Byzantine-Resilient High-Dimensional Federated Learning

Deepesh Data and Suhas Diggavi, *Fellow, IEEE*

***Abstract*—We study stochastic gradient descent (SGD) with local iterations in the presence of Byzantine clients, motivated by federated learning. The clients, instead of communicating with the server in every iteration, maintain their local models, which they update by taking several SGD iterations based on their own datasets and then communicate the net update with the server, thereby achieving communication efficiency. Furthermore, only a subset of clients communicates with the server at synchronization times. The Byzantine clients may collude and send arbitrary vectors to the server to disrupt the learning process. To combat the adversary, we employ an efficient high-dimensional robust mean estimation algorithm at the server to filter-out corrupt vectors; and to analyze the outlier-filtering procedure, we develop a novel matrix concentration result that may be of independent interest. We provide convergence analyses for both strongly-convex and non-convex smooth objectives in the heterogeneous data setting. We believe that ours is the first Byzantine-resilient local SGD algorithm and analysis with non-trivial guarantees. We corroborate our theoretical results with experiments for neural network training.**

Keywords: Federated learning; Byzantine attacks; local iterations; robust mean estimation

I. Introduction

In the *federated learning* (FL) paradigm [1]–[4], several clients (e.g., mobiles devices, organizations, etc.) collaboratively learn a machine learning model, where the training process is facilitated by the data held by the participating clients (without data centralization) and is coordinated by a central server (e.g., the service provider). Due to its many advantages over the traditional centralized learning [5] (e.g., training a machine learning model without collecting the clients' data, which, in addition to reducing the communication load on the network, provides a basic level of privacy to clients' data), FL has emerged as an active area of research recently; see [6] for a detailed survey. Stochastic gradient descent (SGD) has become a de facto standard in optimization for training machine learning models at such a large scale [3], [6], [7], where clients iteratively communicate the gradient updates with the central server, which aggregates the gradients, updates the learning model, and sends the aggregated gradient back to the clients. The promise of FL comes with its own set of challenges [6]: (i) optimizing with *heterogeneous* data at different clients – the local datasets at clients may be "non-i.i.d.", i.e., can be thought of as being generated from different underlying distributions; (ii) slow and unreliable network connections between server and clients, so communication in every iteration may not be feasible; (iii) availability of only a subset of clients for training at a given time (maybe due to low connectivity, as clients may be in different geographic locations); and (iv) robustness against malicious/Byzantine clients who may send incorrect gradient updates to the server to disrupt the training process. In this paper, we propose and analyze an SGD algorithm that *simultaneously* addresses all these challenges. First we setup the problem, put our work in context with the related work, and then summarize our contributions.

We consider an empirical risk minimization problem, where data is stored at R clients, each having a different dataset (with no probabilistic assumption on data generation); client $r \in [R]$ has dataset $\mathcal{D}_r$. Let $F_r : \mathbb{R}^d \to \mathbb{R}$ denote the local loss function associated with the dataset $\mathcal{D}_r$, which is defined as $F_r(\boldsymbol{x}) \triangleq \mathbb{E}_{i \in_U [n_r]}[F_{r,i}(\boldsymbol{x})]$, where $n_r = |\mathcal{D}_r|$, i is uniformly distributed over $[n_r] \triangleq \{1, 2, \ldots, n_r\}$, and $F_{r,i}(\boldsymbol{x})$ is the loss associated with the i'th data point at client r with respect to (w.r.t.) $\boldsymbol{x}$. Our goal is to solve the following minimization problem:

$$\arg\min_{\boldsymbol{x} \in \mathcal{C}} \Big(F(\boldsymbol{x}) \triangleq \frac{1}{R} \sum_{r=1}^{R} \mathbb{E}_{i \in_U [n_r]}[F_{r,i}(\boldsymbol{x})] \Big), \tag{1}$$

where $\mathcal{C} \subseteq \mathbb{R}^d$ denotes the parameter space that is either equal to $\mathbb{R}^d$ or a compact and convex set.

In the absence of the above-mentioned FL challenges, we can minimize (1) using distributed *vanilla* SGD, where in any iteration, server broadcasts the current model parameters to all clients, each of them then samples a stochastic gradient from its local dataset and sends it back to the server, who aggregates the received gradients and updates the global model. However, this simple solution does not satisfy the FL challenges, as *every* client communicates with the server (i.e., no sampling of clients) in *every* SGD iteration (i.e., no local iterations), and furthermore, this solution breaks down even with a single malicious client [8].

Related Work. Recent work have proposed variants of the above-described vanilla SGD that address *some* of the FL

Parts of this work have appeared in the International Conference of Machine Learning (ICML) 2021 and the IEEE International Symposium of Information Theory (ISIT) 2021.

This work was supported in part by NSF grants #2139304, #2007714 and Army Research Laboratory grant under Cooperative Agreement W911NF-17-2-0196. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

Deepesh Data is with Meta Platforms, Inc., Bellevue, WA 98005, USA, and Suhas Diggavi is with the University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA (deepesh.data@gmail.com, suhasdiggavi@ucla.edu).

Part of this work was done when Deepesh Data was at UCLA.

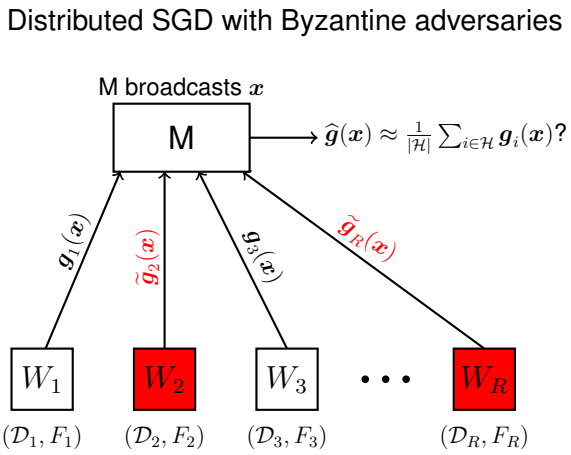


Fig. 1 In the master-worker architecture for distributed optimization, each of the R workers (denoted by W_i) stores local datasets – worker r stores $\mathcal{D}_r$ with an associated local loss function F_r. We are in a heterogeneous data setting, where the local datasets $\mathcal{D}_r$'s are arbitrary and are not necessarily generated from the same distribution. Master (denoted by M) wants to learn a machine learning model through SGD which minimizes the average of local loss functions; see (1). The adversarial nodes are denoted in red color. Let $\mathcal{H}$ denote the set of honest workers. In any SGD iteration, master broadcasts the current model parameter vector $\boldsymbol{x}$ to all workers. Each honest worker i computes the stochastic gradient $\boldsymbol{g}_i(\boldsymbol{x})$ and sends it back to the master; corrupt nodes may send arbitrary vectors. Master wants to compute $\widehat{\boldsymbol{g}}(\boldsymbol{x}) \approx \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}} \boldsymbol{g}_i(\boldsymbol{x})$ in order to update the model parameter vector. Computing $\widehat{\boldsymbol{g}}(\boldsymbol{x})$ and providing convergence analyses for strongly-convex and non-convex objectives is the subject of this paper.

challenges. The algorithms in [9]–[16] work under different heterogeneity assumptions but do not provide any robustness to malicious clients. On the other hand, [8], [17]–[23] provide robustness, but with no local iterations or sampling of clients; furthermore, they assume homogeneous (either same or i.i.d.) data across all clients. A different line of work [24]–[30] use different techniques to provide robustness, and that without local iterations or sampling of clients: [24], [25], [26], [28] use coding across datasets, which is hard to implement in FL; [29] change the objective function and adds a regularizer term to combat the adversary; [30] effectively reduce the heterogeneous problem to a homogeneous problem by clustering, and then learning happens within each cluster having homogeneous data.

Lately, there have been some works [31]–[33] that studied Byzantine robust optimization in the *homogeneous* data setting (without local iterations) and did convergence analyses with momentum updates, matching rates with that of vanilla SGD. It is important to note that the use of momentum updates in these papers help defend against time-coupled attacks in which an adversary strategically constructs an attack over time (i.e., builds the attack over the execution of gradient descent); these attacks are very difficult to combat. The first paper to recognize and defend time-coupled attacks (without using momentum) was [19], that proposed a defense algorithm in the unrealistic distributed setting where clients sample stochastic gradients from the *same* dataset, and analyzed it for convex functions using martingale-based analysis. A similar technique was extended to the non-convex case (under the same assumptions) in [34]. Both these papers also assume that the stochastic gradients have bounded noise, almost surely (as opposed to be bounded in expectation). This assumption was removed recently in [31]–[33], [35], which recognized that time-coupled attacks can also be handled by using momentum updates together with simple robust aggregation method at the server. Among these papers, the analyses in [31]–[33] are again confined to the setting where all clients sample stochastic gradients from the *same* dataset, which is orthogonal to the inherently non-i.i.d. data setup of FL. Recently, [35] proposed a new technique of bucketing to extend the aforementioned momentum analysis that can be combined with existing robust aggregators (in the homogeneous case) to the heterogeneous case. The resulting convergence rate is qualitatively better, in the sense that it coincides with the SGD rate in the absence of Byzantine corruption. It is worth noting that *none* of these papers incorporated local iterations into their algorithms and analyses, which is one of the main ingredient in FL to achieve communication efficiency.

We are only aware of one paper [36], that analyzed SGD in FL setting (i.e., including local iterations), but the approximation error (even in the Byzantine-free setting) of their solution could be as large as $\mathcal{O}(D^2 + G^2)$, where G is the gradient bound and D is the diameter of the parameter space that contains the optimal parameters $\boldsymbol{x}^*$ and all the local parameters $\boldsymbol{x}_r^t$ ever emerged at any client $r \in [R]$ in any iteration $t \in [T]$; this, in our opinion, makes their bound vacuous. In optimization, one would ideally like to have convergence rates depend on D with a factor that decays with the number of iterations, e.g., with $\frac{1}{T}$ or $\frac{1}{\sqrt{T}}$, as also in Theorem 1. In Section VI, we also empirically demonstrate the poor learning performance of their algorithm.

Our Contributions. In this paper, we tackle heterogeneity assuming that the gradient dissimilarity among local datasets is bounded (see (6), which is the same heterogeneity assumption in [35]), and propose and analyze a Byzantine-resilient SGD algorithm (Algorithm 1) with local iterations and client sampling under the bounded variance assumption for SGD (see (2)). We provide convergence analyses for strongly-convex and non-convex smooth objectives. Our convergence results are summarized below, where b is the mini-batch size for stochastic gradients, σ^2 is the variance bound, κ^2 captures the gradient dissimilarity, H is the number of local iterations in between any two consecutive synchronization indices, K is the number of clients sampled at synchronization times, ϵ is the fraction of communicating Byzantine clients at synchronization times, and ϵ' is any constant such that $\epsilon + \epsilon' \leq \frac{1}{3}$.

For strongly-convex objectives, our algorithm can find approximate optimal parameters exponentially (in $\frac{T}{H}$) fast, and for non-convex objectives, it can reach to an approximate stationary point with a speed of $\frac{1}{T/H}$. See Theorem 1 for convergence results. The approximation error Γ essentially consists of two types of error terms: $\Gamma_1 = \mathcal{O}\left(\frac{H\sigma^2}{b\epsilon'}\left(1 + \frac{3d}{2K}\right)(\epsilon + \epsilon')\right)$ and $\Gamma_2 = \mathcal{O}(H\kappa^2)$, where Γ_1 arises due to stochastic sampling of gradients and Γ_2 arises due to dissimilarity in the local datasets. Observe that Γ_1 decreases as we increase the batch size b of stochastic gradients and becomes zero if we take full-batch gradients (which implies $\sigma = 0$), as is the case in

Theorem 2. Note that even though the variance (and gradient dissimilarity) of accumulation of H gradients blows up by a factor of H^2, still both Γ_1 and Γ_2 have a *linear* dependence on the number of local iterations H. See a detailed discussion in Section II-B on the approximation error analysis and the convergence rates, and also for the reason behind obtaining rates that are off by a factor of H when compared to *vanilla* SGD – looking ahead, the reason is working with weak assumptions.

To tackle the malicious behavior of Byzantine clients, we borrow tools from recent advances in high-dimensional robust statistics [37]–[40]; in particular, we use the polynomial-time outlier-filtering procedure from [39], which was developed for robust mean estimation (RME) in high dimensions. In order to use their algorithm (described in Algorithm 2) in our setting that combines Byzantine resilience with local iterations, we develop a novel matrix concentration result (see Theorem 3), which may be of independent interest. As far as we know, this is the first concentration result for stochastic gradients with local iterations on heterogeneous data.

We believe that ours is the first work that combines *local iterations* with *Byzantine-resilience* for SGD and achieves non-trivial results under weak assumptions, while employing the RME algorithm for filtering corrupt updates. RME algorithms are provably superior than the existing algorithms based on median, trimmed-mean, etc., in high-dimensions; see also Section III for a detailed discussion on this. Unlike existing works, we also analyze our algorithm on *heterogeneous* data and allow *sampling of clients*. This required us to derive a nobel matrix concentration result in the general FL setting. Note that the earlier work that provide robustness (even without local iterations or sampling of clients) either assume homogeneous data across clients [8], [17]–[20], [22], [23] or require strong assumptions, such as the bounded gradient assumption on local functions [21].

Paper organization. We describe our algorithm and state the convergence results in Section II. In Section III, we describe our main technical tool, a new matrix concentration result for analyzing the robust accumulated gradient estimation procedure. We prove the convergence results in Section IV and Section V. We provide empirical evaluation of our method in Section VI. We instantiate our assumptions in the statistical heterogeneous data model in Section VII. Omitted details/proofs are provided in the appendices.

II. Problem Setup and Our Results

In this section, we state our assumptions, describe the adversary model and our algorithm, and state our convergence results followed by important remarks about them.

Assumption 1 (Bounded local variances)**.** *The stochastic gradients sampled from any local dataset have uniformly bounded variance over $\mathcal{C}$ for all clients, i.e., there exists a finite σ, such that for all $\boldsymbol{x} \in \mathcal{C}, r \in [R]$, we have*

$$\mathbb{E}_{i \in_U [n_r]} \|\nabla F_{r,i}(\boldsymbol{x}) - \nabla F_r(\boldsymbol{x})\|^2 \leq \sigma^2. \tag{2}$$

It will be helpful to formally define mini-batch stochastic gradients, where instead of computing stochastic gradients based on just one data point, each client samples $b \geq 1$ data points (without replacement) from its local dataset and computes the average of b gradients. For any $\boldsymbol{x} \in \mathbb{R}^d, r \in [R], b \in [n_r]$, consider the following set

$$\mathcal{F}_r^{\otimes b}(\boldsymbol{x}) := \left\{ \frac{1}{b} \sum_{i \in \mathcal{H}_b} \nabla F_{r,i}(\boldsymbol{x}) : \mathcal{H}_b \in \binom{[n_r]}{b} \right\}. \tag{3}$$

Note that $\boldsymbol{g}_r(\boldsymbol{x}) \in_U \mathcal{F}_r^{\otimes b}(\boldsymbol{x})$ is a mini-batch stochastic gradient with batch size b at client r. It is not hard to see the following, which hold for all $\boldsymbol{x} \in \mathcal{C}, r \in [R]$:

$$\mathbb{E}\left[\boldsymbol{g}_r(\boldsymbol{x})\right] = \nabla F_r(\boldsymbol{x}), \tag{4}$$

$$\mathbb{E}\left\|\boldsymbol{g}_r(\boldsymbol{x}) - \nabla F_r(\boldsymbol{x})\right\|^2 \leq \sigma^2/b. \tag{5}$$

Assumption 2 (Bounded gradient dissimilarity)**.** *The difference of the local gradients $\nabla F_r(\boldsymbol{x}), r \in [R]$ and the global gradient $\nabla F(\boldsymbol{x}) = \frac{1}{R}\sum_{r=1}^R \nabla F_r(\boldsymbol{x})$ is uniformly bounded over $\mathbb{R}^d$ for all clients, i.e., there exists a finite κ, such that*

$$\|\nabla F_r(\boldsymbol{x}) - \nabla F(\boldsymbol{x})\|^2 \leq \kappa^2, \quad \forall \boldsymbol{x} \in \mathcal{C}, r \in [R]. \tag{6}$$

Assumption 1 has been standard in the SGD literature. Assumption 2 has also been used earlier to bound heterogeneity in datasets; see, for example, [41], [42], which study decentralized SGD (without adversaries), and more recently [35], which study distributed SGD with adversaries, all with momentum. Note that when clients compute full-batch gradients, we have $\sigma = 0$ in Assumption 1; similarly, when all clients have access to the same dataset as in [8], [19], we have $\kappa = 0$ in Assumption 2. Note that (6) can be seen as a *deterministic* condition on local datasets, under which we derive our results.

A Note on Assumption 2. In the presence of Byzantine adversaries, since we do not know which clients are corrupt, we have to make some structural assumption on the data that can provide relationships among gradients sampled at different nodes for reliable decoding, and Assumption 2 is a natural way to achieve that. There are many alternatives to establish this relationship, e.g., by assuming homogeneous (same or i.i.d.) data across clients [8], [17]–[20], [22], [23] or by explicitly introducing redundancy in the system via coding-theoretic solutions [24], [25], [28]; however, these approaches fall short of in the FL setting.

Assuming bounded gradients of local functions (i.e., $\|\nabla F_r(\boldsymbol{x})\| \leq G$ for some finite G) is a common assumption in literature with heterogeneous data; see, for example, [13], [15, without adversaries] and [21, with adversaries]. Note that under this assumption, we can trivially bound the heterogeneity among local datasets by $\|\nabla F_r(\boldsymbol{x}) - \nabla F_s(\boldsymbol{x})\| \leq 2G$. So, assuming bounded gradients not only simplifies the analysis but also obscures the effect of heterogeneity on the convergence bounds, which Assumption 2 clearly brings out.[1]

Bounds on σ^2 and κ^2 in the Statistical Heterogeneous Model. Since all our results (matrix concentration and convergence) are given in terms of σ and κ, to show a

[1]See [12] for a detailed discussion on the inappropriateness of making bounded gradient assumption in heterogeneous data settings and how it obscures the effect of heterogeneity on convergence rates (even without robustness).

clear dependence of our results on the dimensionality of the problem, we bound these quantities in the *statistical heterogeneous* data model under different distributional assumptions on local gradients; see Section VII for more details, where we prove the following: For the SGD variance bound, we show that if local gradients have sub-Gaussian distribution, then $\sigma = \mathcal{O}(\sqrt{d\log(d)})$. For the gradient dissimilarity bound, we show that if either the local gradients have sub-exponential distribution and each worker has at least $n = \Omega(d\log(nd))$ data points or local gradients have sub-Gaussian distribution and $n \in \mathbb{N}$ is arbitrary, then $\kappa \leq \kappa_{\text{mean}} + \mathcal{O}(\sqrt{d\log(nd)/n})$, where κ_{mean} denotes the distance of the expected local gradients from the global gradient. Note that we make distributional assumptions on data generation *only* to derive bounds on σ, κ; otherwise, all our results hold for arbitrary datasets satisfying (5), (6).

Adversary Model. Throughout the paper, we assume that ϵ denotes the fraction of the K *communicating* clients that are corrupt, i.e., at most ϵK (out of K) clients that communicate with the server at synchronization indices may be corrupt, where $K \leq R$ is the number of clients chosen at synchronization indices. This translates to, in the *worst case*, having $\frac{\epsilon K}{R}$ fraction (i.e., a total of ϵK) of corrupt nodes in the entire system, as in the worst-case, all the corrupt nodes can be selected in a communication round; however, in practice, due to several constraints, such as the unreliable network connection (for which the adversary has no control over), we cannot expect that the server will select all corrupt nodes in all iterations. The corrupt clients may collude and arbitrarily deviate from their pre-specified programs: at synchronization indices, instead of sending the true stochastic gradients (or local models), corrupt clients may send adversarially chosen vectors to the server (they may not even send anything if they wish, in which case, the server can treat them as *erasures* and replace them with a fixed value). Note that, in the erasure case, server knows which clients are corrupt; whereas, in the Byzantine problem, server does not have this information. Note that our theoretical results hold against a worst case adversary, who is aware of the aggregation rule used by the server and has access to local gradients/models at all clients; with all this knowledge, such an adversary may conduct an *adaptive* attack, and our proposed method safeguards against such adaptive adversaries.

A. Main Results

Let $\mathcal{I}_T = \{t_1, t_2, \ldots, t_k, \ldots\}$, with $t_1 = 0$, denote the set of synchronization indices (where $\max_{i\geq 1} |t_{i+1} - t_i| = H$) when the server *arbitrarily* selects a subset of $K \leq R$ clients (denoted by $\mathcal{K} \subseteq [R]$) and sends the global model (denoted by $\boldsymbol{x}$) to them; each client $r \in \mathcal{K}$ updates its local model $\boldsymbol{x}_r$ by taking SGD steps based on its local dataset until the next synchronization time, when all clients in $\mathcal{K}$ send their local models to the server. Note that some of these clients may be corrupt and may send arbitrary vectors.[2] Server employs the decoding RAGE and update the global model $\boldsymbol{x}$ based on that. We present our Byzantine-resilient SGD algorithm with local iterations in Algorithm 1.

Algorithm 1 Byzantine-Resilient SGD with Local Iterations

1: **Initialize.** Set $t := 0$, $\boldsymbol{x}_r^0 := \boldsymbol{0}, \forall r \in [R]$, and $\boldsymbol{x} := \boldsymbol{0}$. Here, $\boldsymbol{x}$ denotes the global model and $\boldsymbol{x}_r^0$ denotes the local model at client r at time 0. Fix a constant step-size η and a mini-batch size b.
2: **while** $(t \leq T)$ **do**
3: Server selects an arbitrary subset $\mathcal{K} \subseteq [R]$ of $|\mathcal{K}| = K$ clients and sends $\boldsymbol{x}$ to all clients in $\mathcal{K}$.
4: **All clients** $r \in \mathcal{K}$ **do in parallel:**
5: Set $\boldsymbol{x}_r^t = \boldsymbol{x}$.
6: **while** (true) **do**
7: Take a mini-batch stochastic gradient $\boldsymbol{g}_r(\boldsymbol{x}_r^t) \in_U \mathcal{F}^{\otimes b}(\boldsymbol{x}_r^t)$ and update the local model:
$$\boldsymbol{x}_r^{t+1} \leftarrow \boldsymbol{x}_r^t - \eta \boldsymbol{g}_r(\boldsymbol{x}_r^t)); \quad t \leftarrow (t+1).$$
8: **if** $(t \in \mathcal{I}_T)$ **then**
9: Let $\widetilde{\boldsymbol{x}}_r^t = \begin{cases} \boldsymbol{x}_r^t & \text{if client } r \text{ is honest,} \\ \divideontimes & \text{if client } r \text{ is corrupt,} \end{cases}$,
where $\divideontimes$ is an arbitrary vector in $\mathbb{R}^d$.
10: Send $\widetilde{\boldsymbol{x}}_r^t$ to the server and break the inner **while** loop.
11: **end if**
12: **end while**
13: **At Server:**
14: Receive $\{\widetilde{\boldsymbol{x}}_r, r \in \mathcal{K}\}$ from the clients in $\mathcal{K}$.
15: For every $r \in \mathcal{K}$, let $\widetilde{\boldsymbol{g}}_{r,\text{accu}} := (\widetilde{\boldsymbol{x}}_r - \boldsymbol{x})/\eta$.
16: Apply the decoding algorithm RAGE (see Algorithm 2 on page 7) on $\{\widetilde{\boldsymbol{g}}_{r,\text{accu}}, r \in \mathcal{K}\}$. Let
$$\widehat{\boldsymbol{g}}_{\text{accu}} := \text{RAGE}(\widetilde{\boldsymbol{g}}_{r,\text{accu}}, r \in \mathcal{K}).$$
17: Update the global model $\boldsymbol{x} \leftarrow \Pi_{\mathcal{C}}(\boldsymbol{x} - \eta\widehat{\boldsymbol{g}}_{\text{accu}})$, where $\Pi_{\mathcal{C}}$ denotes the projection operator onto the set $\mathcal{C}$.
18: **end while**

Our convergence results are for both strongly-convex and non-convex smooth objectives, and we state them in the following theorem. Since our main focus in this paper is on Byzantine resilience and also combining it with local iterations, to avoid the technical complications arising due to the projection operator (in line 17), we prove our results assuming that the parameter space $\mathcal{C}$ is equal to $\mathbb{R}^d$. The analysis involving the projection can be done using the techniques in [18].

Before stating the results, we need some definitions first.

- **L-smoothness:** A function $F : \mathcal{C} \to \mathbb{R}$ is called L-smooth over $\mathcal{C} \subseteq \mathbb{R}^d$, if for every $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}$, we have $\|\nabla F(\boldsymbol{x}) - \nabla F(\boldsymbol{y})\| \leq L\|\boldsymbol{x} - \boldsymbol{y}\|$ (this property is also known as L-

[2]Note that the only disruption that the corrupt clients can cause in the training process is during the gradient aggregation at synchronization indices by sending adversarially chosen vectors to the server, and we give unlimited power to the adversary for that. Because of this and for the purpose of analysis, we can assume, without loss of generality, that in between the synchronization indices, the corrupt clients sample stochastic gradients and update their local parameters honestly.

Lipschitz gradients). This is also equivalent to $F(\boldsymbol{y}) \leq F(\boldsymbol{x}) + \langle \nabla F(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle + \frac{L}{2}\|\boldsymbol{x} - \boldsymbol{y}\|^2$.

- **μ-strong convexity:** A function $F : \mathcal{C} \to \mathbb{R}$ is called μ-strongly convex over $\mathcal{C} \subseteq \mathbb{R}^d$ (for $\mu \geq 0$), if for every $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}$, we have $F(\boldsymbol{y}) \geq F(\boldsymbol{x}) + \langle \nabla F(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle + \frac{\mu}{2}\|\boldsymbol{x} - \boldsymbol{y}\|^2$.

Theorem 1 (Mini-Batch Local Stochastic Gradient Descent)**.** *Let $\mathcal{K}_t$ denote the set of K clients that are active at any given time $t \in [0:T]$ and ϵ denote the fraction of corrupt clients in $\mathcal{K}_t$. For a global objective function $F : \mathbb{R}^d \to \mathbb{R}$, let Algorithm 1 generate a sequence of iterates $\{\boldsymbol{x}_r^t : t \in [0:T], r \in \mathcal{K}_t\}$ when running with a fixed step-size $\eta = \frac{1}{8HL}$. Fix any $\epsilon' > 0, \epsilon \geq 0, \gamma > 1/2$ such that $\epsilon \leq \frac{1}{3} - \gamma\epsilon'$ holds. Then with probability $1 - \frac{T}{H}\exp(-\frac{(2\gamma-1)\epsilon'^2(1-\epsilon)K}{8})$, the sequence of average iterates $\{\boldsymbol{x}^t = \frac{1}{K}\sum_{r\in\mathcal{K}_t} \boldsymbol{x}_r^t : t \in [0:T]\}$ satisfy the following convergence guarantees:*

- **Strongly-convex:** *If F is L-smooth for $L \geq 0$, and μ-strongly convex for $\mu > 0$, we get:*

$$\mathbb{E}\left\|\boldsymbol{x}^T - \boldsymbol{x}^*\right\|^2 \leq \left(1 - \frac{\mu}{16HL}\right)^T \left\|\boldsymbol{x}^0 - \boldsymbol{x}^*\right\|^2 + \frac{13}{\mu^2}\Gamma.$$

- **Non-convex:** *If F is L-smooth for $L \geq 0$, we get:*

$$\frac{1}{T}\sum_{t=0}^{T} \mathbb{E}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2 \leq \frac{\left[\mathbb{E}[F(\boldsymbol{x}^0)] - \mathbb{E}[F(\boldsymbol{x}^*)]\right]}{T/16HL} + \frac{9}{2}\Gamma.$$

In both the bounds above, $\Gamma = \left(\frac{3\Upsilon^2}{H} + \frac{11H\sigma^2}{b} + 36H\kappa^2\right)$ with $\Upsilon^2 = \mathcal{O}\left(\sigma_0^2(\epsilon + \epsilon')\right)$, where $\sigma_0^2 = \frac{25H^2\sigma^2}{b\epsilon'}\left(1 + \frac{3d}{2K}\right) + 28H^2\kappa^2$, and expectation is taken over the sampling of mini-batch stochastic gradients.

We prove the strongly-convex part of Theorem 1 in Section IV and the non-convex part in Section V. In addition to other complications arising due to handling Byzantine clients together with local iterations, our proof deviates from the standard proofs for local SGD: We need to show two recurrences, which arise because at synchronization indices, server performs decoding to filter-out the corrupt clients, while at other indices there is no decoding, as there is no communication. The proof of the first recurrence is significantly more involved than that of the other one.

Failure Probability. The failure probability of our algorithm is at most $\frac{T}{H}\exp(-\frac{(2\gamma-1)\epsilon'^2(1-\epsilon)K}{8})$, which holds for any $\epsilon' > 0, \epsilon \geq 0, \gamma > 1/2$ such that $\epsilon \leq \frac{1}{3} - \gamma\epsilon'$. This bound though scales linearly with T, also goes down exponentially with K. As a result, in settings such as federated learning, where number of clients could be large (e.g., in tens/hundreds of millions) and server samples about a thousand, we can get a very small probability of error, even if run our algorithm for a long time. As a concrete scenario, say, the total number of devices is $R = 10$ million and the server selects $K = 1250$ of them. Then, even if we want robustness against one million malicious clients, by choosing $\gamma = 100$ and $\epsilon' = \frac{1}{\gamma}(\frac{1}{3} - \frac{1}{10})$, the probability of failure of our algorithm would still be less than $\frac{T}{H}e^{-30}$, which even if $T = 10^6$ and $H = 1$, would still be less than 10^{-7}. Note that the bound on probability of error in Theorem 1 is a worst-case bound, and in practice, our algorithm succeeds with moderate parameter values; see, for example, Section VI for our experimental setup and the results.

Note that the error probability is due to the *stochastic* sampling of gradients, and if we want a "zero" probability of error, we can run full-batch GD, for which we get the following result (yielding the approximation error of $\Gamma = \mathcal{O}(H\kappa^2)$).

Theorem 2 (Full-Batch Local Gradient Descent)**.** *In the same setting as that of Theorem 1, except for that we running Algorithm 1 with a fixed step-size $\eta = \frac{1}{5HL}$, and in any iteration, instead of sampling mini-batch stochastic gradients, every honest client takes full-batch gradients from their local datasets. If $\epsilon \leq \frac{1}{3}$, then with probability 1, the sequence of average iterates $\{\boldsymbol{x}^t = \frac{1}{K}\sum_{r\in\mathcal{K}_t} \boldsymbol{x}_r^t : t \in [0:T]\}$ satisfy the following convergence guarantees:*

- **Strongly-convex:** *If F is L-smooth for $L \geq 0$ and μ-strongly convex for $\mu > 0$, we get:*

$$\|\boldsymbol{x}^T - \boldsymbol{x}^*\|^2 \leq \left(1 - \frac{\mu}{10HL}\right)^T \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2 + \frac{14}{\mu^2}\Gamma_{GD}. \tag{7}$$

- **Non-convex:** *If F is L-smooth for $L \geq 0$, we get:*

$$\frac{1}{T}\sum_{t=0}^{T} \left\|\nabla F(\boldsymbol{x}^t)\right\|^2 \leq \frac{10HL}{T}\left[F(\boldsymbol{x}^0) - F(\boldsymbol{x}^*)\right] + \frac{24}{5}\Gamma_{GD}. \tag{8}$$

In (7), (8), $\Gamma_{GD} = \frac{2\Upsilon_{GD}^2}{H} + 25H\kappa^2$, where $\Upsilon_{GD} = \mathcal{O}\left(H\kappa\sqrt{\epsilon}\right)$.

We provide a compete proof of Theorem 2 in Appendix D. For this, we also give a much simplified proof for the matrix concentration result of Theorem 3, which is required to prove convergence.

