z^m_{r,l} \sim \mathcal{D}_m)\right\} \cup \left\{\bigcup_{n\in[M],s\in[r-1],l\in[K]} \mathrm{supp}\mathcal{O}_{F_n}(I^n_{s,l}; z^n_{s,l} \sim \mathcal{D}_n)\right\},$$

2. *for all $r \in [R], k \in [K], m \in [M]$, $I^m_{r,k}$ is a deterministic function (which is same across all the machines) of*

$$\left\{\bigcup_{l\in[k-1]} \mathcal{O}_{F_m}(I^m_{r,l}; z^m_{r,l} \sim \mathcal{D}_m)\right\} \cup \left\{\bigcup_{n\in[M],s\in[r-1],l\in[K]} \mathcal{O}_{F_n}(I^n_{s,l}; z^n_{s,l} \sim \mathcal{D}_n)\right\},$$

3. *at the r^{th} communication round the machines only communicate vectors in*

$$\left\{\bigcup_{n\in[M],s\in[r],l\in[K]} \mathrm{supp}\mathcal{O}_{F_n}(I^n_{s,l}; z^n_{s,l} \sim \mathcal{D}_n)\right\}.$$

We denote this class of algorithms by $\boldsymbol{\mathcal{A}_{ZR}}$. Furthermore, if all the oracle inputs are the same between two communication rounds, i.e., $I^m_{r,k} = I_r \in \mathcal{I}$ for all $m \in [M], k \in [K], r \in [R]$, then we say that the algorithm is centralized, and denote this class of algorithms by $\boldsymbol{\mathcal{A}^{cent}_{ZR}} \subset \mathcal{A}_{ZR}$.

This class captures a very wide variety of distributed optimization algorithms, including mini-batch SGD [5], accelerated mini-batch SGD [27], local SGD [16], as well as all the variance-reduced algorithms [21, 19, 20]. Algorithms that are not distributed zero-respecting are those whose iterates have components in directions about which the algorithm has no information, meaning that in some sense, it is just "wild guessing". We have also defined the smaller class of centralized algorithms which includes algorithms such as mini-batch SARAH [24] and mini-batch STORM [26].

Additional Definitions for the Partial Participation Setting.

We define $\mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta)$ and $\mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)$ that are natural generalizations of $\mathcal{F}^1_M(L, \Delta, \zeta)$ and $\mathcal{F}^2_M(L, \Delta, \tau)$ to the partial participation setting as follows.

Definition 6. *Consider any $\zeta, \Delta, L \geq 0$. And for all m in the support of $\mathcal{P}$, assume that $F_m \in \mathcal{F}(L)$, $\sup_{x\in\mathbb{R}^d} \mathbb{E}_{n\sim\mathcal{P}} \|\nabla F_n(x) - \nabla F(x)\|^2 \leq \zeta^2$ and $F(0) - \inf_{x\in\mathbb{R}^d} F(x) \leq \Delta$. Then we say that our problem is in $\boldsymbol{\mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta)}$.*

Definition 7. *Consider any $\tau, \Delta, L \geq 0$ and $\tau \leq 2L$. And for all m in the support of $\mathcal{P}$, assume $F_m \in \mathcal{F}(L)$ are twice-differentiable, $\sup_{m\in \text{support}(\mathcal{P}), x\in\mathbb{R}^d} \|\nabla^2 F_m(x) - \nabla^2 F(x)\| \leq \tau$, and $F(0) - \inf_{x\in\mathbb{R}^d} F(x) \leq \Delta$. Then we say that our problem is in $\boldsymbol{\mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)}$.*

B Proof of Lower Bounds

In this section we prove Theorems 2.1, 3.2, 3.4 and D.2. All of these results share the communication complexity terms $\min\{\Delta\tau/\epsilon, \zeta^2/\epsilon\}$. We'd show that any algorithm in $\mathcal{A}_{ZR}$ no-matter whether it uses an exact or stochastic oracle, with or without partial participation, and for any number of oracle queries K between communication rounds must incur these many communication rounds. To do so, we'd use the non-convex hard instance proposed by [32] and split it across different machines similar to [33, 15]. Specifically, we consider the following functions (where we assume for simplicity d is even):

$$F(x) := \frac{F_1(x) + F_2(x)}{2}, \tag{B.1}$$

$$F_1(x) := -\psi(x)\phi(x_1) + \sum_{i=1}^{d/2-1} \left[\psi(-x_{2i})\phi(-x_{2_i+1}) - \psi(x_{2i})\phi(x_{2i+1})\right], \tag{B.2}$$

$$F_2(x) := \sum_{i=1}^{d/2} \left[\psi(-x_{2i-1})\phi(-x_{2i}) - \psi(x_{2i-1})\phi(x_{2i})\right], \tag{B.3}$$

where the component functions $\psi(\cdot)$ and $\phi(\cdot)$ are defined as follows,

$$\psi(t) = \begin{cases} 0, & t \leq 1/2, \\ \exp\left(1 - \frac{1}{(2t-1)^2}\right), & t > 1/2. \end{cases} \quad \text{and} \quad \phi(t) = \sqrt{e} \int_{-\infty}^{t} e^{-\frac{1}{2}\tau^2} d\tau. \tag{B.4}$$

The functions F_1, F_2 have the following interesting property: Let E_k be the span of first k basis vectors, i.e., span$(e_1, \ldots, e_k)$. Note that when $x_k \in E_k$ and k is odd, we have

$$\nabla F_1(x_k) \in E_k \text{ and } \nabla F_2(x_k) \in E_{k+1},$$

while when k is even,

$$\nabla F_1(x_k) \in E_{k+1} \text{ and } \nabla F_2(x_k) \in E_k.$$

In our construction, half the machines will have the function F_1, and the other half will have the function F_2 (assume M is even, we'd see later it only changes the lower bound by a factor of $M - 1/M$). First, we initialize all the M machines at 0 and optimize using any distributed zero-respecting algorithm (see Definition 5). Then, the only way to access the next coordinate is to query the gradient of one of two functions—F_1 if the next coordinate is odd and F_2 if the next coordinate is even. This means that, between two rounds of communication, at least one set of machines can't make any progress, and the other set of machines only learns about at most one new coordinate. Thus, the machines are forced to communicate at least $d - 1$ times to be able to span $\mathbb{R}^d$. More formally, we can prove the following lemma:

Lemma 1. *For any vector $v \in \mathbb{R}^d$, define $\mathrm{supp}(v) = \{i \in [d] : v_i \neq 0\}$. Let x_R be the output of any algorithm $A \in \mathcal{A}_{ZR}$ equipped with oracles $\{\mathcal{O}_{F_m}\}_{m\in[M]}$ on each machine, initialized at 0 and optimizing the problem with F_1 on the first half machines and F_2 on the secocnd half. Then after R rounds of communication,*

$$\mathrm{supp}(x_R) \in E_R.$$

The proof of this lemma is identical to Lemma 9 in [15]. We'd use this observation along with some properties of the hard instance to show our lower bound. In particular, we note the following properties for the function $F(\cdot)$.

Lemma 2 (Lemma 3 in [32])**.** *The function F satisfies the following:*

i. We have $F(0) - \inf_x F(x) \leq \Delta_0 d$, where $\Delta_0 = 12$.

ii. For all $x \in \mathbb{R}^d$, $\|\nabla F(x)\| \leq 23\sqrt{d}$.

iii. For every $p \geq 1$, the p-th order derivatives of F are l_p-Lipschitz continuous, where $l_p \leq \exp\left(\frac{5}{2}p\log p + cp\right)$ for an numerical constant $c < \infty$. In particular $l_1 = 152$ (c.f., Lemma 2.2 in [31]).

Note that these properties imply the following for F (c.f., Lemma 2 in [32].).

Lemma 3. *For all $x \in E_k$, where $k < d$, $\|\nabla F(x)\| \geq 1$.*

In other words, if the model vector x doesn't span $\mathbb{R}^d$, it will be forced to have a large gradient. And our distributed problem structure forces the iterates to lie in E_R after R communication rounds, as highlighted in Lemma 1. Formalizing this idea results in the following communication complexity lower bound:

Theorem B.1 (Communication complexity second-order)**.** *Any algorithm $A \in \mathcal{A}_{zr}$ optimizing a problem in $\mathcal{F}_M^2(L, \Delta, \tau)$, $\forall\, \tau, \Delta \geq 0$, $2L \geq \tau$ and with $K > 0$ intermittent accesses to $\{\mathcal{O}_{F_m}^{n,L,0}\}_{m\in[M]}$ on all the clients needs communication rounds,*

$$R \geq c_1 \cdot \frac{\Delta\tau}{\epsilon}$$

to output x_R^A such that $\mathbb{E}[\left\|\nabla F(x_R^A)\right\|^2] \leq \epsilon$ where $\epsilon < c_2\tau\Delta$ and c_1, c_2 are numerical constants.

Proof. Let Δ_0, l_1 be the numerical constants as in Lemma 2. Given accuracy parameter $0 < \epsilon < \frac{\tau\Delta}{4\Delta_0 l_1}$ we define the following functions defined on $\mathbb{R}^{d+1} \to \mathbb{R}$,

$$F_1^\star(x) := \frac{\tau\lambda^2}{4l_1} F_1\left(\frac{x_{1:d}}{\lambda}\right) + \frac{L}{4}x_{d+1}^2,\ F_2^\star(x) := \frac{\tau\lambda^2}{4l_1} F_2\left(\frac{x_{1:d}}{\lambda}\right) + \frac{L}{4}x_{d+1}^2,$$

where $\lambda := \frac{4l_1}{\tau} \cdot \sqrt{\epsilon}$, and $x_{1:d} \in \mathbb{R}^d$ denotes $x \in \mathbb{R}^{d+1}$ restricted to the first d dimensions. For $M > 2$ we put $F_1^\star$ on the first $\lfloor M/2 \rfloor$ machines, $F_2^\star$ on the next $\lfloor M/2 \rfloor$ machines, and if M is odd we put the zero function on the last machine. This only worsens the result by a factor of $\left(\frac{M-1}{M}\right)^2$ as we'd see below, so we can assume without loss of generality that M is even. We define

$$F^\star(x) := \frac{F_1^\star(x) + F_2^\star(x)}{2} = \frac{\tau\lambda^2}{4l_1} F\left(\frac{x_{1:d}}{\lambda}\right) + \frac{L}{4}x_{d+1}^2$$

as the average objective of M machines. Further choosing $d = \left\lfloor \frac{\tau\Delta}{4\Delta_0 l_1 \epsilon} \right\rfloor \geq 1$ guarantees that (due to Lemma 2),

$$F^\star(0) - \inf_x F^\star(x) = F(0) - \inf_{x\in\mathbb{R}^d} F(x) \leq \frac{\tau\lambda^2\Delta_0}{l_1} \cdot d = \frac{4l_1\epsilon\Delta_0}{\tau} \left\lfloor \frac{\tau\Delta}{4\Delta_0 l_1\epsilon} \right\rfloor \leq \Delta.$$

Also, each of our objectives is L smooth as $\tau \leq L$. The second order heterogeneity of our problem is bounded by τ as for all x,

$$\frac{1}{2}\left\|\nabla^2 F_1^\star(x) - \nabla^2 F_2^\star(x)\right\| = \frac{\tau}{8l_1}\left\|\nabla^2 F_1\left(\frac{x}{\lambda}\right) - \nabla^2 F_2\left(\frac{x}{\lambda}\right)\right\| \leq \tau.$$

Thus, $F_1^\star, F_2^\star$ characterize a distributed optimization problem which satisfies all our assumptions. Now, we initialize our algorithm at 0. Then using Lemma 1 we know that for all $r \in [R]$, the output of the algorithm after r communication rounds, i.e., $x_r \in E_r$. In particular for $r \in [d-1]$ using Lemma 3 this implies that

$$\mathbb{E}\left[\left\|\nabla F^\star(x_r)\right\|^2\right] \geq \left(\frac{\tau\lambda}{4l_1}\right)^2 \geq \epsilon.$$

Thus, if we want to achieve ϵ-stationarity, we need to communicate at least $d-1$ times. In other words,

$$R \geq d - 1 \geq \frac{1}{8\Delta_0 l_1} \cdot \frac{\tau\Delta}{\epsilon}.$$

This concludes the proof of the theorem with $c_1 = \frac{1}{8\Delta_0 l_1}$ and $c_2 = \frac{1}{4\Delta_0 l_1}$. □

Similarly while optimizing problems in $\mathcal{F}_M^1(L, \Delta, \zeta)$ we can get the following communication lower bound.

Theorem B.2 (Communication complexity first-order)**.** *Any algorithm $A \in \mathcal{A}_{zr}$ optimizing a problem in $\mathcal{F}_M^1(L, \Delta, \zeta)$, $\forall\, \tau, \Delta \geq 0$, $\Delta L \geq \zeta$ and with $K > 0$ intermittent accesses to $\{\mathcal{O}_{F_m}^{n,L,0}\}_{m\in[M]}$ on all the clients needs communication rounds,*

$$R \geq c_1 \cdot \frac{\zeta^2}{\epsilon}$$

to output x_R^A such that $\mathbb{E}[\left\|\nabla F(x_R^A)\right\|^2] \leq \epsilon$ where $\epsilon < c_2\zeta^2$ and c_1, c_2 are numerical constants.

Proof. Let Δ_0, l_1 be the numerical constants as in Lemma 2. Given accuracy parameter $0 < \epsilon < \frac{\zeta^2}{\Delta_0 l_1}$ we define the following functions,

$$F_1^\star(x) := \frac{\zeta^2\lambda^2}{\Delta l_1} F_1\left(\frac{x}{\lambda}\right),\ F_2^\star(x) := \frac{\zeta^2\lambda^2}{\Delta l_1} F_2\left(\frac{x}{\lambda}\right),$$

where $\lambda := \frac{\Delta l_1}{\zeta^2} \cdot \sqrt{\epsilon}$. For $M > 2$ we put $F_1^\star$ on the first $\lfloor M/2 \rfloor$ machines, $F_2^\star$ on the next $\lfloor M/2 \rfloor$ machines, and if M is odd we put the zero function on the last machine. This only worsens the result by a factor of $\left(\frac{M-1}{M}\right)^2$ as we'd see below, so we can assume without loss of generality that M is even. We define

$$F^\star(x) := \frac{F_1^\star(x) + F_2^\star(x)}{2}$$

as the average objective of M machines. Further choosing $d = \left\lfloor \frac{e^c\zeta^2}{\Delta_0 l_1 \epsilon} \right\rfloor \geq 1$ guarantees that (due to Lemma 2),

$$F^\star(0) - \inf_x F^\star(x) \leq \frac{\zeta^2\lambda^2\Delta_0}{\Delta l_1} \cdot d = \frac{\Delta l_1 \epsilon \Delta_0}{\zeta^2} \left\lfloor \frac{\zeta^2}{\Delta_0 l_1 \epsilon} \right\rfloor \leq \Delta.$$

Also, each of our objectives is L smooth as $\zeta^2/\Delta \leq L$. The first order heterogeneity of our problem is bounded by ζ^2 as for all x (upto numerical constants),

$$\begin{aligned}
\frac{1}{M}\sum_{m\in[M]} \|\nabla F_m(x) - F(x)\|^2 &= \frac{1}{2}\|\nabla F_1^\star(x) - \nabla F_2^\star(x)\|^2, \\
&= \frac{\epsilon}{2}\left\|\nabla F_1\left(\frac{x}{\lambda}\right) - \nabla F_2\left(\frac{x}{\lambda}\right)\right\|^2, \\
&\leq (23)^2\epsilon d, \\
&= (23)^2\epsilon\left\lfloor\frac{\zeta^2}{\Delta_0 l_1 \epsilon}\right\rfloor, \\
&\leq \frac{(23)^2}{\Delta_0 l_1}\cdot\zeta^2 \leq \zeta^2,
\end{aligned}$$

where the last step follows from noting that $\Delta_0 = 12, l_1 = 152$.

Thus, $F_1^\star, F_2^\star$ characterize a distributed optimization problem in $\mathcal{F}_M^1(L, \Delta, \zeta)$. Now, we initialize our algorithm at 0. Then using Lemma 1 we know that for all $r \in [R]$, the output of the algorithm after r communication rounds, i.e., $x_r \in E_r$. In particular for $r \in [d-1]$ using Lemma 3 this implies that

$$\mathbb{E}\left[\|\nabla F^\star(x_r)\|^2\right] \geq \left(\frac{\zeta^2\lambda}{\Delta l_1}\right)^2 \geq \epsilon.$$

Thus if we want to achieve, ϵ-stationarity we need to communicate at least $d-1$ times. In other words,

$$R \geq d - 1 \geq \frac{1}{2\Delta_0 l_1} \cdot \frac{\zeta^2}{\epsilon}.$$

This concludes the proof of the theorem with $c_1 = \frac{1}{2\Delta_0 l_1}$ and $c_2 = \frac{1}{\Delta_0 l_1}$ □

Note that Theorems B.2 and B.1 imply a non-trivial lower bound even if the clients are allowed infinite oracle accesses between two communication rounds, i.e., $K \to \infty$ in the intermittent communication setting. Next, we combine these results with known first-order oracle complexity lower bounds to get the stated theorem statements. We begin by first re-stating theorem 3.2.

