2021 60th IEEE Conference on Decision and Control (CDC)
December 13-15, 2021. Austin, Texas

Federated Learning with Incrementally Aggregated Gradients

Aritra Mitra*, Rayana Jaafar*, George J. Pappas and Hamed Hassani

2021 60th IEEE Conference on Decision and Control (CDC) | 978-1-6654-3659-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/CDC45484.2021.9683443

***Abstract*— We consider the standard federated learning (FL) framework where a set of clients coordinate with a central server to train a statistical model. In a single-machine centralized setting, it is well-known that for smooth and strongly convex finite-sum optimization problems, one can design algorithms that guarantee *exact* linear convergence to the global minimum without computing full (batch) gradients at every iteration. Despite its popularity, an analog of the above result does not exist in FL. Motivated by this gap, we consider a setting where the local loss function of each client can be expressed as a finite sum of smooth component functions. For this setting, we propose a novel computationally-efficient FL algorithm called `FedTrack` that rests on two key ideas: (i) using the most recently communicated versions of the clients' gradients in the local update rule, and (ii) incrementally aggregating gradients of the component functions of each client. While the first idea serves to overcome the effect of heterogeneity across the clients' local loss functions, the second helps to significantly reduce the overall number of gradient computations. For both strongly convex and non-convex local loss functions, we prove that the convergence guarantees of `FedTrack` match their centralized counterparts (up to constants). In particular, for the strongly convex setting, we show that `FedTrack` guarantees exact linear convergence to the global minimum deterministically.**

I. Introduction

Federated Learning (FL) is a machine learning framework where a set of clients collaborate towards training a common statistical model under the orchestration of a central server [1]. In the canonical FL setting, each client has a local loss function defined over its private data, and the goal is to minimize a global loss function that equals the average of the clients' local loss functions. Some of the key challenges intrinsic to the FL setting that complicate solving this task are: (i) *privacy*, which precludes exchanging raw client data; (ii) *objective heterogeneity*, that arises due to differences in the clients' data sets; and (iii) *communication-efficiency*, which dictates the need to reduce the number of communication rounds between the clients and the server.

To mitigate the communication bottleneck in particular, a typical FL algorithm operates in rounds, where between two successive rounds, each client performs certain local processing steps *in isolation* to update its local model. At the end of the round, clients upload their local models to the server, the server aggregates these models to generate a global model, and the global model is then sent back to each client. The clients initiate local processing from this common global model, and the process repeats itself.

Related Work: Due to the combined effect of intermittent communication (as described above) and objective heterogeneity, popular FL algorithms such as `FedAvg` [1], [2] and `FedProx` [3] suffer from a "client-drift" phenomenon: the local iterates of the clients drift towards the minima of their individual loss functions. Recent work has shown that a diminishing learning rate (step-size) is necessary for these algorithms to guarantee convergence to the global minimum [4], [5]. As a consequence, `FedAvg` and `FedProx` fail to guarantee linear convergence to the global minimum. In [6], the authors develop `SCAFFOLD`, where clients employ control variates to achieve variance reduction, leading to improved convergence rates over `FedAvg` and `FedProx` for general stochastic optimization problems. In our recent work [5], we propose `FedLin` - a novel FL algorithm that exploits past gradients and client-specific learning rates to guarantee linear convergence to the global minimum, despite arbitrary objective heterogeneity, and the presence of slow, straggling devices. However, `FedLin` requires each client to compute a full (batch) gradient of its loss function in each local step, making the approach computationally expensive.

In this paper, we consider the specific setting where the loss function of each client can be expressed as a *finite sum* of smooth component functions. Our **motivation** for doing so is twofold. First, the finite-sum setting is of particular relevance for solving empirical risk minimization (ERM) problems where the true risk function is approximated as a finite sample average. Second, for this class of problems, in the single-machine case, one can design *efficient* linearly converging algorithms that do not require computing full gradients at every iteration. As far as we are aware, no such algorithm exists in the context of FL. We take the first steps towards bridging this gap in our current work.

To give context to our contributions on this front, we first briefly review popular techniques for the centralized finite-sum setting. We start with the `SGD` method which samples one component uniformly at random from the finite set in each iteration. While variance reduction techniques such as `SAG` [7] and `SAGA` [8] have the same iteration complexity as `SGD`, they improve upon the latter's convergence rate by maintaining a memory of the most recent gradient value of each component function. Nonetheless, as the sampling procedure in `SAG` and `SAGA` is stochastic, all convergence guarantees are in expectation. The incremental aggregated gradient (`IAG`) method [9] differs from `SAG` in only one aspect: the component functions are processed in a *deterministic order*. In [10], the authors proved that for smooth and

*Aritra Mitra and Rayana Jaafar contributed equally.

The authors are with the Department of Electrical and Systems Engineering, University of Pennsylvania. Email: {rayanaj,amitra20, pappasg, hassani}@seas.upenn.edu. This work was supported by NSF CPS Grant 1837253, NSF CAREER award CIF 1943064, and the Air Force Office of Scientific Research Young Investigator Program (AFOSR-YIP) under award FA9550-20-1-0111.

strongly convex finite-sum settings, IAG guarantees linear convergence to the global minimum deterministically.

Contributions: The main contribution of this paper is to propose and analyze a novel computationally-efficient FL algorithm that we call FedTrack, for the setting where the loss function of each client can be expressed as a finite sum. The key technical idea behind FedTrack is to carefully combine two gradient estimators: a *global gradient estimator* that keeps track of the gradient of the global loss function, and a *local gradient estimator* at each client that keeps track of the gradient of the client's local loss function. In particular, each client computes the full gradient of its loss function *only once* in each round; at every other local step, it samples just one component function in a deterministic order, exactly as in the IAG method. Thus, FedTrack drastically reduces the computational complexity of FedLin [5] where each client computes a full gradient at *every* local step.[1]

When all component functions across clients are smooth, and each client's loss function is strongly convex, we prove that FedTrack guarantees linear convergence *exactly* to the global minimum, i.e., with zero residual error. Apart from [5], this result should also be compared with the very recent work [12], where despite computing full gradients at every local step, linear convergence is guaranteed only to a neighborhood of the global minimum. Importantly, the dependence of the convergence rate of our algorithm on the overall condition number κ is linear, as opposed to the quadratic dependence exhibited by IAG [10]. Thus, we show that computing the full gradient once in a while does have quantifiable benefits. Next, for the general non-convex setting, we show that FedTrack guarantees convergence to a first-order stationary point, while matching the corresponding centralized rate (up to constants). As our final result, we show that when the overall loss function is potentially non-convex, but satisfies the *Polyak-Lojasiewicz* (PL) condition, FedTrack once again guarantees linear convergence to an optimal solution. *To sum up, for the finite sum setting, FedTrack is the first computationally-efficient FL algorithm that guarantees exact linear convergence to the global minimum deterministically.*