B. Important Remarks About Theorem 1

Analysis of the Approximation Error. In Theorem 1, the approximation error Γ essentially consists of two types of error terms: $\Gamma_1 = \mathcal{O}\left(\frac{H\sigma^2}{b\epsilon'}\left(1 + \frac{3d}{2K}\right)(\epsilon + \epsilon')\right)$ and $\Gamma_2 = \mathcal{O}(H\kappa^2)$, where Γ_1 arises due to stochastic sampling of gradients and Γ_2 arises due to dissimilarity in the local datasets. Observe that Γ_1 decreases as we increase the batch size b of stochastic gradients and becomes zero if we take full-batch gradients (which implies $\sigma = 0$), as is the case in Theorem 2. Note that even though the variance (and gradient dissimilarity) of accumulation of H gradients blows up by a factor of H^2, still both Γ_1 and Γ_2 have a *linear* dependence on the number of local iterations H. Observe that since we are working with heterogeneous datasets, the presence of gradient dissimilarity bound κ^2 (which captures the heterogeneity) in the approximation error is inevitable, and will always show up when bounding the deviation of the true "global" gradient from the decoded one in the presence of Byzantine clients, even when $H = 1$; see also Figure 2 for a pictorial intuition.

Convergence Rates. In the strongly-convex case, Algorithm 1 approximately finds the optimal parameters $\boldsymbol{x}^*$ (within Γ error) with $\left(1 - \frac{\mu}{16HL}\right)^T$ speed. Note that $\left(1 - \frac{\mu}{16HL}\right)^T \leq$

$\exp^{-\frac{\mu}{16L}\frac{T}{H}}$, which implies an exponentially fast (in T/H) convergence rate. In the non-convex case, Algorithm 1 reaches to a stationary point (within Γ error) with a speed of $\frac{1}{T/H}$. Note that the convergence rates of *vanilla* SGD (i.e., without local iterations and in Byzantine-free settings) are exponential (in T) and $\frac{1}{T}$ for strongly-convex and non-convex objectives, respectively; whereas, our convergence rates are affected by the number of local iterations H. The reason for this is precisely because we need $\eta \leq \frac{1}{8HL}$ to bound the drift in local parameters across clients; see Lemma 2. Instead, if we had assumed a stronger bounded gradient assumption (which trivially bounds the heterogeneity, as explained on page 3), then Lemma 2 would hold for a constant step-size (e.g., $\eta = \frac{1}{2L}$ would suffice), which would lead to vanilla SGD like convergence rates.

III. Robust Accumulated Gradient Estimation

In this section, first we discuss the inadequacy of traditional methods (such as coordinate-wise median and trimmed-mean) for filtering corrupt gradients in our setting, and then we motivate and describe the robust accumulated gradient estimation (RAGE) procedure that we use in Algorithm 1 as a subroutine at every synchronization index. Then we prove our new matrix concentration result that is required to establish the performance guarantee of RAGE.

Inadequacy of Median and Trimmed-Mean. Coordinate-wise median (med) and trimmed-mean (trimmean) are the two widely used robust estimation procedures that are easy to describe and implement, and they have been employed earlier for robust gradient aggregation in distributed optimization; see, for example, [18], [22, i.i.d. data setting] and [36, FL setting]. Below we argue that these methods give poor performance in FL settings for learning high-dimensional models; we also validate this claim through experiments in Section VI.

- For the simple task of robust mean estimation with inputs coming a unit covariance distribution, med and trimmean have an error that scales with the dimension as $\sqrt{d}$ [37], [39]; when we apply these methods in each SGD iteration, this error translates to a large sub-optimality gap in the convergence rate.
- The adversary may corrupt samples in a way that they preserve the norm of the original uncorrupted samples, but have different adversarially chosen directions (these are called directional attacks); since the performance of these methods are based on the magnitude of the samples, they cannot distinguish between the corrupt and uncorrupt samples. We also implement directional attacks in Section VI to show the efficacy of our method empirically.
- When taking coordinate-wise median, for estimating each coordinate, we use only a *single* sample and discard the rest. This is not a good idea in large-scale settings with non-i.i.d. data, such as FL, where there are potentially millions of clients, and if we somehow are able to use samples from *all* (or most of the) honest clients, we could get a significant reduction in variance of stochastic gradients. In med, we do not take advantage of this variance reduction, which leads to a performance degradation, which may be detrimental for performance due to heterogeneity in data. The same reason also applies to the robust gradient aggregation method (Krum) adopted in [8], which also uses only one of the input gradients and discards the rest, giving poor performance.

Robust Mean Estimation. The above limitations of traditional methods motivate us to employ modern tools from high-dimensional robust statistics [37], [39], [40]. In particular, we use the polynomial-time outlier-filtering procedure for high-dimensional robust mean estimation (RME) from [39] for robust gradient aggregation in Algorithm 1. For clear exposition of the ideas behind their algorithm, we use a version of their algorithm as described in Algorithm 2, which is from [43]. The crucial observation in these RME algorithms is that if the empirical mean of the samples is far from their true mean, then the empirical covariance matrix has high largest eigenvalue. So, the idea is to iteratively filter out samples that have large projection on the principal eigenvector of the empirical covariance matrix, and keep on doing it until the largest eigenvalue of the empirical covariance matrix becomes sufficiently small (line 7). This is done via a soft-removal method, where we assign weights (confidence score) to the samples and down-weighting those that have large projection (line 10) – in each iteration t, at least one sample (whose projection $\tau_i^{(t)}$ is the maximum) gets 0 weight. In the end, take the weighted average of the surviving samples.[3]

The RME algorithms overcome most of the above-mentioned limitations of traditional methods, except for that their guarantees are not directly applicable to our setting. This is because the error guarantee of RME algorithms are given in terms of concentration of the good samples around their sample mean, which is easy to bound if good samples come from the *same* distribution. Note that our setup significantly deviates from this, where not only the input samples (which are accumulated gradients) come from *different* distributions (as clients have heterogeneous data), but each of them is also a sum of H stochastic gradients (due to local iterations). Since local iterations cause local parameters to *drift* from each other, bounding the concentration of good samples requires bounding this drift.

To this end, we develop a novel matrix concentration inequality that first shows an existence of a large subset of uncorrupted accumulated stochastic gradients and then bounds their concentration around the sample mean; see (9) in Theorem 3 below. As far as we know, this is the first matrix concentration result in an FL setting.

First we setup the notation. Let Algorithm 1 generate a sequence of iterates $\{\boldsymbol{x}_r^t : t \in [0:T], r \in \mathcal{K}_t\}$ when running with a fixed step-size $\eta \leq \frac{1}{8HL}$, where $\mathcal{K}_t$ denotes the set of K clients that are active at time $t \in [0:T]$. Take any two consecutive synchronization indices $t_k, t_{k+1} \in \mathcal{I}_T$. Note that $|t_{k+1} - t_k| \leq H$. For an honest client $r \in \mathcal{K}_{t_k}$, let $\boldsymbol{g}_{r,\text{accu}}^{t_k,t_{k+1}} := \sum_{t=t_k}^{t_{k+1}-1} \boldsymbol{g}_r(\boldsymbol{x}_r^t)$ denote the sum of local mini-batch stochastic gradients sampled by client r between time

[3]Note that the outlier-filtering procedure described in Algorithm 2 is intuitive and easy to understand. There are better algorithms that are also more efficient and can achieve better guarantees; see, for example, [44]. All these algorithms require the same bounded matrix concentration assumption that we show in Theorem 3, thus making them applicable to use as a subroutine in Algorithm 1 without requiring any modification in our analysis.

Algorithm 2 Robust Accumulated Gradient Estimation (RAGE) [39], [43]

Input: K vectors $\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_K \in \mathbb{R}^d$ such that there is a subset of them $\mathcal{S} \subset [K]$ with $|\mathcal{S}| \geq \frac{2K}{3}$ having bounded covariance $\lambda_{\max}\left(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}(\boldsymbol{g}_i - \boldsymbol{g}_{\mathcal{S}})(\boldsymbol{g}_i - \boldsymbol{g}_{\mathcal{S}})^T\right) \leq \sigma_0^2$, where $\boldsymbol{g}_{\mathcal{S}} = \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\boldsymbol{g}_i$.
For any $\boldsymbol{w} \in [0,1]^K$ with $\|\boldsymbol{w}\|_1 > 0$, define

$$\boldsymbol{\mu}(\boldsymbol{w}) = \sum_{i=1}^{K} \frac{w_i}{\|\boldsymbol{w}\|_1}\boldsymbol{g}_i$$

$$\boldsymbol{\Sigma}(\boldsymbol{w}) = \sum_{i=1}^{K} \frac{w_i}{\|\boldsymbol{w}\|_1}(\boldsymbol{g}_i - \boldsymbol{\mu}(\boldsymbol{w}))(\boldsymbol{g}_i - \boldsymbol{\mu}(\boldsymbol{w}))^T$$

Let $\boldsymbol{w}^{(0)} = [\frac{1}{K}, \ldots, \frac{1}{K}]$ be a length K vector.
Let $C \geq 11$ be a universal constant.
Let $\boldsymbol{\Sigma}^{(0)} = \boldsymbol{\Sigma}(\boldsymbol{w}^{(0)})$.
Let $t = 0$.
while $\lambda_{\max}(\boldsymbol{\Sigma}(\boldsymbol{w}^{(t)})) > C\sigma_0^2$ **do**
  Let $\boldsymbol{v}^{(t)}$ be the principal eigenvector of $\boldsymbol{\Sigma}(\boldsymbol{w}^{(t)})$.
  For $i \in [K]$, define $\tau_i^{(t)} = \left\langle \boldsymbol{v}^{(t)}, \boldsymbol{g}_i - \boldsymbol{\mu}(\boldsymbol{w}^{(t)})\right\rangle^2$.
  For $i \in [K]$, compute $w_i^{(t+1)} = \left(1 - \frac{\tau_i^{(t)}}{\tau_{\max}^{(t)}}\right) w_i^{(t)}$, where $\tau_{\max}^{(t)} = \max_{i:w_i^{(t)}>0} \tau_i^{(t)}$.
  $t = t + 1$
end while
return $\widehat{\boldsymbol{g}} = \sum_{i=1}^{K} \frac{w_i^{(t)}}{\|\boldsymbol{w}^{(t)}\|_1}\boldsymbol{g}_i$.

t_k and t_{k+1}, where $\boldsymbol{g}_r(\boldsymbol{x}_r^t) \in_U \mathcal{F}_r^{\otimes b}(\boldsymbol{x}_r^t)$ satisfies (4), (5). At iteration t_{k+1}, every honest client $r \in \mathcal{K}_{t_k}$ reports its local model $\boldsymbol{x}_r^{t_{k+1}}$ to the server, from which server computes $\boldsymbol{g}_{r,\text{accu}}^{t_k,t_{k+1}}$ (see line 15 of Algorithm 1), whereas, the corrupt clients may report arbitrary and adversarially chosen vectors in $\mathbb{R}^d$. Server does not know the identities of the corrupt clients, and its goal is to produce an estimate $\widehat{\boldsymbol{g}}_{\text{accu}}^{t_k,t_{k+1}}$ of the average accumulated gradients from honest clients as best as possible.

Theorem 3 (Matrix Concentration)**.** *Suppose an ϵ fraction of K clients that communicate with the server are corrupt. In the setting described above, suppose we are given $K \leq R$ accumulated gradients $\widetilde{\boldsymbol{g}}_{r,\text{accu}}^{t_k,t_{k+1}}, r \in \mathcal{K}_{t_k}$ in $\mathbb{R}^d$, where $\widetilde{\boldsymbol{g}}_{r,\text{accu}}^{t_k,t_{k+1}} = \boldsymbol{g}_{r,\text{accu}}^{t_k,t_{k+1}}$ if r'th client is honest, otherwise can be arbitrary. For any $\epsilon' > 0, \gamma > 1/2$, with probability $1 - \exp(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8})$, there exists a subset $\mathcal{S} \subseteq \mathcal{K}_{t_k}$ of* uncorrupted *gradients of size $(1-(\epsilon+\gamma\epsilon'))K$ s.t.*

$$\lambda_{\max}\left(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}(\boldsymbol{g}_i - \boldsymbol{g}_{\mathcal{S}})(\boldsymbol{g}_i - \boldsymbol{g}_{\mathcal{S}})^T\right) \leq \frac{25H^2\sigma^2}{b\epsilon'}\left(1 + \frac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right) + 28H^2\kappa^2, \quad (9)$$

where, for $i \in \mathcal{S}$, $\boldsymbol{g}_i = \boldsymbol{g}_{i,\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{g}_{\mathcal{S}} = \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\boldsymbol{g}_{i,\text{accu}}^{t_k,t_{k+1}}$, and $\lambda_{\max}$ denotes the largest eigenvalue.

Theorem 3 establishes the concentration results required for the RME algorithm (described in Algorithm 2) that we employ in Algorithm 1. This RME algorithm takes a collection of vectors as input, out of which an unknown large subset (at least a $\frac{2}{3}$-fraction) is promised to be well-concentrated around its sample mean, and outputs an estimate of the sample mean. The formal guarantee is given as follows:

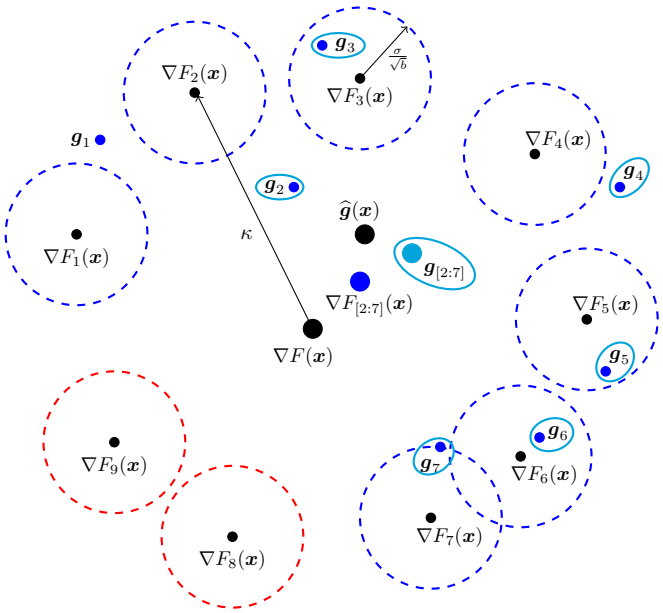


Fig. 2 We have total 9 workers, out of which 2 workers (numbered 8, 9) are Byzantine. Since different workers have different datasets, their true local gradients (denoted by $\nabla F_i(\boldsymbol{x})$) are placed in different locations. The blue dashed circles (numbered 1 to 7) are centered at the true local gradients of honest workers, and have their radius equal to the standard deviation $\sigma/\sqrt{b}$, which implies that their stochastic gradient samples $\boldsymbol{g}_i$ may not lie inside the blue circles. The red dashed circles correspond to the Byzantine workers, and we do not have any control over them. Let $\{\boldsymbol{g}_2, \ldots, \boldsymbol{g}_7\}$ be the subset $\mathcal{S}$ of uncorrupted gradients ensured by the first part of Theorem 3. Let the robust gradient estimator in the second part of Theorem 3 outputs $\widehat{\boldsymbol{g}}(\boldsymbol{x})$ as an estimate of $\boldsymbol{g}_{[2:7]} := \frac{1}{6}\sum_{i=2}^{7}\boldsymbol{g}_i$. To bound the approximation error $\mathbb{E}\|\widehat{\boldsymbol{g}}(\boldsymbol{x}) - \nabla F(\boldsymbol{x})\|$, note that $\mathbb{E}\|\widehat{\boldsymbol{g}}(\boldsymbol{x}) - \nabla F(\boldsymbol{x})\| \leq \mathbb{E}\|\widehat{\boldsymbol{g}}(\boldsymbol{x}) - \boldsymbol{g}_{[2:7]}(\boldsymbol{x})\| + \mathbb{E}\|\boldsymbol{g}_{[2:7]} - \nabla F_{[2:7]}(\boldsymbol{x})\| + \|\nabla F_{[2:7]}(\boldsymbol{x}) - \nabla F(\boldsymbol{x})\|$, where the first term can be bounded by $\mathcal{O}(\sigma_0\sqrt{\epsilon+\epsilon'})$, the second term can be bounded by the square root of $\sigma^2/6b$, which comes from the variance bound for sampling, and the third term can be bounded by κ, which is the gradient dissimilarity bound from (6). Note that the κ term is inevitable because, in the presence of a constant number of Byzantine workers, intuitively, $\nabla F_{[2:7]}(\boldsymbol{x})$ will shift away from $\nabla F(\boldsymbol{x})$ by a constant fraction of κ.

Theorem 4 (Outlier-Filtering Algorithm [39])**.** *Under the same setting and notation of Theorem 3, if $(\epsilon + \gamma\epsilon') \leq \frac{1}{3}$, then we can find an estimate $\widehat{\boldsymbol{g}}$ of $\boldsymbol{g}_{\mathcal{S}}$ in polynomial-time with probability 1, such that $\|\widehat{\boldsymbol{g}} - \boldsymbol{g}_{\mathcal{S}}\| \leq \mathcal{O}\left(\sigma_0\sqrt{\epsilon+\epsilon'}\right)$, where $\sigma_0^2 = \frac{25H^2\sigma^2}{b\epsilon'}\left(1 + \frac{3d}{2K}\right) + 28H^2\kappa^2$.*

Note that, instead of the RME algorithm, if we use med or trimmean, we would get an extra multiplicative factor of $\sqrt{d}$ in the upper-bound on $\|\widehat{\boldsymbol{g}} - \boldsymbol{g}_{\mathcal{S}}\|$ above. This would translate to an extra multiplicative factor of d in the error term $\mathcal{O}\left(\left(\frac{H\sigma^2}{b\epsilon'}\left(1 + \frac{3d}{2K}\right) + H\kappa^2\right)(\epsilon+\epsilon')\right)$ in our approximation error of Theorem 1. Therefore, effectively, we save a factor of d in the approximation error of our convergence results by using RME algorithms for outlier-filtering.

A. Proof-Sketch of Theorem 3 – Matrix Concentration

In order to prove Theorem 3, first we show the following result, which states that if we have m independent distributions

each having bounded variance, and we take one sample from each of them, then there exists a large subset of these samples that has bounded variance as well.

Lemma 1. *Suppose there are m independent distributions $p_1, p_2, \ldots, p_m$ in $\mathbb{R}^d$ such that $\mathbb{E}_{\boldsymbol{y}\sim p_i}[\boldsymbol{y}] = \boldsymbol{\mu}_i, i \in [m]$ and each p_i has a bounded variance in all directions, i.e., $\mathbb{E}_{\boldsymbol{y}\sim p_i}[\langle \boldsymbol{y} - \boldsymbol{\mu}_i, \boldsymbol{v}\rangle^2] \leq \sigma_{p_i}^2, \forall \boldsymbol{v} \in \mathbb{R}^d, \|\boldsymbol{v}\| = 1$. Take any $\epsilon' > 0$ and $\gamma > 1/2$. Then, given m independent samples $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_m$, where $\boldsymbol{y}_i \sim p_i$, with probability $1 - \exp(-\frac{(2\gamma-1)^2\epsilon'^2 m}{8})$, there is a subset $\mathcal{S}$ of $(1-\gamma\epsilon')m$ points such that*

$$\lambda_{\max}\left(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\widetilde{\boldsymbol{y}}_i\widetilde{\boldsymbol{y}}_i^T\right) \leq \frac{4\sigma_{p_{\max}}^2}{\epsilon'}\left(1 + \frac{d}{(1-\gamma\epsilon')m}\right),$$

where $\widetilde{\boldsymbol{y}}_i = \boldsymbol{y}_i - \boldsymbol{\mu}_i$ and $\sigma_{p_{\max}}^2 = \max_{i\in[m]}\sigma_{p_i}^2$.

Lemma 1 is proved in Appendix A.

The important thing to note here is that the m samples come from *different* distributions, which makes it distinct from existing results, such as [45, Proposition B.1], which shows concentration of i.i.d. samples.

Now we give a proof-sketch of Theorem 3 with the help of Lemma 1. A complete proof is provided in Appendix B.

Let $t_k, t_{k+1} \in \mathcal{I}_T$ be any two consecutive synchronization indices. For $i \in \mathcal{K}_{t_k}$ corresponding to an honest client, let $Y_i^{t_k}, Y_i^{t_k+1}, \ldots, Y_i^{t_{k+1}-1}$ be a sequence of $(t_{k+1} - t_k) \leq H$ (dependent) random variables, where for any $t \in [t_k : t_{k+1} - 1]$, the random variable Y_i^t is distributed as

$$Y_i^t \sim \text{Unif}\Big(\mathcal{F}_i^{\otimes b}\big(\boldsymbol{x}_i^t(\boldsymbol{x}_i^{t_k}, Y_i^{t_k}, \ldots, Y_i^{t-1})\big)\Big). \tag{10}$$

Here, Y_i^t corresponds to the mini-batch stochastic gradient sampled from the set $\mathcal{F}_i^{\otimes b}\big(\boldsymbol{x}_i^t(\boldsymbol{x}_i^{t_k}, Y_i^{t_k}, \ldots, Y_i^{t-1})\big)$, which itself depends on the local parameters $\boldsymbol{x}_i^{t_k}$ (which is a deterministic quantity) at the last synchronization index and the past realizations of $Y_i^{t_k}, \ldots, Y_i^{t-1}$. This is because the evolution of local parameters $\boldsymbol{x}_i^t$ depends on $\boldsymbol{x}_i^{t_k}$ and the choice of gradients in between time indices t_k and $t-1$. Now define $Y_i := \sum_{t=t_k}^{t_{k+1}-1} Y_i^t$. Let p_i be the distribution of Y_i, which we will take when using Lemma 1.

It is not hard to show that for any honest client $i \in \mathcal{K}_{t_k}$, we have $\mathbb{E}\|Y_i - \mathbb{E}[Y_i]\|^2 \leq \frac{H^2\sigma^2}{b}$. It is also easy to see that the hypothesis of Lemma 1 is satisfied with $\boldsymbol{\mu}_i = \mathbb{E}[Y_i], \sigma_{p_i}^2 = \frac{H^2\sigma^2}{b}$ for all honest clients $i \in \mathcal{K}_{t_k}$, i.e., we have $\mathbb{E}_{\boldsymbol{y}_i\sim p_i}[\langle \boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i], \boldsymbol{v}\rangle^2] \leq \frac{H^2\sigma^2}{b}, \forall \boldsymbol{v} \in \mathbb{R}^d, \|\boldsymbol{v}\| = 1$.

We are given K different accumulated gradients (each is a summation of H gradients), out of which at least $(1-\epsilon)K$ are according to the correct distribution. By considering only the uncorrupted gradients (i.e., taking $m = (1-\epsilon)K$), we have from Lemma 1 that there exists a subset $\mathcal{S} \subseteq \mathcal{K}_{t_k}$ of size $(1-\gamma\epsilon')(1-\epsilon)K \geq (1-(\epsilon+\gamma\epsilon'))K$ that satisfies (in the following, $\widetilde{\boldsymbol{y}}_i = \boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i]$)

$$\lambda_{\max}\left(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\widetilde{\boldsymbol{y}}_i\widetilde{\boldsymbol{y}}_i^T\right) \leq \widehat{\sigma}_0^2, \tag{11}$$

where $\widehat{\sigma}_0^2 := \frac{4H^2\sigma^2}{b\epsilon'}\left(1 + \frac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right)$.

Note that (11) bounds the deviation of the points in $\mathcal{S}$ from their respective means $\mathbb{E}[\boldsymbol{y}_i]$. However, in (9), we need to bound the deviation of the points in $\mathcal{S}$ from their sample mean $\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\boldsymbol{y}_i$. As it turns out, due to heterogeneity in data and our use of local iterations, this extension is non-trivial and requires some technical work, given next.

From the alternate definition of the largest eigenvalue of symmetric matrices $\mathbf{A} \in \mathbb{R}^{d\times d}$, we have $\lambda_{\max}(\mathbf{A}) = \sup_{\boldsymbol{v}\in\mathbb{R}^d, \|\boldsymbol{v}\|=1} \boldsymbol{v}^T\mathbf{A}\boldsymbol{v}$. With this, (11) is equivalent to

$$\sup_{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|=1} \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i], \boldsymbol{v}\rangle^2 \leq \widehat{\sigma}_0^2. \tag{12}$$

Define $\boldsymbol{y}_{\mathcal{S}} := \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\boldsymbol{y}_i$ to be the sample mean of points in $\mathcal{S}$. Take an arbitrary unit vector $\boldsymbol{v} \in \mathbb{R}^d$. Using some algebraic manipulations provided in Appendix B, we get

$$\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \boldsymbol{y}_i - \boldsymbol{y}_{\mathcal{S}}, \boldsymbol{v}\rangle^2 \leq 6\widehat{\sigma}_0^2 + \frac{4}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\big\|\mathbb{E}[\boldsymbol{y}_j] - \mathbb{E}[\boldsymbol{y}_i]\big\|^2 \tag{13}$$

Using the gradient dissimilarity bound and the L-smoothness of F, we can show that for honest clients $r, s \in \mathcal{K}_{t_k}$, we have $\|\mathbb{E}[\boldsymbol{y}_r] - \mathbb{E}[\boldsymbol{y}_s]\|^2 \leq H\sum_{t=t_k}^{t_{k+1}-1}\left(6\kappa^2 + 3L^2\mathbb{E}\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\|^2\right)$. Using this bound in (13) together with some algebraic manipulations, we get

$$\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \boldsymbol{y}_i - \boldsymbol{y}_{\mathcal{S}}, \boldsymbol{v}\rangle^2 \leq 6\widehat{\sigma}_0^2 + 24H^2\kappa^2 + \frac{12HL^2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\sum_{t=t_k}^{t_{k+1}-1}\mathbb{E}\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\|^2 \tag{14}$$

Now we bound the last term of (14), which is the drift in local parameters at different clients in between any two synchronization indices.

Lemma 2. *If $\eta \leq \frac{1}{8HL}$, we have $\sum_{t=t_k}^{t_{k+1}-1}\mathbb{E}\left\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\right\|^2 \leq 7H^3\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right)$.*

Substituting this in (14) together with some algebraic manipulations provided in Appendix B, we get

$$\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \boldsymbol{y}_i - \boldsymbol{y}_{\mathcal{S}}, \boldsymbol{v}\rangle^2 \leq \frac{25H^2\sigma^2}{b\epsilon'}\left(1 + \frac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right) + 28H^2\kappa^2.$$

Note that this bound holds for all unit vectors $\boldsymbol{v} \in \mathbb{R}^d$. Now substituting $\boldsymbol{g}_{i,\text{accu}}^{t_k,t_{k+1}} = \boldsymbol{y}_i, \boldsymbol{g}_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}} = \boldsymbol{y}_{\mathcal{S}}$ and using the alternate definition of largest eigenvalue proves Theorem 3.

IV. Convergence Proof of the Strongly-Convex Part of Theorem 1

Let $\mathcal{I}_T := \{t_1, t_2, \ldots, t_k, \ldots\}$ with $t_1 = 0$ be the set of synchronization indices at which server selects a subset $\mathcal{K} \subseteq [R]$ of K clients and sends the current global model parameters to them. Upon receiving that, clients in $\mathcal{K}$ performs local SGD steps based on their own local datasets until the next synchronization index, at which they send their local model

parameters to the server. When server has received the updates from clients, it applies the outlier-filtering procedure RAGE (see Algorithm 1) to robustly estimate the average of the uncorrupted accumulated gradients and then updates the global model parameters. We assume that $H = \max_{i\geq 1}(t_{i+1} - t_i)$.

At any iteration $t \in [T]$, let $\mathcal{K}_t \subseteq [R]$ denote the set of clients that are active at time t. Let $\boldsymbol{x}^t := \frac{1}{K}\sum_{r\in\mathcal{K}_t} \boldsymbol{x}_r^t$ denote the average parameter vector of the clients in the active set $\mathcal{K}_t$. Note that, for any $t_i \in \mathcal{I}_T$, the clients in $\mathcal{K}_{t_i}$ remain active at all time indices t such that $t \in [t_i : t_{i+1} - 1]$.

In the following, we denote the decoded gradient at the server at any synchronization time t_{i+1} by $\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}}$, which is an estimate of the average of the accumulated gradients between time t_i and t_{i+1} of the honest clients in $\mathcal{K}_{t_i}$, as in Theorem 3. From Algorithm 1, we can write the parameter update rule for the global model at the synchronization indices as:

$$\boldsymbol{x}^{t_{i+1}} = \boldsymbol{x}^{t_i} - \eta\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}}.$$

Note that at any synchronization index $t_i \in \mathcal{I}_T$, when server selects a subset $\mathcal{K}_{t_i}$ of clients and sends the global parameter vector $\boldsymbol{x}^{t_i}$, all clients in $\mathcal{K}_{t_i}$ set their local model parameters to be equal to the global model parameters, i.e., $\boldsymbol{x}_r^{t_i} = \boldsymbol{x}^{t_i}$ holds for every $r \in \mathcal{K}_{t_i}$.

Now we proceed with proving the strongly-convex part of Theorem 1.

First we derive a recurrence relation for the synchronization indices and then later we extend the proof to all indices. Consider the $(i+1)$'st synchronization index $t_{i+1} \in \mathcal{I}_T$.

$$\begin{aligned}
\boldsymbol{x}^{t_{i+1}} &= \boldsymbol{x}^{t_i} - \eta\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} \\
&= \boldsymbol{x}^{t_i} - \eta\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t) \\
&\quad - \eta\left(\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right)
\end{aligned}$$

For simplicity of notation, define $\mathcal{E} \triangleq \left(\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right)$. Substituting this in the above and using $\boldsymbol{x}^{t_i} = \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\boldsymbol{x}_r^{t_i}$ gives

$$\begin{aligned}
\boldsymbol{x}^{t_{i+1}} &= \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\boldsymbol{x}_r^{t_i} - \eta\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t) - \eta\mathcal{E} \\
&= \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\boldsymbol{x}_r^{t_i} - \eta\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right) - \eta\mathcal{E} \\
&= \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\boldsymbol{x}_r^{t_{i+1}-1} - \eta\nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1})\right) - \eta\mathcal{E} \\
&= \boldsymbol{x}^{t_{i+1}-1} - \eta\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1}) - \eta\mathcal{E} \\
&= \boldsymbol{x}^{t_{i+1}-1} - \eta\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \eta\mathcal{E} \\
&\quad + \eta\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1})\right) \qquad (15)
\end{aligned}$$

Subtracting $\boldsymbol{x}^*$ from both sides gives:

$$\begin{aligned}
\boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^* &= \underbrace{\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^* - \eta\nabla F(\boldsymbol{x}^{t_{i+1}-1})}_{=:\,\boldsymbol{u}} - \eta\mathcal{E} \\
&\quad + \eta\underbrace{\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1})\right)}_{=:\,\boldsymbol{v}} \qquad (16)
\end{aligned}$$

This gives $\boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^* = \boldsymbol{u} + \eta(\boldsymbol{v} - \mathcal{E})$. Taking norm on both sides and then squaring gives

$$\left\|\boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^*\right\|^2 = \|\boldsymbol{u}\|^2 + \eta^2\|\boldsymbol{v} - \mathcal{E}\|^2 + 2\eta\langle\boldsymbol{u}, \boldsymbol{v} - \mathcal{E}\rangle \quad (17)$$

Now we use a simple but powerful trick on inner-products together with the inequality $2\langle\boldsymbol{a},\boldsymbol{b}\rangle \leq \|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2$ and get:

$$\begin{aligned}
2\eta\langle\boldsymbol{u}, \boldsymbol{v} - \mathcal{E}\rangle &= 2\left\langle\sqrt{\frac{\eta\mu}{2}}\boldsymbol{u}, \sqrt{\frac{2\eta}{\mu}}(\boldsymbol{v} - \mathcal{E})\right\rangle \\
&\leq \frac{\eta\mu}{2}\|\boldsymbol{u}\|^2 + \frac{2\eta}{\mu}\|\boldsymbol{v} - \mathcal{E}\|^2 \qquad (18)
\end{aligned}$$

Substituting this back in (17) gives

$$\begin{aligned}
&\left\|\boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^*\right\|^2 \leq \left(1 + \frac{\eta\mu}{2}\right)\|\boldsymbol{u}\|^2 + \eta\left(\eta + \frac{2}{\mu}\right)\|\boldsymbol{v} - \mathcal{E}\|^2 \\
&\leq \left(1 + \frac{\eta\mu}{2}\right)\|\boldsymbol{u}\|^2 + 2\eta\left(\eta + \frac{2}{\mu}\right)\|\boldsymbol{v}\|^2 + 2\eta\left(\eta + \frac{2}{\mu}\right)\|\mathcal{E}\|^2
\end{aligned}$$

Substituting the values of $\boldsymbol{u}, \boldsymbol{v}, \mathcal{E}$ and taking expectation w.r.t. the stochastic sampling of gradients by clients in $\mathcal{K}_{t_i}$ between iterations t_i and t_{i+1} (while conditioning on the past) gives:

$$\begin{aligned}
&\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^*\right\|^2 \\
&\leq \left(1 + \frac{\mu\eta}{2}\right)\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}-1} - \eta\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \boldsymbol{x}^*\right\|^2 \\
&+ 2\eta\left(\eta + \frac{2}{\mu}\right)\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1})\right)\right\|^2 \\
&+ 2\eta\left(\eta + \frac{2}{\mu}\right)\mathbb{E}\left\|\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right\|^2 \qquad (19)
\end{aligned}$$

Now we bound each of the three terms on the RHS of (19) separately in Claim 1, Claim 2, and Claim 3, respectively.