Theorem B.3 (General Lower Bound). *For all $L, \sigma, \Delta \geq 0$, every algorithm $A \in \mathcal{A}_{zr}$, optimizing a problem in $\mathcal{F}_M^1(L, \Delta, \zeta) \cup \mathcal{F}_M^2(L, \Delta, \tau)$ where $\tau/2, \zeta^2/\Delta \leq L$, and with $K > 0$ intermittent accesses to two-point first-order oracles $\{\mathcal{O}_{F_m}^{2,L,\sigma}\}_{m\in[M]}$ on all the machines, outputs x_R^A after $R \geq c_2$ rounds such that,*

$$\mathbb{E}\left[\|\nabla F(x_R^A)\|^2\right] \geq c_1 \cdot \left(\min\left\{\frac{\zeta^2}{R}, \frac{\Delta\tau}{R}\right\} + \frac{\Delta L}{KR} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma\Delta L}{MKR}\right)^{2/3}\right),$$

where c_1, c_2 are numerical constants.

Proof. Note that using Theorems B.2 and B.1 for any problem in $\mathcal{F}_M^1(L\Delta, \zeta) \cup \mathcal{F}_M^2(L, \Delta, \tau)$ we've proven that, the communication complexity is lower bounded by $\min\left\{\frac{\Delta\tau}{\epsilon}, \frac{\zeta^2}{\epsilon}\right\}$ when $\tau/2, \zeta^2/\Delta \leq L$ and $c_2 \cdot \epsilon \leq \cdot \min\{\tau\Delta, \zeta^2\}$ (where $1/c_2$ is the maximum of the numerical constants appearing in B.2 and B.1). This implies the first two terms in the lower bound for $R \geq c_1$.

To get the second term, we put the function F on all the machines and endow the machines with exact oracles, i.e., $\sigma = 0$. Since the oracle is queried at the same input on all the machines, as well as returns the same fixed output, the M machines can be simulated by a single machine. Furthermore, a single query to $\mathcal{O}_F^{2,L,0}$ at two different points $v, w \in \mathbb{R}^d$ is equivalent to querying the oracle $\mathcal{O}_F^{1,L,0}$ two times at v, w. Thus, we can implement any algorithm $A \in \mathcal{A}_{ZR}^{cent}$ which requires K total intermittent accesses to $\mathcal{O}_F^{2,L,0}$ for all $m \in [M]$, by instead considering a single machine with $2K$ intermittent accesses to $\mathcal{O}_F^{1,L,0}$. Due to Carmon et al., we know that the latter problem requires at least $\Delta L/\epsilon$ oracle calls, which implies that our parallel problem requires at least $\Delta L/(K\epsilon)$ communication rounds. This gives the second term.

Finally, due to [31], any zero respecting algorithm optimizing F requires at least $\sigma^2/\epsilon + \sigma\Delta L/\epsilon^{3/2}$ stochastic oracle calls to an active oracle (i.e., an oracle which takes as input both the query point and the random seed, c.f., Section 5.2 in [31]) which is strictly more powerful than $\mathcal{O}_F^{2,L,\sigma}$. Thus, if we put $F_m = F$ on all machines, and give each machine active oracles, then the oracle queries must be lower bounded by $2MKR \geq \sigma^2/\epsilon + \sigma\Delta L/\epsilon^{3/2}$. This in turn proves a lower bound on the queries to the weaker $\mathcal{O}_F^{2,L,\sigma}$ oracles and proves the final two terms.

We choose c_1 as the minimum of the numerical constants coming from Theorems B.2, B.1, [32] and [31]. □

Similarly we can prove Theorem 2.1.

Theorem B.4 (Centralized Lower Bound)**.** *For all $L, \Delta, \sigma \geq 0$, every algorithm $A \in \mathcal{A}_{ZR}^{cent}$ optimizing a problem in $\mathcal{F}_M^1(L, \Delta, \zeta) \cup \mathcal{F}_M^2(L, \Delta, \tau)$ where $\tau/2, \zeta^2/\Delta \leq L$, and with access to an oracle $\mathcal{O}_{F_m}^{2,L,\sigma}$ over $R \geq c_1$ communication rounds must output x_R^A such that*

$$\mathbb{E}\left[\left\|\nabla F(x_R^A)\right\|^2\right] \geq c_2 \cdot \left(\frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma\Delta L}{MKR}\right)^{2/3}\right),$$

where c_1, c_2 are numerical constants.

Proof. The last two oracle complexity terms follow the same way as in Theorem 3.2 due to [31]. We only need to show how to get the higher first term. For this we use the argument in [32]. We put the function F on all the machines and endow the machines with exact oracles, i.e., $\sigma = 0$. Moreover, since this a homogeneous problem, $\tau, \zeta = 0$ for this distributed problem. Furthermore, since the oracle is queried at the same input on all the machines, as well as returns the same fixed output, the M machines can be simulated by a single machine. A single query to $\mathcal{O}_F^{2,L,0}$ at two different points $v, w \in \mathbb{R}^d$ is equivalent to querying the oracle $\mathcal{O}_F^{1,L,0}$ two times at v, w. Thus, we can implement any algorithm $A \in \mathcal{A}_{ZR}^{cent}$ which requires K total intermittent accesses to $\mathcal{O}_F^{2,L,0}$ for all $m \in [M]$, by instead considering a single machine with 2 intermittent accesses to $\mathcal{O}_F^{1,L,0}$. Due to Carmon et al. we know that the latter problem requires at least $\Delta L/\epsilon$ oracle calls, which implies that our parallel problem requires at least $\Delta L/\epsilon$ communication rounds. This gives the first term of the lower bound. □

Finally, for the partial participation case, we need to argue about one additional term. We first re-state the formal result.

Theorem B.5 (Partial participation lower bound)**.** *For all $L, \sigma, \Delta \geq 0$ every algorithm $A \in \mathcal{A}_{zr}$ optimizing a problem in $\mathcal{F}_{\mathcal{P}}^1(L, \Delta, \zeta) \cup \mathcal{F}_{\mathcal{P}}^2(L, \Delta, \tau)$ where $\tau/2, \zeta^2/\Delta \leq L$, and with $K > 0$ intermittent accesses to two-point first-order oracles $\{\mathcal{O}_{F_n}^{2,L,\sigma}\}_{n \in support(\mathcal{P})}$ on all the machines outputs x_R^A after $R \geq c_1$ rounds such that,*

$$\mathbb{E}\left[\left\|\nabla F(x_R^A)\right\|^2\right] \geq c_2 \cdot \left(\min\left\{\frac{\zeta^2}{R}, \frac{\Delta\tau}{R}\right\} + \frac{\Delta L}{KR} + \frac{\sigma^2}{mKR} + \left(\frac{\sigma\Delta L}{mKR}\right)^{2/3} + \frac{\zeta^2}{mR} + \left(\frac{\zeta\Delta L}{mKR}\right)^{2/3}\right),$$

where c_1, c_2 are numerical constants.

Proof. Except for the last two terms, all the other terms follow from the full-participation case lower bound, i.e., Theorem 3.2. To get these terms, we first put an exact two-point oracle on each machine so $\sigma = 0$. Now note that the distributed optimization problem in the partial participation case is just a stochastic optimization problem, where the randomness comes from sampling the machine $n \sim \mathcal{P}$. Moreover, by sampling a machine $n \sim \mathcal{P}$, we can emulate a stochastic gradient oracle, with variance bounded by ζ^2, (due to the first-order heterogeneity condition) and where the stochastic gradients satisfy the mean squared smoothness condition (because the functions on all the machines are L-smooth). Thus for the last term we can simply use the lower bound of [31] for any zero respecting first order algorithm that makes mKR active oracle calls, i.e., the same argument that gave us the variance terms in the lower bound of theorem B.3. And the reason the second last term doesn't have a factor of K, is because we only see mR machines/samples, and through statistical estimation results we know that the sample complexity lower bound (which is stronger than the lower bound for an active oracle) should be $\frac{\zeta^2}{mR}$ (c.f., Lemmas 10, 11 in [31]). □

Finally, prove the centralized lower bound for the partial participation setting in theorem D.2.

Theorem B.6 (Centralized Partial Participation Lower Bound)**.** *For all $L, \Delta, \sigma \geq 0$, every algorithm $A \in \mathcal{A}_{ZR}^{cent}$ optimizing a problem in $\mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta) \cup \mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)$ where $\tau/2, \zeta^2/\Delta \leq L$, and with access to an oracle $\mathcal{O}^{2,L,\sigma}_{F_m}$ over $R \geq c_1$ communication rounds must output x_R^A such that*

$$\mathbb{E}\left[\left\|\nabla F(x_R^A)\right\|^2\right] \geq c_2 \cdot \left(\frac{\Delta L}{R} + \frac{\sigma^2}{mKR} + \left(\frac{\sigma \Delta L}{mKR}\right)^{2/3} + \frac{\zeta^2}{mR} + \left(\frac{\zeta \Delta L}{mR}\right)^{2/3}\right),$$

where c_1, c_2 are numerical constants.

Proof. The first three terms follow from the proof of theorem 2.1. For the last two terms, we consider a similar argument as in the proof of theorem 3.4, i.e., we assume all machines have exact oracles, and hence the only source of randomness is the sampling of machines from the distribution $\mathcal{P}$. The difference with respect to distributed zero respecting algorithms is that centralized algorithms can be simulated by a single query $K = 1$, because they make queries at the same point within a communication round and hence with exact oracles, only a single query is required per machine per communication round. Thus, centralized algorithms can be simulated by mR queries to active oracles with bounded variance ζ^2 and L mean squared smoothness. Thus, the last two terms have a factor of mR as opposed to mKR as in theorem 3.3. This completes the proof. □

C Proof of Theorem 3.1

In this section, we provide the full statement of Theorem 3.1 and its corresponding proofs. More specifically, we choose the input $T = K$ in Algorithm 1 and present the results accordingly.

We first present the full theorem of Theorem 3.1.

Theorem C.1. *Suppose $\{F_m\}_{m \in [M]} \in \mathcal{F}^2_M(L, \Delta, \tau)$ for $L, \Delta, \tau \geq 0$ then,*

(a) if each client $m \in [M]$ has a stochastic oracle $\mathcal{O}^{2,L,\sigma}_{F_m}$, and assuming $\frac{\Delta L}{R} \leq \frac{\sigma^2}{\sqrt{MKb}}$, then the output $\widetilde{x}$ of Algorithm 1 using

$$\beta = \max\left\{\frac{1}{R}, \frac{(\Delta L)^{2/3}(MKb)^{1/3}}{\sigma^{4/3}R^{2/3}}\right\}, b_0 = KR, \eta = c_1 \cdot \min\left\{\frac{1}{L}, \frac{1}{K\tau}, \frac{1}{\sqrt{K}L}, \frac{(\beta MK)^{1/2}}{LK}\right\},$$

satisfies the following

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq c_2 \cdot \left(\frac{\Delta \tau}{R} + \frac{\Delta L}{KR} + \frac{\Delta L}{R\sqrt{Kb}} + \left(\frac{\sigma \Delta L}{MKbR}\right)^{2/3} + \frac{\sigma^2}{MKbR}\right).$$

(b) if each client $m \in [M]$ has a deterministic oracle $\mathcal{O}^{2,L,0}_{F_m}$, then the output $\widetilde{x}$ of Algorithm 1 using $\beta = 1$ and $\eta = \min\left\{\frac{1}{L}, \frac{1}{K\tau}\right\}$ satisfies,

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq c_3 \cdot \left(\frac{\Delta \tau}{R} + \frac{\Delta L}{KR}\right),$$

where c_1, c_2, c_3 are numerical constants.

In addition, if we have $\epsilon^{1/2} \le \sigma\tau/(LM)$, $\epsilon\sigma^2 \le (\Delta L)^2$, and $M\epsilon^{1/2} \le \min\{\sigma, \sigma^3/(L\Delta)\}$, then Algorithm 1 using $K = \sigma L/(M\tau\epsilon^{1/2})$, $b_0 = \sigma^3/(L\Delta M\epsilon^{1/2})$, $\beta = L\epsilon^{1/2}/(\sigma\tau)$ can achieve the ϵ-approximate stationary point with the following communication and gradient complexities

$$R \le c_4\frac{\Delta\tau}{\epsilon} \text{ and } N \le c_5\frac{\Delta L\sigma}{\epsilon^{3/2}},$$

where c_4, c_5 are numerical constants.

Proof of Theorem C.1 and Three Regimes in Figure 1. In the following proof, we assume that each client can use a mini-batch gradient with batch size b, which can give us a more general result. First of all, we will bound the term $\|w^j_{r+1,k} - x_r\|^2$ for each client at local updates. Let's consider the local updates for client j. For $k > 1$, we have

$$\begin{aligned}
\|w^j_{r+1,k} - x_r\|^2 &= \|w^j_{r+1,k-1} - \eta v^j_{r,k-1} - x_r\|^2 \\
&\le \left(1 + \frac{1}{K}\right)\|w^j_{r+1,k-1} - x_r\|^2 + (1+K)\eta^2\|v^j_{r,k-1}\|^2 \\
&\le \left(1 + \frac{1}{K}\right)\|w^j_{r+1,k-1} - x_r\|^2 + 2(1+K)\eta^2\|v^j_{r,k-1} - \nabla F(w^j_{r+1,k-1})\|^2 \\
&\quad + 2(1+K)\eta^2\|\nabla F(w^j_{r+1,k-1})\|^2.
\end{aligned}$$

Therefore, recursively using the above inequality and the fact that $w^j_{r+1,1} = x_r$, we can obtain

$$\begin{aligned}
\|w^j_{r+1,k} - x_r\|^2 &\le 2(1+K)\eta^2\sum_{l=2}^{k}\left(1 + \frac{1}{K}\right)^{k-l}\|v^j_{r,l-1} - \nabla F(w^j_{r+1,l-1})\|^2 \\
&\quad + 2(1+K)\eta^2\sum_{l=2}^{k}\left(1 + \frac{1}{K}\right)^{k-l}\|\nabla F(w^j_{r+1,l-1})\|^2 \\
&\le 2e(1+K)\eta^2\sum_{k=2}^{K}\|v^j_{r,k-1} - \nabla F(w^j_{r+1,k-1})\|^2 + 2e(1+K)\eta^2\sum_{k=2}^{K}\|\nabla F(w^j_{r+1,k-1})\|^2 \\
&= 2e(1+K)\eta^2\sum_{k=1}^{K-1}\|v^j_{r,k} - \nabla F(w^j_{r+1,k})\|^2 + 2e(1+K)\eta^2\sum_{k=1}^{K-1}\|\nabla F(w^j_{r+1,k})\|^2.
\end{aligned} \tag{C.1}$$

Next, we will bound the estimation error between the local gradient estimator and the full gradient $\mathbb{E}\|v^j_{r,k} - \nabla F(w^j_{r+1,k})\|^2$. According to the definition $v^j_{r,k} = \nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k}) + v^j_{r,k-1} - \nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k-1})$, we have

$$\begin{aligned}
&\mathbb{E}\|v^j_{r,k} - \nabla F(w^j_{r+1,k})\|^2 \\
&= \mathbb{E}\big\|\big(v^j_{r,k-1} - \nabla F(w^j_{r+1,k-1})\big) \\
&\qquad + \big(\nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k}) - \nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k-1}) - \nabla F_j(w^j_{r+1,k}) + \nabla F_j(w^j_{r+1,k-1})\big) \\
&\qquad + \big(\nabla F_j(w^j_{r+1,k}) - \nabla F_j(w^j_{r+1,k-1}) + \nabla F(w^j_{r+1,k-1}) - \nabla F(w^j_{r+1,k})\big)\big\|^2 \\
&= \mathbb{E}\big\|\nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k}) - \nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k-1}) - \nabla F_j(w^j_{r+1,k}) + \nabla F_j(w^j_{r+1,k-1})\big\|^2 \\
&\qquad + \mathbb{E}\big\|\big(v^j_{r,k-1} - \nabla F(w^j_{r+1,k-1})\big) \\
&\qquad\qquad + \big(\nabla F_j(w^j_{r+1,k}) - \nabla F_j(w^j_{r+1,k-1}) + \nabla F(w^j_{r+1,k-1}) - \nabla F(w^j_{r+1,k})\big)\big\|^2 \\
&\le \frac{L^2}{b}\mathbb{E}\|w^j_{r+1,k} - w^j_{r+1,k-1}\|^2 + \left(1 + \frac{1}{K}\right)\mathbb{E}\|v^j_{r,k-1} - \nabla F(w^j_{r+1,k-1})\|^2 \\
&\qquad + (1+K)\mathbb{E}\big\|\nabla F_j(w^j_{r+1,k}) - \nabla F_j(w^j_{r+1,k-1}) + \nabla F(w^j_{r+1,k-1}) - \nabla F(w^j_{r+1,k})\big\|^2,
\end{aligned}$$

where the second equality is due to the independence of the random variables, the inequality comes from the fact that the mini-batch gradients consist of b i.i.d. samples, and each client $m \in [M]$ has the stochastic oracle $\mathcal{O}^{2,L,\sigma}_{F_m}$. Therefore, using the above inequality recursively, we can get

$$\begin{aligned}
&\mathbb{E}\|v^j_{r,k} - \nabla F(w^j_{r+1,k})\|^2 \\
&\leq e\mathbb{E}\|v^j_{r,0} - \nabla F(w^j_{r+1,0})\|^2 + \frac{eL^2}{b}\sum_{k=1}^{K}\mathbb{E}\|w^j_{r+1,k} - w^j_{r+1,k-1}\|^2 \\
&\qquad + e(1+K)\sum_{k=1}^{K}\mathbb{E}\big\|\nabla F_j(w^j_{r+1,k}) - \nabla F_j(w^j_{r+1,k-1}) + \nabla F(w^j_{r+1,k-1}) - \nabla F(w^j_{r+1,k})\big\|^2.
\end{aligned} \tag{C.2}$$