II. Problem Formulation

The standard federated learning setup comprises of a set $\mathcal{S} = \{1, \dots, m\}$ of m clients that communicate periodically with a central server to solve the following unconstrained optimization problem:

$$\min_{x \in \mathbb{R}^d} f(x) = \min_{x \in \mathbb{R}^d} \frac{1}{m} \sum_{i \in \mathcal{S}} f_i(x). \tag{1}$$

Here, $f_i(x)$ is the local loss function of client i, and $f(x)$ is the global loss function. Let $x^* \in \operatorname{argmin}_{x \in \mathbb{R}^d} f(x)$. In this paper, we are particularly interested in the scenario where the local loss function $f_i(x)$ of each client i can be expressed as the average of m_i *component* functions, with the j-th component function denoted by $f_{i,j}(x), j \in \mathcal{D}_i = \{1, \dots, m_i\}$. The set $\mathcal{D}_i$ represents the set of data samples for client i. With this notation in place, the resulting optimization problem of interest to us can be stated as follows:

$$\min_{x \in \mathbb{R}^d} f(x) = \min_{x \in \mathbb{R}^d} \frac{1}{m} \sum_{i \in \mathcal{S}} \underbrace{\frac{1}{m_i} \sum_{j \in \mathcal{D}_i} f_{i,j}(x)}_{f_i(x)}. \tag{2}$$

For our analysis, we will assume throughout that each component function $f_{i,j}(x)$ is $L_{i,j}$-smooth, $\forall i \in \mathcal{S}$, and $\forall j \in \mathcal{D}_i$. That is, each $f_{i,j}(x)$ is continuously differentiable, and the gradient map $\nabla f_{i,j} : \mathbb{R}^d \to \mathbb{R}^d$ is $L_{i,j}$-Lipschitz. Let $L_i = \max_{j \in \mathcal{D}_i} L_{i,j}$, and $L = \max_{i \in \mathcal{S}} L_i$. Then, it can be easily verified that each $f_i(x)$ is L_i-smooth and $f(x)$ is L-smooth. Occasionally, we will also assume that each client function $f_i(x)$ is μ-strongly convex. Having introduced the setup, we now proceed to describe our algorithm.

III. Proposed Algorithm: FedTrack

In this section, we will develop our proposed algorithm FedTrack, formally described in Algorithm 1. Like every other FL algorithm, FedTrack proceeds in rounds $t \in \{1, \dots, T\}$, with H local steps performed by each client in parallel in every round. Here, $H \geq 1$ is a positive integer. We will denote by $x_{i,\ell}^{(t)}$ the local iterate of client i at the ℓ-th local step of round t, where $\ell \in \{0, \dots, H-1\}$. At the onset of every round t, we have $x_{i,0}^{(t)} = \bar{x}_t, \forall i \in \mathcal{S}$, i.e., all clients start off from a common global model $\bar{x}_t$, where $\bar{x}_t$ is as defined in line 11 of FedTrack.

To build towards the key ideas underlying our algorithm, let us start by noting that ideally, client i would like to implement the following local update rule:

$$x_{i,\ell+1}^{(t)} = x_{i,\ell}^{(t)} - \eta \underbrace{\left(\frac{1}{m} \sum_{j \in \mathcal{S}} \nabla f_j(x_{i,\ell}^{(t)}) \right)}_{\nabla f(x_{i,\ell}^{(t)})}. \tag{3}$$

There are two immediate challenges that render implementation of the above rule infeasible for our setting. First, as there is no communication between rounds, client i cannot access $\nabla f_j(x_{i,\ell}^{(t)})$, where $j \in \mathcal{S} \backslash \{i\}$. Second, even when it comes to its own local gradient, evaluating the full gradient $\nabla f_i(x_{i,\ell}^{(t)})$ at every local step ℓ can incur significant computational costs. Our goal is to simultaneously overcome each of the above challenges and synthesize a local update rule that behaves approximately as the one in (3). As we discuss next, this is achieved via two carefully constructed gradient estimators.

Global gradient estimator: To tackle the first challenge, the local update rule of FedTrack exploits the use of *past client gradients* to account for objective heterogeneity. Specifically, at the end of communication round $t-1$, each client i computes the full gradient of its local loss function at the global iterate $\bar{x}_t$, and communicates $\nabla f_i(\bar{x}_t)$ to the server (line 13). Note that this is the *only* time in the entire round that a client computes the full gradient of its loss function. Subsequently, the server aggregates the local gradients, and

[1] The idea behind computing the full gradient once in a while is akin to SVRG [11]. However, unlike our method, SVRG samples the components in a stochastic manner, and provides guarantees only in expectation.

Algorithm 1 FedTrack

1: **Input:** Client step-size η, initial iterate $\bar{x}_1 \in \mathbb{R}^d$, initial full gradient $\nabla f(\bar{x}_1)$.
2: **for** $t = 1, \ldots, T$ **do**
3: **for** $i = 1, \ldots, m$ **do**
4: **for** $\ell = 0, \ldots, H-1$ **do**
5: $x_{i,0}^{(t)} = \bar{x}_t$, $\tau_{i,j}^0 = 0, \forall j \in \mathcal{D}_i$
6: $g_{i,\ell}^{(t)} \leftarrow \frac{1}{m_i} \sum_{j \in \mathcal{D}_i} \nabla f_{i,j}\left(x_{i,\tau_{i,j}^\ell}^{(t)}\right)$
7: $x_{i,\ell+1}^{(t)} \leftarrow x_{i,\ell}^{(t)} - \eta\left(\nabla f(\bar{x}_t) - \nabla f_i(\bar{x}_t) + g_{i,\ell}^{(t)}\right)$
8: **end for**
9: Client i transmits $x_{i,H}^{(t)}$ to server
10: **end for**
11: Server transmits $\bar{x}_{t+1} = \frac{1}{m} \sum_{i \in \mathcal{S}} x_{i,H}^{(t)}$ to each client
12: **for** $i = 1, \ldots, m$ **do**
13: Client i transmits $\nabla f_i(\bar{x}_{t+1})$ to server
14: **end for**
15: Server transmits $\nabla f(\bar{x}_{t+1})$ to each client
16: **end for**

transmits the gradient of the global function $\nabla f(\bar{x}_t)$ to all clients (line 15). Thus, equipped with $\nabla f(\bar{x}_t)$, all clients know exactly what the global descent direction looks like at the beginning of round t. Nonetheless, this descent direction is computed at a stale iterate $x_{i,0}^{(t)} = \bar{x}_t$, and not at the current iterate of client i, namely $x_{i,\ell}^{(t)}$. To account for this, a natural idea would be to use the following update rule:

$$x_{i,\ell+1}^{(t)} = x_{i,\ell}^{(t)} - \eta(\nabla f(\bar{x}_t) - \nabla f_i(\bar{x}_t) + \nabla f_i(x_{i,\ell}^{(t)})). \quad (4)$$