Claim 1. *For $\eta < \frac{1}{L}$, we have*

$$\begin{aligned}
&\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}-1} - \eta\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \boldsymbol{x}^*\right\|^2 \\
&\qquad \leq (1 - \mu\eta)\,\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^*\right\|^2. \qquad (20)
\end{aligned}$$

Claim 2. *For $\eta \leq \frac{1}{8HL}$, we have*

$$\begin{aligned}
&\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1}) - \nabla F(\boldsymbol{x}^{t_{i+1}-1})\right)\right\|^2 \\
&\qquad \leq 2\kappa^2 + \frac{7H}{32}\left(\frac{\sigma^2}{b} + 3\kappa^2\right). \qquad (21)
\end{aligned}$$

Claim 3. *If* $\eta \leq \frac{1}{8HL}$, *then with probability at least* $1 - \exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$, *we have*

$$\mathbb{E}\left\|\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right\|^2 \leq 3\Upsilon^2 + \frac{8H^2\sigma^2}{b} + 30H^2\kappa^2, \quad (22)$$

where $\Upsilon^2 = \mathcal{O}\left(\sigma_0^2(\epsilon+\epsilon')\right)$ *and* $\sigma_0^2 = \frac{25H^2\sigma^2}{b\epsilon'}\left(1+\frac{3d}{2K}\right) + 28H^2\kappa^2$.

Claim 1, Claim 2, and Claim 3 are proved in Appendix C.

Using the bounds from (20), (21), (22) in (19) and using $\left(1+\frac{\mu\eta}{2}\right)(1-\mu\eta) \leq \left(1-\frac{\mu\eta}{2}\right)$ for the first term gives

$$\begin{aligned}
\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^*\right\|^2 &\leq \left(1-\frac{\mu\eta}{2}\right)\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^*\right\|^2 \\
&\quad + 2\eta\left(\eta+\frac{2}{\mu}\right)\left(2\kappa^2 + \frac{7H}{32}\left(\frac{\sigma^2}{b}+3\kappa^2\right)\right) \\
&\quad + 2\eta\left(\eta+\frac{2}{\mu}\right)\left(3\Upsilon^2 + \frac{8H^2\sigma^2}{b} + 30H^2\kappa^2\right) \\
&\leq \left(1-\frac{\mu\eta}{2}\right)\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^*\right\|^2 \\
&\quad + \frac{6\eta}{\mu}\left(3\Upsilon^2 + \frac{9H^2\sigma^2}{b} + 33H^2\kappa^2\right), \quad (23)
\end{aligned}$$

where $\Upsilon^2 = \mathcal{O}\left(\sigma_0^2(\epsilon+\epsilon')\right)$ and $\sigma_0^2 = \frac{25H^2\sigma^2}{b\epsilon'}\left(1+\frac{3d}{2K}\right) + 28H^2\kappa^2$. In the last inequality (23) we used $\eta \leq \frac{1}{8LH} \leq \frac{1}{L} \leq \frac{1}{\mu}$, which implies $(\eta+\frac{2}{\mu}) \leq \frac{3}{\mu}$. Note that (23) holds with probability at least $1 - \exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$.

Note that above recurrence in (23) holds only at the synchronization indices $t_i \in \mathcal{I}_T$ for $i = 1, 2, 3, \ldots$. However, in order to establish a recurrence that we can use to prove convergence, we need to show a recurrence relation for all t. Now we give a recurrence at non-synchronization indices.

Take an arbitrary $t \in [T]$ and let $t_i \in \mathcal{I}_T$ be such that $t \in [t_i : t_{i+1} - 1]$; when $H \geq 2$, such t's exist. Note that $\boldsymbol{x}^t = \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}} \boldsymbol{x}_r^t$.

$$\begin{aligned}
\boldsymbol{x}^{t+1} &= \boldsymbol{x}^t - \eta\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\boldsymbol{g}_r(\boldsymbol{x}_r^t) \\
&= \boldsymbol{x}^t - \eta\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\nabla F_r(\boldsymbol{x}_r^t) \\
&\quad - \eta\Big(\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\boldsymbol{g}_r(\boldsymbol{x}_r^t) - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\nabla F_r(\boldsymbol{x}_r^t)\Big) \\
&= \boldsymbol{x}^t - \eta\nabla F(\boldsymbol{x}^t) + \frac{\eta}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^t) - \nabla F_r(\boldsymbol{x}_r^t)\right) \\
&\quad - \frac{\eta}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\boldsymbol{g}_r(\boldsymbol{x}_r^t) - \nabla F_r(\boldsymbol{x}_r^t)\right) \quad (24)
\end{aligned}$$

Now, subtracting $\boldsymbol{x}^*$ from both sides and following the same steps as in from (16) to (19), we get (in the following, expectation is taken w.r.t. the stochastic sampling of gradients at the t'th iteration while conditioning on the past):

$$\begin{aligned}
\mathbb{E}\left\|\boldsymbol{x}^{t+1} - \boldsymbol{x}^*\right\|^2 &\leq \left(1+\frac{\mu\eta}{2}\right)\mathbb{E}\left\|\boldsymbol{x}^t - \boldsymbol{x}^* - \eta\nabla F(\boldsymbol{x}^t)\right\|^2 \\
&\quad + 2\eta\left(\eta+\frac{2}{\mu}\right)\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^t) - \nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2 \\
&\quad + 2\eta\left(\eta+\frac{2}{\mu}\right)\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\boldsymbol{g}_r(\boldsymbol{x}_r^t) - \nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2 \quad (25)
\end{aligned}$$

We can bound the first two terms on the RHS of (25) using (20) and (21), respectively, as $\mathbb{E}\left\|\boldsymbol{x}^t - \eta\nabla F(\boldsymbol{x}^t) - \boldsymbol{x}^*\right\|^2 \leq (1-\mu\eta)\mathbb{E}\left\|\boldsymbol{x}^t - \boldsymbol{x}^*\right\|^2$ and $\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^t) - \nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2 \leq 2\kappa^2 + \frac{7H}{32}\left(\frac{\sigma^2}{b}+3\kappa^2\right)$. To bound the third term on the RHS of (25), we use the fact that variance of the sum of independent random variables is equal to the sum of the variances and that clients sample stochastic gradients $\boldsymbol{g}_r(\boldsymbol{x}_r^t)$ independent of each other; using this fact and (5), we can bound $\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\boldsymbol{g}_r(\boldsymbol{x}_r^t) - \nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2 \leq \frac{\sigma^2}{bK}$. Substituting these in (25) and using $\left(1+\frac{\mu\eta}{2}\right)(1-\mu\eta) \leq \left(1-\frac{\mu\eta}{2}\right)$ for the first term and $(\eta+\frac{2}{\mu}) \leq \frac{3}{\mu}$ (which follows because $\eta \leq \frac{1}{8HL} \leq \frac{1}{L} \leq \frac{1}{\mu}$) give

$$\begin{aligned}
\mathbb{E}\left\|\boldsymbol{x}^{t+1} - \boldsymbol{x}^*\right\|^2 &\leq \left(1-\frac{\mu\eta}{2}\right)\mathbb{E}\left\|\boldsymbol{x}^t - \boldsymbol{x}^*\right\|^2 \\
&\quad + \frac{6\eta}{\mu}\left(2\kappa^2 + \frac{7H}{32}\left(\frac{\sigma^2}{b}+3\kappa^2\right) + \frac{\sigma^2}{bK}\right) \\
&\leq \left(1-\frac{\mu\eta}{2}\right)\mathbb{E}\left\|\boldsymbol{x}^t - \boldsymbol{x}^*\right\|^2 + \frac{6\eta}{\mu}\left(3H\kappa^2 + \frac{2H\sigma^2}{b}\right) \quad (26)
\end{aligned}$$

Note that (26) holds with probability 1.

Now we have a recurrence at the synchronization indices given in (23) and at non-synchronization indices given in (26). Let $\alpha = \left(1-\frac{\mu\eta}{2}\right)$, $\beta_1 = \left(3\Upsilon^2 + \frac{9H^2\sigma^2}{b} + 33H^2\kappa^2\right)$, and $\beta_2 = \left(3H\kappa^2 + \frac{2H\sigma^2}{b}\right)$. Substituting these and using (23) for the synchronization indices and (26) for the rest of the indices, we get:

$$\begin{aligned}
\mathbb{E}\left\|\boldsymbol{x}^T - \boldsymbol{x}^*\right\|^2 &\leq \alpha^T\left\|\boldsymbol{x}^0 - \boldsymbol{x}^*\right\|^2 + \frac{6\eta}{\mu}\left(\sum_{i=0}^{T/H}\sum_{j=1}^{H-1}\alpha^{iH+j}\beta_2 + \sum_{i=0}^{T/H}\alpha^{iH}\beta_1\right) \quad (27) \\
&\leq \alpha^T\left\|\boldsymbol{x}^0 - \boldsymbol{x}^*\right\|^2 + \frac{6\eta}{\mu}\left(\sum_{i=0}^{\infty}\alpha^i\beta_2 + \sum_{i=0}^{\infty}\alpha^{iH}\beta_1\right) \\
&= \alpha^T\left\|\boldsymbol{x}^0 - \boldsymbol{x}^*\right\|^2 + \frac{6\eta}{\mu}\left(\frac{1}{1-\alpha}\beta_2 + \frac{1}{1-\alpha^H}\beta_1\right) \quad (28)
\end{aligned}$$

Since $\alpha = \left(1-\frac{\mu\eta}{2}\right)$, we have $\alpha^H = \left(1-\frac{\mu\eta}{2}\right)^H \overset{(a)}{\leq} \exp(-\frac{\mu\eta H}{2}) \overset{(b)}{\leq} 1 - \frac{\mu\eta H}{2} + \left(\frac{\mu\eta H}{2}\right)^2 \overset{(c)}{\leq} 1 - \frac{\mu\eta H}{2} + \frac{1}{16}\frac{\mu\eta H}{2} = 1 - \frac{15}{16}\frac{\mu\eta H}{2}$. In (a) we used the inequality $(1-\frac{1}{x})^x \leq \frac{1}{e}$ which

holds for any $x > 0$; in (b) we used $\exp(-x) \leq 1 - x + x^2$ which holds for any $x \geq 0$; in (c) we used $\eta \leq \frac{1}{8HL}$ and $\mu \leq L$, which together imply $\frac{\mu\eta H}{2} \leq \frac{1}{16}$. Substituting these in (28) gives

$$\begin{aligned}
&\mathbb{E}\left\|\boldsymbol{x}^T - \boldsymbol{x}^*\right\|^2 \\
&\leq \left(1 - \frac{\mu\eta}{2}\right)^T \left\|\boldsymbol{x}^0 - \boldsymbol{x}^*\right\|^2 + \frac{6\eta}{\mu}\left(\frac{2}{\mu\eta}\beta_2 + \frac{32}{15\mu\eta H}\beta_1\right) \\
&\leq \left(1 - \frac{\mu\eta}{2}\right)^T \left\|\boldsymbol{x}^0 - \boldsymbol{x}^*\right\|^2 + \frac{6\times 32}{15\mu^2}\left(\frac{15}{16}\beta_2 + \frac{1}{H}\beta_1\right) \\
&\leq \left(1 - \frac{\mu\eta}{2}\right)^T \left\|\boldsymbol{x}^0 - \boldsymbol{x}^*\right\|^2 + \frac{13}{\mu^2}\left(\frac{3\Upsilon^2}{H} + \frac{11H\sigma^2}{b} + 36H\kappa^2\right)
\end{aligned} \tag{29}$$

Note that the last term on the RHS of (29) is independent of η, which together with the dependence of η on the first term implies that bigger the η, faster the convergence. Since we need $\eta \leq \frac{1}{8HL}$ for Claim 2 and Claim 3 to hold, we choose $\eta = \frac{1}{8HL}$. Substituting this in (29) yields the convergence rate in the strongly-convex part of Theorem 1.

Error Probability Analysis. Note that (23) holds with probability at least $1 - \exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$ and (26) holds with probability 1. Since to arrive at (27) (which leads to our final bound (29)), we used (23) $\frac{T}{H}$ times and (26) $\left(T - \frac{T}{H}\right)$ times; as a consequence, by union bound, we have that (29) holds with probability at least $1 - \frac{T}{H}\exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$, which is at least $(1-\delta)$, for any $\delta > 0$, provided we run our algorithm for at most $T \leq \delta H \exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$ iterations.

This concludes the proof of the strongly-convex part of Theorem 1.

V. Convergence Proof of the Non-Convex Part of Theorem 1

Let $\mathcal{K}_t \subseteq [R]$ denote the subset of clients of size $|\mathcal{K}_t| = K$ sampled at the t'th iteration. For any $t \in [t_i : t_{i+1} - 1]$, let $\boldsymbol{x}^t = \frac{1}{K}\sum_{k\in\mathcal{K}_{t_i}} \boldsymbol{x}_k^t$ denote the average of the local parameters of clients in the sampling set $\mathcal{K}_{t_i}$.

Similar to the proof given in Section IV for the strongly-convex part of Theorem 1, here also, first we derive a recurrence for the synchronization indices and then for non-synchronization indices.

For the synchronization indices $t_1, t_2, \ldots, t_k, \ldots \in \mathcal{I}_T$, from (15), we have

$$\boldsymbol{x}^{t_{i+1}} = \boldsymbol{x}^{t_{i+1}-1} - \eta\nabla F(\boldsymbol{x}^{t_{i+1}-1}) + \eta C \tag{30}$$

where

$$\begin{aligned}
C = &\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1})\right) \\
&- \left(\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right).
\end{aligned} \tag{31}$$

Now, using the definition of L-smoothness in (30), we have

$$\begin{aligned}
&F(\boldsymbol{x}^{t_{i+1}}) \\
&\leq F(\boldsymbol{x}^{t_{i+1}-1}) + \left\langle\nabla F(\boldsymbol{x}^{t_{i+1}-1}), \boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^{t_{i+1}-1}\right\rangle \\
&\qquad + \frac{L}{2}\left\|\boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^{t_{i+1}-1}\right\|^2 \\
&= F(\boldsymbol{x}^{t_{i+1}-1}) - \eta\left\langle\nabla F(\boldsymbol{x}^{t_{i+1}-1}), \nabla F(\boldsymbol{x}^{t_{i+1}-1}) - C\right\rangle \\
&\qquad + \frac{\eta^2 L}{2}\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - C\right\|^2 \\
&= F(\boldsymbol{x}^{t_{i+1}-1}) - \eta\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2 + \eta\left\langle\nabla F(\boldsymbol{x}^{t_{i+1}-1}), C\right\rangle \\
&\qquad + \frac{\eta^2 L}{2}\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - C\right\|^2 \\
&\overset{(a)}{\leq} F(\boldsymbol{x}^{t_{i+1}-1}) - \eta\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2 \\
&\qquad + \eta\left(\frac{\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2}{4} + \|C\|^2\right) \\
&\qquad + \frac{\eta^2 L}{2}\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - C\right\|^2 \\
&\overset{(b)}{\leq} F(\boldsymbol{x}^{t_{i+1}-1}) - \frac{3\eta}{4}\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2 + \eta\|C\|^2 \\
&\qquad + \eta^2 L\left(\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2 + \|C\|^2\right) \\
&= F(\boldsymbol{x}^{t_{i+1}-1}) - \eta\left(\frac{3}{4} - \eta L\right)\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2 \\
&\qquad + \eta\left(1 + \eta L\right)\|C\|^2
\end{aligned} \tag{32}$$

In (a), we used the inequality $2\langle\boldsymbol{a},\boldsymbol{b}\rangle \leq \tau\|\boldsymbol{a}\|^2 + \frac{1}{\tau}\|\boldsymbol{b}\|^2$, which holds for every $\tau > 0$, and we used $\tau = \frac{1}{2}$ in (a). In (b), we used the inequality $\|\boldsymbol{a}+\boldsymbol{b}\|^2 \leq 2(\|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2)$. For $\eta \leq \frac{1}{8HL} \leq \frac{1}{8L}$, we have $(3/4 - \eta L) \geq 1/2$ and $(1 + \eta L) \leq \frac{9}{8}$. Substituting these in (32) and taking expectation w.r.t. the stochastic sampling of gradients at clients in $\mathcal{K}_{i_t}$ between iterations t_i and t_{i+1} (while conditioning on the past) gives:

$$\begin{aligned}
\mathbb{E}[F(\boldsymbol{x}^{t_{i+1}})] \leq \mathbb{E}[F(\boldsymbol{x}^{t_{i+1}-1})] - \frac{\eta}{2}\mathbb{E}\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2 \\
+ \frac{9\eta}{8}\mathbb{E}\|C\|^2.
\end{aligned} \tag{33}$$

Now we bound $\mathbb{E}\|C\|^2$. Substituting the value of C from (31) gives:

$$\begin{aligned}
\mathbb{E}\|C\|^2 &\leq 2\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1})\right)\right\|^2 \\
&\quad + 2\mathbb{E}\left\|\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right\|^2 \\
&\leq 2\left(2\kappa^2 + \frac{7H}{32}\left(\frac{\sigma^2}{b} + 3\kappa^2\right)\right) \\
&\quad + 2\left(3\Upsilon^2 + \frac{8H^2\sigma^2}{b} + 30H^2\kappa^2\right) \\
&\leq 2\left(3\Upsilon^2 + \frac{9H^2\sigma^2}{b} + 33H^2\kappa^2\right)
\end{aligned} \tag{34}$$

Here, the first inequality used $\|\boldsymbol{a}+\boldsymbol{b}\|^2 \leq 2(\|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2)$ and the second inequality used the bounds from (21) and (22).

Substituting the bound from (34) into (33) gives

$$\mathbb{E}[F(\boldsymbol{x}^{t_{i+1}})] \leq \mathbb{E}[F(\boldsymbol{x}^{t_{i+1}-1})] - \frac{\eta}{2}\mathbb{E}\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2$$

$$+\frac{9\eta}{4}\left(3\Upsilon^2+\frac{9H^2\sigma^2}{b}+33H^2\kappa^2\right) \quad (35)$$

where $\Upsilon^2=\mathcal{O}\left(\sigma_0^2(\epsilon+\epsilon')\right)$ and $\sigma_0^2=\frac{25H^2\sigma^2}{b\epsilon'}\left(1+\frac{3d}{2K}\right)+28H^2\kappa^2$. Note that (35) holds with probability at least $1-\exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$.

Note that the above recurrence in (35) holds only at the synchronization indices $t_i\in\mathcal{I}_T$ for $i=1,2,3,\ldots$. Now we give a recurrence at non-synchronization indices.

We have done a similar calculation in the proof of the strongly-convex part of Theorem 1. Take an arbitrary $t\in[T]$ and let $t_i\in\mathcal{I}_T$ be such that $t\in[t_i:t_{i+1}-1]$; when $H\geq 2$, such t's exist. Note that $\boldsymbol{x}^t=\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\boldsymbol{x}_r^t$.

From (24), we have $\boldsymbol{x}^{t+1}=\boldsymbol{x}^t-\eta\nabla F(\boldsymbol{x}^t)+\eta D$, where

$$D=\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^t)-\nabla F_r(\boldsymbol{x}_r^t)\right)-\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\boldsymbol{g}_r(\boldsymbol{x}_r^t)-\nabla F_r(\boldsymbol{x}_r^t)\right).$$

Using L-smoothness of F, and then performing similar algebraic manipulations that we used in order to arrive at (33), we get:

$$\mathbb{E}[F(\boldsymbol{x}^{t+1})]\leq\mathbb{E}[F(\boldsymbol{x}^t)]-\frac{\eta}{2}\mathbb{E}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2+\frac{9\eta}{8}\mathbb{E}\|D\|^2 \quad (36)$$

Now we bound $\mathbb{E}\|D\|^2$:

$$\begin{aligned}\mathbb{E}\|D\|^2&\leq 2\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^t)-\nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2\\&\quad+2\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\boldsymbol{g}_r(\boldsymbol{x}_r^t)-\nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2\\&\leq 2\left(2\kappa^2+\frac{7H}{32}\left(\frac{\sigma^2}{b}+3\kappa^2\right)+\frac{\sigma^2}{bK}\right)\\&\leq 2\left(3H\kappa^2+\frac{2H\sigma^2}{b}\right) \quad (37)\end{aligned}$$

Here, the second inequality used the same bounds on both the quantities on the RHS of the first inequality that we used to go from (25) to (26).

Substituting the bound on $\mathbb{E}\|D\|^2$ from (37) into (36) gives

$$\mathbb{E}[F(\boldsymbol{x}^{t+1})]\leq\mathbb{E}[F(\boldsymbol{x}^t)]-\frac{\eta}{2}\mathbb{E}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2+\frac{9\eta}{4}\left(3H\kappa^2+\frac{2H\sigma^2}{b}\right) \quad (38)$$

Note that (38) holds with probability 1.

Now we have a recurrence at synchronization indices given in (35) and at non-synchronization indices given in (38). Adding (35) and (38) from $t=0$ to T (use (35) for the synchronization indices and (38) for the rest of the indices) gives:

$$\sum_{t=0}^{T}\mathbb{E}[F(\boldsymbol{x}^{t+1})]\leq\sum_{t=0}^{T}\mathbb{E}[F(\boldsymbol{x}^t)]-\frac{\eta}{2}\sum_{t=0}^{T}\mathbb{E}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2+\frac{9\eta}{4}\left[\frac{T}{H}\left(3\Upsilon^2+\frac{9H^2\sigma^2}{b}+33H^2\kappa^2\right)+\left(T-\frac{T}{H}\right)\left(3H\kappa^2+\frac{2H\sigma^2}{b}\right)\right] \quad (39)$$

Since $\left(T-\frac{T}{H}\right)\leq T$, we can upper-bound the last term by $\frac{9\eta T}{4}\left(\frac{3\Upsilon^2}{H}+\frac{11H\sigma^2}{b}+36H\kappa^2\right)$. Substituting this in (39) and then rearranging, we get:

$$\frac{1}{T}\sum_{t=0}^{T}\mathbb{E}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2\leq\frac{2}{\eta T}\left[\mathbb{E}[F(\boldsymbol{x}^0)]-\mathbb{E}[F(\boldsymbol{x}^{T+1})]\right]+\frac{9}{2}\left(\frac{3\Upsilon^2}{H}+\frac{11H\sigma^2}{b}+36H\kappa^2\right) \quad (40)$$

Note that the last term in (40) is a constant. So, it would be best to take the step-size η to be as large as possible such that it satisfies $\eta\leq\frac{1}{8HL}$. We take $\eta=\frac{1}{8HL}$. Substituting this in (40) and using $F(\boldsymbol{x}^{T+1})\geq F(\boldsymbol{x}^*)$ gives

$$\frac{1}{T}\sum_{t=0}^{T}\mathbb{E}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2\leq\frac{16HL}{T}\left[\mathbb{E}[F(\boldsymbol{x}^0)]-\mathbb{E}[F(\boldsymbol{x}^*)]\right]+\frac{9}{2}\left(\frac{3\Upsilon^2}{H}+\frac{11H\sigma^2}{b}+36H\kappa^2\right), \quad (41)$$

where $\Upsilon^2=\mathcal{O}\left(\sigma_0^2(\epsilon+\epsilon')\right)$ and $\sigma_0^2=\frac{25H^2\sigma^2}{b\epsilon'}\left(1+\frac{3d}{2K}\right)+28H^2\kappa^2$. Note that (41) is the convergence rate in the non-convex part of Theorem 1.

Error Probability Analysis. Note that (35) holds with probability at least $1-\exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$ and (38) holds with probability 1. Since to arrive at (39) (which leads to our final bound (41)), we used (35) $\frac{T}{H}$ times and (38) $\left(T-\frac{T}{H}\right)$ times; as a consequence, by union bound, we have that (41) holds with probability at least $1-\frac{T}{H}\exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$, which is at least $(1-\delta)$, for any $\delta>0$, provided we run our algorithm for at most $T\leq\delta H\exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$ iterations.

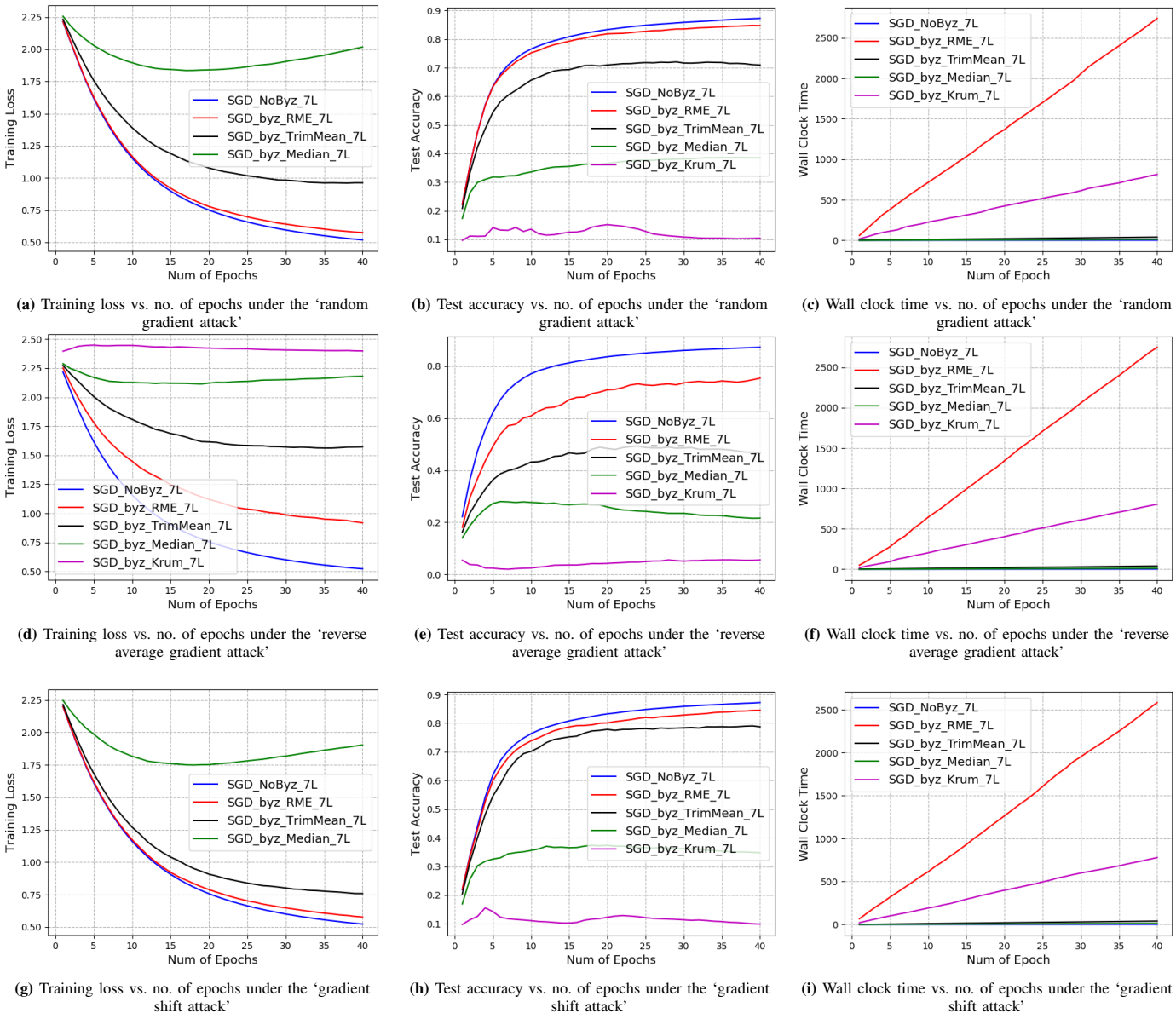
This concludes the proof of the non-convex part of Theorem 1.

VI. Experiments

In this section, we present numerical results on a non-convex objective. Additional implementation details can be found in Appendix F.

Setup. We train a single layer neural network for image classification on the MNIST handwritten digit (from 0-9) dataset. The hidden layer has 25 nodes with ReLU activation function and the output has softmax function. The dimension of the model parameter vector is $19,885$.[4] All clients compute stochastic gradients on a batch-size of 128 in each iteration and communicate the local parameter vectors with the server after

[4] $784\times 25=19,600$ weights between the input and the first layer, 25 bias terms (one for each node in the hidden layer), $25\times 10=250$ weights between the first layer and the output layer, and 10 bias terms (one for each node in the output layer).



(a) Training loss vs. no. of epochs under the 'random gradient attack'

(b) Test accuracy vs. no. of epochs under the 'random gradient attack'

(c) Wall clock time vs. no. of epochs under the 'random gradient attack'

(d) Training loss vs. no. of epochs under the 'reverse average gradient attack'

(e) Test accuracy vs. no. of epochs under the 'reverse average gradient attack'

(f) Wall clock time vs. no. of epochs under the 'reverse average gradient attack'

(g) Training loss vs. no. of epochs under the 'gradient shift attack'

(h) Test accuracy vs. no. of epochs under the 'gradient shift attack'

(i) Wall clock time vs. no. of epochs under the 'gradient shift attack'

Fig. 3 We compare the performance of our method (red) against three methods for robust gradient aggregation, namely, coordinate-wise trimmed-mean (black), coordinate-wise median (green), and Krum (magenta) under three adversarial attacks (**A.1, A.2, A.3**), and plot training loss, test accuracy, and wall clock time against number of epochs. The plot in blue corresponds to running Algorithm 1 with no adversaries and no decoding. In the legends, $7L$ denotes that we are taking $H = 7$ local iterations. See also Footnotes 6, 7, 8.

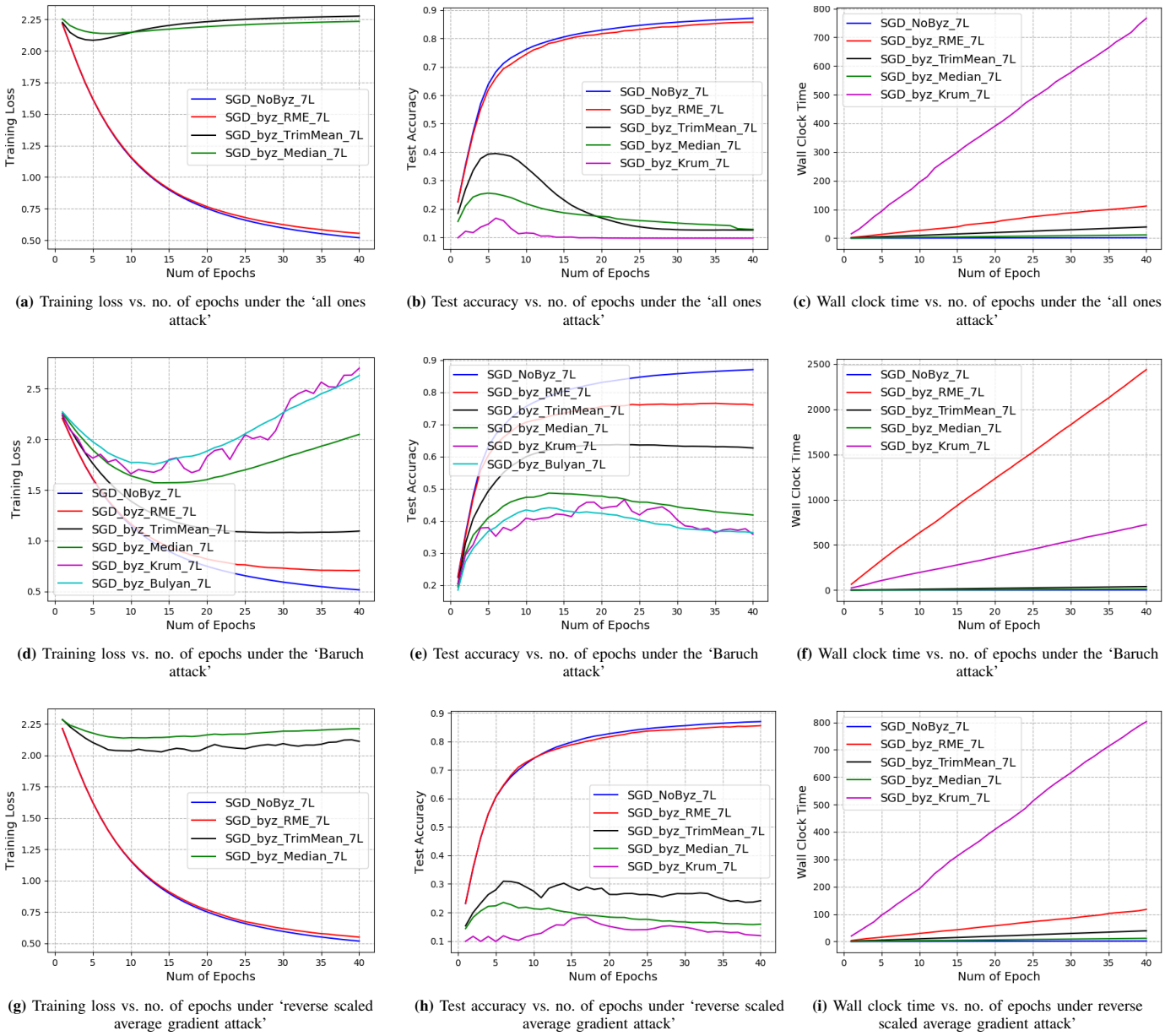
taking $H = 7$ local iterations. For all the defense mechanisms, we start with a step-size $\eta = 0.08$ and decrease its learning rate by a factor of 0.96 when the difference in the corresponding test accuracies in the last 2 consecutive epochs is less than 0.001.