Since $\{F_m\}_{m\in[M]} \in \mathcal{F}^2_M(L, \Delta, \tau)$, by the second-order τ-heterogeneity and Lemma 3 in [21], equation C.2 implies that

$$\begin{aligned}
&\mathbb{E}\|v^j_{r,k} - \nabla F(w^j_{r+1,k})\|^2 \\
&\leq e\mathbb{E}\|v^j_{r,0} - \nabla F(w^j_{r+1,0})\|^2 + \bigg(\frac{eL^2}{b} + 8eK\tau^2\bigg)\sum_{k=1}^{K}\mathbb{E}\|w^j_{r+1,k} - w^j_{r+1,k-1}\|^2 \\
&\leq e\mathbb{E}\|v^j_{r,0} - \nabla F(w^j_{r+1,0})\|^2 + 2\eta^2\bigg(\frac{eKL^2}{b} + 8eK^2\tau^2\bigg)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|v^j_{r,k-1} - \nabla F(w^j_{r+1,k-1})\|^2 \\
&\qquad + 2\eta^2\bigg(\frac{eKL^2}{b} + 8eK^2\tau^2\bigg)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w^j_{r+1,k-1})\|^2,
\end{aligned}$$

where the second inequality is due to the updating rule as well as adding and subtracting the term $\nabla F(w^j_{r+1,k-1})$. As a result, if we choose $\eta \leq 1/(CK\tau)$ and $\eta \leq \sqrt{b}/(C'\sqrt{K}L)$, and the fact that $w^j_{r+1,0} = w^j_{r+1,1} = x_r$, we can obtain

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|v^j_{r,k} - \nabla F(w^j_{r+1,k})\|^2 \leq 2e\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + \frac{1}{6K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w^j_{r+1,k})\|^2. \tag{C.3}$$

Given the above results, we are ready to establish the convergence guarantee of Algorithm 1. For client $\widetilde{m}$ sampled at t-th iteration for the local update, we have

$$\begin{aligned}
F(w^{\widetilde{m}}_{r+1,k+1}) &\leq F(w^{\widetilde{m}}_{r+1,k}) + \langle \nabla F(w^{\widetilde{m}}_{r+1,k}), w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k}\rangle + \frac{L}{2}\|w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k}\|^2 \\
&= F(w^{\widetilde{m}}_{r+1,k}) - \eta\langle \nabla F(w^{\widetilde{m}}_{r+1,k}), v^{\widetilde{m}}_{r,k}\rangle + \frac{\eta^2 L}{2}\|v^{\widetilde{m}}_{r,k}\|^2 \\
&= F(w^{\widetilde{m}}_{r+1,k}) - \eta\langle \nabla F(w^{\widetilde{m}}_{r+1,k}), v^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k}) + \nabla F(w^{\widetilde{m}}_{r+1,k})\rangle + \frac{\eta^2 L}{2}\|v^{\widetilde{m}}_{r,k}\|^2 \\
&\leq F(w^{\widetilde{m}}_{r+1,k}) - \eta\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2 - \eta\langle \nabla F(w^{\widetilde{m}}_{r+1,k}), v^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k})\rangle \\
&\qquad + \eta^2 L\|v^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k})\|^2 + \eta^2 L\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2 \\
&\leq F(w^{\widetilde{m}}_{r+1,k}) - \eta\bigg(\frac{3}{4} - \eta L\bigg)\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2 + \eta(1+\eta L)\|v^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k})\|^2 \\
&\leq F(w^{\widetilde{m}}_{r+1,k}) - \frac{\eta}{2}\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2 + \frac{5}{4}\eta\|v^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k})\|^2,
\end{aligned}$$

where the last inequality is due to the fact that $\eta \leq 1/(4L)$. Therefore, we can obtain that

$$\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2 \leq \frac{2}{\eta}\big(F(w^{\widetilde{m}}_{r+1,k}) - F(w^{\widetilde{m}}_{r+1,k+1})\big) + 3\|v^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k})\|^2.$$

Recall that $w_{r+1,1}^{\widetilde{m}} = x_r$ and $w_{r+1,k+1}^{\widetilde{m}} = x_{r+1}$, averaging from $k = 1, \ldots K$, and taking expectation, we can get

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2 \leq \frac{2}{K\eta}\big(\mathbb{E}F(x_r) - \mathbb{E}F(x_{r+1})\big) + \frac{3}{K}\sum_{k=1}^{K}\mathbb{E}\|v_{r,k}^{\widetilde{m}} - \nabla F(w_{r+1,k}^{\widetilde{m}})\|^2. \tag{C.4}$$

Combining equation C.3 and equation C.4, we can obtain

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2 \leq \frac{2}{K\eta}\big(\mathbb{E}F(x_r) - \mathbb{E}F(x_{r+1})\big) + 6e\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + \\ + \frac{1}{2K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2,$$

which implies that

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2 \leq \frac{4}{K\eta}\big(\mathbb{E}F(x_r) - \mathbb{E}F(x_{r+1})\big) + 12e\mathbb{E}\|v_r - \nabla F(x_r)\|^2. \tag{C.5}$$

Averaging equation C.5 from $t = 0, \ldots, R-1$, we can obtain

$$\frac{1}{RK}\sum_{r=0}^{R-1}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2 \leq \frac{4}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_r)\big) + \frac{12e}{R}\sum_{r=0}^{R-1}\mathbb{E}\|v_r - \nabla F(x_r)\|^2,$$

by the definition of $\widetilde{x}$, we have

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq \frac{4}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_R)\big) + \frac{12e}{R}\sum_{r=0}^{R-1}\mathbb{E}\|v_r - \nabla F(x_r)\|^2. \tag{C.6}$$

Next, we consider the estimation error between v_r and $\nabla F(x_r)$. Recall that we have

$$v_r = \frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r) + (1-\beta)\bigg(v_{r-1} - \frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_{r-1})\bigg),$$

thus we obtain that

$$v_r - \nabla F(x_r) = (1-\beta)\big(v_{r-1} - \nabla F(x_{r-1})\big) + \beta\bigg(\frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \nabla F(x_r)\bigg) \\ + (1-\beta)\bigg(\frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_{r-1}) + \nabla F(x_{r-1}) - \nabla F(x_r)\bigg).$$

Therefore, consider the conditional expectation up to r-th iteration, we have

$$\begin{aligned} \mathbb{E}_r\big\|v_r - \nabla F(x_r)\big\|^2 &\leq (1-\beta)^2\mathbb{E}_r\big\|v_{r-1} - \nabla F(x_{r-1})\big\|^2 \\ &\quad + 2\beta^2\mathbb{E}_r\bigg\|\frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{M}\sum_{j=1}^{M}\nabla F_j(x_r)\bigg\|^2 \\ &\quad + 2(1-\beta)^2\frac{L^2}{MKb}\mathbb{E}_r\big\|x_r - x_{r-1}\big\|^2 \\ &\leq (1-\beta)^2\mathbb{E}_r\big\|v_{r-1} - \nabla F(x_{r-1})\big\|^2 + 2\beta^2\frac{\sigma^2}{MKb} \\ &\quad + 2(1-\beta)^2\frac{L^2}{MKb}\mathbb{E}_r\big\|x_r - x_{r-1}\big\|^2, \end{aligned} \tag{C.7}$$

where the first inequality is due to the fact that the mini-batch gradients consists of b i.i.d. samples and each client has the stochastic oracle $\mathcal{O}_{F_m}^{2,L,\sigma}$, and the last inequality is due to the stochastic oracle $\mathcal{O}_{F_m}^{2,L,\sigma}$. Therefore, taking expectations over all iterations for equation C.7, we can get

$$\begin{aligned}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \leq (1-\beta)^2\mathbb{E}\big\|v_{r-1} - \nabla F(x_{r-1})\big\|^2 + 2\beta^2\frac{\sigma^2}{MKb} \\ + 2(1-\beta)^2\frac{L^2}{MKb}\mathbb{E}\big\|x_r - x_{r-1}\big\|^2. \qquad \text{(C.8)}\end{aligned}$$

Furthermore, we have

$$\begin{aligned}&\beta\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \\ &= \sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - (1-\beta)\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \\ &= \sum_{r=1}^{R}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - (1-\beta)\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - \mathbb{E}\big\|v_R - \nabla F(x_R)\big\|^2 \\ &\quad + \mathbb{E}\big\|v_0 - \nabla F(x_0)\big\|^2 \\ &\leq \sum_{r=1}^{R}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - (1-\beta)^2\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - \mathbb{E}\big\|v_R - \nabla F(x_R)\big\|^2 \\ &\quad + \mathbb{E}\big\|v_0 - \nabla F(x_0)\big\|^2 \\ &\leq 2(1-\beta)^2\frac{L^2}{MKb}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1} - x_r\big\|^2 + 2\beta^2 R\frac{\sigma^2}{MKb} + \mathbb{E}\big\|v_0 - \nabla F(x_0)\big\|^2,\end{aligned}$$

where the last inequality is due to equation C.8. Since we have

$$\mathbb{E}\big\|v_0 - \nabla F(x_0)\big\|^2 = \mathbb{E}\bigg\|\frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_0^j}(x_0) - \nabla F(x_0)\bigg\|^2 \leq \frac{\sigma^2}{Mb_0}.$$

Therefore, we have

$$\beta\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \leq \frac{2(1-\beta)^2L^2}{MKb}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1} - x_r\big\|^2 + 2\beta^2 R\frac{\sigma^2}{MKb} + \frac{\sigma^2}{Mb_0}.$$

This implies that that

$$\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \leq \frac{2(1-\beta)^2L^2}{\beta MKbR}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1} - x_r\big\|^2 + 2\beta\frac{\sigma^2}{MKb} + \frac{\sigma^2}{\beta RMb_0}. \qquad \text{(C.9)}$$

In addition, combining equation C.1 and equation C.3, we can get

$$\begin{aligned}\mathbb{E}\|w_{r+1,k}^j - x_r\|^2 &\leq 8e^2K^2\eta^2\mathbb{E}\|v_r - \nabla F(x_r)\|^2 \\ &\quad + \frac{2e(1+K)\eta^2}{6}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^j)\|^2 + 2e(1+K)\eta^2\sum_{k=1}^{K-1}\|\nabla F(w_{r+1,k}^j)\|^2 \\ &\leq 8e^2K^2\eta^2\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + 10eK^2\eta^2\frac{1}{K}\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w_{r+1,k}^j)\|^2. \qquad \text{(C.10)}\end{aligned}$$

Therefore, we have

$$\begin{aligned}\mathbb{E}\|x_{r+1} - x_r\|^2 &= \mathbb{E}\|w_{r+1,k+1}^{\widetilde{m}} - x_r\|^2 \\ &\leq 8e^2K^2\eta^2\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + 10eK^2\eta^2\frac{1}{K}\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2. \qquad \text{(C.11)}\end{aligned}$$

Thus, plugging equation C.11 into equation C.9, we can get

$$\begin{aligned}\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2 &\le \frac{160L^2K^2\eta^2}{\beta MKb}\frac{1}{R}\sum_{r=0}^{R-1}\Big(\mathbb{E}\|v_r-\nabla F(x_r)\|^2+\frac{1}{K}\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2\Big)\\
&\quad + 2\beta\frac{\sigma^2}{MKb}+\frac{\sigma^2}{\beta RMb_0}\\
&\le \frac{1}{24e+1}\frac{1}{R}\sum_{r=0}^{R-1}\Big(\mathbb{E}\|v_r-\nabla F(x_r)\|^2+\frac{1}{K}\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2\Big)\\
&\quad + 2\beta\frac{\sigma^2}{MKb}+\frac{\sigma^2}{\beta RMb_0},\end{aligned}$$

where the last inequality is due to the fact that $\eta \le \sqrt{\beta MKb}/(C''LK)$. Thus, we have

$$\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2 \le \frac{1}{24e}\frac{1}{R}\sum_{r=0}^{R-1}\frac{1}{K}\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2+4\beta\frac{\sigma^2}{MKb}+2\frac{\sigma^2}{\beta RMb_0}. \tag{C.12}$$

Combining equation C.6 and equation C.12, we can obtain

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le \frac{4}{RK\eta}\big(\mathbb{E}F(x_0)-\mathbb{E}F(x_R)\big)+\frac{1}{2}\mathbb{E}\|\nabla F(\widetilde{x})\|^2+48e\beta\frac{\sigma^2}{MKb}+24e\frac{\sigma^2}{\beta RMb_0},$$

which implies

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le \frac{8}{RK\eta}\big(F(x_0)-F(x^*)\big)+96e\beta\frac{\sigma^2}{MKb}+48e\frac{\sigma^2}{\beta RMb_0}. \tag{C.13}$$

Note that we have the following requirements for the stepsize η: $\eta \le 1/(4L)$, $\eta \le 1/(CK\tau)$, $\eta \le \sqrt{b}/(C'\sqrt{K}L)$, $\eta \le \sqrt{\beta MKb}/(C''LK)$. Plugging these requirements, we can get

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le C_1\Big(\frac{\Delta\tau}{R}+\frac{\Delta L}{KR}+\frac{\Delta L}{R\sqrt{Kb}}+\frac{\Delta L}{R\sqrt{\beta MKb}}+\beta\frac{\sigma^2}{MKb}+\frac{\sigma^2}{\beta RMb_0}\Big). \tag{C.14}$$

Therefore, if we choose $b_0 = KR$ and

$$\beta = \max\left\{\frac{1}{R}, \frac{(\Delta L)^{2/3}(MKb)^{1/3}}{\sigma^{4/3}R^{2/3}}\right\} =: \max\{\beta_1,\beta_2\},$$

we can obtain,

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le C_1\Big(\frac{\Delta\tau}{R}+\frac{\Delta L}{KR}+\frac{\Delta L}{R\sqrt{Kb}}+\frac{\Delta L}{R\sqrt{\beta_2 MKb}}+(\beta_1+\beta_2)\frac{\sigma^2}{MKb}+\frac{\sigma^2}{\beta_1 MKR^2}\Big).$$

which simplifies to,

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le C_1\Big(\frac{\Delta\tau}{R}+\frac{\Delta L}{KR}+\frac{\Delta L}{R\sqrt{Kb}}+\Big(\frac{\sigma\Delta L}{MKbR}\Big)^{2/3}+\frac{\sigma^2}{MKbR}\Big). \tag{C.15}$$

Since we need to ensure that $\beta \le 1$, we require the following assumption for $\beta_2 \le 1$ ($R \ge 1$ w.l.o.g.),

$$\frac{\Delta L}{R} \le \frac{\sigma^2}{\sqrt{MKb}}.$$

This concludes the proof of Theorem C.1 (a).

Deterministic case: Note that if each client $m \in [M]$ has a deterministic oracle $\mathcal{O}_{F_m}^{2,L,0}$, we can choose $\beta = 1$, and according to equation C.6, we can obtain

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le \frac{4}{RK\eta}\big(\mathbb{E}F(x_0)-\mathbb{E}F(x_R)\big)+\frac{12e}{R}\sum_{r=0}^{R-1}\mathbb{E}\|v_r-\nabla F(x_r)\|^2, \tag{C.16}$$

where we have the following requirements of stepsize η: $\eta \leq 1/(4L)$, $\eta \leq 1/(CK\tau)$. Furthermore, we have $\mathbf{v}_t = \nabla F(x_r)$, which implies that

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq C_4\bigg(\frac{\Delta\tau}{R} + \frac{\Delta L}{KR}\bigg).$$

This concludes the proof of Theorem C.1 (b).