However, one problem still remains: we want to avoid computing the full gradient $\nabla f_i(x_{i,\ell}^{(t)})$. To this end, we now discuss how to compute an approximation of $\nabla f_i(x_{i,\ell}^{(t)})$.

Local gradient estimator: FedTrack exploits the structure of the clients' local objective functions and uses the IAG method to approximate $\nabla f_i(x_{i,\ell}^{(t)})$. More precisely, at each local step $\ell \in \{1, \ldots, H-1\}$, client i computes the gradient of only one of its component functions, and uses the most recently computed gradients for the remaining ones to generate an estimate of $\nabla f_i(x_{i,\ell}^{(t)})$. In this way, the number of gradient computations per round is drastically reduced, especially when m_i is large. To formally describe the local gradient estimator for client i at local step ℓ, let $\tau_{i,j}^\ell$ denote the most recent local step when the gradient of the component function $f_{i,j}(x)$ was computed. Then, at each $\ell \in \{1, \ldots, H-1\}$, client i maintains an estimate $g_{i,\ell}^{(t)}$ of $\nabla f_i(x_{i,\ell}^{(t)})$ as follows:

$$g_{i,\ell}^{(t)} \triangleq \frac{1}{m_i} \sum_{j \in \mathcal{D}_i} \nabla f_{i,j}\left(x_{i,\tau_{i,j}^\ell}^{(t)}\right). \quad (5)$$

At $\ell = 0$, $g_{i,0}^{(t)} = \nabla f_i(\bar{x}_t)$, i.e., $\tau_{i,j}^0 = 0, \forall i \in \mathcal{S}, \forall j \in \mathcal{D}_i$. It thus follows that $0 \leq \tau_{i,j}^\ell \leq \ell$. In words, since the gradients of all the component functions of each client $i \in \mathcal{S}$ are computed at $\ell = 0$, the maximum delay encountered in the computation of any given component is at most ℓ at local step ℓ. Replacing $\nabla f_i(x_{i,\ell}^{(t)})$ in (4) with the local gradient estimator $g_{i,\ell}^{(t)}$ from (5), we immediately obtain the local update rule of FedTrack in line 7:

$$x_{i,\ell+1}^{(t)} \leftarrow x_{i,\ell}^{(t)} - \eta\left(\nabla f(\bar{x}_t) - \nabla f_i(\bar{x}_t) + g_{i,\ell}^{(t)}\right). \quad (6)$$

It should be noted that the component functions of every client are sampled in a deterministic order, and the sampling times $\{\tau_{i,j}^\ell\}_{i \in \mathcal{S}, j \in \mathcal{D}_i}$ may very well depend on the communication round t. We deliberately suppress this dependence in our notation for simplicity. To sum up, FedTrack exploits (i) *past client gradients* to overcome the effects of objective heterogeneity, and (ii) uses *incrementally aggregated local gradients* to significantly reduce the number of gradient computations per round.

Having formally introduced our proposed algorithm, we now proceed to show that FedTrack enjoys strong theoretical guarantees. In particular, when the global loss function $f(x)$ is strongly convex, we will show that FedTrack guarantees linear convergence to the global minimum, while imposing no restrictions whatsoever on (i) the level of objective heterogeneity; and (ii) the order in which the component functions at each client are sampled.

Remark 1. *The idea of exploiting past client gradients is related to the gradient-tracking technique [13]–[17] in distributed optimization. However, unlike the distributed setting where each agent can typically communicate with every neighbor at all times, clients can only exchange information via the server at periodic communication rounds in FL; between two rounds, there is no communication whatsoever. Owing to this key difference, our proof techniques and results differ significantly from those in [13]–[17].*

IV. Main Results

In this section, we characterize the performance of FedTrack for the strongly convex and non-convex settings. The proofs of all our results are deferred to Section V.

Theorem 1. *Suppose each $f_{i,j}(x)$ is $L_{i,j}$-smooth and each $f_i(x)$ is μ-strongly convex. Let $\kappa = L/\mu$.[2] Then, with $\eta = \frac{1}{18LH}$, FedTrack guarantees:*

$$f(\bar{x}_{T+1}) - f(x^*) \leq \left(1 - \frac{1}{18\kappa}\right)^T (f(\bar{x}_1) - f(x^*)). \quad (7)$$

Our next result pertains to the general non-convex setting.

Theorem 2. *Suppose each $f_{i,j}(x)$ is $L_{i,j}$-smooth. Then, with $\eta = \frac{1}{18LH}$, FedTrack guarantees:*

$$\min_{t \in [T]} \|\nabla f(\bar{x}_t)\|^2 \leq \frac{36L}{T}\big(f(\bar{x}_1) - f(\bar{x}_{T+1})\big). \quad (8)$$

[2]We remind the reader that $L_i = \max_{j \in \mathcal{D}_i} L_{i,j}$ and $L = \max_{i \in \mathcal{S}} L_i$.

Discussion: To position our results in the context of existing FL literature, we start by noting that algorithms such as `FedAvg` [1] and `FedProx` [3] fail to guarantee linear convergence to the global minimum x^* even when full gradients are computed, and additional assumptions are made on the level of objective heterogeneity. More recent algorithms that relax such assumptions are `SCAFFOLD` [6], `FedSplit` [12], and `FedLin` [5]. In [6], the authors consider the general stochastic oracle model and guarantee linear convergence to a neighborhood of x^*, where the size of the neighborhood depends on the variance of the noise model. In contrast, both [12] and [5] assume that clients compute full gradients of their loss functions at every local step. While [12] guarantees linear convergence only to a neighborhood of x^*, [5] guarantees exact linear convergence to x^*. Our result in Theorem 1 stands out relative to the ones above by establishing that `FedTrack` *simultaneously* enjoys the following properties: (i) guarantees exact linear convergence to x^*; (ii) works under arbitrary levels of heterogeneity across clients' local loss functions; and (iii) does not require computing full gradients at every local step.