Heterogeneous Datasets. The MNIST dataset has $60,000$ training images (with 6000 images of each label) and $10,000$ test images (each having $28 \times 28 = 784$ pixels), and is distributed among the 200 clients in the following *heterogeneous* manner: Each client takes a random permutation of the probability vector $[0.8, 0.1, 0.1, 0, 0, 0, 0, 0, 0, 0]$. Suppose it obtains a vector $\boldsymbol{p}$ such that $p_i = 0.8, p_j = 0.1, p_k = 0.1$ for some distinct $i, j, k \in [0:9]$ and $p_l = 0$ for the rest of the indices, then it selects *uniformly at random* $800, 100, 100$ training images with label i, j, k, respectively.

Adversarial Attacks. We have 12.5% adversarial clients, i.e., 25 out of 200 clients are corrupt, and the corrupt set of clients may change in every iteration. We implement six adversarial attacks:

- **A.1** the 'random gradient attack', where local gradients at clients are replaced by independent Gaussian random vectors having the same norm[5] as the corresponding gradients;
- **A.2** the 'reverse average gradient attack', where corrupt clients send -ve of their average local gradients;
- **A.3** the 'gradient shift attack', where local gradients of corrupt clients are shifted by a scaled (by factor of 50) Gaussian

[5]Note that changing the direction while keeping the norm same is among the worst attacks as the corrupt gradients cannot be filtered out just based on their norms.



(a) Training loss vs. no. of epochs under the 'all ones attack'

(b) Test accuracy vs. no. of epochs under the 'all ones attack'

(c) Wall clock time vs. no. of epochs under the 'all ones attack'

(d) Training loss vs. no. of epochs under the 'Baruch attack'

(e) Test accuracy vs. no. of epochs under the 'Baruch attack'

(f) Wall clock time vs. no. of epochs under the 'Baruch attack'

(g) Training loss vs. no. of epochs under 'reverse scaled average gradient attack'

(h) Test accuracy vs. no. of epochs under 'reverse scaled average gradient attack'

(i) Wall clock time vs. no. of epochs under reverse scaled average gradient attack'

Fig. 4 We compare the performance of our method (red) against four methods for robust gradient aggregation, namely, coordinate-wise trimmed-mean (black), coordinate-wise median (green), Krum (magenta), and Bulyan (cyan) under three adversarial attacks (**A.4, A.5, A.6**), and plot training loss, test accuracy, and wall clock time against number of epochs. The plot in blue corresponds to running Algorithm 1 with no adversaries and no decoding. In the legends, $7L$ denotes that we are taking $H = 7$ local iterations. See also Footnotes 6, 7, 8.

random vector (same for all);

A.4 the 'all ones attack', where gradients of the corrupt clients are replaced by the all ones vector;

A.5 the 'Baruch attack', which was designed in [46] specifically for coordinate-wise trimmed mean (trimmean) [18], Krum [8], and Bulyan [47] defenses; and

A.6 the 'reverse scaled average gradient attack', where corrupt clients compute the -ve of their average local gradients, scale it by the factor of 50, and then send it.

We train our neural network under all the above-described adversarial attacks, and demonstrate in Figure 3 and Figure 4 the performance of our method (red color) against four other methods for robust gradient aggregation, namely, *coordinate-wise trimmed-mean* (black color) and *coordinate-wise median* (green color), which were used in [18], [22], [36], Krum (magenta color), which was proposed in [8], and Bulyan (cyan color), which was proposed in [47]. For reference, we also plot (in blue color) the performance of Algorithm 1 with the same setup as above but without adversaries and with no decoding. For each attack, we plot three curves, one for number of epochs vs. training loss, one for number of epochs vs. test accuracy, and one for number of epochs vs. wall clock time.

Performance (training loss and test accuracy vs. number of epochs). In Figures 3a, 3d, 3g, and Figures 4a, 4d, 4g we compare training loss vs. number of epochs and in Figures 3b, 3e, 3h, and Figures 4b, 4e, 4h we compare test

accuracy vs. number of epochs of our method against the previously mentioned methods under all six adversarial attacks that we have implemented.[6] In particular, for attacks **A.1, A.3, A.4, A.6**, our method (with adversaries) achieves *similar* performance for both training loss and test accuracy as that of running SGD with local iterations but *without* any adversaries and defense mechanism at the server; and for attacks **A.2, A.5**, the performance difference (test accuracy) is around 0.1 at epoch 40, which is still significantly better than all other methods.[7] This conforms to the inadequacy of using these methods in our setting, as described in Section III. Note that the experiments presented in [18], [36] only implemented a benign 'label-flipping' attack, which is a data poisoning attack. This is not a dynamic attack as, unlike gradient attacks, it does not adapt to the learning process over iterations. In contrast, in all our attacks, corrupt clients send adversarial gradients in *every iteration*, making them significantly more malicious than just flipping the labels. As we have mentioned in the related work (on page 2), and we want to emphasize again, that though [36] also studied the same problem as ours, but employed 'coordinate-wise trimmed mean' for robust gradient aggregation, their convergence bound, in our opinion, are vacuous, as the sub-optimality gap in their bounds *always* scales linearly with the diameter of the parameter space. As far as we know, ours is the first theoretical result that combines Byzantine-resilience with local iterations for high-dimensional distributed training on heterogeneous datasets with good empirical performance.

Performance (wall clock time vs. number of epochs). In Figures 3c, 3f, 3i, and Figures 4c, 4f, 4i, we compare wall clock time (i.e., the total time taken by each algorithm over 40 epochs) vs. number of epochs of our method against the previously mentioned methods under all adversarial attacks that we have implemented.[8] It can be seen that, unlike all other methods, the time taken by our method (red in color) changes depending on the attack. This is because our filtering is an iterative method, and in some attacks, it filters out bad updates in much fewer iterations than other attacks. For example, in **A.4, A.6**, our filtering method takes about 7-8$\times$ *less* time than Krum, whereas, in **A.1, A.2, A.3, A.5**, our method takes about $3\times$ more time than Krum.

As mentioned in Appendix F and we would like to emphasize that here, that since we run SVD on the matrix formed by the *same* 1024 randomly chosen coordinates from all update vectors, our decoding algorithm's run-time still has linear dependence on d, because SVD run time is fixed and is independent of d. In contrast, any coordinate-wise decoding algorithm (such as, median or trimmed-mean) do necessarily have to run the algorithm in all d coordinates. Therefore, in large-scale problems, our modified decoding algorithm would be on par with coordinate-wise trimmed-mean and coordinate-wise median, and significantly better than Krum and Bulyan.

[6]We found out that the Bulyan defense mechanism is significantly slower than all other mechanisms. Due to this, we only implemented this for the Baruch-attack, which was specifically designed against Krum/Bulyan algorithms. Since a basic building block of Bulyan is Krum, and Krum performs the worst among all the mechanisms that we implemented, we do not expect Bulyan to perform significantly better than Krum in other attacks as well – note that both Krum and Bulyan are the worst performing defense mechanisms against the Baruch-attack.

[7]We plot the Krum performance in the training loss vs. number of epochs figures only for the attacks **A.2, A.5**; because in all other attacks, the Krum training loss became very high (above 100) even before epoch 40 and would have prevented observing other methods' performance if we had plotted it.

[8]The wall clock time of Bulyan was significantly higher in comparison to all other methods, hence we skipped plotting the wall clock time of Bulyan, as otherwise it would have prevented observing other methods' performance if we had plotted it.

VII. Bounding the Local Variances and Gradient Dissimilarity in the Statistical Heterogeneous Model

In this section, we bound the gradient dissimilarity κ^2 (from (6)) and local variance σ^2 (from (2)) in the statistical model in heterogeneous setting, where different workers may have local data generated from potentially different distributions. The purpose of this section is to provide upper bounds on κ and σ in the statistical model.

Let $q_1, q_2, \ldots, q_R$ denote the R probability distributions from which the local data samples at the workers are drawn. Specifically, the data samples at any worker r are drawn from q_r in an i.i.d. fashion and independently from other workers. For $r \in [R]$, let $\mathcal{Q}_r$ denote the alphabet over which q_r is distributed. For $r \in [R]$, let $f_r : \mathcal{Q}_r \times \mathcal{C} \to \mathbb{R}$ denote the local loss function at worker r, where $f_r(\boldsymbol{z}, \boldsymbol{x})$ is the loss associated with the sample $\boldsymbol{z} \in \mathcal{Q}_r$ w.r.t. the model parameters $\boldsymbol{x} \in \mathcal{C} \subseteq \mathbb{R}^d$. Linear regression is a classic example of this, where, if $\boldsymbol{z} = (\boldsymbol{w}, y)$ denote the pair of a feature vector $\boldsymbol{w} \in \mathbb{R}^d$ and the response $y \in \mathbb{R}$, then $f_r(\boldsymbol{z}, \boldsymbol{x}) = \frac{1}{2}(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - y)^2$. For each worker $r \in [R]$, we assume that for any fixed $\boldsymbol{z} \in \mathcal{Q}_r$, the local loss function $f_r(\boldsymbol{z}, \boldsymbol{x})$ is L-smooth w.r.t. $\boldsymbol{x}$, i.e., for any $\boldsymbol{z} \in \mathcal{Q}_r$, we have $\|\nabla f_r(\boldsymbol{z}, \boldsymbol{x}) - \nabla f_r(\boldsymbol{z}, \boldsymbol{y})\| \leq L\|\boldsymbol{x} - \boldsymbol{y}\|, \forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}$.

Let $\mu_r(\boldsymbol{x}) := \mathbb{E}_{\boldsymbol{z} \sim q_r}[f_r(\boldsymbol{z}, \boldsymbol{x})]$ denote the expected value of $f_r(\boldsymbol{z}, \boldsymbol{x})$, when $\boldsymbol{z}$ is sampled from $\mathcal{Q}_r$ according to q_r. For any $\boldsymbol{x} \in \mathcal{C}$, let $\mu(\boldsymbol{x}) := \frac{1}{R}\sum_{r=1}^{R} \mu_r(\boldsymbol{x})$ denote the average value of $\mu_r(\boldsymbol{x}), r \in [R]$.

We are given n_r i.i.d. samples $\boldsymbol{z}_{r,1}, \boldsymbol{z}_{r,2}, \ldots, \boldsymbol{z}_{r,n_r}$ at the r'th worker from q_r. Fix an arbitrary parameter vector $\boldsymbol{x} \in \mathcal{C}$. Let $\bar{f}_r(\boldsymbol{x}) := \frac{1}{n_r}\sum_{i=1}^{n_r} f_r(\boldsymbol{z}_{r,i}, \boldsymbol{x})$ denote the average loss at worker r on the n_r samples $\boldsymbol{z}_{r,1}, \ldots, \boldsymbol{z}_{r,n_r}$ w.r.t. $\boldsymbol{x}$. Let $\bar{f}(\boldsymbol{x}) := \frac{1}{R}\sum_{r=1}^{r} \bar{f}_r(\boldsymbol{x})$ denote the average loss across all workers. The analogues of (6) and (2) in this statistical heterogeneous model are the following:

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \bar{f}(\boldsymbol{x})\right\|^2 \leq \kappa^2, \quad \forall \boldsymbol{x} \in \mathcal{C}, \quad (42)$$

$$\mathbb{E}_{i \in_U [n_r]} \left\|\nabla f_r(\boldsymbol{z}_{r,i}, \boldsymbol{x}) - \nabla \bar{f}_r(\boldsymbol{x})\right\|^2 \leq \sigma^2, \quad \forall \boldsymbol{x} \in \mathcal{C}. \quad (43)$$

We need to find good upper bounds on κ and σ that hold for all $r \in [R], \boldsymbol{x} \in \mathcal{C}$ with high probability. We provide two bounds on κ, one when the local gradients at workers are assumed to be sub-exponential random vectors, and other when they are sub-Gaussian random vectors. We provide a bound on σ assuming that the local gradients are sub-Gaussian random vectors. These are standard assumptions on gradients in statistical models, where data at all workers are sampled from the *same* distribution in an i.i.d. fashion [17], [20], [22], which is in contrast to our heterogeneous data setting,

where data at different workers may be sampled from *different* distributions. Note that these works minimize the *population risk* with *full batch* gradient descent, whereas, we minimize the *empirical risk* with *stochastic* gradient descent. In particular, [17] and [20] make sub-exponential gradient assumption and give convergence guarantees only for strong-convex objectives. On the other hand, [22] gives convergence guarantees for non-convex objectives, but under a stricter condition of sub-Gaussian distribution on gradients. In this paper, we provide convergence guarantees for both strongly-convex and non-convex objectives. Moreover, as opposed to [17], [20], [22], our results are in a more general heterogeneous data model. Note that we need sub-Gaussian assumption only to bound the variance, which occurs because workers sample stochastic gradients. In case of full batch gradient descent, we only need sub-exponential assumption, as the variance is zero.

Now we state the distributional assumptions on local gradients.

Assumption 3 (Sub-exponential local gradients)**.** *For every $\boldsymbol{x} \in \mathcal{C}$, the local gradient vectors at any worker $r \in [R]$ are sub-exponential random vectors, i.e., there exist non-negative parameters (ν, α) such that*

$$\sup_{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|=1} \mathbb{E}_{\boldsymbol{z}\sim q_r}\left[\exp\left(\lambda\left\langle\nabla f_r(\boldsymbol{z},\boldsymbol{x})-\nabla\mu_r(\boldsymbol{x}),\boldsymbol{v}\right\rangle\right)\right] \leq \exp\left(\lambda^2\nu^2/2\right), \qquad \forall|\lambda|<\frac{1}{\alpha}. \tag{44}$$

Assumption 4 (Sub-Gaussian local gradients)**.** *For every $\boldsymbol{x} \in \mathcal{C}$, the local gradient vectors at any worker $r \in [R]$ are sub-Gaussian random vectors, i.e., there exists a non-negative parameter σ_g such that*

$$\sup_{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|=1} \mathbb{E}_{\boldsymbol{z}\sim q_r}\left[\exp\left(\lambda\left\langle\nabla f_r(\boldsymbol{z},\boldsymbol{x})-\nabla\mu_r(\boldsymbol{x}),\boldsymbol{v}\right\rangle\right)\right] \leq \exp\left(\lambda^2\sigma_g^2/2\right), \qquad \forall\lambda\in\mathbb{R}. \tag{45}$$

Though, as stated above in both the assumptions, local gradients at all workers have the same parameters ((ν, α) for sub-exponential and σ_{g} for sub-Gaussian), this is without loss of generality. In case they have different parameters $((\nu_r, \alpha_r), r \in [R]$ for sub-exponential and $\sigma_r, r \in [R]$ for sub-Gaussian), we can take the final parameters to be the maximum of the respective local parameters – for sub-exponential, we can take $\nu = \max_{r\in[R]} \nu_r$ and $\alpha = \max_{r\in[R]} \alpha_r$, and for sub-Gaussian, we can take $\sigma_{\mathrm{g}} = \max_{r\in[R]} \sigma_r$.

A. Bounding the gradient dissimilarity κ

In this section, we provide an upper bound on $\left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\bar{f}(\boldsymbol{x})\right\|$.

$$\begin{aligned}\left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\bar{f}(\boldsymbol{x})\right\| &\leq \left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\mu_r(\boldsymbol{x})\right\| + \|\nabla\mu_r(\boldsymbol{x})-\nabla\mu(\boldsymbol{x})\| + \left\|\nabla\bar{f}(\boldsymbol{x})-\nabla\mu(\boldsymbol{x})\right\| \\ &\leq \left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\mu_r(\boldsymbol{x})\right\| + \|\nabla\mu_r(\boldsymbol{x})-\nabla\mu(\boldsymbol{x})\| + \frac{1}{R}\sum_{r=1}^{R}\left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\mu_r(\boldsymbol{x})\right\|, \end{aligned} \tag{46}$$

where for the third term, we used $\bar{f}(\boldsymbol{x}) = \frac{1}{R}\sum_{r=1}^{R}\bar{f}_r(\boldsymbol{x})$ and $\mu(\boldsymbol{x}) = \frac{1}{R}\sum_{r=1}^{R}\mu_r(\boldsymbol{x})$, and applied the triangle inequality. It follows from (46) that in order to bound $\left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\bar{f}(\boldsymbol{x})\right\|$ uniformly over $\boldsymbol{x} \in \mathcal{C}$, it suffices to bound $\|\nabla\mu_r(\boldsymbol{x})-\nabla\mu(\boldsymbol{x})\|$ and $\left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\mu_r(\boldsymbol{x})\right\|, \forall r \in [R]$ uniformly over $\boldsymbol{x} \in \mathcal{C}$.

Bounding $\|\nabla\mu_r(\boldsymbol{x})-\nabla\mu(\boldsymbol{x})\|$. Note that $\nabla\mu_r(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{z}\sim q_r}[\nabla f_r(\boldsymbol{z},\boldsymbol{x})]$ is a property of the distribution q_r from which the data samples have been drawn and so is $\nabla\mu(\boldsymbol{x}) = \frac{1}{R}\sum_{r=1}^{R}\nabla\mu_r(\boldsymbol{x})$ the property of $q_1, \ldots, q_R$. Note that $\|\nabla\mu_r(\boldsymbol{x})-\nabla\mu(\boldsymbol{x})\|$ captures heterogeneity among distributions through their expected values, and is equal to zero in the i.i.d. homogeneous data setting of [17], [18], [20], [22]. In order to get a meaningful bound for κ, it is reasonable to assume that this heterogeneity is bounded. We assume a uniform bound on the $\|\nabla\mu_r(\boldsymbol{x})-\nabla\mu(\boldsymbol{x})\|$ for every $\boldsymbol{x} \in \mathcal{C}$.

Assumption 5. *For every worker $r \in [R]$, the population mean of the local gradients has a uniformly bounded deviation from the population mean of the global gradient, i.e.,*

$$\|\nabla\mu_r(\boldsymbol{x})-\nabla\mu(\boldsymbol{x})\| \leq \kappa_{\text{mean}}, \qquad \forall\boldsymbol{x}\in\mathcal{C}. \tag{47}$$

Bounding $\left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\mu_r(\boldsymbol{x})\right\|$. Now we bound the difference between the sample mean and the true mean under both sub-exponential and sub-Gaussian distributional assumptions on local gradients. For that we use standard tools, such as concentration results for sum of independent sub-Gaussian/sub-exponential random variables and ϵ-net arguments. We prove in Lemma 6 and Lemma 7, respectively, in Appendix E that under both the assumptions, with high probability, our bounds are $\left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\mu_r(\boldsymbol{x})\right\| \leq \mathcal{O}\left(\sqrt{\frac{d\log(n_r d)}{n_r}}\right)$ for every $\boldsymbol{x} \in \mathcal{C}$. Note that under the sub-exponential assumption, the bound holds only for sufficiently large n_r such that $n_r = \Omega\left(d\log(n_r d)\right)$, whereas, under the sub-Gaussian assumption, the bound holds for every n_r.

Substituting these bounds in (46) yields the following result, which, for notational convenience, we state for the case when all workers have the same number of data samples. Let $D = \max\{\|\boldsymbol{x}-\boldsymbol{x}'\| : \boldsymbol{x}, \boldsymbol{x}' \in \mathcal{C}\}$ be the diameter of $\mathcal{C}$. Note that $D = \Omega(\sqrt{d})$, and we assume that D can grow at most polynomially in d.

Theorem 5 (Gradient dissimilarity)**.** *Suppose $n := n_r, \forall r \in [R]$, and Assumption 5 holds. Then, the gradient dissimilarity bound under different distributional assumptions is as follows:*

1) [Sub-exponential] *Suppose Assumption 3 holds. Let $n \in \mathbb{N}$ be sufficiently large such that $n = \Omega\left(d\log(nd)\right)$. Then, with probability at least $1 - \frac{R}{(1+nLD)^d}$, the following bound holds for all $r \in [R]$ and $\boldsymbol{x} \in \mathcal{C}$:*

$$\left\|\nabla\bar{f}_r(\boldsymbol{x})-\nabla\bar{f}(\boldsymbol{x})\right\| \leq \kappa_{\text{mean}} + \mathcal{O}\left(\sqrt{\frac{d\log(nd)}{n}}\right). \tag{48}$$

2) [Sub-Gaussian] *Suppose Assumption 4 holds. For every $n \in \mathbb{N}$, with probability at least $1 - \frac{R}{(1+nLD)^d}$, the*

following bound holds for all $r \in [R]$ and $\boldsymbol{x} \in \mathcal{C}$:

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \bar{f}(\boldsymbol{x})\right\| \leq \kappa_{\text{mean}} + \mathcal{O}\left(\sqrt{\frac{d\log(nd)}{n}}\right). \tag{49}$$

Remark 1. *Note that under Assumption 3 (sub-exponential), the gradient dissimilarity bound (48) holds only when each worker has sufficiently large number of samples $n = \Omega\left(d\log(nd)\right)$. On the other hand, under Assumption 4 (sub-Gaussian), the gradient dissimilarity bound (49) holds for every $n \in \mathbb{N}$.*

B. Bounding the local variances

The local variance bound at the r'th worker is $\mathbb{E}_{i\in_U[n_r]}\left\|\nabla f_r(\boldsymbol{z}_{r,i},\boldsymbol{x}) - \nabla \bar{f}_r(\boldsymbol{x})\right\|^2 \leq \sigma^2$ (from (43)). We simplify the LHS:

$$\begin{aligned}
&\mathbb{E}_{i\in_U[n_r]}\left\|\nabla f_r(\boldsymbol{z}_{r,i},\boldsymbol{x}) - \nabla \bar{f}_r(\boldsymbol{x})\right\|^2 \\
&\leq 2\mathbb{E}_{i\in_U[n_r]}\left\|\nabla f_r(\boldsymbol{z}_{r,i},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|^2 \\
&\qquad + 2\mathbb{E}_{i\in_U[n_r]}\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|^2 \\
&\overset{(a)}{=} 2\left\|\nabla f_r(\boldsymbol{z}_{r,1},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|^2 + 2\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|^2 \\
&\overset{(b)}{\leq} 4\left\|\nabla f_r(\boldsymbol{z}_{r,1},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|^2
\end{aligned} \tag{50}$$

For the first term on the RHS of (a), we used that $\boldsymbol{z}_{r,i}, i \in [n_r]$ are i.i.d., and the second term follows because it is independent of $i \in [n_r]$. Inequality (b) follows because $\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|^2 \leq \left\|\nabla f_r(\boldsymbol{z}_{r,1},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|^2$, since the average of i.i.d. samples gives tighter concentration in comparison to if we use just one sample.

Note that bounding $\left\|\nabla f_r(\boldsymbol{z}_{r,1},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|$ is equivalent to bounding $\left\|\nabla f_r(\boldsymbol{z},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|$ for a random $\boldsymbol{z} \sim q_r$. We provide a uniform bound on $\left\|\nabla f_r(\boldsymbol{z},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|$ for a random $\boldsymbol{z} \sim q_r$ in Appendix E-C using the sub-Gaussian gradient assumption. Below we state our final bound on the local variances.

Theorem 6 (Variance bound)**.** *Suppose $n := n_r, \forall r \in [R]$, and Assumption 4 holds. Then, with probability at least $1 - \frac{R}{(1+nLD)^d}$, the following bound holds for all $r \in [R]$:*

$$\mathbb{E}_{i\in_U[n]}\left\|\nabla f_r(\boldsymbol{z}_{r,i},\boldsymbol{x}) - \nabla \bar{f}_r(\boldsymbol{x})\right\|^2 \leq \mathcal{O}\left(d\log(d)\right), \forall \boldsymbol{x} \in \mathcal{C}. \tag{51}$$

Remark 2 (Sub-Gaussian vs. sub-exponential assumption)**.** *Note that, we needed sub-Gaussian assumption on local gradients because we wanted to uniformly bound $\mathbb{E}_{i\in[n_r]}\left\|\nabla f_r(\boldsymbol{z}_{r,i},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|^2$, which is the case when we use* only one *data sample in each SGD iteration. In this paper, we use* mini-batch *SGD with a variable batch size b. So, when the batch-size b is sufficiently large and satisfies $b = \Omega(d\log(bd))$, we can work with the sub-exponential gradient assumption because the large batch size gives a concentration similar to sub-Gaussian. This would give a bound of $\mathcal{O}\left(\frac{d\log(bd)}{b}\right)$ on variance.*

ACKNOWLEDGEMENTS

Deepesh Data would like to thank Navjot Singh for his help with setting up the experiments.

REFERENCES

[1] J. Konecný, "Stochastic, distributed and federated optimization for machine learning," *CoRR*, vol. abs/1707.01155, 2017.

[2] J. Konecný, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *CoRR*, vol. abs/1610.02527, 2016.

[3] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017, pp. 1273–1282.

[4] M. Mohri, G. Sivek, and A. T. Suresh, "Agnostic federated learning," in *International Conference on Machine Learning (ICML)*, 2019, pp. 4615–4625.

[5] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, Q. V. Le, M. Z. Mao, M. Ranzato, A. W. Senior, P. A. Tucker, K. Yang, and A. Y. Ng, "Large scale distributed deep networks," in *Neural Information Processing Systems (NIPS)*, 2012, pp. 1232–1240.

[6] P. Kairouz *et al.*, "Advances and open problems in federated learning," *CoRR*, vol. abs/1912.04977, 2019.

[7] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *COMPSTAT*, 2010, pp. 177–186.

[8] P. Blanchard, E. M. E. Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *NIPS*, 2017, pp. 119–129.

[9] F. Haddadpour, M. M. Kamani, M. Mahdavi, and V. R. Cadambe, "Local SGD with periodic averaging: Tighter analysis and adaptive synchronization," in *Neural Information Processing Systems (NeurIPS)*, 2019, pp. 11 080–11 092.

[10] F. Haddadpour and M. Mahdavi, "On the convergence of local descent methods in federated learning," *CoRR*, vol. abs/1910.14425, 2019. [Online]. Available: http://arxiv.org/abs/1910.14425

[11] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh, "SCAFFOLD: stochastic controlled averaging for federated learning," in *International Conference on Machine Learning (ICML)*, 2020, pp. 5132–5143.

[12] A. Khaled, K. Mishchenko, and P. Richtárik, "Tighter theory for local SGD on identical and heterogeneous data," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, S. Chiappa and R. Calandra, Eds., 2020, pp. 4519–4529.

[13] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-iid data," in *International Conference on Learning Representations (ICLR)*, 2020. [Online]. Available: https://openreview.net/forum?id=HJxNAnVtDS

[14] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Conference on Machine Learning and Systems (MLSys)*, 2020. [Online]. Available: http://arxiv.org/abs/1812.06127

[15] H. Yu, S. Yang, and S. Zhu, "Parallel restarted SGD with faster convergence and less communication: Demystifying why model averaging works for deep learning," in *Conference on Artificial Intelligence (AAAI)*, 2019, pp. 5693–5700.

[16] D. Basu, D. Data, C. Karakus, and S. N. Diggavi, "Qsparse-local-sgd: Distributed SGD with quantization, sparsification and local computations," in *NeurIPS*, 2019, pp. 14 668–14 679.

[17] Y. Chen, L. Su, and J. Xu, "Distributed statistical machine learning in adversarial settings: Byzantine gradient descent," *POMACS*, vol. 1, no. 2, pp. 44:1–44:25, 2017.

[18] D. Yin, Y. Chen, K. Ramchandran, and P. L. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *ICML*, 2018, pp. 5636–5645.

[19] D. Alistarh, Z. Allen-Zhu, and J. Li, "Byzantine stochastic gradient descent," in *Neural Information Processing Systems (NeurIPS)*, 2018, pp. 4618–4628.

[20] L. Su and J. Xu, "Securing distributed gradient descent in high dimensional statistical learning," *POMACS*, vol. 3, no. 1, pp. 12:1–12:41, 2019.

[21] C. Xie, S. Koyejo, and I. Gupta, "Zeno: Distributed stochastic gradient descent with suspicion-based fault-tolerance," in *International Conference on Machine Learning (ICML)*, 2019, pp. 6893–6901.

[22] D. Yin, Y. Chen, K. Ramchandran, and P. L. Bartlett, "Defending against saddle point attack in byzantine-robust distributed learning," in *ICML*, 2019, pp. 7074–7084.

[23] D. Data and S. N. Diggavi, "On byzantine-resilient high-dimensional stochastic gradient descent," in *IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2020, pp. 2628–2633.

[24] L. Chen, H. Wang, Z. B. Charles, and D. S. Papailiopoulos, "DRACO: byzantine-resilient distributed training via redundant gradients," in *International Conference on Machine Learning (ICML)*, 2018, pp. 902–911.

[25] S. Rajput, H. Wang, Z. B. Charles, and D. S. Papailiopoulos, "DETOX: A redundancy-based framework for faster and more robust gradient aggregation," in *NeurIPS*, 2019, pp. 10 320–10 330.

[26] D. Data, L. Song, and S. N. Diggavi, "Data encoding methods for byzantine-resilient distributed optimization," in *ISIT*, 2019, pp. 2719–2723.

[27] D. Data and S. N. Diggavi, "Byzantine-tolerant distributed coordinate descent," in *ISIT*, 2019, pp. 2724–2728.

[28] D. Data, L. Song, and S. N. Diggavi, "Data encoding for byzantine-resilient distributed optimization," *IEEE Transactions on Information Theory*, vol. 67, no. 2, pp. 1117–1140, 2021.

[29] L. Li, W. Xu, T. Chen, G. B. Giannakis, and Q. Ling, "RSA: byzantine-robust stochastic aggregation methods for distributed learning from heterogeneous datasets," in *Conference on Artificial Intelligence (AAAI)*, 2019, pp. 1544–1551.

[30] A. Ghosh, J. Hong, D. Yin, and K. Ramchandran, "Robust federated learning in a heterogeneous environment," *CoRR*, vol. abs/1906.06629, 2019. [Online]. Available: http://arxiv.org/abs/1906.06629

[31] S. P. Karimireddy, L. He, and M. Jaggi, "Learning from history for byzantine robust optimization," in *International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 139. PMLR, 2021, pp. 5311–5319. [Online]. Available: http://proceedings.mlr.press/v139/karimireddy21a.html

[32] E. M. E. Mhamdi, R. Guerraoui, and S. Rouault, "Distributed momentum for byzantine-resilient stochastic gradient descent," in *International Conference on Learning Representations (ICLR)*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=H8UHdhWG6A3

[33] S. Farhadkhani, R. Guerraoui, N. Gupta, R. Pinot, and J. Stephan, "Byzantine machine learning made easy by resilient averaging of momentums," in *International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 2022, pp. 6246–6283. [Online]. Available: https://proceedings.mlr.press/v162/farhadkhani22a.html

[34] Z. Allen-Zhu, F. Ebrahimianghazani, J. Li, and D. Alistarh, "Byzantine-resilient non-convex stochastic gradient descent," in *International Conference on Learning Representations (ICLR)*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=PbEHqvFtcS

[35] S. P. Karimireddy, L. He, and M. Jaggi, "Byzantine-robust learning on heterogeneous datasets via bucketing," in *International Conference on Learning Representations (ICLR)*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=jXKKDEi5vJt

[36] C. Xie, O. Koyejo, and I. Gupta, "SLSGD: secure and efficient distributed on-device machine learning," in *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD, Proceedings, Part II*, 2019, pp. 213–228.