In the following, we discuss how to obtain the result in Figure 1 when each client $m \in [M]$ has a stochastic oracle $\mathcal{O}_{F_m}^{2,L,\sigma}$. We always assume that $\tau \leq L$ and without loss of the generality, we assume $b = 1$, and ignore all the dependence on constants. According to equation C.14, if we choose β, b_0 such that

$$\beta\frac{\sigma^2}{MKb} \leq \epsilon \text{ and } \frac{\sigma^2}{\beta RM\epsilon} \leq b_0, \tag{C.17}$$

we can obtain

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq C_5\bigg(\frac{\Delta\tau}{R} + \frac{\Delta L}{KR} + \frac{\Delta L}{R\sqrt{Kb}} + \frac{\Delta L}{R\sqrt{\beta MKb}} + \epsilon\bigg). \tag{C.18}$$

Therefore, to achieve $\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq \epsilon$, we need the following communication complexity

$$R = C_3\bigg(\frac{\Delta\tau}{\epsilon} + \frac{\Delta L}{K\epsilon} + \frac{\Delta L}{\epsilon\sqrt{Kb}} + \frac{\Delta L}{\epsilon\sqrt{\beta MKb}}\bigg).$$

Furthermore,the gradient complexity of Algorithm 1 is $N = MbKR + bK + Mb_0$. If we have

$$Mb_0 \leq N, \tag{C.19}$$

we have the following gradient complexity:

$$N = C_4 MbKR = C_4\bigg(\frac{MbK\Delta\tau}{\epsilon} + \frac{Mb\Delta L}{\epsilon} + \frac{M\Delta L\sqrt{Kb}}{\epsilon} + \frac{\Delta L\sqrt{MKb}}{\epsilon\sqrt{\beta}}\bigg).$$

Note that we want to keep the $R = \Delta\tau/\epsilon$ while minimizing N, i.e., to obtain N close to $\Delta L\sigma/\epsilon^{3/2}$. Recall that we have

$$R = \frac{\Delta\tau}{\epsilon} + \frac{\Delta L}{\epsilon\sqrt{K}} + \frac{\Delta L}{\epsilon\sqrt{\beta MK}} \text{ and } N = \frac{MK\Delta\tau}{\epsilon} + \frac{M\Delta L\sqrt{K}}{\epsilon} + \frac{\Delta L\sqrt{MK}}{\epsilon\sqrt{\beta}}.$$

To achieve $R = \Delta\tau/\epsilon$, we need

$$K \geq \max\bigg\{\frac{L^2}{\tau^2}, \frac{L^2}{\beta M\tau^2}\bigg\}. \tag{C.20}$$

Green regime: We want to achieve best of both worlds, i.e., $R = \Delta\tau/\epsilon$ and $N = \Delta L\sigma/\epsilon^{3/2}$. According to N, we need to have

$$K \leq \max\bigg\{\frac{L}{\tau}\cdot\frac{\sigma}{M\epsilon^{1/2}}, \frac{\sigma^2}{M^2\epsilon}, \frac{\sigma^2\beta}{M\epsilon}\bigg\}. \tag{C.21}$$

Therefore, combining equation C.20 and equation C.21, we can obtain

$$\epsilon^{1/2} \leq \frac{\sigma\tau}{LM} \text{ and } \beta \geq \frac{L\epsilon^{1/2}}{\sigma\tau}.$$

In addition, according to equation C.17, we have

$$\beta \leq \frac{\epsilon MK}{\sigma^2} \leq \frac{\epsilon N}{R\sigma^2} = \frac{L\epsilon^{1/2}}{\sigma\tau}.$$

Therefore, we can choose $\beta = L\epsilon^{1/2}/(\sigma\tau)$, and this will lead to

$$K = \frac{\sigma L}{M\tau\sqrt{\epsilon}}.$$

In addition, according to equation C.17 and equation C.19, we have

$$b_0 = \frac{\sigma^3}{\Delta L M \epsilon^{1/2}},$$

and we need

$$\frac{\sigma^3}{\Delta L \epsilon^{1/2}} \leq \frac{\sigma^2}{\epsilon} \leq \frac{\Delta L \sigma}{\epsilon^{3/2}},$$

which will hold if we have $\epsilon\sigma^2 \leq (\Delta L)^2$.
To summarize, if we have $\epsilon^{1/2} \leq \sigma\tau/(LM)$, $L\epsilon^{1/2} \leq \sigma\tau$ ($\epsilon \leq \sigma^2$), and $\epsilon\sigma^2 \leq (\Delta L)^2$, we have

$$R = \frac{\Delta\tau}{\epsilon} \text{ and } N = \frac{\Delta L \sigma}{\epsilon^{3/2}}$$

if we choose $K = \sigma L/(M\tau\epsilon^{1/2}) \geq 1$ ($M\epsilon^{1/2} \leq \sigma$), $b_0 = \sigma^3/(L\Delta M\epsilon^{1/2})$, $\beta = L\epsilon^{1/2}/(\sigma\tau)$ (always less than 1 in this regime). This gives us the green regime in Figure 1.
Orange regime: In this regime, we still want to keep the $R = \Delta\tau/\epsilon$ while minimizing N. Since we have $\epsilon^{1/2} \geq \sigma\tau/(LM)$, we cannot make $N = \Delta\sigma L/\epsilon^{3/2}$. Thus, according to equation C.20, we have

$$N = \frac{ML\Delta}{\epsilon} \cdot \frac{L}{\tau} + \frac{\sqrt{M}L\Delta}{\sqrt{\beta}\epsilon} \cdot \frac{L}{\tau} + \frac{ML\Delta}{\epsilon} \cdot \frac{L}{\tau\beta M}.$$

By choosing $\beta = 1/M$, we can get

$$N = \frac{ML\Delta}{\epsilon} \cdot \frac{L}{\tau}.$$

And we have $K = L^2/\tau^2$. Furthermore, according to equation C.17 and equation C.19, we have

$$\frac{\sigma^2\tau^2}{M^2L^2} \leq \epsilon, b_0 = \frac{\sigma^2}{\Delta\tau}, \frac{M\sigma^2}{\Delta\tau} \leq \frac{ML\Delta}{\epsilon} \cdot \frac{L}{\tau}$$

where the first inequality holds due to $\epsilon^{1/2} \geq \sigma\tau/(LM)$ and the last one holds if we have $\epsilon\sigma^2 \leq (L\Delta)^2$.
To summarize, if we have $\epsilon^{1/2} \geq \sigma\tau/(LM)$ and $\epsilon\sigma^2 \leq (\Delta L)^2$, we have

$$R = \frac{\Delta\tau}{\epsilon} \text{ and } N = \frac{ML\Delta}{\epsilon} \cdot \frac{L}{\tau},$$

if we choose $K = L^2/\tau^2$, $b_0 = \sigma^2/(\Delta\tau)$.
Red region: If we have $\epsilon \geq \Delta\tau$, then we only need $R = 1$, and thus we have $N \geq ML^2\Delta^2/\epsilon^2$. □

C.1 Mini-batch STORM

In this section, we present the convergence guarantee of mini-batch STORM for completeness. More specifically, if we choose the number of local update to be one in Algorithm 1, our method will reduce to mini-batch STORM. As a result, we have the following convergence guarantee.

Theorem C.2. *Suppose* $\{F_m\}_{m\in[M]} \in \mathcal{F}_M^2(L, \Delta, \tau)$ *for* $L, \Delta, \tau \geq 0$ *then, if each client* $m \in [M]$ *has a stochastic oracle* $\mathcal{O}_{F_m}^{2,L,\sigma}$*, then the output* $\widetilde{x}$ *of mini-batch STORM using* $\beta = \frac{(\Delta L)^{2/3}(MK)^{1/3}}{\sigma^{4/3}R^{2/3}} \leq 1$, $b_0 = \min\left\{\frac{\sigma^{4/3}(RK)^{2/3}}{(\Delta L)^{2/3}M^{1/3}}, \frac{\sigma^{8/3}(KR)^{1/3}}{(\Delta L)^{4/3}M^{2/3}}\right\}$*, and* $\eta = \min\left\{\frac{1}{L}, \frac{(\beta M)^{1/2}}{LK^{1/2}}\right\}$ *satisfies*

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq c_1 \cdot \left(\frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\Delta\sigma L}{RMK}\right)^{2/3}\right),$$

where c_1 *is a numerical constant.*

Proof of Theorem C.2. The proof of this result directly follows the proof of Theorem C.1. We can just set $K = 1$, let $\tau = L$, and ignoring the $\Delta L/(R\sqrt{Kb})$ term (which appears when local updates $K > 1$) in equation C.15 to get

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq C_1\left(\frac{\Delta L}{R} + \frac{\sigma^2}{MbR} + \left(\frac{\sigma\Delta L}{MbR}\right)^{2/3}\right)$$

provided that

$$\beta = \frac{(\Delta L)^{2/3}(Mb)^{1/3}}{\sigma^{4/3}R^{2/3}} \leq 1.$$

Finally, if we choose the batch size to be the number of updates in the local update algorithms, i.e., $b = K$, we obtain that

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq C_1\left(\frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\Delta\sigma L}{RMK}\right)^{2/3}\right),$$

and we have

$$\beta = \frac{(\Delta L)^{2/3}(MK)^{1/3}}{\sigma^{4/3}R^{2/3}} \leq 1,\ b_0 \ = \ \min\left\{\frac{\sigma^{4/3}(RK)^{2/3}}{(\Delta L)^{2/3}M^{1/3}}, \frac{\sigma^{8/3}(KR)^{1/3}}{(\Delta L)^{4/3}M^{2/3}}\right\}.$$

Note that C_1, C_2 are numerical constants. □

C.2 The Gap in the Stochastic Setting

According to the results in Table 1, there is a gap between the convergence rates of CE-LSGD and CE-LGD, which doesn't go away when $\sigma = 0$. In particular, the brown term in CE-LGD's upper bound, which doesn't depend on σ, matches the corresponding term in the lower bound, but the brown term in CE-LSGD's upper bound is worse by a factor of $1/\sqrt{K}$. This result comes from a more pessimistic choice of step size in the stochastic setting.

To elucidate this further, consider a more general communication model. Recall that each machine makes K queries in the IC setting between two communication rounds. We can instead consider the model where each machine is allowed to make Kb queries but at most at K different inputs. Centralized algorithms will make just Kb queries at the same input. For instance, in this model, MB-SGD or MB-STORM will make R updates with batch size MKb. However, local update algorithms can make K *"mini-batch"* style queries, i.e., make b repeated queries at the current local iterate. This oracle model has been studied for hierarchical parallelism [39]. For instance, let's say each machine has access to a GPU. Then it is preferable that each local update uses the largest batch size $b = b_{max}$ that saturates the GPU's capacity (such as its memory) without additional parallel run-time when compared to $b = 1$. Modern specialized hardware for deep learning (including FPGAs, TPUs, etc.) is designed with such parallelism, and b_{max} is usually much larger than 1 [40]. Thus, if energy usage (i.e., more oracle queries) is a non-concern and getting to an accurate solution as quickly as possible is most important, then it is useful to consider this hierarchical setting. In this setting, we can attain the following convergence guarantee for CE-LSGD.

Theorem C.3. *Suppose* $\{F_m\}_{m\in[M]} \in \mathcal{F}_M^2(L, \Delta, \tau)$ *for* $L, \Delta, \tau \geq 0, \tau \leq 2L$*, each client* $m \in [M]$ *has a stochastic oracle* $\mathcal{O}_{F_m}^{2,L,\sigma}$ *which it uses through b-calls for every single query, and assume that* $\frac{\Delta L}{R} \leq \frac{\sigma^2}{\sqrt{MKb}}$*. Then the output* $\widetilde{x}$ *of Algorithm 1 using* $\beta = \max\left\{\frac{1}{R}, \frac{(\Delta L)^{2/3}(MKb)^{1/3}}{\sigma^{4/3}R^{2/3}}\right\}$*,* $b_0 = KbR$ *and* $\eta = \min\left\{\frac{1}{L}, \frac{1}{K\tau}, \frac{\sqrt{b}}{\sqrt{K}L}, \frac{(\beta MKb)^{1/2}}{LK}\right\}$*, satisfies the following*

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \preceq \frac{\Delta\tau}{R} + \frac{\Delta L}{KR} + \frac{\Delta L}{R\sqrt{Kb}} + \left(\frac{\sigma\Delta L}{MKbR}\right)^{2/3} + \frac{\sigma^2}{MKbR}.$$

When $b = 1$, this reduces to Theorem 3.1 since the third term in the upper bound always dominates the second term. In the exact setting as we show in Appendix C, the last three terms go away altogether. Using arguments similar to the ones given in Appendix B (to prove Theorem 3.2), we can show that every term except the third term is tight in Theorem C.3. We currently don't know how to get rid of the loose third term, but as apparent from the theorem, setting $b = K$ suffices to recover the min-max optimal guarantee even on the stochastic setting. This gap also appears in the partial participation setting, which we study in the next section.

D Proof of Convergence for Algorithm 2

As we discussed before, we can adapt Algorithm 1 to the partial participation setting, and we detail our method in Algorithm 2. Now, we provide the convergence guarantee of Algorithm 2. Same as before, we choose the input $T = K$ in Algorithm 3.3 and present the results accordingly.

Algorithm 2 CE-LSGD for Partial Participation

```
input Initialization $x_0$, communication round $R$, parameters $b_0, b, T, \beta \in [0,1]$, $M$, and $M_0$
Let $x_{-1} = x_0$
for $r = 0, 1, \ldots, R-1$ do
    if $r = 0$ set $\rho = 1, Q = 1, B = b_0, S = M_0$ else set $\rho = \beta, Q = T, B = Q, S = M$
    Sample a subset $\mathcal{S}_r \sim \mathcal{P}^{\otimes S}$ of $S$ clients
    Communicate (send) $(x_r, x_{r-1})$ to clients $m \in \mathcal{S}_r$
    on client $m \in \mathcal{S}_r$ do
        Compute $\nabla F_{m,\mathcal{B}_r^m}(x_r)$ and $\nabla F_{m,\mathcal{B}_r^m}(x_{r-1})$, where $|\mathcal{B}_r^m| = B$
        Communicate (rec) $\big(\nabla F_{m,\mathcal{B}_r^m}(x_r), \nabla F_{m,\mathcal{B}_r^m}(x_{r-1})\big)$ to the server
    end on client
    $v_r = \frac{1}{|\mathcal{S}_r|}\sum_{m\in\mathcal{S}_r} \nabla F_{m,\mathcal{B}_r^m}(x_r) + (1-\rho)\Big(v_{r-1} - \frac{1}{|\mathcal{S}_r|}\sum_{m\in\mathcal{S}_r} \nabla F_{m,\mathcal{B}_r^m}(x_{r-1})\Big)$
    Communicate (send) $(x_r, v_r)$ to client $\widetilde{m}_r$, where $\widetilde{m}_r \sim \mathcal{P}$
    on client $\widetilde{m}$ do
        $w_{r+1,1}^{\widetilde{m}_r} := w_{r+1,0}^{\widetilde{m}_r} := x_r, v_{r,0}^{\widetilde{m}_r} := v_r$
        for $k = 1, \ldots, Q$ do
            Sample $\mathcal{B}_{r,k}^{\widetilde{m}} \sim \mathcal{D}_{\widetilde{m}}^{\otimes b}$, get $\nabla F_{\widetilde{m},\mathcal{B}_{r,k}^{\widetilde{m}}}(w_{r+1,k}^{\widetilde{m}_r})$, $\nabla F_{\widetilde{m},\mathcal{B}_{r,k}^{\widetilde{m}}}(w_{r+1,k-1}^{\widetilde{m}_r})$, where $|\mathcal{B}_{r,k}^{\widetilde{m}}| = b$
            $v_{r,k}^{\widetilde{m}_r} = \nabla F_{\widetilde{m},\mathcal{B}_{r,k}^{\widetilde{m}}}(w_{r+1,k}^{\widetilde{m}_r}) + v_{r,k-1}^{\widetilde{m}_r} - \nabla F_{\widetilde{m},\mathcal{B}_{r,k}^{\widetilde{m}}}(w_{r+1,k-1}^{\widetilde{m}_r})$
            $w_{r+1,k+1}^{\widetilde{m}_r} = w_{r+1,k}^{\widetilde{m}_r} - \eta v_{r,k}^{\widetilde{m}_r}$
        end for
        Communicate (rec) $\big(w_{r+1,Q+1}^{\widetilde{m}_r}\big)$ to the server
    end on client
    Let $x_{r+1} = w_{r+1,Q+1}^{\widetilde{m}_r}$
end for
output Choose $\widetilde{x}$ uniformly from $\{w_{r,k}^{\widetilde{m}_r}\}_{r\in[R],k\in[Q]}$
```

Proof of Theorem 3.3. The proof of the theorem mainly follows the proof in Theorem C.1. As before, we prove the result of using the mini-batch gradient with batch size b, which is a more general result. More specifically, the proof for local updates will not change, and we can get the following result according to equation C.6

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le \frac{4}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_R)\big) + \frac{12e}{R}\sum_{r=0}^{R-1}\mathbb{E}\|v_r - \nabla F(x_r)\|^2. \tag{D.1}$$

For the variance reduction term v_r, we have

$$v_r = \frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_r) + (1-\beta)\Bigg(v_{r-1} - \frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_{r-1})\Bigg),$$

thus we obtain that

$$\begin{aligned} v_r - \nabla F(x_r) &= (1-\beta)\big(v_{r-1} - \nabla F(x_{r-1})\big) + \beta\Bigg(\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \nabla F(x_r)\Bigg) \\ &\quad + (1-\beta)\Bigg(\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}F_{j,\mathcal{B}_r^j}(x_{r-1}) + \nabla F(x_{r-1}) - \nabla F(x_r)\Bigg). \end{aligned}$$

Therefore, consider the conditional expectation up to r, we have

$$
\begin{aligned}
&\mathbb{E}_r\big\|v_r - \nabla F(x_r)\big\|^2 \\
&\quad \le (1-\beta)^2\mathbb{E}_r\big\|v_{r-1} - \nabla F(x_{r-1})\big\|^2 \\
&\qquad + 2\beta^2\mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \nabla F(x_r)\Bigg\|^2 \\
&\qquad\quad + 2(1-\beta)^2\mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}F_{j,\mathcal{B}_r^j}(x_{r-1}) + \nabla F(x_{r-1}) - \nabla F(x_r)\Bigg\|^2 \\
&\quad \le (1-\beta)^2\mathbb{E}_r\big\|v_{r-1} - \nabla F(x_{r-1})\big\|^2 + 2\beta^2\left(\frac{2\sigma^2}{MKb} + \frac{2\zeta^2}{M}\right) \\
&\qquad\quad + 4(1-\beta)^2\left(\frac{L^2}{MKb} + \frac{\tau^2}{M}\right)\mathbb{E}_r\big\|x_r - x_{r-1}\big\|^2, \qquad\qquad \text{(D.2)}
\end{aligned}
$$

where the last inequality is due to the following results. First of all, we have

$$
\begin{aligned}
\mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \nabla F(x_r)\Bigg\|^2 &= \mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_j(x_r) \\
&\qquad + \frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_j(x_r) - \nabla F(x_r)\Bigg\|^2 \\
&\le 2\mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_j(x_r)\Bigg\|^2 \\
&\qquad + 2\mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_j(x_r) - \nabla F(x_r)\Bigg\|^2 \\
&\le 2\frac{\sigma^2}{MKb} + 2\frac{\zeta^2}{M}, \qquad\qquad \text{(D.3)}
\end{aligned}
$$

where the last inequality is due to the independence between $j \in \mathcal{S}_r$ with $|\mathcal{S}_r| = m$, each client j has the stochastic oracle $\mathcal{O}_{F_j}^{2,L,\sigma}$, and $F_j \in \mathcal{F}_{\mathcal{P}}^1(L,\Delta,\zeta)$ with ζ first-order heterogeneity.