In a concurrent work [18], the authors study the finite sum setting like us, and provide linear convergence rate guarantees that hold only in expectation. In contrast, our results hold *deterministically*. Finally, the convergence rate of `FedTrack` has a linear dependence on the condition number κ, unlike the quadratic dependence of the `IAG` method [10].

From Theorem 2, we note that for general non-convex losses, `FedTrack` guarantees convergence to a first-order stationary point at a rate that matches its centralized counterpart (up to constants). To get a finer result, we note that a function $h(\cdot)$ is said to satisfy the Polyak-Lojasiewicz (PL) condition [19] with a constant $\mu > 0$ if for any $x \in \mathbb{R}^d$,

$$\|\nabla h(x)\|^2 \geq 2\mu(h(x) - h(x^*)), \tag{9}$$

where $x^* \in \operatorname{argmin}_{x\in\mathbb{R}^d} h(x)$. The PL condition in (9) generalizes strong convexity, and functions satisfying it are called μ-PL functions. Our final result generalizes Theorem 1, and shows that as long as the global loss function $f(x)$ is μ-PL (but not necessarily convex), `FedTrack` still guarantees linear convergence to an optimal solution.

Theorem 3. *Suppose each $f_{i,j}(x)$ is $L_{i,j}$-smooth and $f(x)$ is μ-PL. Let $x^* \in \operatorname{argmin}_{x\in\mathbb{R}^d} f(x)$. Then, with $\eta = \frac{1}{18LH}$, `FedTrack` guarantees:*

$$f(\bar{x}_{T+1}) - f(x^*) \leq \left(1 - \frac{1}{18\kappa}\right)^T (f(\bar{x}_1) - f(x^*)). \tag{10}$$

V. Convergence Analysis

In this section, we provide the proofs of Theorems 1, 2 and 3. Before diving into the technical details, let us first provide an intuitive argument as to why `FedTrack` works. The key property that comes to our aid in the analysis is smoothness of the component functions of each client. Indeed, smoothness tells us that if $\|x_{i,\ell}^{(t)} - x_{i,\tau_{i,j}^\ell}^{(t)}\|$ is not too large for each $j \in \mathcal{D}_i$, then $g_{i,\ell}^{(t)}$ will approximate $\nabla f_i(x_{i,\ell}^{(t)})$ reasonably well. Smoothness also tells us that if $\|\bar{x}_t - x_{i,\ell}^{(t)}\|$ is small enough, then so is the gap between $g_{i,\ell}^{(t)}$ and $\nabla f_i(\bar{x}_t)$, and that between $\nabla f(x_{i,\ell}^{(t)})$ and $\nabla f(\bar{x}_t)$. When all the above claims hold simultaneously, the local update rule of `FedTrack` in Eq. (6) will be a reasonable approximation of the ideal update rule in Eq. (3), exactly as desired. In what follows, we will formalize this idea.

A. Preliminaries

We begin by noting that if $f(x)$ is L-smooth, then for any two points $x, y \in \mathbb{R}^d$, the following hold:

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \text{ and} \tag{11}$$

$$f(y) - f(x) \leq \langle y - x, \nabla f(x)\rangle + \frac{L}{2}\|y - x\|^2. \tag{12}$$

We will also have occasion to use the fact that for any m vectors $x_1, \ldots, x_m \in \mathbb{R}^d$,

$$\left\|\sum_{i=1}^m x_i\right\|^2 \leq m\sum_{i=1}^m \|x_i\|^2. \tag{13}$$

For our subsequent analysis, we fix a communication round $t \in \{1, \cdots, T\}$. Our general proof strategy will be to bound the change $f(\bar{x}_{t+1}) - f(\bar{x}_t)$ in the global loss function over round t. A quantity that will show up in the above bound is the *local gradient estimation error* $e_{i,\ell}^{(t)}$ of client i at local step ℓ, which we define as follows:

$$\begin{aligned} e_{i,\ell}^{(t)} &\triangleq g_{i,\ell}^{(t)} - \nabla f_i(x_{i,\ell}^{(t)}) \\ &= \frac{1}{m_i}\sum_{j\in\mathcal{D}_i}\left[\nabla f_{i,j}(x_{i,\tau_{i,j}^\ell}^{(t)}) - \nabla f_{i,j}(x_{i,\ell}^{(t)})\right]. \end{aligned} \tag{14}$$

We drop the superscript t on the local iterates $x_{i,\ell}^{(t)}$ and the local gradient errors $e_{i,\ell}^{(t)}$.

B. Bounding Global Function Change

Lemma 1 provides an upper bound on the change in the function value in communication round t, $f(\bar{x}_{t+1}) - f(\bar{x}_t)$.

Lemma 1. *Suppose each $f_{i,j}(x)$ is $L_{i,j}$-smooth. Then, `FedTrack` guarantees:*

$$\begin{aligned} f(\bar{x}_{t+1}) - f(\bar{x}_t) \leq{}& \frac{\eta L}{m}\left(\sum_{i=1}^m\sum_{\ell=0}^{H-1}\|x_{i,\ell} - \bar{x}_t\|\right)\|\nabla f(\bar{x}_t)\| \\ &- \eta H\|\nabla f(\bar{x}_t)\|^2 \\ &+ \frac{\eta}{m}\left(\sum_{i=1}^m\sum_{\ell=0}^{H-1}\|e_{i,\ell}\|\right)\|\nabla f(\bar{x}_t)\| \\ &+ \frac{2\eta^2L^3H}{m}\sum_{i=1}^m\sum_{\ell=0}^{H-1}\|x_{i,\ell} - \bar{x}_t\|^2 \\ &+ 2\eta^2H^2L\|\nabla f(\bar{x}_t)\|^2 + \frac{L\eta^2H}{m}\sum_{i=1}^m\sum_{\ell=0}^{H-1}\|e_{i,\ell}\|^2. \end{aligned} \tag{15}$$