[37] K. A. Lai, A. B. Rao, and S. S. Vempala, "Agnostic estimation of mean and covariance," in *FOCS*, 2016, pp. 665–674.

[38] J. Steinhardt, M. Charikar, and G. Valiant, "Resilience: A criterion for learning in the presence of arbitrary outliers," in *ITCS*, 2018, pp. 45:1–45:21.

[39] I. Diakonikolas, G. Kamath, D. Kane, J. Li, A. Moitra, and A. Stewart, "Robust estimators in high-dimensions without the computational intractability," *SIAM J. Comput.*, vol. 48, no. 2, pp. 742–864, 2019.

[40] I. Diakonikolas and D. M. Kane, "Recent advances in algorithmic high-dimensional robust statistics," *CoRR*, vol. abs/1911.05911, 2019.

[41] H. Yu, R. Jin, and S. Yang, "On the linear speedup analysis of communication efficient momentum SGD for distributed non-convex optimization," in *ICML*, 2019, pp. 7184–7193.

[42] X. Li, W. Yang, S. Wang, and Z. Zhang, "Communication efficient decentralized training with multiple local updates," *CoRR*, vol. abs/1910.09126, 2019.

[43] J. Li, "Robustness in Machine Learning (CSE 599-M); Lecture 5 - Efficient filtering from spectral signatures," 2019. [Online]. Available: https://jerryzli.github.io/robust-ml-fall19.html

[44] Y. Dong, S. B. Hopkins, and J. Li, "Quantum entropy scoring for fast robust mean estimation and improved outlier detection," in *Neural Information Processing Systems (NeurIPS)*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 6065–6075.

[45] M. Charikar, J. Steinhardt, and G. Valiant, "Learning from untrusted data," in *STOC*, 2017, pp. 47–60.

[46] G. Baruch, M. Baruch, and Y. Goldberg, "A little is enough: Circumventing defenses for distributed learning," in *Neural Information Processing Systems (NeurIPS)*, 2019, pp. 8632–8642.

[47] E. M. E. Mhamdi, R. Guerraoui, and S. Rouault, "The hidden vulnerability of distributed learning in byzantium," in *International Conference on Machine Learning (ICML)*, 2018, pp. 3518–3527.

[48] J. D. Batson, D. A. Spielman, and N. Srivastava, "Twice-ramanujan sparsifiers," *SIAM J. Comput.*, vol. 41, no. 6, pp. 1704–1721, 2012.

[49] R. Vershynin, "Introduction to the non-asymptotic analysis of random matrices," *CoRR*, vol. abs/1011.3027, 2010.

Deepesh Data Deepesh Data is currently working as a research scientist at Meta Platforms, Inc. Before that he was a postdoctoral scholar at the University of California, Los Angeles (UCLA), from Mar'18 to Oct'22 and at the Indian Institute of Technology Bombay (IIT-B) from Sep'17 to Feb'18. His research interests are in federated machine learning, differential privacy, cryptography, algorithms, and information theory, with a current focus on privacy-preserving machine learning.

He received M.Sc. and Ph.D. degrees from the School of Technology and Computer Science at the Tata Institute of Fundamental Research (TIFR), Mumbai, India, in 2017, and B.Tech. degree in Computer Science and Engineering from the International Institute of Information Technology, Hyderabad (IIIT-H), India, in 2011. He has received the Best Paper Award from the ACM Conference on Computer and Communications Security (CCS) 2021, ACM India Doctoral Dissertation Award for 2019 (Honorable Mention), TIFR-Sasken Best Ph.D. Thesis Award for 2017-18 in Technology and Computer Sciences, and a Microsoft Research India Ph.D. Fellowship for 2014-17.

Suhas N. Diggavi Suhas Diggavi is currently a Professor of Electrical and Computer Engineering at UCLA. His undergraduate education is from IIT, Delhi and his PhD is from Stanford University. He has worked as a principal member research staff at AT&T Shannon Laboratories and directed the Laboratory for Information and Communication Systems (LICOS) at EPFL. At UCLA, he directs the Information Theory and Systems Laboratory.

His research interests include information theory and its applications to several areas including machine learning, security & privacy, wireless networks, data compression, cyber-physical systems, bio-informatics and neuroscience; more information can be found at http://licos.ee.ucla.edu.

He has received several recognitions for his research from IEEE and ACM, including the 2013 IEEE Information Theory Society & Communications Society Joint Paper Award, the 2021 ACM Conference on Computer and Communications Security (CCS) best paper award, the 2013 ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc) best paper award, the 2006 IEEE Donald Fink prize paper award among others. He was selected as a Guggenheim fellow in 2021. He also received the 2019 Google Faculty Research Award, 2020 Amazon faculty research award and 2021 Facebook/Meta faculty research award. He served as a IEEE Distinguished Lecturer and also served on board of governors for the IEEE Information theory society (2016-2021). He is a Fellow of the IEEE.

He is the Editor-in-Chief of the IEEE BITS Information Theory Magazine and has been an associate editor for IEEE Transactions on Information Theory, ACM/IEEE Transactions on Networking and other journals and special issues, as well as in the program committees of several IEEE conferences. He has also helped organize IEEE and ACM conferences including serving as the Technical Program Co-Chair for 2012 IEEE Information Theory Workshop (ITW), the Technical Program Co-Chair for the 2015 IEEE International Symposium on Information Theory (ISIT) and General co-chair for ACM Mobihoc 2018. He has 8 issued patents.

APPENDIX A
PROOF OF LEMMA 1

As mentioned in Section III-A, Lemma 1 generalizes [45, Proposition B.1], where the m samples $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ are drawn independently from a *single* distribution p with mean $\boldsymbol{\mu}$ and variance bound of σ_p^2, whereas, in our setting, different $\boldsymbol{y}_i$'s may come from different distributions, which may have different means and variances. Lemma 1 can be proved using similar arguments given in the proof of [45, Proposition B.1], and we provide a complete proof of this in this section.

Proof of Lemma 1 relies on the following lemma, which we prove in Appendix A-A.

Lemma 3. *Let p be a distribution on $\mathbb{R}^d$ such that $\mathbb{E}_{\boldsymbol{y}\sim p}[\boldsymbol{y}] = \boldsymbol{\mu}$ and $\mathbb{E}_{\boldsymbol{y}\sim p}[\langle \boldsymbol{y}-\boldsymbol{\mu}, \boldsymbol{v}\rangle^2] \le \sigma^2$ for all unit vectors $\boldsymbol{v} \in \mathbb{R}^d$. Let $\mathbf{M}$ be a symmetric matrix such that $\mathbf{0} \prec \mathbf{M} \prec c\mathbf{I}$ for some constant $c > 0$ and $\operatorname{tr}\big((c\mathbf{I}-\mathbf{M})^{-1}\big) \le \frac{1}{4\sigma_{\text{prev}}^2}$, where $\sigma_{\text{prev}} \ge \sigma$. Take an arbitrary $\epsilon' \in (0,1]$. Then, for $\boldsymbol{y} \sim p$, with probability at least $1-\frac{\epsilon'}{2}$, we have $\big(\mathbf{M}+\epsilon'(\boldsymbol{y}-\boldsymbol{\mu})(\boldsymbol{y}-\boldsymbol{\mu})^T\big) \prec (c+4\sigma^2)\mathbf{I}$ and $\operatorname{tr}\Big(\big((c+4\sigma^2)\mathbf{I}-(\mathbf{M}+\epsilon'(\boldsymbol{y}-\boldsymbol{\mu})(\boldsymbol{y}-\boldsymbol{\mu})^T)\big)^{-1}\Big) \le \frac{1}{4\sigma_{\text{prev}}^2}$.*

Now we continue to prove Lemma 1 with the help of Lemma 3.

Initialize a matrix $\mathbf{M} := \mathbf{0}$, a set $\mathcal{S} := \emptyset$, and $c := 4\sigma_{p_{\max}}^2 d$. Note that the preconditioning of Lemma 3 (i.e., $\mathbf{0} \prec \mathbf{M} \prec c\mathbf{I}$ and $\operatorname{tr}\big((c\mathbf{I}-\mathbf{M})^{-1}\big) \le \frac{1}{4\sigma_{\text{prev}}^2}$) is satisfied with $\sigma_{\text{prev}} = \sigma_{p_{\max}}$. Go through the stream of m samples from $\boldsymbol{y}_1$ to $\boldsymbol{y}_m$. Note that $\sigma_{p_{\max}} \ge \sigma_{p_i}$ holds for all $i \in [m]$. For notational convenience, let $\widetilde{\boldsymbol{y}}_i = \boldsymbol{y}_i - \boldsymbol{\mu}_i$ for $i = 1, 2, \ldots, m$. If $(\mathbf{M}+\epsilon'\widetilde{\boldsymbol{y}}_i\widetilde{\boldsymbol{y}}_i^T)$ satisfies the conclusion of Lemma 3, i.e., $\big(\mathbf{M}+\epsilon'\widetilde{\boldsymbol{y}}_i\widetilde{\boldsymbol{y}}_i^T\big) \prec (c+4\sigma_{p_i}^2)\mathbf{I}$ and $\operatorname{tr}\Big(\big((c+4\sigma_{p_i}^2)\mathbf{I}-(\mathbf{M}+\epsilon'\widetilde{\boldsymbol{y}}_i\widetilde{\boldsymbol{y}}_i^T)\big)^{-1}\Big) \le \frac{1}{4\sigma_{p_{\max}}^2}$ (which we know holds with probability at least $1-\frac{\epsilon'}{2}$), then update $\mathcal{S} \leftarrow \mathcal{S}\cup\{i\}$, $\mathbf{M} \leftarrow \mathbf{M}+\epsilon'\widetilde{\boldsymbol{y}}_i\widetilde{\boldsymbol{y}}_i^T$, and $c \leftarrow c + 4\sigma_{p_i}^2$.[9]

Note that, in the next iteration, when we consider the sample $\boldsymbol{y}_{i+1}$, the preconditioning of Lemma 3 is automatically satisfied: If the conclusion in the i'th step did not hold and we did not update $\mathcal{S}, \mathbf{M}, c$, then the preconditioning of Lemma 3 in the $(i+1)$'st iteration trivially holds, as it used to hold in the i'th iteration. If the conclusion in the i'th step held and we updated $\mathcal{S}, \mathbf{M}, c$, then the preconditioning of Lemma 3 in the $(i+1)$'st iteration holds, as it is the same condition that we checked in the conclusion of the i'th iteration for updating $\mathcal{S}, \mathbf{M}, c$.

When we have gone through the stream of m samples, in the end, we have $c = 4\sigma_{p_{\max}}^2 d + \sum_{i\in\mathcal{S}} 4\sigma_{p_i}^2 \le 4\sigma_{p_{\max}}^2(d+|\mathcal{S}|)$ and $\mathbf{M} \prec \big(4\sigma_{p_{\max}}^2(d+|\mathcal{S}|)\big)\mathbf{I}$, which implies that $\lambda_{\max}(\mathbf{M}) \le 4\sigma_{p_{\max}}^2(d+|\mathcal{S}|)$. Since $\mathbf{M} = \sum_{i\in\mathcal{S}} \epsilon'\widetilde{\boldsymbol{y}}_i\widetilde{\boldsymbol{y}}_i^T$, we have $\lambda_{\max}\left(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\widetilde{\boldsymbol{y}}_i\widetilde{\boldsymbol{y}}_i^T\right) = \frac{1}{\epsilon'|\mathcal{S}|}\lambda_{\max}(\mathbf{M}) \le \frac{4\sigma_{p_{\max}}^2}{\epsilon'}(1+\frac{d}{|\mathcal{S}|})$. It only remains to show that $|\mathcal{S}| \ge (1-\gamma\epsilon')m$ holds with high probability.

[9]Note that we only observe $\boldsymbol{y}_i$'s, not $(\boldsymbol{y}_i-\boldsymbol{\mu}_i)$. In the context of distributed SGD, the $\boldsymbol{y}_i$'s correspond to the stochastic gradients that the server receives from clients, and there, the server does not know the true local gradients at any client – the true local gradient at client i corresponds to the mean $\boldsymbol{\mu}_i$ here. Yet, in each iteration i, we probabilistically add $\epsilon'(\boldsymbol{y}_i-\boldsymbol{\mu}_i)(\boldsymbol{y}_i-\boldsymbol{\mu}_i)^T$ to $\mathbf{M}$. We can do that, because we just want to show an existence of a set $\mathcal{S}$ that satisfies the required properties stated in Lemma 1. This is just for the purpose of analysis, and we are not giving an algorithm to construct $\mathcal{S}$.

By the above discussion, note that for each element i, we add i to $\mathcal{S}$ with probability at least $1-\frac{\epsilon'}{2}$. Since the m samples $\boldsymbol{y}_i, i \in [m]$ are independent of each other, we have that the distribution of $|\mathcal{S}|$ is lower-bounded by the sum of m independent indicator random variables, where each of them is equal to 1 with probability $1-\frac{\epsilon'}{2}$. So, by Chernoff bound, we have $\Pr[|\mathcal{S}| \le (1-\gamma\epsilon')m] \le \exp(-\frac{(2\gamma-1)\epsilon'^2 m}{8})$, which implies that $\Pr[|\mathcal{S}| \ge (1-\gamma\epsilon')m] \ge 1-\exp(-\frac{(2\gamma-1)\epsilon'^2 m}{8})$. This holds for any $\gamma > 1/2$.

We have shown that with probability $1-\exp(-\frac{(2\gamma-1)^2\epsilon'^2 m}{8})$, there exists a subset $\mathcal{S}$ of $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ such that $|\mathcal{S}| \ge (1-\gamma\epsilon')m$ and $\lambda_{\max}\left(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\widetilde{\boldsymbol{y}}_i\widetilde{\boldsymbol{y}}_i^T\right) \le \frac{4\sigma_{p_{\max}}^2}{\epsilon'}\left(1+\frac{d}{(1-\gamma\epsilon')m}\right)$. Substituting $\widetilde{\boldsymbol{y}}_i = \boldsymbol{y}_i - \boldsymbol{\mu}_i$ for $i = 1, 2, \ldots, m$ concludes the proof of Lemma 1.

A. Proof of Lemma 3

A version of this lemma has appeared in [45, Lemma B.2], which, in turn, is essentially the same as [48, Lemma 3.3]. Our proof is along the lines of the proof of [45, Lemma B.2].

For simplicity of notation, let $\widetilde{\boldsymbol{y}} = \boldsymbol{y} - \boldsymbol{\mu}$. Instead of $\big(\mathbf{M}-\epsilon'\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big)$, it will be helpful later to consider $\big(\mathbf{M}-t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big)$ for arbitrary $t \in [0, \epsilon']$.

By the Sherman-Morrison matrix inversion formula, we have that if a square matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ is invertible and $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$ are column vectors such that $(1+\boldsymbol{v}^T\mathbf{A}^{-1}\boldsymbol{u}) \ne 0$, then $(\mathbf{A}+\boldsymbol{u}\boldsymbol{v}^T)$ is invertible and its inverse is equal to $(\mathbf{A}+\boldsymbol{u}\boldsymbol{v}^T)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\boldsymbol{u}\boldsymbol{v}^T\mathbf{A}^{-1}}{1+\boldsymbol{v}^T\mathbf{A}^{-1}\boldsymbol{u}}$.

We want to apply this formula on $\big(((c+4\sigma^2)\mathbf{I}-\mathbf{M}) - t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big)^{-1}$ with $\mathbf{A} = ((c+4\sigma^2)\mathbf{I}-\mathbf{M})$, $\boldsymbol{u} = \sqrt{t}\widetilde{\boldsymbol{y}}$, and $\boldsymbol{v} = -\sqrt{t}\widetilde{\boldsymbol{y}}$. For that, we need to show two things: first, that $((c+4\sigma^2)\mathbf{I}-\mathbf{M})$ is invertible, and second, that $(1-t\widetilde{\boldsymbol{y}}^T((c+4\sigma^2)\mathbf{I}-\mathbf{M})^{-1}\widetilde{\boldsymbol{y}}) \ne 0$. For the first requirement, note that $\mathbf{M} \prec (c+4\sigma^2)\mathbf{I}$, which follows because $\mathbf{M} \prec c\mathbf{I}$ (by assumption), and $\sigma > 0$. This implies that $((c+4\sigma^2)\mathbf{I}-\mathbf{M})$ is invertible. It follows from the analysis below (see the paragraph after (54)) that the second requirement also holds for every $t \in [0, \epsilon']$ with probability at least $1-\frac{\epsilon'}{2}$. Now, applying the Sherman-Morrison matrix inversion formula on $\big(((c+4\sigma^2)\mathbf{I}-\mathbf{M}) - t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big)^{-1}$:

$$\begin{aligned}\big(((c+4\sigma^2)\mathbf{I}-\mathbf{M}) - t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big)^{-1} &= \big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1} \\ &+ t\frac{\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}}{1-t\widetilde{\boldsymbol{y}}^T\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\widetilde{\boldsymbol{y}}} \quad (52)\end{aligned}$$

Taking trace on both sides gives

$$\begin{aligned}&\operatorname{tr}\left(\left(((c+4\sigma^2)\mathbf{I}-\mathbf{M}) - t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\right)^{-1}\right) \\ &= \operatorname{tr}\left(\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\right) \\ &\quad + \frac{\operatorname{tr}\left(\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\right)}{\frac{1}{t}-\widetilde{\boldsymbol{y}}^T\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\widetilde{\boldsymbol{y}}}.\end{aligned}$$

Let $\Phi_c(\mathbf{M}) = \mathrm{tr}\big((c\mathbf{I}-\mathbf{M})^{-1}\big)$. Using $\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$ on the last term and using the fact that trace of a scalar is the scalar itself, we get

$$\Phi_{c+4\sigma^2}(\mathbf{M}+t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T) = \Phi_{c+4\sigma^2}(\mathbf{M}) + \frac{\widetilde{\boldsymbol{y}}^T\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-2}\widetilde{\boldsymbol{y}}}{\frac{1}{t}-\widetilde{\boldsymbol{y}}^T\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\widetilde{\boldsymbol{y}}}. \tag{53}$$

We are given $\Phi_c(\mathbf{M}) \le \frac{1}{4\sigma^2_{\text{prev}}}$, and we want to show $\Phi_{c+4\sigma^2}(\mathbf{M}+t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T) \le \frac{1}{4\sigma^2_{\text{prev}}}$. So, it suffices to prove that $\Phi_{c+4\sigma^2}(\mathbf{M}+t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T) \le \Phi_c(\mathbf{M})$. This, in light of (53), is equivalent to the condition

$$\frac{1}{t} \ge \widetilde{\boldsymbol{y}}^T\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\widetilde{\boldsymbol{y}} + \underbrace{\frac{\widetilde{\boldsymbol{y}}^T\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-2}\widetilde{\boldsymbol{y}}}{\Phi_c(\mathbf{M})-\Phi_{c+4\sigma^2}(\mathbf{M})}}_{=:\Psi}, \tag{54}$$

which, as we show in the analysis below, will hold with probability at least $1-\frac{\epsilon'}{2}$ for all $t \in [0,\epsilon']$. (Assume that (54) holds with probability at least $1-\frac{\epsilon'}{2}$ for all $t \in [0,\epsilon']$. Note that $\Phi_c(\mathbf{M}) > \Phi_{c+4\sigma^2}(\mathbf{M})$ (from Claim 5 below) and $((c+4\sigma^2)\mathbf{I}-\mathbf{M}) \succ \mathbf{0}$ hold. Using these in (54) imply that $\frac{1}{t} > \widetilde{\boldsymbol{y}}^T\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\widetilde{\boldsymbol{y}}$ holds with probability at least $1-\frac{\epsilon'}{2}$ for all $t \in [0,\epsilon']$. Thus the second requirement $(1-t\widetilde{\boldsymbol{y}}^T((c+4\sigma^2)\mathbf{I}-\mathbf{M})^{-1}\widetilde{\boldsymbol{y}}) \neq 0$ also holds, which was necessary for applying the matrix inversion formula on $\big(((c+4\sigma^2)\mathbf{I}-\mathbf{M})-t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big)^{-1}$ to write (52).)

Since Ψ is a scalar, we have $\mathrm{tr}(\Psi) = \Psi$. Taking trace in (54), and using $\mathrm{tr}(\widetilde{\boldsymbol{y}}^T\mathbf{A}\widetilde{\boldsymbol{y}}) = \mathrm{tr}(\mathbf{A}\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T)$, and then taking expectation, we get

$$\mathbb{E}[\Psi] = \mathbb{E}\left[\mathrm{tr}\left(\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\right)\right] + \frac{\mathbb{E}\left[\mathrm{tr}\left(\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-2}\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\right)\right]}{\Phi_c(\mathbf{M})-\Phi_{c+4\sigma^2}(\mathbf{M})}. \tag{55}$$

Since $\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big) \succ \mathbf{0}$, we also have that $\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-i} \succ \mathbf{0}$, for $i=1,2$. Let $\mathbf{A} = \big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-i}$ for any $i \in \{1,2\}$. Now we argue that $\mathbb{E}\left[\mathrm{tr}\left(\mathbf{A}\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\right)\right] \le \sigma^2\mathrm{tr}(\mathbf{A})$, where σ^2 is such that $\mathbb{E}_{\boldsymbol{y}\sim p}[\langle\widetilde{\boldsymbol{y}},\boldsymbol{v}\rangle^2] \le \sigma^2$ for all unit vectors $\boldsymbol{v} \in \mathbb{R}^d$. Note that the last condition is equivalent to $\sup_{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|=1} \boldsymbol{v}^T\left(\mathbb{E}_{\boldsymbol{y}\sim p}[\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T]\right)\boldsymbol{v} \le \sigma^2$, which, in view (60), is equivalent to saying that $\lambda_{\max}\left(\mathbb{E}_{\boldsymbol{y}\sim p}[\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T]\right) \le \sigma^2$.

Claim 4. $\mathbb{E}\left[\mathrm{tr}\left(\mathbf{A}\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\right)\right] \le \sigma^2\mathrm{tr}(\mathbf{A})$.

Proof. The claim follows from the following set of inequalities.

$$\begin{aligned}\mathbb{E}\left[\mathrm{tr}\left(\mathbf{A}\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\right)\right] &\overset{(a)}{=} \mathbb{E}\left[\sum_{i,j}\mathbf{A}_{ij}(\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T)_{ji}\right]\\ &= \sum_{i,j}\mathbf{A}_{ij}\big(\mathbb{E}[\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T]\big)_{ji}\\ &\overset{(b)}{=} \mathrm{tr}\big(\mathbf{A}\mathbb{E}[\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T]\big)\\ &\overset{(c)}{\le} \left\|\mathbb{E}\left[\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\right]\right\|\|\mathbf{A}\|_*\\ &\overset{(d)}{\le} \sigma^2\mathrm{tr}(\mathbf{A})\end{aligned}$$

In (a) and (b), we used the definition of trace: $\mathrm{tr}(\mathbf{AB}) = \sum_i(\mathbf{AB})_{ii} = \sum_{i,j}\mathbf{A}_{ij}\mathbf{B}_{ji}$. In (c), we used $\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA}) \le \|\mathbf{B}\|\|\mathbf{A}\|_*$, where $\|\cdot\|_*$ denotes the nuclear norm, which is equal to the sum of singular values; see Claim 6 on page 21 for a proof. In (d), we used two things, first, since $\mathbf{A} \succeq \mathbf{0}$, we have $\|\mathbf{A}\|_* = \mathrm{tr}(\mathbf{A})$, and second, that $\left\|\mathbb{E}\left[\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\right]\right\| \le \sigma^2$, which follows because $\left\|\mathbb{E}\left[\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\right]\right\| = \lambda_{\max}\left(\mathbb{E}_{\boldsymbol{y}\sim p}[\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T]\right) \le \sigma^2$. □

Using Claim 4 in (55) gives

$$\mathbb{E}[\Psi] \le \sigma^2\mathrm{tr}\left(\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-1}\right) + \sigma^2\frac{\mathrm{tr}\left(\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-2}\right)}{\Phi_c(\mathbf{M})-\Phi_{c+4\sigma^2}(\mathbf{M})}. \tag{56}$$

Claim 5. $\Phi_c(\mathbf{M}) - \Phi_{c+4\sigma^2}(\mathbf{M}) \ge 4\sigma^2\mathrm{tr}\left(\big((c+4\sigma^2)\mathbf{I}-\mathbf{M}\big)^{-2}\right)$.

Proof. Since $(c\mathbf{I}-\mathbf{M}) \succ \mathbf{0}$, let its eigen-decomposition be $(c\mathbf{I}-\mathbf{M}) = \sum_i\lambda_i\boldsymbol{u}_i\boldsymbol{u}_i^T$, where λ_i's are the eigenvalues of $(c\mathbf{I}-\mathbf{M})$ and $\boldsymbol{u}_i$'s are the corresponding eigenvectors. It follows that $\big((c\mathbf{I}-\mathbf{M})+4\sigma^2\mathbf{I}\big) = \sum_i(\lambda_i+4\sigma^2)\boldsymbol{u}_i\boldsymbol{u}_i^T$. These imply that $(c\mathbf{I}-\mathbf{M})^{-1} = \sum_i\frac{1}{\lambda_i}\boldsymbol{u}_i\boldsymbol{u}_i^T$ and $\big((c\mathbf{I}-\mathbf{M})+4\sigma^2\mathbf{I}\big)^{-1} = \sum_i\frac{1}{(\lambda_i+4\sigma^2)}\boldsymbol{u}_i\boldsymbol{u}_i^T$.

Substituting the definition of $\Phi_c(\mathbf{M}) = \mathrm{tr}\big((c\mathbf{I}-\mathbf{M})^{-1}\big)$, we have

$$\begin{aligned}&\Phi_c(\mathbf{M})-\Phi_{c+4\sigma^2}(\mathbf{M})\\ &\quad= \mathrm{tr}\left((c\mathbf{I}-\mathbf{M})^{-1}-\big(((c+4\sigma^2)\mathbf{I}-\mathbf{M})\big)^{-1}\right)\\ &\quad= \mathrm{tr}\left(\sum_i\frac{1}{\lambda_i}\boldsymbol{u}_i\boldsymbol{u}_i^T-\sum_i\frac{1}{(\lambda_i+4\sigma^2)}\boldsymbol{u}_i\boldsymbol{u}_i^T\right)\\ &\quad= 4\sigma^2\mathrm{tr}\left(\sum_i\frac{1}{\lambda_i(\lambda_j+4\sigma^2)}\boldsymbol{u}_i\boldsymbol{u}_i^T\right)\\ &\quad\overset{(g)}{=} 4\sigma^2\sum_i\frac{1}{\lambda_i(\lambda_j+4\sigma^2)}\\ &\quad\overset{(h)}{\ge} 4\sigma^2\sum_i\frac{1}{(\lambda_j+4\sigma^2)^2}\\ &\quad= 4\sigma^2\mathrm{tr}\left(\sum_i\frac{1}{(\lambda_j+4\sigma^2)^2}\boldsymbol{u}_i\boldsymbol{u}_i^T\right)\\ &\quad= 4\sigma^2\mathrm{tr}\left(\big((c\mathbf{I}-\mathbf{M})+4\sigma^2\mathbf{I}\big)^{-2}\right)\end{aligned}$$

Here (g) follows from the fact that trace of a square matrix is equal to the sum of its eigenvalues and (h) follows because $\frac{1}{\lambda} \ge \frac{1}{\lambda+4\sigma^2}$. □

Substituting $\Phi_{c+4\sigma^2}(\mathbf{M}) = \mathrm{tr}\left(\big(((c+4\sigma^2)\mathbf{I}-\mathbf{M})\big)^{-1}\right)$ for the first term in (56) and the bound from Claim 5 for the second term gives $\mathbb{E}[\Psi] \le \sigma^2\left(\Phi_{c+4\sigma^2}(\mathbf{M})+\frac{1}{4\sigma^2}\right)$. Note that Claim 5 trivially implies $\Phi_{c+4\sigma^2}(\mathbf{M}) \le \Phi_c(\mathbf{M})$, where

$\Phi_c(\mathbf{M}) = \text{tr}\big((c\mathbf{I} - \mathbf{M})^{-1}\big) \leq \frac{1}{4\sigma_{\text{prev}}^2}$ (which follows from the hypothesis of Lemma 3). So, we have

$$\mathbb{E}\left[\Psi\right] \leq \sigma^2 \left(\frac{1}{4\sigma_{\text{prev}}^2} + \frac{1}{4\sigma^2}\right) \overset{\text{(h)}}{\leq} \sigma^2 \left(\frac{1}{4\sigma^2} + \frac{1}{4\sigma^2}\right) \leq \frac{1}{2}, \tag{57}$$

where (h) follows from our assumption that $\sigma_{\text{prev}} \geq \sigma$.

Note that Ψ is a non-negative random variable (see (54)). So, by the Markov's inequality, we have $\Pr[\Psi \geq \frac{1}{\epsilon'}] \leq \frac{\mathbb{E}[\Psi]}{1/\epsilon'} \leq \frac{\epsilon'}{2}$, which implies that $\Pr[\Psi \leq \frac{1}{\epsilon'}] \geq 1 - \frac{\epsilon'}{2}$. Substituting the value of Ψ in (54) implies that (54) holds with probability at least $1 - \frac{\epsilon'}{2}$ for all $t \in [0, \epsilon']$. Note that the condition in (54) is equivalent to the condition that $\Phi_{c+4\sigma^2}(\mathbf{M} + t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T) \leq \Phi_c(\mathbf{M})$, where $\Phi_c(\mathbf{M}) \leq \frac{1}{4\sigma_{\text{prev}}^2}$. Thus, with probability at least $1 - \frac{\epsilon'}{2}$, we have that $\text{tr}\left(\left(((c+4\sigma^2)\mathbf{I} - (\mathbf{M} + t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T))\right)^{-1}\right) = \Phi_{c+4\sigma^2}(\mathbf{M} + t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T) \leq \frac{1}{4\sigma_{\text{prev}}^2}$, for every $t \in [0, \epsilon']$.

It only remains to show that $\big(\mathbf{M} + \epsilon'\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big) \prec (c + 4\sigma^2)\mathbf{I}$, which is equivalent to the condition that $\lambda_{\max}\big(\mathbf{M} + \epsilon'\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big) < (c + 4\sigma^2)$. Suppose not, i.e., $\lambda_{\max}\big(\mathbf{M} + \epsilon'\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big) \geq (c + 4\sigma^2)$. Note that we have $\lambda_{\max}(\mathbf{M}) < c$ (by the hypothesis of Lemma 3). Since $\lambda_{\max}\big(\mathbf{M} + t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big)$ is a continuous function of t and $\lambda_{\max}(\mathbf{M}) < c$, $\lambda_{\max}\big(\mathbf{M} + \epsilon'\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big) \geq (c + 4\sigma^2)$, we have from the intermediate value theorem that there exists a $t' \in [0, \epsilon']$ such that $\lambda_{\max}\big(\mathbf{M} + t'\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T\big) = (c + 4\sigma^2)$. This implies that the matrix $\left(((c+4\sigma^2)\mathbf{I} - (\mathbf{M} + t'\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T))\right)^{-1}$ is not invertible (as $\big((c+4\sigma^2)\mathbf{I} - (\mathbf{M} + t'\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T)\big)$ has a zero eigenvalue), implying that $\text{tr}\left(\left(((c+4\sigma^2)\mathbf{I} - (\mathbf{M} + t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T))\right)^{-1}\right)$ is unbounded. But, we have already shown that $\text{tr}\left(\left(((c+4\sigma^2)\mathbf{I} - (\mathbf{M} + t\widetilde{\boldsymbol{y}}\widetilde{\boldsymbol{y}}^T))\right)^{-1}\right) \leq \frac{1}{4\sigma_{\text{prev}}^2} < \infty$, for all $t \in [0, \epsilon']$. A contradiction.

This completes the proof of Lemma 3.