In addition, we have

$$
\begin{aligned}
&\mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\nabla F_{j,\mathcal{B}_r^j}(x_{r-1}) + \nabla F(x_{r-1}) - \nabla F(x_r)\Bigg\|^2 \\
&\quad = \mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\Big(\nabla F_{j,\mathcal{B}_r^j}(x_r) - \nabla F_{j,\mathcal{B}_r^j}(x_{r-1}) + \nabla F_j(x_{r-1}) - \nabla F_j(x_r)\Big)\Bigg\|^2 \\
&\qquad + \mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\Big(\nabla F_j(x_{r-1}) - \nabla F_j(x_r)\Big) - \nabla F(x_{r-1}) + \nabla F(x_r)\Bigg\|^2 \\
&\quad \le \left(\frac{2L^2}{MKb} + \frac{2\tau^2}{M}\right)\mathbb{E}_r\big\|x_r - x_{r-1}\big\|^2,
\end{aligned}
$$

where the first equality is due to the independence of the random variables, and the last inequality comes from the following two derivations:

$$
\begin{aligned}
&\mathbb{E}_r\Bigg\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\Big(\nabla F_{j,\mathcal{B}_r^j}(x_r) - \nabla F_{j,\mathcal{B}_r^j}(x_{r-1}) + \nabla F_j(x_{r-1}) - \nabla F_j(x_r)\Big)\Bigg\|^2 \\
&\quad = \frac{1}{|\mathcal{S}_r|^2}\mathbb{E}_r\sum_{j\in\mathcal{S}_r}\Big\|\nabla F_{j,\mathcal{B}_r^j}(x_r) - \nabla F_{j,\mathcal{B}_r^j}(x_{r-1}) + \nabla F_j(x_{r-1}) - \nabla F_j(x_r)\Big\|^2 \\
&\quad \le \frac{2L^2}{MKb}\mathbb{E}_r\big\|x_r - x_{r-1}\big\|^2,
\end{aligned}
$$

where the equality comes from the independence of each random variable, the inequality is due to the fact Kb samples are i.i.d. and smoothness assumption. On the other hand, we have

$$\begin{aligned}
&\mathbb{E}_r\left\|\frac{1}{|\mathcal{S}_r|}\sum_{j\in\mathcal{S}_r}\Big(\nabla F_j(x_{r-1})-\nabla F_j(x_r)\Big)-\nabla F(x_{r-1})+\nabla F(x_r)\right\|^2\\
&\quad\le\frac{2\tau^2}{M}\mathbb{E}_r\big\|x_r-x_{r-1}\big\|^2,
\end{aligned}$$

where the inequality is due to the independence between $j\in\mathcal{S}_r$, and the second-order τ-heterogeneity. Therefore, taking expectations over all iterations for equation D.2, we can get

$$\begin{aligned}
\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2\le(1-\beta)^2\mathbb{E}\big\|v_{r-1}-\nabla F(x_{r-1})\big\|^2+2\beta^2\bigg(\frac{2\sigma^2}{MKb}+\frac{2\zeta^2}{M}\bigg)\\
+4(1-\beta)^2\bigg(\frac{L^2}{MKb}+\frac{\tau^2}{M}\bigg)\mathbb{E}\big\|x_r-x_{r-1}\big\|^2.
\end{aligned}\tag{D.4}$$

Furthermore, we have

$$\begin{aligned}
&\beta\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2\\
&=\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2-(1-\beta)\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2\\
&=\sum_{r=1}^{R}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2-(1-\beta)\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2-\mathbb{E}\big\|v_R-\nabla F(x_R)\big\|^2\\
&\qquad+\mathbb{E}\big\|v_0-\nabla F(x_0)\big\|^2\\
&\le\sum_{r=1}^{R}\mathbb{E}\big\|v_r-\nabla F(x_R)\big\|^2-(1-\beta)^2\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2-\mathbb{E}\big\|v_r-\nabla F(x_R)\big\|^2\\
&\qquad+\mathbb{E}\big\|v_0-\nabla F(x_0)\big\|^2\\
&\le4(1-\beta)^2\bigg(\frac{L^2}{MKb}+\frac{\tau^2}{M}\bigg)\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1}-x_r\big\|^2+2\beta^2R\bigg(\frac{2\sigma^2}{MKb}+\frac{2\zeta^2}{M}\bigg)+\mathbb{E}\big\|v_0-\nabla F(x_0)\big\|^2,
\end{aligned}$$

where the last inequality is due to equation D.4. Furthermore, according to equation D.3, we have

$$\mathbb{E}\big\|v_0-\nabla F(x_0)\big\|^2=\mathbb{E}\left\|\frac{1}{M_0}\sum_{j\in\mathcal{S}_0}\nabla F_{j,\mathcal{B}_0^j}(x_0)-\nabla F(x_0)\right\|^2\le\frac{2\zeta^2}{M_0}+\frac{2\sigma^2}{M_0b_0}.$$

Therefore, we have

$$\begin{aligned}
\beta\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2\le4(1-\beta)^2\bigg(\frac{L^2}{MKb}+\frac{\tau^2}{M}\bigg)\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1}-x_r\big\|^2\\
+2\beta^2R\bigg(\frac{2\sigma^2}{MKb}+\frac{2\zeta^2}{M}\bigg)+\frac{2\zeta^2}{M_0}+\frac{2\sigma^2}{M_0b_0}.
\end{aligned}$$

This implies that that

$$\begin{aligned}
\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r-\nabla F(x_r)\big\|^2\le4(1-\beta)^2\bigg(\frac{L^2}{\beta MKb}+\frac{\tau^2}{\beta M}\bigg)\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1}-x_r\big\|^2\\
+2\beta\bigg(\frac{2\sigma^2}{MKb}+\frac{2\zeta^2}{M}\bigg)+\frac{2\zeta^2}{\beta RM_0}+\frac{2\sigma^2}{\beta RM_0b_0}.
\end{aligned}\tag{D.5}$$

In addition, according to equation C.11, we have

$$\mathbb{E}\|x_{r+1} - x_r\|^2 \leq 8e^2K^2\eta^2\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + 10eK^2\eta^2\frac{1}{K}\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2. \tag{D.6}$$

Thus, plugging equation D.6 into equation D.5, we can get

$$\begin{aligned}\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 &\leq \frac{160K^2\eta^2}{\beta}\bigg(\frac{L^2}{MKb} + \frac{\tau^2}{M}\bigg)\frac{1}{R}\sum_{r=0}^{R-1}\bigg(\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + \frac{1}{K}\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2\bigg)\\
&\quad + 2\beta\bigg(\frac{2\sigma^2}{MKb} + \frac{2\zeta^2}{M}\bigg) + \frac{2\zeta^2}{\beta RM_0} + \frac{2\sigma^2}{\beta RM_0b_0}\\
&\leq \frac{1}{24e+1}\frac{1}{R}\sum_{r=0}^{R-1}\bigg(\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + \frac{1}{K}\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w^{\widetilde{M}}_{r+1,k})\|^2\bigg)\\
&\quad + 2\beta\bigg(\frac{2\sigma^2}{MKb} + \frac{2\zeta^2}{M}\bigg) + \frac{2\zeta^2}{\beta RM_0} + \frac{2\sigma^2}{\beta RM_0b_0}\end{aligned}$$

where the last inequality is due to the fact that $\eta \leq \sqrt{\beta MKb}/(C''LK)$ and $\eta \leq \sqrt{\beta M}/(C''\tau K)$. Thus, we have

$$\begin{aligned}\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 &\leq \frac{1}{24e}\frac{1}{R}\sum_{r=0}^{R-1}\frac{1}{K}\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2\\
&\quad + 4\beta\bigg(\frac{2\sigma^2}{MKb} + \frac{2\zeta^2}{M}\bigg) + \frac{4\zeta^2}{\beta RM_0} + \frac{4\sigma^2}{\beta RM_0b_0}.\end{aligned} \tag{D.7}$$

Combining equation D.1 and equation D.7, we can obtain

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2 &\leq \frac{4}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_R)\big) + \frac{1}{2}\mathbb{E}\|\nabla F(\widetilde{x})\|^2\\
&\quad + 48e\beta\bigg(\frac{2\sigma^2}{MKb} + \frac{2\zeta^2}{M}\bigg) + 48e\bigg(\frac{\zeta^2}{\beta RM_0} + \frac{\sigma^2}{\beta RM_0b_0}\bigg),\end{aligned}$$

which implies

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2 &\leq \frac{8}{RK\eta}\big(F(x_0) - F(x^*)\big)\\
&\quad + 96e\beta\bigg(\frac{2\sigma^2}{MKb} + \frac{2\zeta^2}{M}\bigg) + 96e\bigg(\frac{\zeta^2}{\beta RM_0} + \frac{\sigma^2}{\beta RM_0b_0}\bigg).\end{aligned} \tag{D.8}$$

Note that we have the following requirement for the stepsize η: $\eta \leq 1/(4L)$, $\eta \leq 1/(CK\tau)$, $\eta \leq \sqrt{b}/(C'\sqrt{K}L)$, $\eta \leq \sqrt{\beta MKb}/(C''LK)$, $\eta \leq \sqrt{\beta M}/(C''\tau K)$. Plugging the requirement of the step-size η, we get (ignoring constants)

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2 &\leq \frac{\Delta\tau}{R} + \frac{\Delta\tau}{R\sqrt{\beta M}} + \frac{\Delta L}{KR} + \frac{\Delta L}{R\sqrt{Kb}} + \frac{\Delta L}{R\sqrt{\beta MKb}} + \beta\frac{\sigma^2}{MKb} + \frac{\sigma^2}{\beta RM_0b_0} + \beta\frac{\zeta^2}{M} + \frac{\zeta^2}{\beta RM_0}\\
&= \frac{\Delta\tau}{R} + \frac{\Delta}{R\sqrt{\beta M}}\bigg(\tau + \frac{L}{\sqrt{Kb}}\bigg) + \frac{\Delta L}{KR} + \frac{\Delta L}{R\sqrt{Kb}} + \beta\frac{\sigma^2}{MKb} + \frac{\sigma^2}{\beta RM_0b_0} + \beta\frac{\zeta^2}{M} + \frac{\zeta^2}{\beta RM_0}\end{aligned}$$

Let $M_0 = MR$ and $b_0 = K$, so that $M_0b_0 = MKR$ (i.e., we can implement the algorithm in the intermittent communication setting) and β be set as follows,

$$\beta = \max\left\{\frac{1}{R}, \left(\frac{\Delta(\tau + L/\sqrt{Kb})\sqrt{M}}{R(\sigma^2/Kb + \zeta^2)}\right)^{2/3}\right\} =: \max\{\beta_1, \beta_2\}.$$

then we have (ignoring numerical constants),

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2 &\leq \frac{\Delta\tau}{R} + \frac{\Delta L}{KR} + \frac{\Delta L}{R\sqrt{Kb}} + \frac{\Delta}{R\sqrt{\beta_2 M}}\bigg(\tau + \frac{L}{\sqrt{Kb}}\bigg) + (\beta_1 + \beta_2)\left(\frac{\sigma^2}{MKb} + \frac{\zeta^2}{m}\right) + \frac{\sigma^2}{\beta_1 MKR^2} + \frac{\zeta^2}{\beta_1 MR^2},\\
&\leq \frac{\Delta\tau}{R} + \frac{\Delta L}{\sqrt{Kb}R} + \frac{\sigma^2}{MKbR} + \left(\frac{\sigma\Delta L}{MKbR}\right)^{2/3} + \frac{\zeta^2}{MR} + \left(\frac{\zeta\Delta\tau}{MR}\right)^{2/3} + \left(\frac{\Delta(\sigma\tau + L\zeta)}{M\sqrt{Kb}R}\right)^{2/3}.\end{aligned}$$

Further note that to get the theorem statement in Theorem 3.3, we need to ensure $\beta_2 \leq 1$ ($R > 1$ w.l.o.g.) which gives the following condition that we state in Theorem 3.3,

$$\frac{\Delta(\tau + L/\sqrt{Kb})\sqrt{M}}{R} \leq \frac{\sigma^2}{Kb} + \zeta^2.$$

Mini-batch STORM: As before, we can get the convergence of mini-batch STORM by setting local update steps to be 1, $\tau = L$, ignoring the $\sqrt{K}$ dependence term, and choosing a mini-batch size of bK compared with the local update algorithm. Therefore, the mini-batch STORM has the following convergence guarantee for partial participation

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq \frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma\Delta L}{M\sqrt{Kb}R}\right)^{2/3} + \frac{\zeta^2}{MR} + \left(\frac{\zeta\Delta L}{MR}\right)^{2/3}.$$

Theorem D.1. *Suppose for all m in support of $\mathcal{P}$, $F_m \in \mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta) \cap \mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)$, if each client m has a stochastic oracle $\mathcal{O}^{2,L,\sigma}_{F_m}$, and assuming that $\frac{\Delta L}{R} \preceq \frac{\sigma^2}{\sqrt{M}K} + \frac{\zeta^2}{\sqrt{M}}$, then the output $\widetilde{x}$ of* MB-STORM *using $b_0 = K$, $M_0 = MR$, $\beta = \max\left\{\frac{1}{R}, \left(\frac{\Delta L\sqrt{M}}{R(\sigma^2/K+\zeta^2)}\right)^{2/3}\right\}$, and $\eta = \frac{1}{KL} \cdot \min\left\{1, \sqrt{\beta M}\right\}$ satisfies*

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \preceq \frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma\Delta L}{M\sqrt{K}R}\right)^{2/3} + \frac{\zeta^2}{MR} + \left(\frac{\zeta\Delta L}{MR}\right)^{2/3}.$$

Furthermore, we prove the following lower bound showing that the convergence rate of MB-STORM is *almost optimal*.

Theorem D.2. *For all $L, \sigma, \tau, \Delta, \zeta \geq 0$, $\tau \leq 2L$, $\zeta \leq \sqrt{\Delta L}$, every algorithm $A \in \mathcal{A}^{cent}_{ZR}$ optimizing a problem in $\mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta) \cup \mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)$ with $K > 0$ intermittent accesses to two-point first-order oracles $\{\mathcal{O}^{2,L,\sigma}_{F_m}\}_{m \in support(\mathcal{P})}$ on all the machines outputs x^A_R after $R \succeq 1$ rounds such that,*

$$\mathbb{E}\left[\left\|\nabla F(x^A_R)\right\|^2\right] \succeq \frac{\Delta L}{R} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma\Delta L}{MKR}\right)^{2/3} + \frac{\zeta^2}{MR} + \left(\frac{\zeta\Delta L}{MR}\right)^{2/3}.$$

Deterministic case: Note that if each client m has a deterministic oracle $\mathcal{O}^{2,L,0}_{F_m}$, suppose $b = b_0 = 1$, we can choose

$$\beta = \max\left\{\frac{1}{R}, \left(\frac{\Delta\tau\sqrt{M}}{\zeta^2 R}\right)^{2/3}\right\},\ M_0 = MR,\ \eta = \min\left\{\frac{1}{L}, \frac{1}{K\tau}, \frac{\sqrt{\beta M}}{\tau K}\right\},$$

and we can get (ignoring the dependence on some numerical constants)

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq \frac{\Delta\tau}{R} + \frac{\Delta L}{KR} + \frac{\zeta^2}{MR} + \left(\frac{\zeta\Delta\tau}{MR}\right)^{2/3}.$$

Oracle complexity in Table 2 for mini-batch STORM.