Proof. From the local update rule of `FedTrack` in (6), we have

$$\begin{aligned} x_{i,H} = \bar{x}_t - \eta \sum_{\ell=0}^{H-1} (\nabla f_i(x_{i,\ell}) + e_{i,\ell}) \\ - \eta H (\nabla f(\bar{x}_t) - \nabla f_i(\bar{x}_t)), \forall i \in \mathcal{S}. \end{aligned} \tag{16}$$

Thus,

$$\begin{aligned} \bar{x}_{t+1} = \frac{1}{m}\sum_{i=1}^{m} x_{i,H} &= \bar{x}_t - \frac{\eta}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}(\nabla f_i(x_{i,\ell}) + e_{i,\ell}) \\ &\quad - \frac{\eta H}{m}\sum_{i=1}^{m}(\nabla f(\bar{x}_t) - \nabla f_i(\bar{x}_t)) \\ &= \bar{x}_t - \frac{\eta}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}(\nabla f_i(x_{i,\ell}) + e_{i,\ell}), \end{aligned} \tag{17}$$

Hence, since $f(x)$ is L-smooth, then by equations (12) and (17), we have

$$\begin{aligned} f(\bar{x}_{t+1}) - f(\bar{x}_t) &\leq \langle \bar{x}_{t+1} - \bar{x}_t, \nabla f(\bar{x}_t)\rangle + \frac{L}{2}\|\bar{x}_{t+1} - \bar{x}_t\|^2 \\ &= -\frac{\eta}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}\langle \nabla f_i(x_{i,\ell}), \nabla f(\bar{x}_t)\rangle \\ &\quad - \frac{\eta}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}\langle e_{i,\ell}, \nabla f(\bar{x}_t)\rangle \\ &\quad + \frac{L}{2}\left\|\frac{\eta}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}(\nabla f_i(x_{i,\ell}) + e_{i,\ell})\right\|^2 \\ &\leq \underbrace{-\frac{\eta}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}\langle \nabla f_i(x_{i,\ell}), \nabla f(\bar{x}_t)\rangle}_{T_1} \\ &\quad \underbrace{-\frac{\eta}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}\langle e_{i,\ell}, \nabla f(\bar{x}_t)\rangle}_{T_2} \\ &\quad \underbrace{+L\left\|\frac{\eta}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}\nabla f_i(x_{i,\ell})\right\|^2}_{T_3} \\ &\quad \underbrace{+L\left\|\frac{\eta}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1} e_{i,\ell}\right\|^2}_{T_4}, \end{aligned} \tag{18}$$

where the last step follows from equation (13).

We now proceed to bound each of the four terms that appear in (18) separately. For the first and third terms, we directly state the upper bound without proof. A detailed derivation of these upper bounds can be found in [5]. For the first term, we have

$$\begin{aligned} T_1 \leq \frac{\eta L}{m}\left(\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}\|x_{i,\ell} - \bar{x}_t\|\right)\|\nabla f(\bar{x}_t)\| \\ - \eta H \|\nabla f(\bar{x}_t)\|^2. \end{aligned} \tag{19}$$

For the second term, an application of the Cauchy-Schwartz inequality yields

$$\begin{aligned} T_2 &\leq \eta\left\|\frac{1}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1} e_{i,\ell}\right\|\|\nabla f(\bar{x}_t)\| \\ &\leq \frac{\eta}{m}\left(\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}\|e_{i,\ell}\|\right)\|\nabla f(\bar{x}_t)\|, \end{aligned} \tag{20}$$

where the last step follows from a direct application of the triangle inequality. For the third term, we have

$$T_3 \leq \frac{2\eta^2 L^3 H}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}\|x_{i,\ell} - \bar{x}_t\|^2 + 2\eta^2 H^2 L \|\nabla f(\bar{x}_t)\|^2. \tag{21}$$

For the fourth term, two applications of equation (13) yields

$$T_4 \leq \frac{L\eta^2 H}{m}\sum_{i=1}^{m}\sum_{\ell=0}^{H-1}\|e_{i,\ell}\|^2. \tag{22}$$

Plugging the bounds of equations (19), (20), (21) and (22) in equation (18) establishes the final result. □

C. Bounding Gradient Error

To simplify the bound in equation (15), it is apparent that we must derive an upper-bound on the local gradient error $e_{i,\ell}$. To this end, we will make use of the following lemma.

Lemma 2. *Suppose each $f_{i,j}(x)$ is $L_{i,j}$-smooth. Then, for all $\ell \in \{0, \ldots, H-1\}$ and for all $i \in \mathcal{S}$, the gradient error in Eq. (14) can be bounded as follows:*

$$\|e_{i,\ell}\| \leq \eta L H \|\nabla f(\bar{x}_t)\| + 3\eta L^2 H \max_{0 \leq b \leq \ell-1}\|x_{i,b} - \bar{x}_t\|. \tag{23}$$

Proof. From equation (14), we have

$$\begin{aligned} \|e_{i,\ell}\| &= \left\|\frac{1}{m_i}\sum_{j \in \mathcal{D}_i}\left(\nabla f_{i,j}(x_{i,\tau_{i,j}^{\ell}}) - \nabla f_{i,j}(x_{i,\ell})\right)\right\| \\ &\overset{(a)}{\leq} \frac{1}{m_i}\sum_{j \in \mathcal{D}_i}\left\|\nabla f_{i,j}(x_{i,\tau_{i,j}^{\ell}}) - \nabla f_{i,j}(x_{i,\ell})\right\| \\ &\overset{(b)}{\leq} \frac{1}{m_i}\sum_{j \in \mathcal{D}_i} L_{i,j}\|x_{i,\tau_{i,j}^{\ell}} - x_{i,\ell}\|. \end{aligned} \tag{24}$$