Claim 6. *For any two dimension compatible matrices $\mathbf{A}, \mathbf{B}$, we have $\text{tr}(\mathbf{AB}) \leq \|\mathbf{A}\|\|\mathbf{B}\|_*$, where $\|\cdot\|$ is the matrix norm induced by the ℓ_2-norm, and $\|\cdot\|_*$ is the nuclear norm, which is equal to the sum of singular values of $\mathbf{B}$.*

Proof. Let $r = \text{rank}(\mathbf{B})$, and let $\sigma_i, i = 1, 2, \ldots, r$ denote the non-zero singular values of $\mathbf{B}$. By the singular value decomposition, we have $\mathbf{B} = \sum_{i=1}^r \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$, where $\boldsymbol{u}_i, \boldsymbol{v}_i$ are the left and right singular vectors, respectively, corresponding to the singular value σ_i. Note that $\boldsymbol{u}_i, \boldsymbol{v}_i$ for every $i = 1, \ldots, r$ are unit norm vectors.

$$\begin{aligned}
\text{tr}(\mathbf{AB}) &= \text{tr}(\mathbf{A}\sum_{i=1}^r \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T) \\
&= \sum_{i=1}^r \sigma_i \text{tr}(\mathbf{A}\boldsymbol{u}_i \boldsymbol{v}_i^T) \\
&\qquad \text{(Since tr(A +B) = tr(A) + tr(B))} \\
&= \sum_{i=1}^r \sigma_i \text{tr}(\boldsymbol{v}_i^T \mathbf{A} \boldsymbol{u}_i) \qquad \text{(Since tr(AB) = tr(BA))} \\
&= \sum_{i=1}^r \sigma_i \boldsymbol{v}_i^T \mathbf{A} \boldsymbol{u}_i \qquad \text{(Since } \boldsymbol{v}_i^T \mathbf{A} \boldsymbol{u}_i \text{ is a scalar)} \\
&\overset{\text{(a)}}{\leq} \sum_{i=1}^r \sigma_i \|\boldsymbol{v}_i\| \|\mathbf{A}\| \|\boldsymbol{u}_i\| \\
&= \|\mathbf{A}\| \sum_{i=1}^r \sigma_i \\
&\text{(Since } \|\boldsymbol{u}_i\| = 1, \|\boldsymbol{v}_i\| = 1, \text{ for every } i = 1, \ldots, r) \\
&= \|\mathbf{A}\| \|\mathbf{B}\|_*
\end{aligned}$$

In (a), first we used the Cauchy-Schwarz inequality to write $\boldsymbol{v}_i^T \mathbf{A} \boldsymbol{u}_i \leq \|\boldsymbol{v}_i\| \|\mathbf{A}\boldsymbol{u}_i\|$ and then used the definition of matrix norm to write $\|\mathbf{A}\boldsymbol{u}_i\| \leq \|\mathbf{A}\| \|\boldsymbol{u}_i\|$. Note that if $\mathbf{A}, \mathbf{B}$ are positive semi-definite, then equality holds in (a) above if and only if $\mathbf{B}$ is a multiple of $\boldsymbol{u}\boldsymbol{u}^T$, where $\boldsymbol{u}$ is the eigenvector corresponding to the largest eigenvalue of $\mathbf{A}$.

This completes the proof of Claim 6. □

Appendix B
Complete Proof of Theorem 3

Let $t_k, t_{k+1} \in \mathcal{I}_T$ be any two consecutive synchronization indices. For $i \in \mathcal{K}_{t_k}$ corresponding to an honest client, let $Y_i^{t_k}, Y_i^{t_k+1}, \ldots, Y_i^{t_{k+1}-1}$ be a sequence of $(t_{k+1} - t_k) \leq H$ (dependent) random variables, where, for any $t \in [t_k : t_{k+1} - 1]$, the random variable Y_i^t is distributed as

$$Y_i^t \sim \text{Unif}\Big(\mathcal{F}_i^{\otimes b}\big(\boldsymbol{x}_i^t(\boldsymbol{x}_i^{t_k}, Y_i^{t_k}, \ldots, Y_i^{t-1})\big)\Big). \tag{58}$$

Here, Y_i^t corresponds to the stochastic sampling of mini-batch gradients from the set $\mathcal{F}_i^{\otimes b}\big(\boldsymbol{x}_i^t(\boldsymbol{x}_i^{t_k}, Y_i^{t_k}, \ldots, Y_i^{t-1})\big)$, which itself depends on the local parameters $\boldsymbol{x}_i^{t_k}$ (which is a deterministic quantity) at the last synchronization index and the past realizations of $Y_i^{t_k}, \ldots, Y_i^{t-1}$. This is because the evolution of local parameters $\boldsymbol{x}_i^t$ depends on $\boldsymbol{x}_i^{t_k}$ and the choice of gradients in between time indices t_k and $t-1$. Now define $Y_i := \sum_{t=t_k}^{t_{k+1}-1} Y_i^t$; and let p_i be the distribution of Y_i. This is the distribution p_i we will take when using Lemma 1.

Claim 7. *For any honest client $i \in \mathcal{K}_{t_k}$, we have $\mathbb{E}\|Y_i - \mathbb{E}[Y_i]\|^2 \leq \frac{H^2\sigma^2}{b}$, where expectation is taken over sampling stochastic gradients by client i between the synchronization indices t_k and t_{k+1}.*

Proof. Take an arbitrary honest client $i \in \mathcal{K}_{t_k}$.

$$\begin{aligned}
\mathbb{E}\|Y_i - \mathbb{E}[Y_i]\|^2 &= \mathbb{E}\left\|\sum_{t=t_k}^{t_{k+1}-1} \left(Y_i^t - \mathbb{E}[Y_i^t]\right)\right\|^2 \\
&\overset{\text{(a)}}{\leq} (t_{k+1} - t_k) \sum_{t=t_k}^{t_{k+1}-1} \mathbb{E}\|Y_i^t - \mathbb{E}[Y_i^t]\|^2 \\
&\overset{\text{(b)}}{\leq} \frac{H^2\sigma^2}{b},
\end{aligned}$$

where (a) follows from the Jensen's inequality; in (b) we used $(t_{k+1} - t_k) \leq H$ and that $\mathbb{E}\|Y_i^t - \mathbb{E}[Y_i^t]\|^2 \leq \frac{\sigma^2}{b}$ for all $j \in [H]$, which follows from the explanation below:

$\mathbb{E}\|Y_i^t - \mathbb{E}[Y_i^t]\|^2$

$$
\begin{aligned}
&= \sum_{\boldsymbol{y}_i^{t_k},\ldots,\boldsymbol{y}_i^{t-1}} \Pr\left[Y_i^j = \boldsymbol{y}_i^j, j \in [t_k : t-1]\right] \\
&\qquad \times \mathbb{E}\left[\|Y_i^t - \mathbb{E}[Y_i^t]\|^2 \mid Y_i^j = \boldsymbol{y}_i^j, j \in [t_k : t-1]\right] \\
&\overset{(c)}{\leq} \sum_{\boldsymbol{y}_i^{t_k},\ldots,\boldsymbol{y}_i^{t-1}} \Pr\left[Y_i^j = \boldsymbol{y}_i^j, j \in [t_k : t-1]\right] \cdot \frac{\sigma^2}{b} \\
&= \frac{\sigma^2}{b}.
\end{aligned}
$$

Note that $Y_i^t \sim \text{Unif}\Big(\mathcal{F}_i^{\otimes b}\big(\boldsymbol{x}_i^t(\boldsymbol{x}_i^{t_k}, Y_i^{t_k}, \ldots, Y_i^{t-1})\big)\Big)$. So, when we fix the values $Y_i^{t_k} = \boldsymbol{y}_i^{t_k}, \ldots, Y_i^{t-1} = \boldsymbol{y}_i^{t-1}$, the parameter vector $\boldsymbol{x}_i^t\big(\boldsymbol{x}_i^{t_k}, Y_i^{t_k} \ldots, Y_i^{t-1}\big)$ becomes a deterministic quantity. Now we can use the variance bound (5) in order to bound $\mathbb{E}\left[\|Y_i^t - \mathbb{E}[Y_i^t]\|^2 \mid Y_i^j = \boldsymbol{y}_i^j, j \in [t_k : t-1]\right] \leq \frac{\sigma^2}{b}$. This is what we used in (c). □

It is easy to see that the hypothesis of Lemma 1 is satisfied with $\boldsymbol{\mu}_i = \mathbb{E}[Y_i], \sigma_{p_i}^2 = \frac{H^2\sigma^2}{b}$ for all honest clients $i \in \mathcal{K}_{t_k}$ (note that p_i is the distribution of Y_i):

$$
\begin{aligned}
\mathbb{E}_{\boldsymbol{y}_i \sim p_i}[\langle \boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i], \boldsymbol{v}\rangle^2] &\overset{(d)}{\leq} \mathbb{E}[\|\boldsymbol{y}_i - \mathbb{E}_{\boldsymbol{y}_i \sim p_i}[\boldsymbol{y}_i]\|^2] \cdot \|\boldsymbol{v}\|^2 \\
&\overset{(e)}{\leq} \frac{H^2\sigma^2}{b},
\end{aligned}
$$

where (d) follows from the Cauchy-Schwarz inequality and (e) follows from Claim 7 and $\|\boldsymbol{v}\| \leq 1$.

We are given K different (summations of H) gradients, out of which at least $(1-\epsilon)K$ are according to the correct distribution. By considering only the uncorrupted gradients (i.e., taking $m = (1-\epsilon)K$), we have from Lemma 1 that there exists a subset $\mathcal{S} \subseteq \mathcal{K}_{t_k}$ of K gradients of size $(1-\gamma\epsilon')(1-\epsilon)K \geq (1-(\epsilon+\gamma\epsilon'))K$ that satisfies

$$
\begin{aligned}
&\lambda_{\max}\left(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}(\boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i])(\boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i])^T\right) \\
&\qquad \leq \frac{4H^2\sigma^2}{b\epsilon'}\left(1 + \frac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right). \qquad (59)
\end{aligned}
$$

Note that (59) bounds the deviation of the points in $\mathcal{S}$ from their respective means $\mathbb{E}[\boldsymbol{y}_i]$. However, in (9), we need to bound the deviation of the points in $\mathcal{S}$ from their sample mean $\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\boldsymbol{y}_i$. As it turns out, due to our use of local iterations, this will require some technical work.

From the alternate definition of the largest eigenvalue of symmetric matrices $\mathbf{A} \in \mathbb{R}^{d\times d}$, we have

$$
\lambda_{\max}(\mathbf{A}) = \sup_{\boldsymbol{v}\in\mathbb{R}^d, \|\boldsymbol{v}\|=1} \boldsymbol{v}^T\mathbf{A}\boldsymbol{v}. \qquad (60)
$$

Applying this with $\mathbf{A} = \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}(\boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i])(\boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i])^T$, we can equivalently write (59) as

$$
\begin{aligned}
&\sup_{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|=1}\left(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i], \boldsymbol{v}\rangle^2\right) \\
&\qquad \leq \underbrace{\frac{4H^2\sigma^2}{b\epsilon'}\left(1 + \frac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right)}_{=:\,\widehat{\sigma}_0^2}. \qquad (61)
\end{aligned}
$$

Define $\boldsymbol{y}_{\mathcal{S}} := \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\boldsymbol{y}_i$ to be the sample mean of the points in $\mathcal{S}$. Take an arbitrary $\boldsymbol{v} \in \mathbb{R}^d$ such that $\|\boldsymbol{v}\| = 1$.

$$
\begin{aligned}
&\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \boldsymbol{y}_i - \boldsymbol{y}_{\mathcal{S}}, \boldsymbol{v}\rangle^2 \\
&\quad = \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}[\langle \boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i], \boldsymbol{v}\rangle + \langle \mathbb{E}[\boldsymbol{y}_i] - \boldsymbol{y}_{\mathcal{S}}, \boldsymbol{v}\rangle]^2 \\
&\quad \leq \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \boldsymbol{y}_i - \mathbb{E}[\boldsymbol{y}_i], \boldsymbol{v}\rangle^2 + \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \mathbb{E}[\boldsymbol{y}_i] - \boldsymbol{y}_{\mathcal{S}}, \boldsymbol{v}\rangle^2 \\
&\qquad\qquad \text{(using } (a+b)^2 \leq 2a^2 + 2b^2\text{)}
\end{aligned}
$$

Using (61) to bound the first term, we get

$$
\begin{aligned}
&\leq 2\widehat{\sigma}_0^2 + \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\left\langle \mathbb{E}[\boldsymbol{y}_i] - \frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\boldsymbol{y}_j, \boldsymbol{v}\right\rangle^2 \\
&= 2\widehat{\sigma}_0^2 + \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\left[\frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\langle \boldsymbol{y}_j - \mathbb{E}[\boldsymbol{y}_i], \boldsymbol{v}\rangle\right]^2 \\
&\leq 2\widehat{\sigma}_0^2 + \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\langle \boldsymbol{y}_j - \mathbb{E}[\boldsymbol{y}_i], \boldsymbol{v}\rangle^2 \\
&\qquad\qquad \text{(using the Jensen's inequality)} \\
&\leq 2\widehat{\sigma}_0^2 + \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{2}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\langle \boldsymbol{y}_j - \mathbb{E}[\boldsymbol{y}_j], \boldsymbol{v}\rangle^2 \\
&\qquad + \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{2}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\langle \mathbb{E}[\boldsymbol{y}_j] - \mathbb{E}[\boldsymbol{y}_i], \boldsymbol{v}\rangle^2 \\
&\qquad \text{(adding/subtracting } \mathbb{E}[\boldsymbol{y}_j] \text{ and using } (a+b)^2 \leq 2a^2 + 2b^2\text{)} \\
&\leq 2\widehat{\sigma}_0^2 + \frac{4}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\langle \boldsymbol{y}_j - \mathbb{E}[\boldsymbol{y}_j], \boldsymbol{v}\rangle^2 \\
&\qquad + \frac{4}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\|\mathbb{E}[\boldsymbol{y}_j] - \mathbb{E}[\boldsymbol{y}_i]\|^2 \\
&\qquad \text{(using the Cauchy-Schwarz inequality and that } \|\boldsymbol{v}\| \leq 1\text{)} \\
&\leq 6\widehat{\sigma}_0^2 + \frac{4}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\|\mathbb{E}[\boldsymbol{y}_j] - \mathbb{E}[\boldsymbol{y}_i]\|^2 \qquad (62)
\end{aligned}
$$

Claim 8. *For any $r, s \in \mathcal{K}_{t_k}$, we have*

$$
\|\mathbb{E}[\boldsymbol{y}_r] - \mathbb{E}[\boldsymbol{y}_s]\|^2 \leq H\sum_{t=t_k}^{t_{k+1}-1}\left(6\kappa^2 + 3L^2\mathbb{E}\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\|^2\right), \qquad (63)
$$

where expectations in $\mathbb{E}[\boldsymbol{y}_r]$ and $\mathbb{E}[\boldsymbol{y}_s]$ are taken over sampling stochastic gradients between the synchronization indices $t_k, \ldots, t_{k+1}$ by client r and client s, respectively.

Proof. Note that we can equivalently write $\mathbb{E}[\boldsymbol{y}_r] = \mathbb{E}[Y_r]$ and $\mathbb{E}[\boldsymbol{y}_s] = \mathbb{E}[Y_s]$.

$$
\begin{aligned}
\|\mathbb{E}[Y_r] - \mathbb{E}[Y_s]\|^2 &= \|\mathbb{E}[Y_r] - \mathbb{E}[Y_s]\|^2 \\
&= \left\|\sum_{t=t_k}^{t_{k+1}-1}\left(\mathbb{E}[Y_r^t] - \mathbb{E}[Y_s^t]\right)\right\|^2 \\
&\leq (t_{k+1} - t_k)\sum_{t=t_k}^{t_{k+1}-1}\left\|\mathbb{E}[Y_r^t] - \mathbb{E}[Y_s^t]\right\|^2 \qquad (64)
\end{aligned}
$$

By definition of Y_s^t from (58), we have $Y_s^t \sim \text{Unif}\Big(\mathcal{F}_s^{\otimes b}\big(\boldsymbol{x}_s^t\big(\boldsymbol{x}_s^{t_k}, Y_s^{t_k}, \ldots, Y_s^{t-1}\big)\big)\Big)$, which implies using (4) that $\mathbb{E}[Y_s^t] = \mathbb{E}\left[\nabla F_s\big(\boldsymbol{x}_s^t\big(\boldsymbol{x}_s^{t_k}, Y_s^{t_k}, \ldots, Y_s^{t-1}\big)\big)\right]$, where on the RHS, expectation is taken over $(Y_s^{t_k}, \ldots, Y_s^{t-1})$. To make the notation less cluttered, in the following, for any $s \in \mathcal{K}_{t_k}$, we write $\boldsymbol{x}_s^t$ to denote $\boldsymbol{x}_s^t\big(\boldsymbol{x}_s^{t_k}, Y_s^{t_k}, \ldots, Y_s^{t-1}\big)$ with the understanding that expectation is always taken over the sampling of stochastic gradients between t_k and t_{k+1}. With these substitutions, the t'th term from (65) can be written as:

$$\begin{aligned}
\left\|\mathbb{E}[Y_r^t] - \mathbb{E}[Y_s^t]\right\|^2 &= \left\|\mathbb{E}\left[\nabla F_r(\boldsymbol{x}_r^t) - \nabla F_s(\boldsymbol{x}_s^t)\right]\right\|^2 \\
&\overset{(a)}{\leq} \mathbb{E}\left\|\nabla F_r\left(\boldsymbol{x}_r^t\right) - \nabla F_s\left(\boldsymbol{x}_s^t\right)\right\|^2 \qquad (65) \\
&\overset{(b)}{\leq} 3\mathbb{E}\left\|\nabla F_r\left(\boldsymbol{x}_r^t\right) - \nabla F\left(\boldsymbol{x}_r^t\right)\right\|^2 \\
&\quad + 3\mathbb{E}\left\|\nabla F_s\left(\boldsymbol{x}_s^t\right) - \nabla F\left(\boldsymbol{x}_s^t\right)\right\|^2 \\
&\quad + 3\mathbb{E}\left\|\nabla F\left(\boldsymbol{x}_r^t\right) - \nabla F\left(\boldsymbol{x}_s^t\right)\right\|^2 \\
&\overset{(c)}{\leq} 6\kappa^2 + 3L^2\mathbb{E}\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\|^2. \qquad (66)
\end{aligned}$$

Here, (a) and (b) both follow from the Jensen's inequality. (c) used the gradient dissimilarity bound from (6) to bound the first two terms[10] and L-Lipschitzness of ∇F to bound the last term. Substituting the bound from (66) back in (65) and using $(t_{k+1} - t_k) \leq H$ proves Claim 8. □

Using the bound from (63) in (62) gives

$$\begin{aligned}
\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \boldsymbol{y}_i - \boldsymbol{y}_{\mathcal{S}}, \boldsymbol{v}\rangle^2 &\leq 6\widehat{\sigma}_0^2 \\
&\quad + \frac{4}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}} H\sum_{t=t_k}^{t_{k+1}-1}\left(6\kappa^2 + 3L^2\mathbb{E}\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\|^2\right) \\
&= 6\widehat{\sigma}_0^2 + 24H^2\kappa^2 \\
&\quad + \frac{12HL^2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\sum_{t=t_k}^{t_{k+1}-1}\mathbb{E}\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\|^2 \qquad (67)
\end{aligned}$$

Now we bound the last term of (67), which is the drift in local parameters at different clients in between any two synchronization indices.

Lemma 4. *For any $r, s \in \mathcal{K}_{t_k}$, if $\eta \leq \frac{1}{8HL}$, we have*

$$\sum_{t=t_k}^{t_{k+1}-1}\mathbb{E}\left\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\right\|^2 \leq 7H^3\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right), \qquad (68)$$

where expectation is taken over sampling stochastic gradients at clients r, s between the synchronization indices t_k and t_{k+1}.

Proof. For any $t \in [t_k : t_{k+1} - 1]$ and $r, s \in \mathcal{K}_{t_k}$, define $D_{r,s}^t = \mathbb{E}\left\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\right\|^2$. Note that at synchronization time t_k, all clients in the active set $\mathcal{K}_{t_k}$ have the same parameters, i.e., $\boldsymbol{x}_r^{t_k} = \boldsymbol{x}^{t_k}$ for every $r \in \mathcal{K}_{t_k}$. This together with $\boldsymbol{x}_\gamma^{t_k} = \boldsymbol{x}_\gamma^{t_k} - \eta\sum_{j=t_k}^{t-1}\boldsymbol{g}_\gamma(\boldsymbol{x}_\gamma^j)$ (which holds for $\gamma = r, s$), implies

$$\begin{aligned}
D_{r,s}^t &= \mathbb{E}\left\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\right\|^2 \\
&= \eta^2\mathbb{E}\left\|\sum_{j=t_k}^{t-1}\left(\boldsymbol{g}_r(\boldsymbol{x}_r^j) - \boldsymbol{g}_s(\boldsymbol{x}_s^j)\right)\right\|^2 \\
&\qquad\qquad \text{(Since } \boldsymbol{x}_r^{t_k} = \boldsymbol{x}^{t_k}, \forall r \in \mathcal{K}_{t_k}\text{)} \\
&\leq \eta^2(t - t_k)\sum_{j=t_k}^{t-1}\mathbb{E}\left\|\boldsymbol{g}_r(\boldsymbol{x}_r^j) - \boldsymbol{g}_s(\boldsymbol{x}_s^j)\right\|^2 \\
&\leq \eta^2 H\sum_{j=t_k}^{t-1}\Big(3\mathbb{E}\left\|\boldsymbol{g}_r(\boldsymbol{x}_r^j) - \nabla F_r(\boldsymbol{x}_r^j)\right\|^2 \\
&\qquad + 3\mathbb{E}\left\|\boldsymbol{g}_s(\boldsymbol{x}_s^j) - \nabla F_s(\boldsymbol{x}_s^j)\right\|^2 \\
&\qquad + 3\mathbb{E}\left\|\nabla F_r(\boldsymbol{x}_r^j) - \nabla F_s(\boldsymbol{x}_s^j)\right\|^2\Big) \qquad (69)
\end{aligned}$$

To bound the first and the second terms we use the variance bound from (5).[11] We can bound the third term in the same way as we bounded it in (65) and obtained (66). This gives

$$\begin{aligned}
D_{r,s}^t &\leq \eta^2 H\sum_{j=t_k}^{t-1}\left(\frac{6\sigma^2}{b} + 18\kappa^2 + 9L^2\mathbb{E}\|\boldsymbol{x}_r^j - \boldsymbol{x}_s^j\|^2\right) \\
&\leq \frac{6H^2\sigma^2\eta^2}{b} + 18H^2\eta^2\kappa^2 + 9L^2H\eta^2\sum_{j=t_k}^{t-1}D_{r,s}^j \\
&\qquad\qquad \text{(Since } D_{r,s}^j = \mathbb{E}\left\|\boldsymbol{x}_r^j - \boldsymbol{x}_s^j\right\|^2\text{)}
\end{aligned}$$

Taking summation from $t = t_k$ to $t_{k+1} - 1$ gives

$$\begin{aligned}
\sum_{t=t_k}^{t_{k+1}-1}D_{r,s}^t &\leq \sum_{t=t_k}^{t_{k+1}-1}\frac{6H^2\sigma^2\eta^2}{b} + \sum_{t=t_k}^{t_{k+1}-1}18H^2\eta^2\kappa^2 \\
&\qquad + \sum_{t=t_k}^{t_{k+1}-1}9L^2H\eta^2\sum_{j=t_k}^{t-1}D_{r,s}^j \\
&\leq \frac{6H^3\sigma^2\eta^2}{b} + 18H^3\eta^2\kappa^2 \\
&\qquad + 9L^2H^2\eta^2\sum_{t=t_k}^{t_{k+1}-1}D_{r,s}^t.
\end{aligned}$$

After rearranging terms, we get

$$(1 - 9L^2H^2\eta^2)\sum_{t=t_k}^{t_{k+1}-1}D_{r,s}^t \leq \frac{6H^3\sigma^2\eta^2}{b} + 18H^3\eta^2\kappa^2. \qquad (70)$$

If we take $\eta \leq \frac{1}{8HL}$, we get $\left(1 - 9\eta^2L^2H^2\right) \geq \frac{6}{7}$. Substituting this in the LHS of (70) yields $\sum_{t=t_k}^{t_{k+1}-1}D_{r,s}^t \leq \frac{7H^3\sigma^2\eta^2}{b} + 21H^3\eta^2\kappa^2$, which proves Lemma 4. □

Substituting the bound from (68) for the last term in (67) gives

$$\begin{aligned}
\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\langle \boldsymbol{y}_i - \boldsymbol{y}_{\mathcal{S}}, \boldsymbol{v}\rangle^2 &\leq 6\widehat{\sigma}_0^2 + 24H^2\kappa^2 \\
&\quad + \frac{12HL^2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}\left(7H^3\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right)\right)
\end{aligned}$$

[10] Note that though $\boldsymbol{x}_r^t$'s are random quantities, we can still bound $\mathbb{E}\left\|\nabla F_r(\boldsymbol{x}_r^t) - \nabla F_s(\boldsymbol{x}_s^t)\right\|^2 \leq \kappa^2$ because the gradient dissimilarity bound (6) holds uniformly over the entire domain.

[11] Note that $\boldsymbol{x}_r^j$'s are random quantities, however, since the variance bound (5) holds uniformly over the entire domain, we can bound $\mathbb{E}\left\|\boldsymbol{g}_r(\boldsymbol{x}_r^j) - \nabla F_r(\boldsymbol{x}_r^j)\right\|^2 \leq \frac{\sigma^2}{b}$.

$$= 6\widehat{\sigma}_0^2 + 24H^2\kappa^2 + 84H^4L^2\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right)$$
$$\leq 6\widehat{\sigma}_0^2 + 28H^2\kappa^2 + \frac{21H^2\sigma^2}{16b} \qquad \text{(Using } \eta \leq \tfrac{1}{8LH}\text{)}$$
$$\leq \frac{24H^2\sigma^2}{b\epsilon'}\left(1 + \frac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right) + \frac{21H^2\sigma^2}{16b} + 28H^2\kappa^2$$
$$\text{(Since } \widehat{\sigma}_0^2 = \tfrac{4H^2\sigma^2}{b\epsilon'}\left(1 + \tfrac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right)\text{)}$$
$$\leq \frac{25H^2\sigma^2}{b\epsilon'}\left(1 + \frac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right) + 28H^2\kappa^2. \tag{71}$$

In the last inequality we used $\frac{21}{16} \leq \frac{1}{\epsilon'} \leq \frac{1}{\epsilon'}\left(1 + \frac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right)$, where the first inequality follows because $\epsilon' \leq \frac{1}{3\gamma} < \frac{2}{3}$. Note that (71) holds for every unit vector $\boldsymbol{v} \in \mathbb{R}^d$. Using this and substituting $\boldsymbol{g}_{i,\text{accu}}^{t_k,t_{k+1}} = \boldsymbol{y}_i, \boldsymbol{g}_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}} = \boldsymbol{y}_{\mathcal{S}}$ in (71), we get

$$\sup_{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|=1} \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\left\langle \boldsymbol{g}_{i,\text{accu}}^{t_k,t_{k+1}} - \boldsymbol{g}_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2$$
$$\leq \frac{25H^2\sigma^2}{b\epsilon'}\left(1 + \frac{d}{(1-(\epsilon+\gamma\epsilon'))K}\right) + 28H^2\kappa^2.$$

This, in view of the alternate definition of the largest eigenvalue given in (60), is equivalent to (9), which proves Theorem 3.

Appendix C
Omitted Details from Section IV

We prove Claim 1, Claim 2, and Claim 3 below.

A. Proof of Claim 1

Expand the LHS.

$$\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^* - \eta\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2$$
$$= \mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^*\right\|^2 + \eta^2\mathbb{E}\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2$$
$$+ 2\eta\mathbb{E}\left\langle \boldsymbol{x}^* - \boldsymbol{x}^{t_{i+1}-1}, \nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\rangle \tag{72}$$

We can bound the second term on the RHS using L-smoothness of F, which implies that $\|\nabla F(\boldsymbol{x})\|^2 \leq 2L(F(\boldsymbol{x}) - F(\boldsymbol{x}^*))$ holds for every $\boldsymbol{x} \in \mathbb{R}^d$; see Fact 1 on page 26. We can bound the third term on the RHS using μ-strong convexity of F as follows: $\left\langle \boldsymbol{x}^* - \boldsymbol{x}^{t_{i+1}-1}, \nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\rangle \leq F(\boldsymbol{x}^*) - F(\boldsymbol{x}^{t_{i+1}-1}) - \frac{\mu}{2}\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^*\|^2$. Substituting these back in (72) gives:

$$\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^* - \eta\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2$$
$$\leq (1-\mu\eta)\,\mathbb{E}\left\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^*\right\|^2$$
$$- 2\eta(1-\eta L)\mathbb{E}\left(F(\boldsymbol{x}^{t_{i+1}-1}) - F(\boldsymbol{x}^*)\right) \tag{73}$$

Since $\eta < \frac{1}{L}$, we have $(1-\eta L) > 0$. We also have $F(\boldsymbol{x}^{t_{i+1}-1}) \geq F(\boldsymbol{x}^*)$. Using these together, we can ignore the last term in the RHS of (73). This proves Claim 1.

B. Proof of Claim 2

By definition, we have $\boldsymbol{x}^{t_{i+1}-1} = \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}} \boldsymbol{x}^{t_{i+1}-1}$.

$$\mathbb{E}\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1}) - \nabla F(\boldsymbol{x}^{t_{i+1}-1})\right)\right\|^2$$
$$\leq \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1}) - \nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2$$
$$\leq \frac{2}{K}\sum_{r\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1}) - \nabla F(\boldsymbol{x}_r^{t_{i+1}-1})\right\|^2$$
$$+ \frac{2}{K}\sum_{r\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\nabla F(\boldsymbol{x}_r^{t_{i+1}-1}) - \nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2$$
$$\overset{(a)}{\leq} \frac{2}{K}\sum_{r\in\mathcal{K}_{t_i}}\kappa^2 + \frac{2}{K}\sum_{r\in\mathcal{K}_{t_i}}L^2\mathbb{E}\left\|\boldsymbol{x}_r^{t_{i+1}-1} - \boldsymbol{x}^{t_{i+1}-1}\right\|^2$$
$$= 2\kappa^2 + \frac{2L^2}{K}\sum_{r\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\boldsymbol{x}_r^{t_{i+1}-1} - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\boldsymbol{x}_s^{t_{i+1}-1}\right\|^2$$
$$\leq 2\kappa^2 + \frac{2L^2}{K}\sum_{r\in\mathcal{K}_{t_i}}\frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\boldsymbol{x}_r^{t_{i+1}-1} - \boldsymbol{x}_s^{t_{i+1}-1}\right\|^2 \tag{74}$$
$$\overset{(b)}{\leq} 2\kappa^2 + \frac{2L^2}{K}\sum_{r\in\mathcal{K}_{t_i}}\frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\left(7H^3\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right)\right)$$
$$= 2\kappa^2 + 14L^2H^3\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right)$$
$$\overset{(c)}{\leq} 2\kappa^2 + \frac{7H}{32}\left(\frac{\sigma^2}{b} + 3\kappa^2\right)$$

In (a) we used the gradient dissimilarity bound from (6) to bound the first term and L-Lipschitz gradient property of F to bound the second term. For (b), note that we have already bounded $\sum_{t=t_i}^{t_{i+1}-1}\mathbb{E}\left\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\right\|^2 \leq 7H^3\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right)$ in (68) in Lemma 4. Since each term in the summation is trivially bounded by the same quantity, which we used in (b) to bound $\mathbb{E}\left\|\boldsymbol{x}_r^{t_{i+1}-1} - \boldsymbol{x}_s^{t_{i+1}-1}\right\|^2 \leq 7H^3\eta^2\left(\frac{\sigma^2}{b} + 3\kappa^2\right)$. In (c) we used $\eta \leq \frac{1}{8HL}$.

C. Proof of Claim 3

Let $\mathcal{S} \subseteq \mathcal{K}_{t_i}$ denote the subset of honest clients of size $(1-(\epsilon+\epsilon'))K$, whose average accumulated gradient between time t_i and t_{i+1} that server approximates at time t_{i+1} in Theorem 3. Let the average accumulated gradient be denoted by $\boldsymbol{g}_{\mathcal{S},\text{accu}}^{t_i,t_{i+1}} = \frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\boldsymbol{g}_{r,\text{accu}}^{t_i,t_{i+1}}$, where $\boldsymbol{g}_{r,\text{accu}}^{t_i,t_{i+1}} = \sum_{t=t_i}^{t_{i+1}-1}\boldsymbol{g}_r(\boldsymbol{x}_r^t)$, and server approximates it by $\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}}$. Note that $\mathcal{S}$ exists with probability at least $1 - \exp\left(-\frac{(2\gamma-1)^2\epsilon'^2(1-\epsilon)K}{8}\right)$. To make the notation less cluttered, for every $r \in \mathcal{K}_{t_i}$, define $\nabla F_r^{t_i,t_{i+1}} := \sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)$.

$$\mathbb{E}\left\|\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\nabla F_r^{t_i,t_{i+1}}\right\|^2$$
$$\leq 3\mathbb{E}\left\|\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\boldsymbol{g}_{r,\text{accu}}^{t_i,t_{i+1}}\right\|^2$$

$$+ 3\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}} \boldsymbol{g}_{r,\text{accu}}^{t_i,t_{i+1}} - \frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}} \nabla F_r^{t_i,t_{i+1}}\right\|^2$$
$$+ 3\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}} \nabla F_r^{t_i,t_{i+1}} - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}} \nabla F_s^{t_i,t_{i+1}}\right\|^2 \quad (75)$$

Now we bound each term on the RHS of (75).