We need following many communication rounds to achieve ϵ stationarity:

$$\frac{\Delta L}{\epsilon} + \frac{\sigma^2}{MK\epsilon} + \frac{\sigma\Delta L}{MK^{1/2}\epsilon^{3/2}} + \frac{\zeta^2}{M\epsilon} + \frac{\zeta\Delta L}{M\epsilon^{3/2}}.$$

Recalling the assumptions for table 2, since $\epsilon^{1/2} \preceq \zeta/m$ we can ignore the first term, since $\epsilon^{1/2} \preceq \Delta L/\sigma$ we can ignore the second term, and since $\epsilon^{1/2} \preceq \Delta\tau/\zeta \preceq \Delta L/\zeta$ we can also ignore the fourth term. This leaves us the following communication complexity,

$$\frac{\sigma\Delta L}{MK^{1/2}\epsilon^{3/2}} + \frac{\zeta\Delta L}{M\epsilon^{3/2}}.$$

Thus to get the best communication complexity of order $1/\epsilon^{3/2}$ we choose $K \cong \max\{1, \sigma^2/\zeta^2\}$ which simplifies to,

$$R \cong \frac{(\sigma + \zeta)\Delta L}{M\epsilon^{3/2}}$$

Then the oracle complexity is of the order

$$M \cdot \frac{\sigma \Delta L}{M\epsilon^{3/2}} + M\frac{\sigma^2}{\zeta^2} \cdot \frac{\Delta L \zeta}{M\epsilon^{3/2}},$$

where we chose $K \cong 1$ for the first term and $K \cong \sigma^2/\zeta^2$ for the second term. This simplifies to,

$$N \cong \frac{\sigma^2 \Delta L}{\zeta \epsilon^{3/2}} + \frac{\sigma \Delta L}{\epsilon^{3/2}} \cong \frac{\sigma \Delta L}{\epsilon^{3/2}} \cdot \left(1 + \frac{\sigma}{\zeta}\right).$$

Oracle complexity in Table 2 for CE-LSGD.
We need the following many communication rounds to achieve ϵ stationarity:

$$\frac{\Delta \tau}{\epsilon} + \frac{\Delta L}{\sqrt{K}\epsilon} + \frac{\sigma^2}{MK\epsilon} + \frac{\sigma \Delta L}{MK\epsilon^{3/2}} + \frac{\zeta^2}{M\epsilon} + \frac{\zeta \Delta \tau}{M\epsilon^{3/2}} + \frac{\Delta(\zeta L + \sigma \tau)}{M\sqrt{K}\epsilon^{3/2}}.$$

Recalling the assumptions for table 2, since $\epsilon^{1/2} \preceq \zeta/m$ we can ignore the first and second terms, since $\epsilon^{1/2} \preceq \Delta L/\sigma$ we can ignore the third term, and since $\epsilon^{1/2} \preceq \Delta\tau/\zeta \preceq \Delta L/\zeta$ we can also ignore the fifth term. This gives us the following simplified communication complexity,

$$\frac{\sigma \Delta L}{MK\epsilon^{3/2}} + \frac{\zeta \Delta \tau}{M\epsilon^{3/2}} + \frac{\Delta(\zeta L + \sigma \tau)}{M\sqrt{K}\epsilon^{3/2}}.$$

Note that because of the second term we are bound to have a communication complexity of order $1/\epsilon^{3/2}$, just like MB-STORM. What needs to be figured out, is how to correctly balance the K in other terms. If we choose $K \cong 1$, we will get both communication and oracle complexity of the order $1/\epsilon^{3/2}$. All we need to do is account for the relative scaling of the problem parameters now. In particular we choose K such that

$$\frac{\zeta \Delta \tau}{M\epsilon^{3/2}} \cong \frac{\sigma \Delta L}{MK\epsilon^{3/2}}, \frac{\zeta \Delta \tau}{m\epsilon^{3/2}} \cong \frac{\Delta(\zeta L + \sigma \tau)}{M\sqrt{K}\epsilon^{3/2}},$$

Then we need to ensure

$$K \cong \max\left\{\frac{\sigma L}{\zeta \tau}, \frac{L^2}{\tau^2}, \frac{\sigma^2}{\zeta^2}\right\}.$$

This ensures that,

$$R \cong \frac{\zeta \Delta \tau}{M\epsilon^{3/2}},$$

and the oracle complexity is upper bounded by,

$$\begin{aligned} N \cong MK\frac{\zeta \Delta \tau}{M\epsilon^{3/2}} &\preceq \frac{\sigma \Delta L}{\epsilon^{3/2}} + \frac{\zeta \Delta L}{\epsilon^{3/2}} \cdot \frac{L}{\tau} + \frac{\sigma \Delta \tau}{\epsilon^{3/2}} \cdot \frac{\sigma}{\zeta}, \\ &\cong \frac{\zeta \Delta L}{\epsilon^{3/2}} \cdot \frac{L}{\tau} + \frac{\sigma \Delta L}{\epsilon^{3/2}} \cdot \left(1 + \frac{\sigma \tau}{\zeta L}\right), \end{aligned}$$

which recovers the oracle complexity in Table 2.

□

D.1 The Gap in the Partial Participation Setting

Accroding to the discussion in Section 3.2, we can most likely improve our upper bound for CE-LSGD as well. For instance, note that the guarantee for MB-STORM, which follows from our general analysis for CE-LSGD, has a gap w.r.t. the centralized lower bound in the third term, i.e., the red term in Table 1. This gap is most likely a result of our analysis and can be seen in the rate for CE-LSGD (red terms in Table 1). However, if we consider the hierarchical setting described in Section C.2, where each oracle query is made b-times to get a gradient estimate, then we can recover the following guarantees for CE-LSGD and MB-STORM.

Theorem D.3. *Suppose for all m in support of $\mathcal{P}$, $F_m \in \mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta) \cap \mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)$, each client $m \in [M]$ has a stochastic oracle $\mathcal{O}^{2,L,\sigma}_{F_m}$ which it uses through b-calls for every single query, and assume that $\frac{\Delta\tau}{R} + \frac{\Delta L}{\sqrt{KbR}} \preceq \frac{\sigma^2}{\sqrt{MKb}} + \frac{\zeta^2}{\sqrt{M}}$. Then the output $\widetilde{x}$ of Algorithm 2 using $b_0 = Kb$, $M_0 = MR$, $\beta = \max\left\{\frac{1}{R}, \left(\frac{\Delta(\tau + L/\sqrt{Kb})\sqrt{M}}{R(\sigma^2/Kb + \zeta^2)}\right)^{2/3}\right\}$, and $\eta \;=\; \min\left\{\frac{1}{L}, \frac{1}{K\tau}, \frac{\sqrt{b}}{\sqrt{K}L}, \frac{\sqrt{\beta M}}{\sqrt{Kb}L}, \frac{\sqrt{\beta M}}{\tau K}\right\}$ satisfies*

$$\mathbb{E}\left\|\nabla F(\widetilde{x})\right\|^2 \preceq \frac{\Delta\tau}{R} + \frac{\Delta L}{KR} + \frac{\Delta L}{\sqrt{Kb}R} + \frac{\sigma^2}{MKR} + \left(\frac{\sigma\Delta L}{MKR}\right)^{2/3} + \frac{\zeta^2}{MR} + \left(\frac{\zeta\Delta\tau}{MR}\right)^{2/3} + \left(\frac{\Delta(\sigma\tau + L\zeta)}{M\sqrt{Kb}R}\right)^{2/3}.$$

Theorem D.4. *Suppose for all m in support of $\mathcal{P}$, $F_m \in \mathcal{F}^1_{\mathcal{P}}(L, \Delta, \zeta) \cap \mathcal{F}^2_{\mathcal{P}}(L, \Delta, \tau)$, if each client m has a stochastic oracle $\mathcal{O}^{2,L,\sigma}_{F_m}$ which it uses through b-calls for every single query, and assume that $\frac{\Delta L}{R} \preceq \frac{\sigma^2}{\sqrt{MKb}} + \frac{\zeta^2}{\sqrt{M}}$. Then the output $\widetilde{x}$ of* MB-STORM *using $b_0 = Kb$, $M_0 = MR$, $\beta = \max\left\{\frac{1}{R}, \left(\frac{\Delta L\sqrt{M}}{R(\sigma^2/Kb + \zeta^2)}\right)^{2/3}\right\}$, and $\eta \;=\; \min\left\{\frac{1}{KL}, \frac{\sqrt{\beta M}}{KL}\right\}$ satisfies*

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \preceq \frac{\Delta L}{R} + \frac{\sigma^2}{MKbR} + \left(\frac{\sigma\Delta L}{M\sqrt{Kb}R}\right)^{2/3} + \frac{\zeta^2}{MR} + \left(\frac{\zeta\Delta L}{MR}\right)^{2/3}.$$

If we use similar arguments as in Appendix B (for proving Theorem D.2), we can then show that the upper bound for MB-STORM in Theorem D.4 is optimal except for the third term. As for CE-LSGD, by choosing $b = K$, the gap in the optimization term (brown term in Table 1) goes away, just like the full-participation setting.

E CE-LSGD with HvP

In this section we present a version of our algorithm 1 for the online setting. A motivating example for this discussion is the distributed stochastic optimization (DSO) problem, where for each client $m \in [M]$, $F_m(\cdot) := \mathbb{E}_{z\sim\mathcal{D}_m}[f(\cdot; z)]$ and only client m can sample from $\mathcal{D}_m$. In this model, if for all $z \sim \text{supp}(\mathcal{D}_m)$, $f(\cdot; z) \in \mathcal{F}(L, \Delta)$ and $\mathbb{E}_{z\sim\mathcal{D}_m}[\|\nabla f(\cdot; z) - \nabla F(\cdot; z)\|^2] \leq \sigma^2$, then we can implement $\mathcal{O}^{2,L,\sigma}_{F_m}$ at points $x, y \in \mathbb{R}^d$ by first sampling $z \sim \mathcal{D}_m$ and then returning $(f(x; z), f(y; z), \nabla f(x; z), \nabla f(y; z))$. DSO captures problems in cross-device Federated learning (FL) [16, 17] where the functions $f(\cdot; z)$ are loss functions and z denoting a data-sample is observed in an online fashion. The devices don't store the data for future queries so all the queries must be made as soon as $f(\cdot; z)$ becomes available. Most variance-reduced algorithms only require access to $f(\cdot; z)$ at the current and previous models. Thus, the two-point oracle can be implemented even in the online setting by always storing two models on memory. In certain settings, though, this is not possible as the model sizes are too big, and two different models can not be stored on the device. To alleviate this, we propose an extension of our Algorithm 1, which uses a stochastic Hessian vector product oracle instead of the multi-point oracle to implement variance reduction [36].

For ease of presentation, we first introduce some definitions. We assume that for all $m \in [M]$,

$$F_m \in \mathcal{F}(L, L_2, \Delta) := \Bigg\{ G \in \mathcal{F}(L) \text{s.t. } G \text{ is twice-differentiable, } G(0) - \inf_{x\in RR^d} G(x) \leq \Delta$$

$$\text{and } \sup_{x,y\in\mathbb{R}^d} \|\nabla^2 G(x) - \nabla^2 G(y)\| \leq L_2\|x - y\| \Bigg\}.$$

Similarly, we denote $F \in \mathcal{F}(L, L_2, \Delta)$ and define the problem class $\boldsymbol{\mathcal{F}^2_M(L, L_2, \Delta, \tau)}$ for $\{F_m\}_{m\in[M]}$ with bounded second-order τ-heterogeneity.

Definition 8 (Stochastic Hessian-vector Product Oracle)**.** *Given a function $G \in \mathcal{F}(L, L_2, \Delta)$, $\mathcal{Q}^{L,\sigma}_G : (\mathbb{R}^d)^2 \times \mathcal{Z} \to \mathbb{R} \times (\mathbb{R}^d)^2$ is a stochastic Hessian-vector Product oracle if for some distribution $\mathcal{D}$ on $\mathcal{Z}$, and for any $x, v \in \mathbb{R}^d$, the oracle samples a random seed $z \sim \mathcal{D}$ and returns $\mathcal{Q}^{L,\sigma}_G(x, v, z) \;=\; (f(x; z),\; g(x; z),\; h(x; z)v)$ such that:*

(a) $\mathbb{E}_{z\in\mathcal{D}}[(f(x;z), g(x;z), h(x;z)v)] = (G(x), \nabla G(x), \nabla^2 G(x)v)$

(b) $\mathbb{E}_{z\sim\mathcal{D}}\left[\|g(x;z) - \nabla G(x)\|^2\right] \le \sigma^2$*,*

(c) for all $x, y \in \mathbb{R}^d$*,* $\mathbb{E}_{z\sim\mathcal{D}}[\|g(x;z) - g(y;z)\|] \le L\|x-y\|$*,*

(d) $\mathbb{E}_{z\sim\mathcal{D}}\left[\left\|h(x;z)v - \nabla^2 G(x)v\right\|^2\right] \le L^2\|v\|^2$*.*

Note that this stochastic oracle doesn't require simultaneous queries at the same z. We state the result of our algorithm with HvP below. Same as before, we choose the input $T = K$ in Algorithm 3

Theorem E.1. *Suppose* $\{F_m\}_{m\in[M]} \in \mathcal{F}_M^2(L, L_2, \Delta, \tau)$*, and each client* $m \in [M]$ *has a stochastic HvP oracle* $\mathcal{Q}_{F_m}^{L,\sigma}$*, then Algorithm 3 using* $\eta = c_1 \cdot \min\{1/L, 1/(K\tau)\}$ *satisfies*

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le c_2\left(\frac{L\Delta}{RK} + \frac{\tau\Delta}{R} + \epsilon\right).$$

Furthermore, if we choose $1/R \le \beta = \eta K\sqrt{\epsilon}\cdot\max\{M^{1/2}\epsilon^{1/4}L_2^{1/2}/\sigma, L/\sigma\} \le 1$ *and assume* $\epsilon^{1/2}M \le L^2/L_2$*, with probability at least* $7/8$*, Algorithm 3 uses the following number of oracle calls to achieve* ϵ*-approximate stationary point*

$$N = c_3\left(\frac{\Delta\sigma L}{\epsilon^{3/2}} + \frac{\eta\Delta L^2K^2}{\epsilon} + \frac{M\eta\Delta L^2K}{\epsilon} + \frac{M\Delta L}{\epsilon} + \frac{M\Delta K\tau}{\epsilon}\right),$$

where c_1, c_2, c_3 *are numerical constants. In addition, if we have* $\epsilon^{1/4}M \le \sigma^{1/2}$*,* $\epsilon^{1/4}L \le \tau\sigma^{1/2}$*, and* $\epsilon \le \sigma^2$*, then Algorithm 3 using* $K = \sigma^{1/2}/\epsilon^{1/4}$ *can achieve the* ϵ*-approximate stationary point with the following communication and oracle complexities*

$$R \le c_4\frac{\Delta\tau}{\epsilon} \text{ and } N = c_5\frac{\Delta L\sigma}{\epsilon^{3/2}},$$

where c_4, c_5 *are numerical constants.*

To interpret this result we can consider the simpler distributed stochastic optimization setting, where $F_m = \mathbb{E}_{z\sim\mathcal{D}_m}[f(x;z)]$ and $f(\cdot;z)$ is L-Lipschitz. In this setting, we can easily implement the HvP oracle. Then, Algorithm 3 attains the same order of communication and oracle complexities as Algorithm 1 (see Theorem C.1) without the requirement of simultaneous queries.

In this section, we provide the proof of Theorem E.1.