In the steps above, (a) follows from the triangle inequality and (b) follows from the fact that each $f_{i,j}$ is $L_{i,j}$-smooth (see (11)). Noting that the gradient sampling times satisfy $0 \leq \tau_{i,j}^{\ell} \leq \ell$, a repetitive application of the triangle inequality

leads to the following:

$$\begin{aligned}\|e_{i,\ell}\| &\leq \frac{1}{m_i}\sum_{j\in\mathcal{D}_i} L_{i,j}\|x_{i,\ell} - x_{i,\tau_{i,j}^{\ell}}\| \\ &= \frac{1}{m_i}\sum_{j\in\mathcal{D}_i} L_{i,j}\left\|\sum_{k=\tau_{i,j}^{\ell}}^{\ell-1} x_{i,k+1} - x_{i,k}\right\| \\ &\leq \frac{1}{m_i}\sum_{j\in\mathcal{D}_i} L_{i,j}\sum_{k=\tau_{i,j}^{\ell}}^{\ell-1}\|x_{i,k+1} - x_{i,k}\| \\ &\overset{(a)}{\leq} L_i\sum_{k=0}^{\ell-1}\|x_{i,k+1} - x_{i,k}\| \\ &\overset{(b)}{\leq} \eta L_i\sum_{k=0}^{\ell-1}\big(\|\nabla f(\bar{x}_t)\| + \|\nabla f_i(x_{i,k}) - \nabla f_i(\bar{x}_t)\| \\ &\qquad + \|e_{i,k}\|\big) \\ &\overset{(c)}{\leq} \eta L_i\sum_{k=0}^{\ell-1}\big(\|\nabla f(\bar{x}_t)\| + L_i\|x_{i,k} - \bar{x}_t\| + \|e_{i,k}\|\big).\end{aligned} \tag{25}$$

In the steps above, (a) follows from the fact that $\tau_{i,j}^{\ell} \geq 0$ and $L_i = \max_{j\in\mathcal{D}_i} L_{i,j}$, (b) follows directly from the local update rule of `FedTrack` (see Eq. (6)) and (c) follows from the L_i-smoothness of $f_i(x)$ (see (11)). Applying equation (24) on the term $\|e_{i,k}\|$ in (25), we get

$$\begin{aligned}\|e_{i,\ell}\| &\leq \eta L_i\sum_{k=0}^{\ell-1}\Bigg(\|\nabla f(\bar{x}_t)\| + L_i\|x_{i,k} - \bar{x}_t\| \\ &\quad + \frac{1}{m_i}\sum_{j\in\mathcal{D}_i} L_{i,j}\|x_{i,k} - x_{i,\tau_{i,j}^{k}}\|\Bigg) \\ &\overset{(a)}{\leq} \eta L_i\sum_{k=0}^{\ell-1}\Bigg(\|\nabla f(\bar{x}_t)\| + L_i\|x_{i,k} - \bar{x}_t\| \\ &\quad + \frac{1}{m_i}\sum_{j\in\mathcal{D}_i} L_{i,j}\big(\|x_{i,k} - \bar{x}_t\| + \|\bar{x}_t - x_{i,\tau_{i,j}^{k}}\|\big)\Bigg) \\ &\overset{(b)}{\leq} \eta L_i\sum_{k=0}^{\ell-1}\Bigg(\|\nabla f(\bar{x}_t)\| + L_i\|x_{i,k} - \bar{x}_t\| \\ &\quad + \frac{2}{m_i}\sum_{j\in\mathcal{D}_i} L_{i,j}\max_{0\leq b\leq k}\|x_{i,b} - \bar{x}_t\|\Bigg) \\ &\leq \eta L_i\sum_{k=0}^{\ell-1}\Bigg(\|\nabla f(\bar{x}_t)\| + 3L_i\max_{0\leq b\leq k}\|x_{i,b} - \bar{x}_t\|\Bigg) \\ &\overset{(c)}{\leq} \eta LH\|\nabla f(\bar{x}_t)\| + 3\eta L^2 H\max_{0\leq b\leq \ell-1}\|x_{i,b} - \bar{x}_t\|.\end{aligned} \tag{26}$$

In the steps above, (a) follows from a direct application of the triangle inequality after adding and subtracting the global iterate $\bar{x}_t$, (b) follows from the fact that $0 \leq \tau_{i,j}^{k} \leq k$ and (c) follows from the fact that $\ell \leq H$ and $L = \max_{i\in\mathcal{S}} L_i$. □

D. Bounding Client Drift (**Convex Setting**)

To further simplify the bound in equation (15), we must derive an upper bound on the client drift $\|x_{i,\ell} - \bar{x}_t\|$. To this end, we will make use of the following lemma from [5].

Lemma 3. *Suppose $f(x)$ is L-smooth and convex. Then, for any $\eta \in (0, 1)$ satisfying $\eta \leq \frac{1}{L}$, and any two points $x, y \in \mathbb{R}^d$, we have*

$$\|y - x - \eta(\nabla f(y) - \nabla f(x))\| \leq \|y - x\|. \tag{27}$$

Using the above lemma, we now provide a bound on the client drift for μ-strongly convex client functions.

Lemma 4. *Suppose each $f_{i,j}(x)$ is $L_{i,j}$-smooth and each $f_i(x)$ is μ-strongly convex. Moreover, suppose $\eta \leq 1/(3LH)$. Then, `FedTrack` guarantees the following bound for each $i \in \mathcal{S}$, and $\forall \ell \in \{0, \ldots, H-1\}$:*

$$\|x_{i,\ell} - \bar{x}_t\| \leq 4\eta H\|\nabla f(\bar{x}_t)\|. \tag{28}$$

Proof. Fix any $i \in \mathcal{S}$. From the local update rule of `FedTrack` in line 7, we have

$$\begin{aligned}\|x_{i,\ell+1} - \bar{x}_t\| &= \|x_{i,\ell} - \bar{x}_t - \eta\left(\nabla f_i(x_{i,\ell}) - \nabla f_i(\bar{x}_t)\right) \\ &\quad - \eta\nabla f(\bar{x}_t) - \eta e_{i,\ell}\| \\ &\leq \|x_{i,\ell} - \bar{x}_t - \eta\left(\nabla f_i(x_{i,\ell}) - \nabla f_i(\bar{x}_t)\right)\| \\ &\quad + \eta\|\nabla f(\bar{x}_t)\| + \eta\|e_{i,\ell}\| \\ &\leq \|x_{i,\ell} - \bar{x}_t\| + \eta\|\nabla f(\bar{x}_t)\| + \eta\|e_{i,\ell}\|,\end{aligned} \tag{29}$$

where the last inequality follows from Lemma 3 since $\eta \leq 1/L \leq 1/L_i$, $\forall i \in \mathcal{S}$. Combining equations (23) and (29), for $\eta \leq 1/(3LH)$ we obtain

$$\begin{aligned}\|x_{i,\ell+1} - \bar{x}_t\| &\leq \|x_{i,\ell} - \bar{x}_t\| + (\eta + \eta^2 LH)\|\nabla f(\bar{x}_t)\| \\ &\quad + 3\eta^2 L^2 H\max_{0\leq b\leq \ell-1}\|x_{i,b} - \bar{x}_t\| \\ &\leq \|x_{i,\ell} - \bar{x}_t\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &\quad + \eta L\max_{0\leq b\leq \ell-1}\|x_{i,b} - \bar{x}_t\|.\end{aligned} \tag{30}$$