Bounding the first term on the RHS of (75). We can bound this using the second part of Theorem 3 as follows (note that given the first part of Theorem 3 is satisfied, the second part provides deterministic approximation guarantees, which implies that it also holds in expectation):

$$\mathbb{E}\left\|\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}} \boldsymbol{g}_{r,\text{accu}}^{t_i,t_{i+1}}\right\|^2 \leq \Upsilon^2, \quad (76)$$

where $\Upsilon^2 = \mathcal{O}\left(\sigma_0^2(\epsilon+\epsilon')\right)$ and $\sigma_0^2 = \frac{25H^2\sigma^2}{b\epsilon'}\left(1+\frac{3d}{2K}\right) + 28H^2\kappa^2$.

Bounding the second term on the RHS of (75). We can bound this using the variance bound (5).

$$\begin{aligned}
&\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\left(\boldsymbol{g}_{r,\text{accu}}^{t_i,t_{i+1}} - \nabla F_r^{t_i,t_{i+1}}\right)\right\|^2 \\
&= \mathbb{E}\left\|\sum_{t=t_i}^{t_{i+1}-1}\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\left(\boldsymbol{g}_r(\boldsymbol{x}_r^t) - \nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2 \\
&\overset{(a)}{\leq} (t_{i+1}-t_i)\sum_{t=t_i}^{t_{i+1}-1}\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\left(\boldsymbol{g}_r(\boldsymbol{x}_r^t) - \nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2 \\
&\overset{(b)}{\leq} H\sum_{t=t_i}^{t_{i+1}-1}\frac{1}{|\mathcal{S}|^2}\mathbb{E}\left\|\sum_{r\in\mathcal{S}}\left(\boldsymbol{g}_r(\boldsymbol{x}_r^t) - \nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2 \\
&\overset{(c)}{=} H\sum_{t=t_i}^{t_{i+1}-1}\frac{1}{|\mathcal{S}|^2}\sum_{r\in\mathcal{S}}\mathbb{E}\left\|\boldsymbol{g}_r(\boldsymbol{x}_r^t) - \nabla F_r(\boldsymbol{x}_r^t)\right\|^2 \\
&\overset{(d)}{\leq} H\sum_{t=t_i}^{t_{i+1}-1}\frac{1}{|\mathcal{S}|}\frac{\sigma^2}{b} \\
&\overset{(e)}{\leq} \frac{4H^2\sigma^2}{3bK}.
\end{aligned} \quad (77)$$

In (a) we used the Jensen's inequality. In (b) used $|t_{i+1}-t_i| \leq H$. In (c) we used (4) (which states that $\mathbb{E}[\boldsymbol{g}_r(\boldsymbol{x})] = \nabla F_r(\boldsymbol{x})$ holds for every honest client $r \in [R]$ and $\boldsymbol{x} \in \mathbb{R}^d$) together with that the stochastic gradients at different clients are sampled independently, and then we used the fact that the variance of independent random variables is equal to the sum of the variances. Note that $\text{Var}(\boldsymbol{g}_r(\boldsymbol{x}_r^t)) = \mathbb{E}\left\|\boldsymbol{g}_r(\boldsymbol{x}_r^t) - \nabla F_r(\boldsymbol{x}_r^t)\right\|^2$. In (d) we used the variance bound (5). In (e) we used $|\mathcal{S}| \geq (1-(\epsilon+\gamma\epsilon'))K \geq \frac{2K}{3}$, where the last inequality uses $(\epsilon+\gamma\epsilon') \leq \frac{1}{3}$.

Bounding the third term on the RHS of (75).

$$\begin{aligned}
&\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F_r^{t_i,t_{i+1}} - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\nabla F_s^{t_i,t_{i+1}}\right\|^2 \\
&= \mathbb{E}\left\|\sum_{t=t_i}^{t_{i+1}-1}\left(\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F_r(\boldsymbol{x}_r^t) - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\nabla F_s(\boldsymbol{x}_s^t)\right)\right\|^2 \\
&\overset{(a)}{\leq} H\sum_{t=t_i}^{t_{i+1}-1}\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F_r(\boldsymbol{x}_r^t) - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\nabla F_s(\boldsymbol{x}_s^t)\right\|^2
\end{aligned} \quad (78)$$

In (a), first we used the Jensen's inequality and then substituted $|t_{i+1}-t_i| \leq H$. In order to bound (78), it suffices to bound $\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F_r(\boldsymbol{x}_r^t) - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\nabla F_s(\boldsymbol{x}_s^t)\right\|^2$ for every $t \in [t_i : t_{i+1}-1]$. We bound this in the following. Take an arbitrary $t \in [t_i : t_{i+1}-1]$.

$$\begin{aligned}
&\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F_r(\boldsymbol{x}_r^t) - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\nabla F_s(\boldsymbol{x}_s^t)\right\|^2 \\
&\leq 3\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\left(\nabla F_r(\boldsymbol{x}_r^t) - \nabla F(\boldsymbol{x}_r^t)\right)\right\|^2 \\
&\quad + 3\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F(\boldsymbol{x}_r^t) - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\nabla F(\boldsymbol{x}_s^t)\right\|^2 \\
&\quad + 3\mathbb{E}\left\|\frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}_s^t) - \nabla F_s(\boldsymbol{x}_s^t)\right)\right\|^2
\end{aligned}$$

The first and third terms can both be bounded by $3\kappa^2$ (using Jensen's inequality and bounded gradient dissimilarity bound (6)). For the second term, we can add and subtract $\nabla F(\boldsymbol{x}^t) = \frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F(\boldsymbol{x}^t) = \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\nabla F(\boldsymbol{x}^t)$ and then using $\|\boldsymbol{a}+\boldsymbol{b}\|^2 \leq 2\|\boldsymbol{a}\|^2 + 2\|\boldsymbol{b}\|^2$, we get the following

$$\begin{aligned}
&\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F_r^{t_i,t_{i+1}} - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\nabla F_s^{t_i,t_{i+1}}\right\| \\
&\leq 3\kappa^2 + 3\kappa^2 + 6\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F(\boldsymbol{x}_r^t) - \nabla F(\boldsymbol{x}^t)\right\|^2 \\
&\quad + 6\mathbb{E}\left\|\frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}_s^t) - \nabla F(\boldsymbol{x}^t)\right)\right\|^2 \\
&\leq 6\kappa^2 + \frac{6}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\mathbb{E}\left\|\nabla F(\boldsymbol{x}_r^t) - \nabla F(\boldsymbol{x}^t)\right\|^2 \\
&\quad + \frac{6}{K}\sum_{s\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\nabla F(\boldsymbol{x}_s^t) - \nabla F(\boldsymbol{x}^t)\right\|^2 \\
&\leq 6\kappa^2 + \frac{6}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}L^2\mathbb{E}\left\|\boldsymbol{x}_r^t - \boldsymbol{x}^t\right\|^2 \\
&\quad + \frac{6}{K}\sum_{s\in\mathcal{K}_{t_i}}L^2\mathbb{E}\left\|\boldsymbol{x}_s^t - \boldsymbol{x}^t\right\|^2 \\
&= 6\kappa^2 + \frac{6L^2}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\mathbb{E}\left\|\boldsymbol{x}_r^t - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\boldsymbol{x}_s^t\right\|^2 \\
&\quad + \frac{6L^2}{K}\sum_{r\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\boldsymbol{x}_r^t - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\boldsymbol{x}_s^t\right\|^2
\end{aligned}$$

$$\leq 6\kappa^2 + \frac{6L^2}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\boldsymbol{x}_r^t-\boldsymbol{x}_s^t\right\|^2 + \frac{6L^2}{K}\sum_{r\in\mathcal{K}_{t_i}}\frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\boldsymbol{x}_r^t-\boldsymbol{x}_s^t\right\|^2$$

Substituting this back in (78) gives:

$$\begin{aligned}
\mathbb{E}\left\|\frac{1}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\nabla F_r^{t_i,t_{i+1}} - \frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\nabla F_s^{t_i,t_{i+1}}\right\|^2 &\leq H\sum_{t=t_i}^{t_{i+1}-1}6\kappa^2 \\
&+ H\sum_{t=t_i}^{t_{i+1}-1}\frac{6L^2}{|\mathcal{S}|}\sum_{r\in\mathcal{S}}\frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\boldsymbol{x}_r^t-\boldsymbol{x}_s^t\right\|^2 \\
&+ H\sum_{t=t_i}^{t_{i+1}-1}\frac{6L^2}{K}\sum_{r\in\mathcal{K}_{t_i}}\frac{1}{K}\sum_{s\in\mathcal{K}_{t_i}}\mathbb{E}\left\|\boldsymbol{x}_r^t-\boldsymbol{x}_s^t\right\|^2 \\
&\overset{(a)}{\leq} 6H^2\kappa^2 + 6HL^2\left(7H^3\eta^2\left(\frac{\sigma^2}{b}+3\kappa^2\right)\right) \\
&\quad + 6HL^2\left(7H^3\eta^2\left(\frac{\sigma^2}{b}+3\kappa^2\right)\right) \\
&= 6H^2\kappa^2 + 84L^2H^4\eta^2\left(\frac{\sigma^2}{b}+3\kappa^2\right) \\
&\overset{(b)}{\leq} 10H^2\kappa^2 + \frac{21H^2\sigma^2}{16b}. \qquad (79)
\end{aligned}$$

In (a) we used $t_{i+1}-t_i \leq H$ and the bound $\sum_{t=t_i}^{t_{i+1}-1}\mathbb{E}\left\|\boldsymbol{x}_r^t-\boldsymbol{x}_s^t\right\|^2 \leq 7H^3\eta^2\left(\frac{\sigma^2}{b}+3\kappa^2\right)$, which holds when $\eta \leq \frac{1}{8HL}$; we have already shown this in (68) in Lemma 4. In (b) we used $\eta \leq \frac{1}{8HL}$.

Substituting the bounds from (76), (77), (79) in (75) gives

$$\begin{aligned}
\mathbb{E}\left\|\widehat{\boldsymbol{g}}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\nabla F_r^{t_i,t_{i+1}}\right\|^2 &\leq 3\Upsilon^2 + \frac{4H^2\sigma^2}{bK} + 3\left(10H^2\kappa^2+\frac{21H^2\sigma^2}{16b}\right) \\
&\leq 3\Upsilon^2 + \frac{4H^2\sigma^2}{bK} + 30H^2\kappa^2 + \frac{4H^2\sigma^2}{b} \\
&= 3\Upsilon^2 + \frac{8H^2\sigma^2}{b} + 30H^2\kappa^2,
\end{aligned}$$

where $\Upsilon^2 = \mathcal{O}\left(\sigma_0^2(\epsilon+\epsilon')\right)$ and $\sigma_0^2 = \frac{25H^2\sigma^2}{b\epsilon'}\left(1+\frac{3d}{2K}\right) + 28H^2\kappa^2$.

This completes the proof of Claim 3.

D. A Useful Fact

Fact 1. *Let $F:\mathbb{R}^d\to\mathbb{R}$ be an L-smooth function with a global minimizer $\boldsymbol{x}^*$. Then, for every $\boldsymbol{x}\in\mathbb{R}^d$, we have*

$$\|\nabla F(\boldsymbol{x})\|^2 \leq 2L(F(\boldsymbol{x})-F(\boldsymbol{x}^*)).$$

Proof. By definition of L-smoothness, we have $F(\boldsymbol{y}) \leq F(\boldsymbol{x}) + \langle\nabla F(\boldsymbol{x}), \boldsymbol{y}-\boldsymbol{x}\rangle + \frac{L}{2}\|\boldsymbol{y}-\boldsymbol{x}\|^2$ holds for every $\boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^d$. Fix an arbitrary $\boldsymbol{x}\in\mathbb{R}^d$ and take infimum over $\boldsymbol{y}$ on both sides:

$$\begin{aligned}
\inf_{\boldsymbol{y}} F(\boldsymbol{y}) &\leq \inf_{\boldsymbol{y}}\left(F(\boldsymbol{x}) + \langle\nabla F(\boldsymbol{x}), \boldsymbol{y}-\boldsymbol{x}\rangle + \frac{L}{2}\|\boldsymbol{y}-\boldsymbol{x}\|^2\right) \\
&\overset{(a)}{=} \inf_{\boldsymbol{v}:\|\boldsymbol{v}\|=1}\inf_{t}\left(F(\boldsymbol{x}) + t\langle\nabla F(\boldsymbol{x}),\boldsymbol{v}\rangle + \frac{Lt^2}{2}\right) \\
&\overset{(b)}{=} \inf_{\boldsymbol{v}:\|\boldsymbol{v}\|=1}\left(F(\boldsymbol{x}) - \frac{1}{2L}\langle\nabla F(\boldsymbol{x}),\boldsymbol{v}\rangle^2\right) \\
&\overset{(c)}{=} \left(F(\boldsymbol{x}) - \frac{1}{2L}\|\nabla F(\boldsymbol{x})\|^2\right)
\end{aligned}$$

The value of t that minimizes the RHS of (a) is $t = -\frac{1}{L}\langle\nabla F(\boldsymbol{x}),\boldsymbol{v}\rangle$, this implies (b); (c) follows from the Cauchy-Schwarz inequality: $\langle\boldsymbol{u},\boldsymbol{v}\rangle \leq \|\boldsymbol{u}\|\|\boldsymbol{v}\|$, where equality is achieved whenever $\boldsymbol{u}=\boldsymbol{v}$. Now, substituting $\inf_{\boldsymbol{y}} F(\boldsymbol{y}) = F(\boldsymbol{x}^*)$ yields the result. □

APPENDIX D
FULL-BATCH LOCAL GRADIENT DESCENT – PROOF OF THEOREM 2

In this section, we focus on the case when in each local iteration clients compute *full-batch* gradients (instead of computing mini-batch stochastic gradients) in Algorithm 1 and prove Theorem 2.

Note that the robust accumulated gradient estimation (RAGE) result of Theorem 3 (which is for stochastic gradients) is one of the main ingredients behind the convergence analyses of Theorem 1. So, in order to prove Theorem 2, first we need to show a RAGE result for full-batch gradients. Note that we can obtain such a result by substituting $\sigma=0$ in both the parts of Theorem 3; however, this would give a loose bound on the approximation error in the second part. In the following, we get a tighter bound (both for RAGE and the convergence rates in Theorem 2) by working directly with full-batch gradients. To get a RAGE result for full-batch gradients, we do a much simplified analysis than what we did before to prove Theorem 3, and the resulting result is stated and proved below in Theorem 7.

Note that, in order to prove Theorem 3, we showed an existence of a subset $\mathcal{S}$ of honest clients (from the set $\mathcal{K}$ of clients who communicate with the server) from whom the accumulated stochastic gradients are well-concentrated, as stated in form of a matrix concentration bound (9) in Theorem 3. It turns out that for full-batch gradients, an analogous result can be proven directly (as there is no randomness due to stochastic gradients); and below we provide such a result. Note that Theorem 3 is a probabilistic statement, where we show that with high probability, there exists a large subset $\mathcal{S}\subseteq\mathcal{K}$ of honest clients whose stochastic accumulated gradients are well-concentrated. In contrast, in the following result, we can deterministically take the set of *all* honest clients in $\mathcal{K}$ to be that subset for which we can directly show the concentration.

First we setup the notation to state our main result on RAGE for full-batch gradients. Let $\mathcal{K}_t \subseteq [R]$ denote the subset of clients of size K that are active at any time $t\in[0:T]$. Let Algorithm 1 generate a sequence of iterates $\{\boldsymbol{x}_r^t : t\in[0:T], r\in\mathcal{K}_t\}$ when run with a fixed step-size η satisfying $\eta\leq\frac{1}{5HL}$ while minimizing a global objective function $F:\mathbb{R}^d\to\mathbb{R}$, where in any iteration, instead of sampling mini-batch stochastic gradients, every honest client takes full-batch gradients from their local datasets. Take any

two consecutive synchronization indices $t_k, t_{k+1} \in \mathcal{I}_T$. Note that $|t_{k+1} - t_k| \le H$. For an honest client $r \in \mathcal{K}_{t_k}$, let $\nabla F_{r,\text{accu}}^{t_k,t_{k+1}} := \sum_{t=t_k}^{t_{k+1}-1} \nabla F_r(\boldsymbol{x}_r^t)$ denote the sum of local full-batch gradients taken by client r between time t_k and t_{k+1}. Note that at iteration t_{k+1}, every honest client $r \in \mathcal{K}_{t_k}$ reports its local parameters $\boldsymbol{x}_r^{t_{k+1}}$ to the server, from which server can compute $\nabla F_{r,\text{accu}}^{t_k,t_{k+1}}$, whereas, corrupt clients may report arbitrary and adversarially chosen vectors in $\mathbb{R}^d$. The goal of the server is to produce an estimate $\nabla \widehat{F}_{\text{accu}}^{t_k,t_{k+1}}$ of the average accumulated gradients from honest clients as best as possible.

Theorem 7 (Robust Accumulated Gradient Estimation for Full-Batch Gradient Descent)**.** *Suppose an ϵ fraction of clients who communicate with the server are corrupt. In the setting and notation described above, suppose we are given $K \le R$ accumulated full-batch gradients $\nabla \widetilde{F}_{r,\text{accu}}^{t_k,t_{k+1}}, r \in \mathcal{K}_{t_k}$ in $\mathbb{R}^d$, where $\nabla \widetilde{F}_{r,\text{accu}}^{t_k,t_{k+1}} = \nabla F_{r,\text{accu}}^{t_k,t_{k+1}}$ if the r'th client is honest, otherwise can be arbitrary. Let $\mathcal{S} \subseteq \mathcal{K}_{t_k}$ be the subset of* all *honest clients in $\mathcal{K}_{t_k}$ and $\nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}} := \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \nabla F_{i,\text{accu}}^{t_k,t_{k+1}}$ be the sample average of uncorrupted full-batch gradients. If $\epsilon \le \frac{1}{3}$, then with probability* 1*, we can find an estimate $\nabla \widehat{F}_{\text{accu}}^{t_k,t_{k+1}}$ of $\nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}$ in polynomial-time, such that $\left\|\nabla \widehat{F}_{\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}\right\| \le \mathcal{O}\left(H\kappa\sqrt{\epsilon}\right)$.*

Proof. Let $\Xi := \left(\nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}\right)$. First we prove that

$$\lambda_{\max}\Big(\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \Xi\,\Xi^T\Big) \le 11H^2\kappa^2. \tag{80}$$

In view of the alternate characterization the largest eigenvalue given in (60), this is equivalent to showing

$$\sup_{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|=1} \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \le 11H^2\kappa^2, \tag{81}$$

which we prove below. Define $F_{\text{accu}}^{t_k,t_{k+1}} := \sum_{t=t_k}^{t_{k+1}-1} F(\boldsymbol{x}^t)$, where $\boldsymbol{x}^t = \frac{1}{K}\sum_{r\in\mathcal{K}_{t_k}} \boldsymbol{x}_r^t$ for any $t \in [t_k : t_{k+1} - 1]$. Take an arbitrary unit vector $\boldsymbol{v} \in \mathbb{R}^d$.

$$\begin{aligned}
&\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \\
&\le \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \\
&\qquad + \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \\
&\qquad\qquad \text{(Using } \|a+b\|^2 \le 2\|a\|^2 + 2\|b\|^2\text{)} \\
&= \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \\
&\qquad + 2\left\langle \nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \\
&= \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \\
&\qquad + 2\left[\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle\right]^2 \\
&\le \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \\
&\qquad + \frac{2}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \\
&= \frac{4}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 \\
&\le \frac{4}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\| \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\text{accu}}^{t_k,t_{k+1}}\right\|^2 \\
&\qquad\qquad \text{(Using } \langle \boldsymbol{u},\boldsymbol{v}\rangle \le \|\boldsymbol{u}\|\|\boldsymbol{v}\| \text{ and that } \|\boldsymbol{v}\| = 1\text{)} \\
&= \frac{4}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\| \sum_{t=t_k}^{t_{k+1}-1} \left(\nabla F_i(\boldsymbol{x}_i^t) - \nabla F(\boldsymbol{x}^t)\right)\right\|^2 \\
&\qquad\qquad \text{(Since } F_{\text{accu}}^{t_k,t_{k+1}} = \textstyle\sum_{t=t_k}^{t_{k+1}-1} F(\boldsymbol{x}^t)\text{)} \\
&\le \frac{4}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} (t_{k+1}-t_k) \sum_{t=t_k}^{t_{k+1}-1} \left\|\nabla F_i(\boldsymbol{x}_i^t) - \nabla F(\boldsymbol{x}^t)\right\|^2 \\
&\qquad\qquad \text{(Using Jensen's inequality)} \\
&\le \frac{4H}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \sum_{t=t_k}^{t_{k+1}-1} \Big(2\left\|\nabla F_i(\boldsymbol{x}_i^t) - \nabla F(\boldsymbol{x}_i^t)\right\|^2 \\
&\qquad\qquad +2\left\|\nabla F(\boldsymbol{x}_i^t) - \nabla F(\boldsymbol{x}^t)\right\|^2\Big) \\
&\overset{\text{(a)}}{\le} \frac{4H}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \sum_{t=t_k}^{t_{k+1}-1} \left(2\kappa^2 + 2L^2\left\|\boldsymbol{x}_i^t - \boldsymbol{x}^t\right\|^2\right) \\
&\le 8H^2\kappa^2 + 8HL^2 \sum_{t=t_k}^{t_{k+1}-1} \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \left\|\boldsymbol{x}_i^t - \frac{1}{K}\sum_{j\in\mathcal{K}_{t_k}} \boldsymbol{x}_j^t\right\|^2 \\
&\qquad\qquad \text{(Since } \boldsymbol{x}^t = \tfrac{1}{K}\textstyle\sum_{j\in\mathcal{K}_{t_k}} \boldsymbol{x}_j^t\text{)} \\
&\le 8H^2\kappa^2 + 8HL^2 \sum_{t=t_k}^{t_{k+1}-1} \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \frac{1}{K}\sum_{j\in\mathcal{K}_{t_k}} \left\|\boldsymbol{x}_i^t - \boldsymbol{x}_j^t\right\|^2
\end{aligned} \tag{82}$$

The last inequality follows from the Jensen's inequality. In (a) we used (6) to bound $\|\nabla F_i(\boldsymbol{x}_i^t) - \nabla F(\boldsymbol{x}_i^t)\|^2 \le \kappa^2$ and L-Lipschitz gradient property of F to bound $\|\nabla F(\boldsymbol{x}_i^t) - \nabla F(\boldsymbol{x}^t)\| \le L\|\boldsymbol{x}_i^t - \boldsymbol{x}^t\|$.

Now we bound the last term of (82).

Lemma 5. *For any $r, s \in \mathcal{K}_{t_k}$, if $\eta \le \frac{1}{5HL}$, we have*

$$\sum_{t=t_k}^{t_{k+1}-1} \left\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\right\|^2 \le 7\eta^2H^3\kappa^2. \tag{83}$$

Proof. Note that we have shown a similar result (but, in expectation) in Lemma 4 (on page 23), which is for stochastic gradients. We will simplify that proof to prove Lemma 5, which is for full-batch deterministic gradients.

Take an arbitrary $t \in [t_k : t_{k+1} - 1]$. Following the proof of Lemma 4 until (69) and removing the factor of 3 inside the summation (the factor of 3 appeared because we applied the Jensen's inequality earlier to separate the deterministic gradient term and the stochastic gradient terms) would give

$$\left\|\boldsymbol{x}_r^t - \boldsymbol{x}_s^t\right\|^2 \le \eta^2 H \sum_{j=t_k}^{t-1} \left\|\nabla F_r(\boldsymbol{x}_r^j) - \nabla F_s(\boldsymbol{x}_s^j)\right\|^2. \tag{84}$$

Following the remaining proof of Lemma 4 from (69) until the end and substituting $\sigma = 0$ gives the desired result. □

Substituting the bound from (83) into (82) gives

$$\begin{aligned}\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}\left\langle \nabla F_{i,\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}, \boldsymbol{v}\right\rangle^2 &\le 8H^2\kappa^2 + 56H^4L^2\eta^2\kappa^2 \\ &\le 8H^2\kappa^2 + \frac{56}{25}H^2\kappa^2 \qquad \text{(Substituting } \eta \le \tfrac{1}{5HL}\text{)} \\ &\le 11H^2\kappa^2. \end{aligned} \tag{85}$$

Note that (85) holds for an arbitrary unit vector $\boldsymbol{v} \in \mathbb{R}^d$, implying that (81) holds true. Since (81) and (80) are equivalent, we have thus shown (80).

Now apply Theorem 4 with $\mathcal{S}$ being the set of all honest clients, and $\boldsymbol{g}_{i,\text{accu}}^{t_k,t_{k+1}} = \nabla F_{i,\text{accu}}^{t_k,t_{k+1}}$, $\boldsymbol{g}_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}} = \nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}$ $\widehat{\boldsymbol{g}}_{\text{accu}}^{t_k,t_{k+1}} = \nabla \widehat{F}_{\text{accu}}^{t_k,t_{k+1}}$, $\epsilon' = 0$, and $\sigma_0^2 = 11H^2\kappa^2$. We would get that we can find an estimate $\nabla \widehat{F}_{\text{accu}}^{t_k,t_{k+1}}$ of $\nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}$ in polynomial-time, such that $\left\|\nabla \widehat{F}_{\text{accu}}^{t_k,t_{k+1}} - \nabla F_{\mathcal{S},\text{accu}}^{t_k,t_{k+1}}\right\| \le \mathcal{O}\left(H\kappa\sqrt{\epsilon}\right)$ holds with probability 1. □

Theorem 2 can be proved with appropriate modifications in the proof of Theorem 1, and for completeness, we prove it below.

A. Convergence Proof of the Strongly-Convex Part of Theorem 2

Let $\mathcal{K}_t \subseteq [R]$ denote the subset of clients of size $|\mathcal{K}_t| = K$ that are active at the t'th iteration. For any $t \in [t_i : t_{i+1}-1]$, let $\boldsymbol{x}^t = \frac{1}{K}\sum_{k\in\mathcal{K}_{t_i}} \boldsymbol{x}_k^t$ denote the average of the local parameters of clients in the sampling set $\mathcal{K}_{t_i}$.

Following the proof of the strongly-convex part of Theorem 1 given in Appendix C until (19) gives

$$\begin{aligned}&\left\|\boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^*\right\|^2 \\ &\le \left(1+\frac{\mu\eta}{2}\right)\left\|\boldsymbol{x}^{t_{i+1}-1} - \eta\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \boldsymbol{x}^*\right\|^2 \\ &+ 2\eta\left(\eta+\frac{2}{\mu}\right)\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1})\right)\right\|^2 \\ &+ 2\eta\left(\eta+\frac{2}{\mu}\right)\left\|\widehat{F}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right\|^2 \end{aligned} \tag{86}$$

We have already bounded the first term in Claim 1 (on page 9) by

$$\begin{aligned}&\left\|\boldsymbol{x}^{t_{i+1}} - \eta\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \boldsymbol{x}^*\right\|^2 \\ &\qquad\le (1-\eta\mu)\left\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^*\right\|^2. \end{aligned} \tag{87}$$

In order to bound the second term, we follow the proof of Claim 2 exactly until (74), and then to bound $\left\|\boldsymbol{x}_r^{t_{i+1}-1} - \boldsymbol{x}_s^{t_{i+1}-1}\right\|^2$ for every $r, s \in \mathcal{K}_{t_i}$, we use the bound from (83) in Lemma 5 and use $\eta \le \frac{1}{5HL}$, which gives

$$\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F_r(\boldsymbol{x}^{t_{i+1}-1}) - \nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1})\right)\right\|^2 \le 3H\kappa^2. \tag{88}$$

To bound the third term in the RHS of (86), we can simplify the proof of Claim 3: Firstly, note that with full-batch gradients, the variance σ^2 becomes zero; secondly, as shown in Theorem 7, the robust estimation of accumulated gradients holds with probability 1. Following the proof of Claim 3 with these changes and using $\eta \le \frac{1}{5HL}$, we get

$$\left\|\widehat{F}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right\|^2 \le 2\Upsilon_{\text{GD}}^2 + 20H^2\kappa^2, \tag{89}$$

where $\Upsilon_{\text{GD}} = \mathcal{O}\left(H\kappa\sqrt{\epsilon}\right)$. Substituting all these bounds from (87)-(89) into (86) and simplifying further using $\left(1+\frac{\mu\eta}{2}\right)(1-\mu\eta) \le \left(1-\frac{\mu\eta}{2}\right)$ and $\left(\eta+\frac{2}{\mu}\right) \le \frac{3}{\mu}$ gives

$$\begin{aligned}\left\|\boldsymbol{x}^{t_{i+1}} - \boldsymbol{x}^*\right\|^2 &\le \left(1-\frac{\mu\eta}{2}\right)\left\|\boldsymbol{x}^{t_{i+1}-1} - \boldsymbol{x}^*\right\|^2 \\ &\quad + \frac{6\eta}{\mu}\left(2\Upsilon_{\text{GD}}^2 + 23H^2\kappa^2\right) \end{aligned} \tag{90}$$

Note that (90) gives a recurrence at the synchronization indices. Now we give a recurrence at non-synchronization indices. Take an arbitrary $t \in [T]$ and let $t_i \in \mathcal{I}_T$ be such that $t \in [t_i : t_{i+1} - 1]$; when $H \ge 2$, such t's exist. Following the steps that we used to arrive at (25), we get the following (note that the last term on the RHS of (25) is zero, as $\boldsymbol{g}_r(\boldsymbol{x}_r^t) = \nabla F_r(\boldsymbol{x}_r^t)$ holds for every $r \in [R]$ and $t \in [T]$; this will also save us the factor of 2 in the previous term as we don't have to use the Jensen's inequality to get to (25)):

$$\begin{aligned}\left\|\boldsymbol{x}^{t+1} - \boldsymbol{x}^*\right\|^2 &\le \left(1+\frac{\mu\eta}{2}\right)\left\|\boldsymbol{x}^t - \boldsymbol{x}^* - \eta\nabla F(\boldsymbol{x}^t)\right\|^2 \\ &\quad + \eta\left(\eta+\frac{2}{\mu}\right)\left\|\frac{1}{K}\sum_{r\in\mathcal{K}_t}\left(\nabla F(\boldsymbol{x}^t) - \nabla F_r(\boldsymbol{x}_r^t)\right)\right\|^2 \end{aligned} \tag{91}$$

Substituting the bounds from (87) and (88) into (91) and simplifying the coefficients as above, we get

$$\left\|\boldsymbol{x}^{t+1} - \boldsymbol{x}^*\right\|^2 \le \left(1-\frac{\mu\eta}{2}\right)\left\|\boldsymbol{x}^t - \boldsymbol{x}^*\right\|^2 + \frac{3\eta}{\mu}(3H\kappa^2) \tag{92}$$

Now we have a recurrence at the synchronization indices given in (90) and at non-synchronization indices given in (92). Let $\alpha = \left(1-\frac{\mu\eta}{2}\right)$, $\beta_1 = \left(2\Upsilon_{\text{GD}}^2 + 23H^2\kappa^2\right)$, and $\beta_2 = \left(\frac{3}{2}H\kappa^2\right)$. Following the same steps that we used to arrive at (28) gives:

$$\begin{aligned}\left\|\boldsymbol{x}^T - \boldsymbol{x}^*\right\|^2 &\le \alpha^T\left\|\boldsymbol{x}^0 - \boldsymbol{x}^*\right\|^2 \\ &\quad + \frac{6\eta}{\mu}\left(\frac{1}{1-\alpha}\beta_2 + \frac{1}{1-\alpha^H}\beta_1\right) \end{aligned} \tag{93}$$

Since $\alpha = \left(1-\frac{\mu\eta}{2}\right)$, we have $\alpha^H = \left(1-\frac{\mu\eta}{2}\right)^H \overset{\text{(a)}}{\le} \exp\left(-\frac{\mu\eta H}{2}\right) \overset{\text{(b)}}{\le} 1 - \frac{\mu\eta H}{2} + \left(\frac{\mu\eta H}{2}\right)^2 \overset{\text{(c)}}{\le} 1 - \frac{\mu\eta H}{2} + \frac{1}{10}\frac{\mu\eta H}{2} = 1 - \frac{9}{10}\frac{\mu\eta H}{2}$. In (a) we used the inequality $(1-\frac{1}{x})^x \le \frac{1}{e}$ which holds for any $x > 0$; in (b) we used $\exp(-x) \le 1 - x + x^2$ which holds for any $x \ge 0$; in (c) we used $\eta \le \frac{1}{5HL}$ and $\mu \le L$, which imply $\frac{\mu\eta H}{2} \le \frac{1}{10}$. Substituting these in (93) gives

$$\left\|\boldsymbol{x}^T - \boldsymbol{x}^*\right\|^2$$

$$\leq \left(1-\frac{\mu\eta}{2}\right)^T \left\|\boldsymbol{x}^0-\boldsymbol{x}^*\right\|^2 + \frac{6\eta}{\mu}\left(\frac{2}{\mu\eta}\beta_2 + \frac{20}{9\mu\eta H}\beta_1\right)$$
$$\leq \left(1-\frac{\mu\eta}{2}\right)^T \left\|\boldsymbol{x}^0-\boldsymbol{x}^*\right\|^2 + \frac{6\times 20}{9\mu^2}\left(\frac{9}{10}\beta_2 + \frac{1}{H}\beta_1\right)$$
$$\leq \left(1-\frac{\mu\eta}{2}\right)^T \left\|\boldsymbol{x}^0-\boldsymbol{x}^*\right\|^2 + \frac{14}{\mu^2}\left(\frac{2\Upsilon_{\text{GD}}^2}{H} + 25H\kappa^2\right), \tag{94}$$

where $\Upsilon_{\text{GD}} = \mathcal{O}\left(H\kappa\sqrt{\epsilon}\right)$. Substituting the value of $\eta = \frac{1}{5HL}$ yields the convergence rate (7) in the strongly-convex part of Theorem 2. Note that (94) holds with probability 1.