Proof of Theorem E.1. In the following discussion, we use $\{C_i\}_{i=1}^{16}$ to denote numerical constants. First of all, we will bound the estimation error $\mathbb{E}\left\|v_{r,k}^j - \nabla F(w_{r+1,k}^j)\right\|^2$. Consider the local updates for client j. We have

$$v_{r,k}^j = v_{r,k-1}^j + \sum_{l=1}^{b_{r,k}} \nabla^2 f_j(w_{r+1,k}^{j,l-1}, z_l)(w_{r+1,k}^{j,l} - w_{r+1,k}^{j,l-1}),$$

where $w_{r+1,k}^{j,l}$ is defined as

$$w_{r+1,k}^{j,l} = \frac{l}{b_{r,k}}w_{r+1,k}^j + \left(1 - \frac{l}{b_{r,k}}\right)w_{r+1,k-1}^j \text{ for } l \in \{0,\dots,b_{r,k}\}.$$

Therefore, we can get

$$\begin{aligned} v_{r,k}^j - \nabla F(w_{r+1,k}^j) &= v_{r,k-1}^j + \sum_{l=1}^{b_{r,k}} \nabla^2 f_j(w_{r+1,k}^{j,l-1}, z_l)(w_{r+1,k}^{j,l} - w_{r+1,k}^{j,l-1}) - \nabla F(w_{r+1,k}^j) \\ &= v_{r,k-1}^j - \nabla F(w_{r+1,k-1}^j) \\ &\quad + \sum_{l=1}^{b_{r,k}} \left(\nabla^2 f_j(w_{r+1,k}^{j,l-1}, z_l) - \nabla^2 F_j(w_{r+1,k}^{j,l-1})\right)(w_{r+1,k}^{j,l} - w_{r+1,k}^{j,l-1}) \\ &\quad + \sum_{l=1}^{b_{r,k}} \nabla^2 F_j(w_{r+1,k}^{j,l-1})(w_{r+1,k}^{j,l} - w_{r+1,k}^{j,l-1}) + \nabla F(w_{r+1,k-1}^j) - \nabla F(w_{r+1,k}^j). \end{aligned}$$

Algorithm 3 CE-LSGD with Hessian-vector Product

```
input Initialization $x_0$, iteration number $R$, step size $\eta$, parameters $T$, $B_0$, $\beta \in [0, 1]$
Let $x_{-1} = x_0$
for $r = 0, 1, \ldots, R-1$ do
    if $r = 0$ set $\rho = 1, Q = 1, B = B_0$ else set $\rho = \beta$, $Q = T$ and $B = B_r$, where $B_r = C_1 \max\left\{L^2\|x_r - x_{r-1}\|^2/(M\beta\epsilon), L_2\|x_r - x_{r-1}\|^2/(\beta\sqrt{\epsilon}), \sigma^2\beta/(M\epsilon)\right\}$
    Communicate (send) $(x_r, x_{r-1})$ to clients
    on client $m \in [M]$ do
        Compute $\nabla F_{m,\mathcal{B}_r^m}(x_r)$, where $|\mathcal{B}_r^m| = B$, and $H_r^m = HVP(x_r, x_{r-1}, B, m)$
        Communicate (rec) $\left(\nabla F_{m,\mathcal{B}_r^m}(x_r), H_r^m\right)$ to the server
    end on client
    $v_r = \frac{\beta}{M}\sum_{m=1}^M \nabla F_{m,\mathcal{B}_r^m}(x_r) + (1-\beta)\left(v_{r-1} - \frac{1}{M}\sum_{m=1}^M H_r^m\right)$
    Communicate (send) $(x_r, v_r)$ to client $\widetilde{m}_r$, where $\widetilde{m}_r \sim Unif\left([M]\right)$
    on client $\widetilde{m}$ do
        $w_{r+1,1}^{\widetilde{m}_r} := w_{r+1,0}^{\widetilde{m}_r} := x_r, v_{r,0}^{\widetilde{m}_r} := v_r$
        for $k = 1, \ldots, Q$ do
            Let $b_{r,k} = C_2 K \cdot \max\left\{\eta^2 L^2 K, L_2\|w_{r+1,k}^{\widetilde{m}_r} - w_{r+1,k-1}^{\widetilde{m}_r}\|^2/\sqrt{\epsilon}\right\}$ when $k > 1$
            $v_{r,k}^{\widetilde{m}_r} = v_{r,k-1}^{\widetilde{m}_r} + HVP(w_{r+1,k}^{\widetilde{m}_r}, w_{r+1,k-1}^{\widetilde{m}_r}, b_{r,k}, \widetilde{m})$
            $w_{r+1,k+1}^{\widetilde{m}_r} = w_{r+1,k}^{\widetilde{m}_r} - \eta v_{r,k}^{\widetilde{m}_r}$
        end for
        Communicate (rec) $\left(w_{r+1,Q+1}^{\widetilde{m}_r}\right)$ to the server
    end on client
    Let $x_{r+1} = w_{r+1,Q+1}^{\widetilde{m}_r}$
end for
output Choose $\widetilde{x}$ uniformly from $\{w_{r,k}^{\widetilde{m}_r}\}_{r \in [R], k \in [Q]}$
```

Algorithm 4 Hessian-vector Products (HVP) Estimator

```
input Parameters $x, x_{\text{prev}}$, batch size $b_0$, client index $j$
Let $b = \lceil b_0 \rceil$
Let $x^l = \frac{l}{b}x + \left(1 - \frac{l}{b}\right)x_{\text{prev}}$ for $l \in \{0, \ldots, b\}$
$H = \sum_{l=1}^b \nabla^2 f(x^{l-1}; z_l)(x^l - x^{l-1})$, where $z_l \sim_{i.i.d.} D_j$
output $H$
```

Thus we can obtain that

$$
\begin{aligned}
&\mathbb{E}\left\|v_{r,k}^j - \nabla F(w_{r+1,k}^j)\right\|^2 \\
&\le \mathbb{E}\Big\|v_{r,k-1}^j - \nabla F(w_{r+1,k-1}^j) \\
&\qquad\qquad + \sum_{l=1}^{b_{r,k}} \nabla^2 F_j(w_{r+1,k}^{j,l-1})(w_{r+1,k}^{j,l} - w_{r+1,k}^{j,l-1}) + \nabla F(w_{r+1,k-1}^j) - \nabla F(w_{r+1,k}^j)\Big\|^2 \\
&\quad + \mathbb{E}\left\|\sum_{l=1}^{b_{r,k}} \left(\nabla^2 f_j(w_{r+1,k}^{j,l-1}, z_l) - \nabla^2 F_j(w_{r+1,k}^{j,l-1})\right)(w_{r+1,k}^{j,l} - w_{r+1,k}^{j,l-1})\right\|^2 \\
&\le \left(1 + \frac{1}{K}\right)\mathbb{E}\left\|v_{r,k-1}^j - \nabla F(w_{r+1,k-1}^j)\right\|^2 \\
&\quad + (1+K)\mathbb{E}\left\|\sum_{l=1}^{b_{r,k}} \nabla^2 F_j(w_{r+1,k}^{j,l-1})(w_{r+1,k}^{j,l} - w_{r+1,k}^{j,l-1}) + \nabla F(w_{r+1,k-1}^j) - \nabla F(w_{r+1,k}^j)\right\|^2 \\
&\quad + \mathbb{E}\left\|\sum_{l=1}^{b_{r,k}} \left(\nabla^2 f_j(w_{r+1,k}^{j,l-1}, z_l) - \nabla^2 F_j(w_{r+1,k}^{j,l-1})\right)(w_{r+1,k}^{j,l} - w_{r+1,k}^{j,l-1})\right\|^2.
\end{aligned}
$$

In addition, we have

$$\begin{aligned}
&\mathbb{E}\Big\|\sum_{l=1}^{b_{r,k}}\nabla^2 F_j(w_{r+1,k}^{j,l-1})(w_{r+1,k}^{j,l}-w_{r+1,k}^{j,l-1})+\nabla F(w_{r+1,k-1}^{j})-\nabla F(w_{r+1,k}^{j})\Big\|^2\\
&\quad=\mathbb{E}\Big\|\sum_{l=1}^{b_{r,k}}\nabla^2 F_j(w_{r+1,k}^{j,l-1})(w_{r+1,k}^{j,l}-w_{r+1,k}^{j,l-1})+\sum_{l=1}^{b_{r,k}}\big(\nabla F_j(w_{r+1,k-1}^{j,l-1})-\nabla F_j(w_{r+1,k}^{j,l})\big)\\
&\qquad\qquad+\sum_{l=1}^{b_{r,k}}\big(\nabla F(w_{r+1,k-1}^{j,l-1})-\nabla F(w_{r+1,k}^{j,l})-\nabla F_j(w_{r+1,k-1}^{j,l-1})+\nabla F_j(w_{r+1,k}^{j,l})\big)\Big\|^2\\
&\quad\le b_{r,k}^2\frac{L_2^2\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\|^4}{2b_{r,k}^4}+2\tau^2\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\|^2\\
&\quad=\frac{L_2^2}{2b_{r,k}^2}\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\|^4+2\tau^2\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\|^2,
\end{aligned}$$

where the inequality is due to each client $m \in [M]$ has a stochastic HvP oracle $\mathcal{Q}_{F_m}^{L,\sigma}$ and $\{F_m\}_{m\in[M]} \in \mathcal{F}_M^2(L, L_2, \Delta, \tau)$. On the other hand, we have

$$\begin{aligned}
&\mathbb{E}\Big\|\sum_{l=1}^{b_{r,k}}\big(\nabla^2 f_j(w_{r+1,k}^{j,l-1},z_l)-\nabla^2 F_j(w_{r+1,k}^{j,l-1})\big)(w_{r+1,k}^{j,l}-w_{r+1,k}^{j,l-1})\big\|^2\\
&\quad=\frac{1}{b_{r,k}^2}\sum_{l=1}^{b_{r,k}}\mathbb{E}\big\|\big(\nabla^2 f_j(w_{r+1,k}^{j,l-1},z_l)-\nabla^2 F_j(w_{r+1,k}^{j,l-1})\big)(w_{r+1,k}^{j}-w_{r+1,k-1}^{j})\Big\|^2\\
&\quad\le\frac{L^2}{b_{r,k}}\big\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\big\|^2,
\end{aligned}$$

where the equality is due to the independence of random variables and the definition of $w_{r+1,k}^{j,l}$, and the inequality comes from the stochastic HvP oracle $\mathcal{Q}_{F_m}^{L,\sigma}$. Combining these results, we can obtain

$$\begin{aligned}
\mathbb{E}\big\|v_{r,k}^{j}-\nabla F(w_{r+1,k}^{j})\big\|^2\le\Big(1+\frac{1}{K}\Big)\mathbb{E}\big\|v_{r,k-1}^{j}-\nabla F(w_{r+1,k-1}^{j})\big\|^2+2(1+K)\tau^2\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\|^2\\
+(1+K)\frac{L_2^2}{2b_{r,k}^2}\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\|^4+\frac{L^2}{b_{r,k}}\big\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\big\|^2.
\end{aligned}$$

Therefore, using the above inequality recursively, we can obtain

$$\begin{aligned}
\mathbb{E}\big\|v_{r,k}^{j}-\nabla F(w_{r+1,k}^{j})\big\|^2\le e\mathbb{E}\big\|v_{r,0}^{j}-\nabla F(w_{r+1,0}^{j})\big\|^2+\sum_{k=1}^{K}\frac{eL^2}{b_{r,k}}\mathbb{E}\big\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\big\|^2\\
+\sum_{k=1}^{K}\frac{eKL_2^2}{b_{r,k}^2}\big\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\big\|^4+4eK\tau^2\sum_{k=1}^{K}\big\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\big\|^2.
\end{aligned}\tag{E.1}$$

Next, let's consider the global variance reduction term v_r. Recall that, we have

$$v_r=\beta\frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r)+(1-\beta)v_{r-1}+(1-\beta)\frac{1}{M}\sum_{j=1}^{M}\sum_{l=1}^{B_r}\nabla^2 f_j(x_r^{l-1},z_l^j)(x_r^l-x_r^{l-1}),$$

where x_r^l is defined as

$$x_r^l=\frac{l}{B_r}x_r+\Big(1-\frac{l}{B_r}\Big)x_{r-1}\text{ for }l\in\{0,\dots,B_r\}.$$

Therefore, we have

$$\begin{aligned} v_r - \nabla F(x_r) = {} & (1-\beta)\big(v_{r-1} - \nabla F(x_{r-1})\big) + \beta\bigg(\frac{1}{M}\sum_{j=1}^{M} \nabla F_{j,\mathcal{B}_r^j}(x_r) - \nabla F(x_r)\bigg) \\ & + (1-\beta)\big(\frac{1}{M}\sum_{j=1}^{M}\sum_{l=1}^{B_r} \big(\nabla^2 f_j(x_r^{l-1}, z_l^j) - \nabla^2 F_j(x_r^{l-1})\big)(x_r^l - x_r^{l-1})\big) \\ & + (1-\beta)\big(\frac{1}{M}\sum_{j=1}^{M}\sum_{l=1}^{B_r} \nabla^2 F_j(x_r^{l-1})(x_r^l - x_r^{l-1}) + \nabla F(x_{r-1}) - \nabla F(x_r)\big). \end{aligned}$$

Thus we can obtain

$$\begin{aligned} \mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \le {} & (1-\beta)^2\bigg(1+\frac{\beta}{2}\bigg)\mathbb{E}\big\|v_{r-1} - \nabla F(x_{r-1})\big\|^2 + \beta^2\frac{\sigma^2}{MB_r} \\ & + \bigg(1+\frac{2}{\beta}\bigg)(1-\beta)^2\frac{L_2^2}{4{B_r}^2}\|x_r - x_{r-1}\|^4 + (1-\beta)^2\frac{L^2}{MB_r}\|x_r - x_{r-1}\|^2. \end{aligned} \tag{E.2}$$

Suppose we choose $b_{r,k}$ as follows (here j is random sampled as in line 10 of Algorithm 3):

$$b_{r,k} = C_1 K \cdot \max\bigg\{\eta^2 L^2 K, \frac{L_2\big\|w_{r+1,k}^j - w_{r+1,k-1}^j\big\|^2}{\sqrt{\epsilon}}\bigg\}.$$

Therefore, plugging $b_{r,k}$ into equation E.1, we can obtain

$$\begin{aligned} & \mathbb{E}\big\|v_{r,k}^j - \nabla F(w_{r+1,k}^j)\big\|^2 \qquad \text{(E.3)} \\ & \le e\mathbb{E}\big\|v_{r,0}^j - \nabla F(w_{r+1,0}^j)\big\|^2 + \sum_{k=1}^{K}\frac{eL^2}{b_{r,k}}\big\|w_{r+1,k}^j - w_{r+1,k-1}^j\big\|^2 \\ & \quad + \sum_{k=1}^{K}\frac{eKL_2^2}{b_{r,k}^2}\big\|w_{r+1,k}^j - w_{r+1,k-1}^j\big\|^4 + 4eK\tau^2\sum_{k=1}^{K}\big\|w_{r+1,k}^j - w_{r+1,k-1}^j\big\|^2 \\ & \le e\mathbb{E}\big\|v_{r,0}^j - \nabla F(w_{r+1,0}^j)\big\|^2 + (4eK^2\tau^2\eta^2 + 1/24)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|v_{r,k-1}^j\|^2 + \epsilon \\ & \le e\mathbb{E}\big\|v_{r,0}^j - \nabla F(w_{r+1,0}^j)\big\|^2 + (8eK^2\tau^2\eta^2 + 1/12)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|v_{r,k-1}^j - \nabla F(w_{r+1,k}^j)\|^2 \\ & \quad + (8eK^2\tau^2\eta^2 + 1/12)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^j)\|^2 + \epsilon. \qquad \text{(E.4)} \end{aligned}$$

If we choose $\eta \le C_2/(K\tau)$ and use the fact that $w_{r+1,0}^j = w_{r+1,1}^j = x_r$, we can obtain

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|v_{r,k}^j - \nabla F(w_{r+1,k}^j)\|^2 \le 2e\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + \frac{1}{6K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^j)\|^2 + \epsilon. \tag{E.5}$$

Thus, according to equation C.4, and plugging the result in equation E.5, we can get

$$\begin{aligned} \frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^j)\|^2 & \le \frac{2}{K\eta}\big(\mathbb{E}F(x_r) - \mathbb{E}F(x_{r+1,K+1}^j)\big) + \frac{3}{K}\sum_{k=1}^{K}\mathbb{E}\|v_{r,k}^j - \nabla F(w_{r+1,k}^j)\|^2 \\ & \le \frac{2}{K\eta}\big(\mathbb{E}F(x_r) - \mathbb{E}F(x_{r+1,K+1}^j)\big) + 6e\mathbb{E}\big\|v_{r,0}^j - \nabla F(w_{r+1,0}^j)\big\|^2 + 3\epsilon \\ & \quad + \frac{1}{2K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^j)\|^2. \end{aligned}$$

Therefore, we can get

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^{j})\|^2 \leq \frac{4}{K\eta}\big(\mathbb{E}F(x_r) - \mathbb{E}F(x_{r+1})\big) + 12e\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + 6\epsilon. \quad \text{(E.6)}$$

Averaging equation E.6 from $t = 0, \ldots, R-1$, we can obtain

$$\frac{1}{RK}\sum_{r=0}^{R-1}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^{j})\|^2 \leq \frac{4}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_R)\big) + \frac{6e}{R}\sum_{r=0}^{R-1}\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + 6\epsilon,$$

by the definition of $\widetilde{x}$, we have

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \leq \frac{4}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_R)\big) + \frac{6e}{R}\sum_{r=0}^{R-1}\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + 6\epsilon. \quad \text{(E.7)}$$