We now proceed with a proof by induction to establish the final result. The induction claim is the following

$$\|x_{i,\ell} - \bar{x}_t\| \leq 4\eta\ell\|\nabla f(\bar{x}_t)\|, \tag{31}$$

for all $\ell \in \{0, \ldots, H-1\}$. First, we prove the base cases for $\ell = 0$ and $\ell = 1$. For $\ell = 0$, we have

$$\begin{aligned}\|x_{i,1} - \bar{x}_t\| &\leq \|x_{i,0} - \bar{x}_t\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &= 2\eta\|\nabla f(\bar{x}_t)\| \leq 4\eta\|\nabla f(\bar{x}_t)\|,\end{aligned}$$

where we used $x_{i,0} = \bar{x}_t$, $\forall i \in \mathcal{S}$. Similarly, for $\ell = 1$, we have

$$\begin{aligned}\|x_{i,2} - \bar{x}_t\| &\leq \|x_{i,1} - \bar{x}_t\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &\leq 4\eta\|\nabla f(\bar{x}_t)\| \leq 8\eta\|\nabla f(\bar{x}_t)\|.\end{aligned}$$

For the induction hypothesis, suppose

$$\|x_{i,\ell} - \bar{x}_t\| \leq 4\eta\ell\|\nabla f(\bar{x}_t)\|, \tag{32}$$

for $\ell \in \{0, \dots, k\}$. Hence, we may write

$$\begin{aligned}\|x_{i,k+1} - \bar{x}_t\| &\leq \|x_{i,k} - \bar{x}_t\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &\quad + \eta L \max_{0\leq b\leq k-1} \|x_{i,b} - \bar{x}_t\| \\ &\overset{(a)}{\leq} 4\eta k\|\nabla f(\bar{x}_t)\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &\quad + \eta L(4\eta(k-1))\|\nabla f(\bar{x}_t)\| \\ &\overset{(b)}{\leq} 4\eta k\|\nabla f(\bar{x}_t)\| + 4\eta\|\nabla f(\bar{x}_t)\| \\ &\leq 4\eta(k+1)\|\nabla f(\bar{x}_t)\|.\end{aligned}$$

In the above steps, (a) follows from the induction hypothesis (32), and (b) follows from the fact that $2\eta L(k-1) \leq 3\eta LH \leq 1$. This completes the induction step and establishes the result of Lemma 4. □

As an immediate result of Lemma 4, we now provide a bound on the gradient error $\|e_{i,\ell}\|$. The proof follows from using Eq. (28) to bound the term $\|x_{i,b} - \bar{x}_t\|$ in Eq. (23).

Lemma 5. *Suppose each $f_{i,j}(x)$ is $L_{i,j}$-smooth and each $f_i(x)$ is μ-strongly convex. Moreover, suppose $\eta \leq 1/(12LH)$. Then, `FedTrack` guarantees the following bound for each $i \in \mathcal{S}$, and $\forall \ell \in \{0, \dots, H-1\}$:*

$$\|e_{i,\ell}\| \leq 2\eta LH\|\nabla f(\bar{x}_t)\|. \tag{33}$$

E. Proof of Theorem 1

We are now ready to complete the proof of Theorem 1. Combining the bounds in Lemma's 1, 4, and 5, we obtain

$$\begin{aligned}f(\bar{x}_{t+1}) - f(\bar{x}_t) &\leq -\eta H\left(1 - 4\eta LH\right)\|\nabla f(\bar{x}_t)\|^2 \\ &\quad + 4\eta^2 LH^2\|\nabla f(\bar{x}_t)\|^2 + 36\eta^4 L^3 H^4\|\nabla f(\bar{x}_t)\|^2 \\ &\leq -\eta H\left(1 - 9\eta LH\right)\|\nabla f(\bar{x}_t)\|^2,\end{aligned} \tag{34}$$

where the last step follows from the fact that $12\eta LH \leq 1$. From (34) and (9), we obtain

$$f(\bar{x}_{t+1}) - f(x^*) \leq \left(1 - 2\eta\mu H\left(1 - 9\eta LH\right)\right)\left(f(\bar{x}_t) - f(x^*)\right). \tag{35}$$

With $\eta = \dfrac{1}{18LH}$, the above inequality becomes

$$f(\bar{x}_{t+1}) - f(x^*) \leq \left(1 - \frac{1}{18\kappa}\right)\left(f(\bar{x}_t) - f(x^*)\right), \tag{36}$$

where $\kappa = \frac{L}{\mu}$. Using the above inequality recursively leads to the claim of the theorem.

*F. Bounding Client Drift (**Non-Convex Setting**)*

We now proceed to analyze the non-convex setting. We begin by noting that the upper-bounds on the global function change and the local gradient error provided in Lemma's 1 and 2, respectively, only require smoothness of the clients' component functions $f_{i,j}(x)$ and make no convexity assumption whatsoever. Hence, Lemma's 1 and 2 also hold for the non-convex setting. However, the claim of Lemma 4 assumes convexity of the client functions $f_i(x)$, and hence, is not applicable for the non-convex setting. To this end, Lemma 6 provides a bound on the client drift without any assumption of convexity.

Lemma 6. *Suppose each $f_{i,j}(x)$ is $L_{i,j}$-smooth. Moreover, suppose $\eta \leq 1/(4LH)$. Then, `FedTrack` guarantees the following bound for each $i \in \mathcal{S}$, and $\forall \ell \in \{0, \dots, H-1\}$:*

$$\|x_{i,\ell} - \bar{x}_t\| \leq 4\eta H\|\nabla f(\bar{x}_t)\|. \tag{37}$$

Proof. Fix any $i \in \mathcal{S}$. From the local update rule of `FedTrack` in line (6), we have

$$\begin{aligned}\|x_{i,\ell+1} - \bar{x}_t\| &= \|x_{i,\ell} - \bar{x}_t - \eta\left(\nabla f_i(x_{i,\ell}) - \nabla f_i(\bar{x}_t)\right) \\ &\quad - \eta\nabla f(\bar{x}_t) - \eta e_{i,\ell}\| \\ &\leq \|x_{i,\ell} - \bar{x}_t\| + \eta\|\left(\nabla f_i(x_{i,\ell}) - \nabla f_i(\bar{x}_t)\right)\| \\ &\quad + \eta\|\nabla f(\bar{x}_t)\| + \eta\|e_{i,\ell}\| \\ &\overset{(a)}{\leq} (1 + \eta L_i)\|x_{i,\ell} - \bar{x}_t\| + \eta\|\nabla f(\bar{x}_t)\| \\ &\quad + \eta\|e_{i,\ell}\|,\end{aligned} \tag{38}$$