B. Convergence Proof of the Non-Convex Part of Theorem 2

Following the proof of the non-convex part of Theorem 1 given in Section V until (33) and using $\eta \leq \frac{1}{5HL}$ gives:

$$F(\boldsymbol{x}^{t_{i+1}}) \leq F(\boldsymbol{x}^{t_{i+1}-1}) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2 + \frac{6\eta}{5}\|C\|^2, \tag{95}$$

where $C = \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^{t_{i+1}-1}) - \nabla F_r(\boldsymbol{x}_r^{t_{i+1}-1})\right) - \left(\widehat{F}_{\text{accu}}^{t_i,t_{i+1}} - \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\sum_{t=t_i}^{t_{i+1}-1}\nabla F_r(\boldsymbol{x}_r^t)\right)$.

Using the bounds from (88) and (89), together with the Jensen's inequality, we can bound $\|C\|^2$ as follows:

$$\|C\|^2 \leq 2(3H\kappa^2) + 2(2\Upsilon_{\text{GD}}^2 + 20H^2\kappa^2)$$
$$\leq 2(2\Upsilon_{\text{GD}}^2 + 23H^2\kappa^2) \tag{96}$$

Substituting the bound from (96) into (95) gives:

$$F(\boldsymbol{x}^{t_{i+1}}) \leq F(\boldsymbol{x}^{t_{i+1}-1}) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{x}^{t_{i+1}-1})\right\|^2 + \frac{12\eta}{5}\left(2\Upsilon_{\text{GD}}^2 + 23H^2\kappa^2\right), \tag{97}$$

where $\Upsilon_{\text{GD}} = \mathcal{O}\left(H\kappa\sqrt{\epsilon}\right)$.

Note that above recurrence in (97) holds only at the synchronization indices. Now we give a recurrence at non-synchronization indices.

We have done a similar calculations in the non-convex part of Theorem 1 in Section V.

Take an arbitrary $t \in [T]$ and let $t_i \in \mathcal{I}_T$ be such that $t \in [t_i : t_{i+1}-1]$; when $H \geq 2$, such t's exist. Following the same steps until (36) and using $\eta \leq \frac{1}{5HL}$ gives:

$$F(\boldsymbol{x}^{t+1}) \leq F(\boldsymbol{x}^t) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2 + \frac{6\eta}{5}\|D\|^2, \tag{98}$$

where $D = \frac{1}{K}\sum_{r\in\mathcal{K}_{t_i}}\left(\nabla F(\boldsymbol{x}^t) - \nabla F_r(\boldsymbol{x}_r^t)\right)$.

Using the bound from (88), we have $\|D\|^2 \leq 3H\kappa^2$. Substituting this in (98) gives:

$$F(\boldsymbol{x}^{t+1}) \leq F(\boldsymbol{x}^t) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2 + \frac{6\eta}{5}(3H\kappa^2) \tag{99}$$

Now we have a recurrence at the synchronization indices given in (97) and at non-synchronization indices given in (99). Adding (97) and (99) from $t=0$ to T (use (97) for the synchronization indices and (99) for the rest of the indices) gives:

$$\sum_{t=0}^{T} F(\boldsymbol{x}^{t+1}) \leq \sum_{t=0}^{T} F(\boldsymbol{x}^t) - \frac{\eta}{2}\sum_{t=0}^{T}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2$$
$$+ \frac{12\eta}{5}\left[\frac{T}{H}\left(2\Upsilon_{\text{GD}}^2 + 23H^2\kappa^2\right) + \left(T - \frac{T}{H}\right)\left(\frac{3}{2}H\kappa^2\right)\right] \tag{100}$$

After rearranging and simplifying the last constant terms, we get:

$$\frac{1}{T}\sum_{t=0}^{T}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2 \leq \frac{2}{\eta T}\left[F(\boldsymbol{x}^0) - F(\boldsymbol{x}^{T+1})\right] + \frac{24}{5}\left(\frac{2\Upsilon_{\text{GD}}^2}{H} + 25H\kappa^2\right) \tag{101}$$

Note that the last term in (101) is a constant. So, it would be best to take the step-size η to be as large as possible such that it satisfies $\eta \leq \frac{1}{5HL}$. We take $\eta = \frac{1}{5HL}$. Substituting this in (101) and using $F(\boldsymbol{x}^{T+1}) \geq F(\boldsymbol{x}^*)$ gives

$$\frac{1}{T}\sum_{t=0}^{T}\left\|\nabla F(\boldsymbol{x}^t)\right\|^2 \leq \frac{10HL}{T}\left[F(\boldsymbol{x}^0) - F(\boldsymbol{x}^*)\right] + \frac{24}{5}\left(\frac{2\Upsilon_{\text{GD}}^2}{H} + 25H\kappa^2\right), \tag{102}$$

where $\Upsilon_{\text{GD}} = \mathcal{O}\left(H\kappa\sqrt{\epsilon}\right)$. This yields the convergence rate (8) in the non-convex part of Theorem 2. Note that (102) holds with probability 1.

This concludes the proof of Theorem 2.

Appendix E
Omitted Details from Section VII

In this section, we bound $\left\|\nabla\bar{f}_r(\boldsymbol{x}) - \nabla\mu_r(\boldsymbol{x})\right\|$ under both the sub-exponential and sub-Gaussian gradient distributional assumptions. First we give some definitions.

Definition 1 (Sub-exponential distribution)**.** *A random variable Z with mean $\mu = \mathbb{E}[Z]$ is* sub-exponential *if there are non-negative parameters (ν, α) such that*

$$\mathbb{E}\left[\exp\left(\lambda(Z-\mu)\right)\right] \leq \exp\left(\lambda^2\nu^2/2\right), \qquad \forall|\lambda| < \frac{1}{\alpha}.$$

A random vector Z with mean $\mu = \mathbb{E}[Z]$ is sub-exponential *if its projection on every unit vector is sub-exponential, i.e., there are non-negative parameters (ν, α) such that*

$$\sup_{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|=1}\mathbb{E}\left[\exp\left(\lambda\langle Z-\mu, \boldsymbol{v}\rangle\right)\right] \leq \exp\left(\lambda^2\nu^2/2\right), \forall|\lambda| < \frac{1}{\alpha}.$$

Now we state a concentration inequality for sums of independent sub-exponential random variables.

Fact 2 (Sub-exponential concentration inequality)**.** *Suppose $X_1, X_2, \ldots, X_n$ are independent random variables, where for every $i \in [n]$, X_i is sub-exponential with parameters (ν_i, α_i) and mean μ_i. Then $\sum_{i=1}^n X_i$ is sub-exponential with parameters (ν, α), where $\nu^2 = \sum_{i=1}^n \nu_i^2$ and $\alpha = \max_{1\leq i\leq n}\alpha_i$. Moreover, we have*

$$\Pr\left[\sum_{i=1}^{n}(X_i - \mu_i) \geq t\right] \leq \exp\left(-\frac{1}{2}\min\left\{\frac{t^2}{\nu^2}, \frac{t}{\alpha}\right\}\right), \ \forall t \geq 0 \tag{103}$$

Definition 2 (Sub-Gaussian distribution)**.** *A random variable Z with mean $\mu = \mathbb{E}[Z]$ is* sub-Gaussian *if there is a non-negative parameter σ_g such that*

$$\mathbb{E}\left[\exp\left(\lambda(Z-\mu)\right)\right] \leq \exp\left(\lambda^2\sigma_g^2/2\right), \ \forall \lambda \in \mathbb{R}.$$

A random vector Z with mean $\mu = \mathbb{E}[Z]$ is sub-Gaussian *if its projection on every unit vector is sub-Gaussian, i.e., there is a non-negative parameter σ_g such that*

$$\sup_{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|=1} \mathbb{E}\left[\exp\left(\lambda\langle Z-\mu, \boldsymbol{v}\rangle\right)\right] \leq \exp\left(\lambda^2\sigma_g^2/2\right), \ \forall \lambda \in \mathbb{R}.$$

Now we state a concentration inequality for sums of independent sub-Gaussian random variables.

Fact 3 (Sub-Gaussian concentration inequality)**.** *Suppose $X_1, X_2, \ldots, X_n$ are independent random variables, where for every $i \in [n]$, X_i is sub-Gaussian with parameter $\sigma_i > 0$ and mean μ_i. Then $\sum_{i=1}^n X_i$ is sub-Gaussian with parameter $\sigma_g = \sqrt{\sum_{i=1}^n \sigma_i^2}$. Moreover, we have*

$$\Pr\left[\sum_{i=1}^n (X_i - \mu_i) \geq t\right] \leq \exp\left(-t^2/2\sigma_g^2\right), \ \forall t \geq 0. \quad (104)$$

Let $D = \max\{\|\boldsymbol{x}-\boldsymbol{x}'\| : \boldsymbol{x}, \boldsymbol{x}' \in \mathcal{C}\}$ be the diameter of $\mathcal{C}$. Note that $\mathcal{C}$ is contained in $\mathcal{B}^d_{D/2}$, which is the Euclidean ball of radius $\frac{D}{2}$ in d dimensions that contains $\mathcal{C}$. Note that $D = \Omega(\sqrt{d})$, and we assume that D can grow at most polynomially in d.

Now we state two lemmas (which will be used to prove Theorem 5), each of which uniformly bounds $\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|$ over all $\boldsymbol{x} \in \mathcal{C}$ under different distributional assumptions on gradients. We prove these one by one in subsequent subsections.

Lemma 6 (Sub-exponential gradients)**.** *Suppose Assumption 3 holds. Take an arbitrary $r \in [R]$. Let $n_r \in \mathbb{N}$ be sufficiently large such that $n_r = \Omega\left(d\log(n_r d)\right)$. Then, with probability at least $1 - \frac{1}{(1+n_r LD)^d}$, we have*

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 3\nu\sqrt{\frac{8d\log(1+n_r LD)}{n_r}}, \quad \forall x \in \mathcal{C}. \quad (105)$$

Lemma 7 (Sub-Gaussian gradients)**.** *Suppose Assumption 4 holds. Take an arbitrary $r \in [R]$. For any $n_r \in \mathbb{N}$, with probability at least $1 - \frac{1}{(1+n_r LD)^d}$, we have*

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 3\sigma_g\sqrt{\frac{8d\log(1+n_r LD)}{n_r}}, \quad \forall x \in \mathcal{C}. \quad (106)$$

Proof of Theorem 5. In order to prove Theorem 5, we need to show two bounds, one (stated in (48)) under the sub-exponential gradient assumption, and the other (stated in (49)) under the sub-Gaussian assumption. We can show (48) using Lemma 6 and (49) using Lemma 7. Here we only show (48); and (49) can be shown similarly.

Using Assumption 5 (i.e., $\|\nabla \mu_r(\boldsymbol{x}) - \nabla \mu(\boldsymbol{x})\| \leq \kappa_{\text{mean}}, \forall \boldsymbol{x} \in \mathcal{C}$) in (46) gives

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \bar{f}(\boldsymbol{x})\right\| \leq \left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| + \kappa_{\text{mean}} + \frac{1}{R}\sum_{r=1}^R \left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|. \quad (107)$$

Note that (105) holds for any fixed worker $r \in [R]$. By the union bound, we have that with probability at least $1 - \frac{R}{(1+n_r LD)^d}$, for every $r \in [R]$, we have $\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 3\nu\sqrt{\frac{8d\log(1+n_r LD)}{n_r}}, \forall \boldsymbol{x} \in \mathcal{C}$.

Let $n_r = n, \forall r \in [R]$. Using these in (107), we get that with probability at least $1 - \frac{R}{(1+n_r LD)^d}$, for every worker $r \in [R]$, we have $\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \bar{f}(\boldsymbol{x})\right\| \leq \kappa_{\text{mean}} + \mathcal{O}\left(\sqrt{\frac{d\log(nd)}{n}}\right), \forall \boldsymbol{x} \in \mathcal{C}$, which proves (48). This completes the proof of Theorem 5. □

A. Proof of Lemma 6 (sub-exponential gradients)

We prove Lemma 6 with the help of the following result, which holds for any fixed $\boldsymbol{x} \in \mathcal{C}$. Then we extend this bound to all $\boldsymbol{x} \in \mathcal{C}$ using an ϵ-net argument. These are standard calculations and have appeared in literature [17], [22].

Lemma 8. *Suppose Assumption 3 holds. Take an arbitrary $r \in [R]$. For any $\delta \in (0,1)$ and $n_r \in \mathbb{N}$, define $\Delta = \sqrt{2}\nu\sqrt{\frac{d\log 5 + \log(1/\delta)}{n_r}}$. If n_r is such that $\Delta \leq \frac{\nu^2}{\alpha}$, then, for any fixed $\boldsymbol{x} \in \mathcal{C}$, with probability at least $1-\delta$, we have*

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 2\sqrt{2}\nu\sqrt{\frac{d\log 5 + \log(1/\delta)}{n_r}}, \quad (108)$$

where randomness is due to the sub-exponential distribution of local gradients.

Proof. Let $\mathcal{B}^d = \{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\| \leq 1\}$. Let $\mathcal{V} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{N_{1/2}}\}$ denote an $\frac{1}{2}$-net of $\mathcal{B}^d$, which implies that for every $\boldsymbol{v} \in \mathcal{B}^d$, there exists a $\boldsymbol{v}' \in \mathcal{V}$ such that $\|\boldsymbol{v}-\boldsymbol{v}'\| \leq \frac{1}{2}$. We have from [49, Lemma 5.2] that $N_{1/2} = |\mathcal{V}| \leq 5^d$.

Fix an arbitrary $\boldsymbol{x} \in \mathcal{C}$. Note that there exists a $\boldsymbol{v}^* \in \mathcal{B}^d$ (namely, $\boldsymbol{v}^* = \frac{\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})}{\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\|}$) such that $\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| = \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \boldsymbol{v}^*\right\rangle$. By the property of $\mathcal{V}$, there exists an index $i^* \in [N_{1/2}]$ such that $\|\boldsymbol{v}^* - \boldsymbol{v}_{i^*}\| \leq \frac{1}{2}$. Now we bound $\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|$.

$$\begin{aligned}
&\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \\
&= \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \boldsymbol{v}^*\right\rangle \\
&= \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \boldsymbol{v}_{i^*}\right\rangle \\
&\qquad + \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \boldsymbol{v}^* - \boldsymbol{v}_{i^*}\right\rangle \\
&\leq \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \boldsymbol{v}_{i^*}\right\rangle \\
&\qquad + \left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \|\boldsymbol{v}^* - \boldsymbol{v}_{i^*}\| \\
&\leq \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \boldsymbol{v}_{i^*}\right\rangle + \frac{1}{2}\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \\
&\leq \max_{\boldsymbol{v}\in\mathcal{V}} \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \boldsymbol{v}\right\rangle + \frac{1}{2}\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|
\end{aligned}$$

By collecting similar terms together, we get

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 2\max_{\boldsymbol{v}\in\mathcal{V}} \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \boldsymbol{v}\right\rangle \quad (109)$$

Note that the RHS of (109) is a non-negative number (because LHS is). Note also that, since

$\mathcal{V} \subset \mathcal{B}^d$, for every $\boldsymbol{v} \in \mathcal{V}$, we have $\|\boldsymbol{v}\| \leq 1$. This implies that $\max_{\boldsymbol{v}\in\mathcal{V}} \langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \boldsymbol{v} \rangle \leq \max_{\boldsymbol{v}\in\mathcal{V}} \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle$. Using this in (109), we get

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 2 \max_{\boldsymbol{v}\in\mathcal{V}} \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle. \tag{110}$$

Fix any $\boldsymbol{v} \in \mathcal{V}$. It follows from Assumption 3 that $\left\langle \nabla f_r(\boldsymbol{z}, \boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle$, where $\boldsymbol{z} \sim q_r$, is a sub-exponential random variable (with mean zero) with parameters (ν, α). From Fact 2 (stated on page 29), we have that $\sum_{i=1}^{n_r} \left\langle \nabla f_r(\boldsymbol{z}_{r,i}, \boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle$ (where $\boldsymbol{z}_{r,i} \sim q_r, i \in [n_r]$ are i.i.d.) is a sub-exponential random variable with parameters $(\sqrt{n_r}\nu, \alpha)$.

Now, apply the concentration bound from (103) with $t = n_r\Delta$. Substituting this and the parameters $(\sqrt{n_r}\nu, \alpha)$, the bound becomes $\exp(-\frac{1}{2}\min\{\frac{n_r^2\Delta^2}{n_r\nu^2}, \frac{n_r\Delta}{\alpha}\}) \overset{\text{(a)}}{=} \exp(-\frac{1}{2}\frac{n_r\Delta^2}{\nu^2})$, where (a) follows because $\Delta \leq \frac{\nu^2}{\alpha}$. This gives

$$\Pr\left[\sum_{i=1}^{n_r} \left\langle \nabla f_r(\boldsymbol{z}_{r,i}, \boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle \geq n_r\Delta\right] \leq \exp\left(-\frac{n_r\Delta^2}{2\nu^2}\right). \tag{111}$$

Note that $\sum_{i=1}^{n_r} \left\langle \nabla f_r(\boldsymbol{z}_{r,i}, \boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle = n_r \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle$. Using this in (111) yields

$$\Pr\left[\left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle \geq \Delta\right] \leq \exp\left(-\frac{n_r\Delta^2}{2\nu^2}\right) \tag{112}$$

This implies that

$$\begin{aligned}
&\Pr\left[\max_{\boldsymbol{v}\in\mathcal{V}} \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle \geq \Delta\right] \\
&\qquad \leq \sum_{\boldsymbol{v}\in\mathcal{V}} \Pr\left[\left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle \geq \Delta\right] \\
&\qquad \leq |\mathcal{V}| \exp\left(-\frac{n_r\Delta^2}{2\nu^2}\right) \leq 5^d \exp\left(-\frac{n_r\Delta^2}{2\nu^2}\right) \\
&\qquad = \exp\left(-\frac{n_r\Delta^2}{2\nu^2} + d\log 5\right)
\end{aligned} \tag{113}$$

Together with (110), which implies that

$$\begin{aligned}
&\Pr\left[\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \geq t\right] \\
&\qquad \leq \Pr\left[2\max_{\boldsymbol{v}\in\mathcal{V}} \left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \right\rangle \geq t\right]
\end{aligned}$$

holds for every $t > 0$, (113) gives

$$\begin{aligned}
\Pr\left[\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \geq 2\Delta\right] &\leq \exp\left(-\frac{n_r\Delta^2}{2\nu^2} + d\log 5\right) \\
&\leq \delta,
\end{aligned} \tag{114}$$

where in the last inequality we used $\Delta = \sqrt{2}\nu\sqrt{\frac{d\log 5 + \log(1/\delta)}{n_r}}$.

This completes the proof of Lemma 8. □

Proof of Lemma 6. We have from Lemma 8 that for each fixed $\boldsymbol{x} \in \mathcal{C}$, with probability at least $1-\delta$, we have

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 2\nu\sqrt{\frac{2d\log 5 + 2\log(1/\delta)}{n_r}}. \tag{115}$$

To extend this argument uniformly over the entire set $\mathcal{C}$, we use another covering argument. Recall that D is the diameter of $\mathcal{C}$. Note that $\mathcal{C}$ is contained in $\mathcal{B}^d_{D/2}$, which is the Euclidean ball of radius $\frac{D}{2}$ in d dimensions that contains $\mathcal{C}$. For some $\delta_0 > 0$, let $\mathcal{C}_{\delta_0} = \{\boldsymbol{x}_0, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{N_{\delta_0}}\}$ be the δ_0-net of $\mathcal{C}$. It follows from [49, Lemma 5.2] that $N_{\delta_0} \leq \left(1 + \frac{D}{\delta_0}\right)^d$.

Applying the union bound in (115), we get that with probability at least $1-\delta$, we have for all $\boldsymbol{x}_i \in \mathcal{C}_{\delta_0}$,

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}_i) - \nabla \mu_r(\boldsymbol{x}_i)\right\| \leq 2\nu\sqrt{\frac{2d\log 5 + 2\log\left(\frac{N_{\delta_0}}{\delta}\right)}{n_r}}. \tag{116}$$

We want to bound $\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\|$ for all $\boldsymbol{x} \in \mathcal{C}$. Take any $\boldsymbol{x} \in \mathcal{C}$. Since $\mathcal{C}_{\delta_0}$ is a δ_0-net of $\mathcal{C}$, there exists an $\boldsymbol{x}' \in \mathcal{C}_{\delta_0}$ such that $\|\boldsymbol{x} - \boldsymbol{x}'\| \leq \delta_0$.

$$\begin{aligned}
&\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \\
&\quad \leq \underbrace{\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \bar{f}_r(\boldsymbol{x}')\right\|}_{=:\, T_1} + \underbrace{\left\|\nabla \mu_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}')\right\|}_{=:\, T_2} \\
&\qquad + \left\|\nabla \bar{f}_r(\boldsymbol{x}') - \nabla \mu_r(\boldsymbol{x}')\right\|
\end{aligned} \tag{117}$$

Now we bound each term on the RHS of (117).

$$\begin{aligned}
T_1 &= \left\|\frac{1}{n_r}\sum_{i=1}^{n_r} (\nabla f_r(\boldsymbol{z}_{r,i}, \boldsymbol{x}) - \nabla f_r(\boldsymbol{z}_{r,i}, \boldsymbol{x}'))\right\| \\
&\leq \frac{1}{n_r}\sum_{i=1}^{n_r} \|\nabla f_r(\boldsymbol{z}_{r,i}, \boldsymbol{x}) - \nabla f_r(\boldsymbol{z}_{r,i}, \boldsymbol{x}')\| \\
&\leq L\|\boldsymbol{x} - \boldsymbol{x}'\| \leq L\delta_0
\end{aligned}$$

$$\begin{aligned}
T_2 &= \|\mathbb{E}_{\boldsymbol{z}\sim q_r}[\nabla f_r(\boldsymbol{z}, \boldsymbol{x}) - \nabla f_r(\boldsymbol{z}, \boldsymbol{x};)]\| \\
&\leq \mathbb{E}_{\boldsymbol{z}\sim q_r} \|\nabla f_r(\boldsymbol{z}, \boldsymbol{x}) - \nabla f_r(\boldsymbol{z}, \boldsymbol{x};)\| \\
&\leq \mathbb{E}_{\boldsymbol{z}\sim q_r} L\|\boldsymbol{x} - \boldsymbol{x}'\| \leq L\delta_0
\end{aligned}$$

Substituting the above bounds on T_1, T_2 in (117) and bounding the third term of (117) using (116) gives

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 2L\delta_0 + 2\nu\sqrt{\frac{2d\log 5 + 2\log\left(\frac{N_{\delta_0}}{\delta}\right)}{n_r}}. \tag{118}$$

Note that $N_{\delta_0} \leq \left(1 + \frac{D}{\delta_0}\right)^d$. Take $\delta = 1/\left(1 + \frac{D}{\delta_0}\right)^d$. If we take $\delta_0 = \frac{1}{n_r L}$, which implies $\delta = \frac{1}{(1+n_r LD)^d}$, we would get $2d\log 5 + 2\log\left(\frac{N_{\delta_0}}{\delta}\right) \leq 4d + 4d\log(1+n_r LD) \leq 8d\log(1+n_r LD)$. Substituting these in above gives

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq \frac{2}{n_r} + \frac{2\nu}{\sqrt{n_r}}\sqrt{8d\log(1+n_r LD)}. \tag{119}$$

When $n_r \geq \frac{1}{2\nu^2 d\log(1+n_r LD)}$ (which is a very small number less than 1), with probability at least $1-\frac{1}{(1+n_r LD)^d}$, we have

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 3\nu\sqrt{\frac{8d\log(1+n_r LD)}{n_r}}, \quad \forall x \in \mathcal{C}. \tag{120}$$

Lower bound on n_r. Note that Lemma 8 requires $\Delta \leq \frac{\nu^2}{\alpha}$, where $\Delta = \sqrt{2}\nu\sqrt{\frac{d\log 5+\log(1/\delta)}{n_r}}$. Substituting the value of $\delta = \frac{1}{(1+n_r LD)^d}$ gives $n_r \geq \frac{2\alpha^2}{\nu^2}\left(d\log 5 + d\log(1+n_r LD)\right)$, which is $\Omega(d\log(n_r LD))$ for constant α, ν. Treating the smoothness parameter L a constant, we get $n_r = \Omega(d\log(n_r d))$ to be requirement on the sample size at the r'th worker for the bound in Lemma 6 to hold.

This completes the proof of Lemma 6. □

B. Proof of Lemma 7 (sub-Gaussian gradients)

We prove Lemma 7 with the help of the following result, which holds for any fixed $\boldsymbol{x} \in \mathcal{C}$.

Lemma 9. *Suppose Assumption 4 holds. Take an arbitrary $r \in [R]$. For any $\delta \in (0,1)$ and $n_r \in \mathbb{N}$, with probability at least $1-\delta$, we have for any fixed $\boldsymbol{x} \in \mathcal{C}$:*

$$\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \leq 2\sqrt{2}\sigma_g\sqrt{\frac{d\log 5+\log(1/\delta)}{n_r}}, \tag{121}$$

where randomness is due to the sub-Gaussian distribution of local gradients.

Proof. Follow the proof of Lemma 8 exactly until (110). Then instead of the sub-exponential assumption, use the sub-Gaussian assumption (Assumption 4) on local gradients. Then apply the concentration bound from (104) with $t = n_r\Delta$. This gives that for any fixed $\boldsymbol{v} \in \mathcal{V}$ and any $\Delta \geq 0$, we have

$$\Pr\left[\left\langle \nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x}), \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}\right\rangle \geq \Delta\right] \leq \exp\left(-\frac{n_r\Delta^2}{2\sigma_g^2}\right). \tag{122}$$

Now following the proof of Lemma 8 from (112) to (114) gives

$$\Pr\left[\left\|\nabla \bar{f}_r(\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\right\| \geq 2\Delta\right] \leq \exp\left(-\frac{n_r\Delta^2}{2\sigma_g^2} + d\log 5\right) \leq \delta, \tag{123}$$

where in the last inequality we used $\Delta = \sqrt{2}\sigma_g\sqrt{\frac{d\log 5+\log(1/\delta)}{n_r}}$. □

We can extend the bound from Lemma 9 to all $\boldsymbol{x} \in \mathcal{C}$ (and prove Lemma 7) using an ϵ-net argument exactly in the same way as used in the proof of Lemma 6. So, to avoid repetition, we do not show this extension here.

C. Bounding the local variances

In Section VII-B, we showed that in order to bound $\mathbb{E}_{i\in_U[n_r]}\left\|\nabla f_r(\boldsymbol{z}_{r,i},\boldsymbol{x}) - \nabla \bar{f}_r(\boldsymbol{x})\right\|^2$ uniformly over all $\boldsymbol{x} \in \mathcal{C}$, it suffices to bound $\|\nabla f_r(\boldsymbol{z},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\|$ for a random $\boldsymbol{z} \sim q_r$ uniformly over all $\boldsymbol{x} \in \mathcal{C}$.

Bounding $\|\nabla f_r(\boldsymbol{z},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\|$. To bound this, we need sub-Gaussian assumption on local gradients (we can also bound this using sub-exponential assumption, but that will give a bound that scales as $\widetilde{\Omega}(d)$ as opposed to $\widetilde{\Omega}(\sqrt{d})$). Note that Lemma 7 holds for any $n_r \in \mathbb{N}$. In particular, it also holds for $n_r = 1$. So, under Assumption 4, with probability at least $1-\frac{1}{(1+n_r LD)^d}$, we have

$$\|\nabla f_r(\boldsymbol{z},\boldsymbol{x}) - \nabla \mu_r(\boldsymbol{x})\| \leq 3\sigma_g\sqrt{8d\log(1+LD)}, \quad \forall x \in \mathcal{C}, \tag{124}$$

where $\boldsymbol{z} \sim q_r$, and probability is over the randomness due to the sub-Gaussian distribution of local gradients. So, with probability at least $1-\frac{1}{(1+n_r LD)^d}$, we have

$$\mathbb{E}_{i\in_U[n_r]}\left\|\nabla f_r(\boldsymbol{z}_{r,i},\boldsymbol{x}) - \nabla \bar{f}_r(\boldsymbol{x})\right\|^2 \leq 288\sigma_g^2 d\log(1+LD), \quad \forall \boldsymbol{x} \in \mathcal{C}. \tag{125}$$

Note that (125) holds for a fixed worker $r \in [R]$. By taking the union bound over all workers $r \in [R]$ proves Theorem 6.

Appendix F
Additional Experimental Details

There are some implementation issues about the decoding algorithm (as described in Algorithm 2) that could be important in the deployment of the algorithm. In the following, we describe these issues and also explain our approach in the implementation to address them.

- Note that the stopping criterion (see line 7) in our decoding algorithm described in Algorithm 2 requires the matrix concentration bound σ_0^2 that we show in Theorem 3 in terms of the SGD variance bound σ^2 (see (2)) and the bounded gradient dissimilarity κ^2 (see (6)). Since these are properties of the local datasets stored at clients, which is challenging to determine in a adversarial federated learning setting. In order to mitigate this, we observe two things:
  1) the only place where Algorithm 2 uses this matrix concentration bound is in the stopping criterion (in line 7); and
  2) in each iteration of the while loop, at least one sample gets its weight reduced to zero.

  Since we know an upper bound on the fraction of corrupt samples, these two observations suggest replacing the stopping condition in line 7 with the condition that break the while loop when the number of samples whose weights become zero is more than the number of corrupt samples. This is what we used as a stopping criterion (in line 7) in our implementation of Algorithm 2.
- Note that each iteration of the while loop (line 7) of Algorithm 2 requires computing the principal eigenvector of the covariance matrix (line 8), which can be done using the singular value decomposition (SVD) algorithm.

This, however, could be computationally expensive. To mitigate this, we choose uniformly at random 1024 coordinates from the all gradient vectors (same 1024 random coordinates from all the gradients), and run the decoding algorithm only on them. Suppose $\mathcal{A}$ denotes the set of indices of the surviving gradients (i.e., whose weight are not zero when the filtering algorithm terminates), then we will discard all those full gradients whose indices are outside the set $\mathcal{A}$.

Furthermore, we observed performance boost when replacing the line 13 of Algorithm 2 (i.e., $\widehat{\boldsymbol{g}} = \sum_{i=1}^{K} \frac{w_i^{(t)}}{\|\boldsymbol{w}^{(t)}\|_1} \boldsymbol{g}_i$) with $\widehat{\boldsymbol{g}} = \sum_{i \in \mathcal{A}} \frac{1}{|\mathcal{A}|} \boldsymbol{g}_i$, where $\mathcal{A}$ contains the identities of the surviving samples; in other words, we replaced the weighted average with the uniform average.