Suppose we choose B_r as follows:

$$B_r = C_3 \cdot \max\left\{\frac{L^2\big\|x_r - x_{r-1}\big\|^2}{M\beta\epsilon}, \frac{L_2\big\|x_r - x_{r-1}\big\|^2}{\beta\sqrt{\epsilon}}\right\}, \quad \text{(E.8)}$$

Therefore, plugging B_r into equation E.2, we have

$$\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \leq (1-\beta/2)^2\mathbb{E}\big\|v_{r-1} - \nabla F(x_{r-1})\big\|^2 + 2\beta^2\frac{\sigma^2}{MB_r} + 2\beta\epsilon,$$

Furthermore, we have

$$\begin{aligned}
&\frac{\beta}{2}\sum_{r=0}^{t-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \\
&= \sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - (1-\beta/2)\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \\
&= \sum_{r=1}^{R}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - (1-\beta/2)\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - \mathbb{E}\big\|v_R - \nabla F(x_R)\big\|^2 \\
&\quad + \mathbb{E}\big\|v_0 - \nabla F(x_0)\big\|^2 \\
&\leq \sum_{r=1}^{R}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - (1-\beta/2)^2\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 - \mathbb{E}\big\|v_R - \nabla F(x_R)\big\|^2 \\
&\quad + \mathbb{E}\big\|v_0 - \nabla F(x_0)\big\|^2 \\
&\leq 2\beta^2\sum_{r=0}^{R-1}\frac{\sigma^2}{MB_r} + 2R\beta\epsilon + \frac{\sigma^2}{MB_0},
\end{aligned}$$

which implies

$$\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|v_r - \nabla F(x_r)\big\|^2 \leq 2\beta\frac{1}{R}\sum_{r=0}^{R-1}\frac{\sigma^2}{MB_r} + 2\epsilon + \frac{\sigma^2}{R\beta MB_0}. \quad \text{(E.9)}$$

Finally combining equation E.7 and equation E.9, we have

$$\begin{aligned}
\mathbb{E}\|\nabla F(\widetilde{x})\|^2 &\leq \frac{4}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_r)\big) + \frac{6e}{R}\sum_{r=0}^{R-1}\mathbb{E}\|v_r - \nabla F(x_r)\|^2 + 6\epsilon \\
&\leq \frac{4}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_r)\big) + 12e\beta\frac{1}{R}\sum_{r=0}^{R-1}\frac{\sigma^2}{MB_r} + 12e\epsilon + \frac{6e\sigma^2}{R\beta MB_0}.
\end{aligned}$$

If we have the following

$$B_r = \frac{\sigma^2\beta}{M\epsilon} \text{ and } B_0 = \frac{\sigma^2}{R\beta M\epsilon}, \tag{E.10}$$

we can obtain

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le \frac{4}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_r)\big) + 30e\epsilon.$$

Note that we have $\eta \le 1/(4L)$ and $\eta \le C_2/(K\tau)$. Therefore, we have

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \le C_4\bigg(\frac{L\Delta}{RK} + \frac{\tau\Delta}{R} + \epsilon\bigg), \tag{E.11}$$

where $\Delta = F(x_0) - F(x^*)$. Therefore, to achieve ϵ accuracy, we have $R \le C_5\big(\Delta L/(K\epsilon) + \Delta\tau/\epsilon\big)$, where $\{C_i\}_{i=1}^5$ are numerical constants.

Next, we are going to derive the number of oracle calls, i.e., gradient complexity and the number Hessian-vector product queries. According to Algorithm 3, the number of Hessian-vector product queries will be at the same order of the number of stochastic gradient evaluations. Therefore, we only need to determine the gradient complexity of Algorithm 3.

First of all, according to the requirement in equation E.10, we have the following gradient complexity per client on the global updates

$$\sum_{r=1}^{R} B_r + B_0 = R\frac{\sigma^2\beta}{M\epsilon} + \frac{\sigma^2}{R\beta M\epsilon} \le C_6\frac{\Delta\sigma^2\beta}{MK\eta\epsilon^2}, \tag{E.12}$$

where the last inequality comes from that $\beta \ge 1/R$ and $R \le C_7\Delta/(K\eta\epsilon)$. In addition, according to equation E.8, we have

$$B_r = C_3 \cdot \max\bigg\{\frac{L^2\|x_r - x_{r-1}\|^2}{M\beta\epsilon}, \frac{L_2\|x_r - x_{r-1}\|^2}{\beta\sqrt{\epsilon}}\bigg\}.$$

Furthermore, we have

$$\begin{aligned}
\mathbb{E}\|x_r - x_{r-1}\|^2 &= \mathbb{E}\|x_{r,K+1}^j - x_{r-1}\|^2 \\
&\le 4eK\eta^2\sum_{k=1}^{K-1}\mathbb{E}\|v_{r-1,k}^j - \nabla F(w_{r,k}^j)\|^2 + 4eK\eta^2\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w_{r,k}^j)\|^2 \\
&\le 4eK\eta^2\sum_{k=1}^{K-1}\big(2e\mathbb{E}\|v_{r-1} - \nabla F(x_{r-1})\|^2 + \frac{1}{6K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r,k}^j)\|^2 + \epsilon\big) \\
&\qquad + 4eK\eta^2\sum_{k=1}^{K-1}\mathbb{E}\|\nabla F(w_{r,k}^j)\|^2 \\
&\le 4e^2K^2\eta^2\mathbb{E}\|v_{r-1} - \nabla F(x_{r-1})\|^2 + 4eK^2\eta^2\epsilon \\
&\qquad + \big(4eK^2\eta^2 + eK\eta^2\big)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r,k}^j)\|^2,
\end{aligned}$$

where the first inequality is due to equation C.1 and the second one comes from equation E.5. Therefore, averaging over R, by equation E.9 and equation E.11, we can obtain that

$$\frac{1}{R}\sum_{r=1}^{R}\mathbb{E}\|x_r - x_{r-1}\|^2 \le 13e^2K^2\eta^2\epsilon.$$

Therefore, we have (the extra gradient complexity is due to line 1 in Algorithm 4)

$$\begin{aligned}\mathbb{E}\left[\sum_{r=1}^{R}(B_r+1)\right] &\le C_3\Big(\frac{L^2}{M\beta\epsilon}+\frac{L_2}{\beta\sqrt{\epsilon}}\Big)\sum_{r=1}^{R}\mathbb{E}\big\|x_r-x_{r-1}\big\|^2+R\\ &\le 13eC_3\Big(\frac{L^2}{M\beta\epsilon}+\frac{L_2}{\beta\sqrt{\epsilon}}\Big)RK^2\eta^2\epsilon+R\\ &\le C_8\eta\Big(\frac{\Delta L^2K}{M\beta\epsilon}+\frac{\Delta L_2K}{\beta\sqrt{\epsilon}}\Big)+C_8\frac{\Delta L}{K\epsilon}+C_8\frac{\Delta\tau}{\epsilon}. \qquad \text{(E.13)}\end{aligned}$$

To choose the **optimal value of** β, let's consider equation E.12 and equation E.13. Note that given equation E.13, we can use Markov's inequality to show that (we will specify the probability later)

$$\sum_{r=1}^{R}(B_r+1)\le C_8\eta\Big(\frac{\Delta L^2K}{M\beta\epsilon}+\frac{\Delta L_2K}{\beta\sqrt{\epsilon}}\Big)+C_8\frac{\Delta L}{K\epsilon}+C_8\frac{\Delta\tau}{\epsilon}. \qquad \text{(E.14)}$$

Therefore, combining equation E.12 and equation E.14, we have the following gradient complexity for global updates

$$\sum_{r=1}^{R}(B_r+1)+B_0=C_9\Big(\frac{\Delta\sigma^2\beta}{MK\eta\epsilon^2}+\eta\frac{\Delta L^2K}{M\epsilon\beta}+\eta\frac{\Delta L_2K}{\beta\sqrt{\epsilon}}+\frac{\Delta L}{K\epsilon}+\frac{\Delta\tau}{\epsilon}\Big).$$

Solving for the β to achieve the smallest gradient complexity in terms of the dependence of ϵ, we can get $\beta=\eta K\sqrt{\epsilon}\cdot\max\{M^{1/2}\epsilon^{1/4}L_2^{1/2}/\sigma, L/\sigma\}$. Therefore, equation E.12 implies that

$$\sum_{r=1}^{R}B_r+B_0\le C_6\Big(\frac{\Delta\sigma L}{M\epsilon^{3/2}}+\frac{\Delta\sigma L_2^{1/2}}{M^{1/2}\epsilon^{5/4}}\Big), \qquad \text{(E.15)}$$

and equation E.13 implies that

$$\mathbb{E}\left[\sum_{r=1}^{R}(B_r+1)\right]\le C_8\Big(\frac{\Delta\sigma L}{M\epsilon^{3/2}}+\frac{\Delta\sigma L_2^{1/2}}{M^{1/2}\epsilon^{5/4}}+\frac{\Delta L}{K\epsilon}+\frac{\Delta\tau}{\epsilon}\Big) \qquad \text{(E.16)}$$

In addition, if we have $M\le L^2/(\epsilon^{1/2}L_2)$, we will have

$$\mathbb{E}\left[\sum_{r=0}^{R}(B_r+1)\right]\le C_{10}\Big(\frac{\Delta\sigma L}{M\epsilon^{3/2}}+\frac{\Delta L}{K\epsilon}+\frac{\Delta\tau}{\epsilon}\Big). \qquad \text{(E.17)}$$

Next, let's consider the local updates, we have

$$b_{r,k}^{j}=C_1K\cdot\max\left\{\eta^2L^2,\frac{L_2\big\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\big\|^2}{\sqrt{\epsilon}}\right\}.$$

Therefore, we have

$$\sum_{r=1}^{R}\sum_{k=1}^{K}(b_{r,k}+1)=\eta^2L^2K^2R\le C_{11}\frac{\Delta\eta KL^2}{\epsilon},$$

where the inequality comes from the fact that $R\le C_7\Delta/(K\eta\epsilon)$. In addition, we have

$$\begin{aligned}\big\|w_{r+1,k}^{j}-w_{r+1,k-1}^{j}\big\|^2 &=\eta^2\|v_{r,k-1}^{j}\|^2\\ &\le\eta^2\big(\big\|v_{r,k-1}^{j}-\nabla F(w_{r+1,k-1}^{j})+\nabla F(w_{r+1,k-1}^{j})\big\|^2\big)\\ &\le 2\eta^2\big\|v_{r,k-1}^{j}-\nabla F(w_{r+1,k-1}^{j})\big\|^2+2\eta^2\big\|\nabla F(w_{r+1,k-1}^{j})\big\|^2\\ &\le 2K\eta^2\big\|v_{r-1}-\nabla F(x_{r-1})\big\|^2+2K\eta^2\frac{1}{K}\sum_{k=1}^{K}\|\nabla F(w_{r+1,k}^{j})\|^2.\end{aligned}$$

Therefore, averaging over R, by equation E.9 and equation E.11, we can obtain that (recall that we have $\eta = \min\{1/(4L), C_2/(K\tau)\}$)

$$\frac{1}{R}\sum_{r=1}^{R}\sum_{k=1}^{K}\mathbb{E}\left\|w_{r+1,k}^{j} - w_{r+1,k-1}^{j}\right\|^2 \leq 2K^2\eta^2\epsilon + 2K^2\eta^2\epsilon \leq 4K^2\eta^2\epsilon.$$

Thus, we have

$$\mathbb{E}\left[\sum_{r=1}^{R}\sum_{k=1}^{K}(b_{r,k}+1)\right] \leq C_1\frac{L_2}{\sqrt{\epsilon}}RK^3\eta^2\epsilon + RK \leq C_{12}\left(\frac{\eta\Delta L_2K^2}{\sqrt{\epsilon}} + \frac{\Delta L}{\epsilon} + \frac{\Delta K\tau}{\epsilon}\right).$$

Hence, the local updates will contribute to the following gradient complexity per client in expectation:

$$\begin{aligned}\mathbb{E}\left[\sum_{r=1}^{R}\sum_{k=1}^{K}(b_{r,k}+1)\right] &= C_{13}\left(\frac{\Delta\eta KL^2}{\epsilon} + \frac{\eta\Delta L_2K^2}{\sqrt{\epsilon}} + \frac{\Delta L}{\epsilon} + \frac{\Delta K\tau}{\epsilon}\right)\\ &\leq C_{13}\left(\frac{\Delta\eta KL^2}{\epsilon} + \frac{\eta\Delta L^2K^2}{M\epsilon} + \frac{\Delta L}{\epsilon} + \frac{\Delta K\tau}{\epsilon}\right),\end{aligned} \tag{E.18}$$

where the inequality comes from the requirement that $M \leq L^2/(\epsilon^{1/2}L_2)$. As a result, let G denote the total number of gradient complexity, combining equation E.15, equation E.16 and equation E.18, we have

$$\mathbb{E}[G] = C_{14}\left(\frac{\Delta\sigma L}{M\epsilon^{3/2}} + \frac{\Delta\eta KL^2}{\epsilon} + \frac{\eta\Delta L^2K^2}{M\epsilon} + \frac{\Delta L}{\epsilon} + \frac{\Delta K\tau}{\epsilon}\right).$$

Therefore, we have

$$\mathbb{E}[N] = C_{15}\left(\frac{\Delta\sigma L}{\epsilon^{3/2}} + \frac{\eta\Delta L^2K^2}{\epsilon} + \frac{M\eta\Delta L^2K}{\epsilon} + \frac{M\Delta L}{\epsilon} + \frac{M\Delta K\tau}{\epsilon}\right).$$

Therefore, using Markov's inequality, we have with probability at least $7/8$,

$$N = C_{16}\left(\frac{\Delta\sigma L}{\epsilon^{3/2}} + \frac{\eta\Delta L^2K^2}{\epsilon} + \frac{M\eta\Delta L^2K}{\epsilon} + \frac{M\Delta L}{\epsilon} + \frac{M\Delta K\tau}{\epsilon}\right).$$

Note that we require $\beta \leq 1$, which implies that

$$M^{1/2}\epsilon^{3/4}K \leq \frac{\sigma L}{L_2^{1/2}} \text{ and } \epsilon^{1/2}K \leq \sigma.$$

Since we have $M \leq L^2/(\epsilon^{1/2}L_2)$, we can reduce to the following requirement

$$\epsilon^{1/2}K \leq \sigma. \tag{E.19}$$

Next, we are going to show that under certain conditions, CE-LSGD-HvP is able to achieve the optimal communication complexity. In the following discussion, we ignore the dependence on the numerical constants for simplicity. Recall that, we have the following communication complexity:

$$R = \frac{\Delta L}{K\epsilon} + \frac{\Delta\tau}{\epsilon}.$$

If we want to achieve $N = \Delta\sigma L/\epsilon^{3/2}$ gradient complexities, we need to have

$$K \leq \frac{L\sigma}{ML\epsilon^{1/2}} \text{ and } K \leq \frac{(\sigma L)^{1/2}}{L^{1/2}\epsilon^{1/4}}.$$

Recall that we have the following requirements $M \leq L^2/(\epsilon^{1/2}L_2)$ and $\epsilon^{1/2}K \leq \sigma$.
Case 1: if we have

$$M \geq \frac{\sigma^{1/2}}{\epsilon^{1/4}},$$

we can get

$$K = \frac{\sigma}{M\epsilon^{1/2}}.$$

We still need the requirement $M \leq L^2/(\epsilon^{1/2}L_2)$. Furthermore, we can get

$$R = \frac{\Delta ML}{\sigma\epsilon^{1/2}} + \frac{\Delta\tau}{\epsilon}.$$

And we have

$$N = \frac{\Delta\sigma L}{\epsilon^{3/2}} + \frac{M\Delta L}{\epsilon} + \frac{\Delta L\sigma\tau}{L\epsilon^{3/2}}.$$

If we further have $M \leq \sigma L/(L\epsilon^{1/2})$, we can get

$$N = \frac{\Delta\sigma L}{\epsilon^{3/2}} + \frac{\Delta L\sigma\tau}{L\epsilon^{3/2}}.$$

Note that we also need $R \geq 1/\beta$, which implies $\Delta L \geq \sigma\epsilon^{1/2}$.
Case 2: if we have

$$M \leq \frac{\sigma^{1/2}}{\epsilon^{1/4}},$$

we can get

$$K = \frac{\sigma^{1/2}}{\epsilon^{1/4}}.$$

We still need the requirements $M \leq L^2/(\epsilon^{1/2}L_2)$ and $\epsilon^{1/2} \leq \sigma$. Furthermore, we can get

$$R = \frac{\Delta L}{\epsilon^{3/4}\sigma^{1/2}} + \frac{\Delta\tau}{\epsilon}.$$

And we have

$$N = \frac{\Delta\sigma L}{\epsilon^{3/2}} + \frac{M\Delta L}{\epsilon} + \frac{M\Delta\sigma^{1/2}\tau}{\epsilon^{5/4}}.$$

If we further have $M \leq \sigma/\epsilon^{1/2}$, we can get

$$N = \frac{\Delta\sigma L}{\epsilon^{3/2}} + \frac{\Delta\sigma\tau}{\epsilon^{3/2}}.$$

□