where step (a) follows from the smoothness of the client functions (see (11)). Combining equations (23) and (38), for $\eta \leq 1/(4LH)$, we obtain

$$\begin{aligned}\|x_{i,\ell+1} - \bar{x}_t\| &\leq (1 + \eta L_i)\|x_{i,\ell} - \bar{x}_t\| \\ &\quad + (\eta + \eta^2 LH)\|\nabla f(\bar{x}_t)\| \\ &\quad + 3\eta^2 L^2 H \max_{0\leq b\leq \ell-1} \|x_{i,b} - \bar{x}_t\| \\ &\overset{(a)}{\leq} (1 + \eta L)\|x_{i,\ell} - \bar{x}_t\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &\quad + \eta L \max_{0\leq b\leq \ell-1} \|x_{i,b} - \bar{x}_t\| \\ &= \|x_{i,\ell} - \bar{x}_t\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &\quad + \eta L\big(\|x_{i,\ell} - \bar{x}_t\| + \max_{0\leq b\leq \ell-1} \|x_{i,b} - \bar{x}_t\|\big),\end{aligned} \tag{39}$$

where step (a) follows from the fact that $\eta LH \leq 3\eta LH \leq 1$. We now proceed with a proof by induction to establish the final result. The induction claim is the following

$$\|x_{i,\ell} - \bar{x}_t\| \leq 4\eta\ell\|\nabla f(\bar{x}_t)\|, \tag{40}$$

for all $\ell \in \{0, \cdots, H-1\}$. Just as we did in the proof of Lemma 4, we prove the base case for $\ell = 0$ and $\ell = 1$. For $\ell = 0$, we have

$$\begin{aligned}\|x_{i,1} - \bar{x}_t\| &\leq (1 + \eta L)\|x_{i,0} - \bar{x}_t\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &= 2\eta\|\nabla f(\bar{x}_t)\| \leq 4\eta\|\nabla f(\bar{x}_t)\|,\end{aligned}$$

where we used $x_{i,0} = \bar{x}_t$, $\forall i \in \mathcal{S}$. Similarly, for $\ell = 1$, we have

$$\begin{aligned}\|x_{i,2} - \bar{x}_t\| &\leq (1 + \eta L)\|x_{i,1} - \bar{x}_t\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &\leq 6\eta\|\nabla f(\bar{x}_t)\| \leq 8\eta\|\nabla f(\bar{x}_t)\|.\end{aligned}$$

For the induction hypothesis, suppose

$$\|x_{i,\ell} - \bar{x}_t\| \leq 4\eta\ell\|\nabla f(\bar{x}_t)\|, \tag{41}$$

for $\ell \in \{0, \ldots, k\}$. Hence, we may write

$$\begin{aligned}\|x_{i,k+1} - \bar{x}_t\| &\leq \|x_{i,k} - \bar{x}_t\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &\quad + \eta L\big(\|x_{i,k} - \bar{x}_t\| + \max_{0\leq b\leq k-1} \|x_{i,b} - \bar{x}_t\|\big) \\ &\overset{(a)}{\leq} 4\eta k\|\nabla f(\bar{x}_t)\| + 2\eta\|\nabla f(\bar{x}_t)\| \\ &\quad + \eta L(8\eta k\|\nabla f(\bar{x}_t)\|) \\ &\overset{(b)}{\leq} 4\eta k\|\nabla f(\bar{x}_t)\| + 4\eta\|\nabla f(\bar{x}_t)\| \\ &\leq 4\eta(k+1)\|\nabla f(\bar{x}_t)\|.\end{aligned}$$

In the above steps, (a) follows directly from the induction hypothesis (41), and (b) follows from the fact that $4\eta Lk \leq 4\eta LH \leq 1$. This completes the induction step and establishes the result of Lemma 6. □

Note that we were able to get the same bound on the client drift term, both with and without the convexity assumption. The only difference is that the non-convex case imposes a tighter bound on the step-size η. As an immediate result, the claim of Lemma 5 also holds for the case where the client functions $f_i(x)$ are not necessarily convex.

G. Proof of Theorem 2

We are now ready to complete the proof of Theorem 2. Combining the bounds in Lemma's 1, 5 and 6, we obtain

$$f(\bar{x}_{t+1}) - f(\bar{x}_t) \leq -\eta H\left(1 - 9\eta LH\right)\|\nabla f(\bar{x}_t)\|^2. \quad (42)$$

Plugging $\eta = \dfrac{1}{18LH}$ in (42), we get

$$\|\nabla f(\bar{x}_t)\|^2 \leq 36L\big(f(\bar{x}_t) - f(\bar{x}_{t+1})\big). \quad (43)$$

Summing the above inequality from $t = 1$ to $t = T$, we obtain as desired

$$\begin{aligned}\min_{t\in[T]} \|\nabla f(\bar{x}_t)\|^2 &\leq \frac{1}{T}\sum_{t=1}^{T} \|\nabla f(\bar{x}_t)\|^2 \\ &\leq \frac{36L}{T}\big(f(\bar{x}_1) - f(\bar{x}_{T+1})\big).\end{aligned} \quad (44)$$

H. Proof of Theorem 3

As in the proof of Theorem 2, we note that Lemma's 1, 5 and 6 do not require convexity and hence, equation (42) holds for the setting where the client functions are not necessarily convex. In section V-E, we note that the bound on the quantity $f(\bar{x}_{t+1}) - f(x^*)$ obtained in equation (35) only made use of the PL condition (9). It follows that equation (35) also holds for the case where the global function $f(x)$ is μ-PL, immediately leading to the assertion of Theorem 3.

VI. Conclusion

We studied a federated learning setting where the local loss function of each client can be expressed as a finite sum of smooth component functions. For this setting, we proposed a novel, computationally-efficient algorithm `FedTrack` that employs a global gradient estimator to overcome objective heterogeneity, and a local gradient estimator at each client that significantly reduces the number of overall gradient computations. For smooth loss functions, we showed that `FedTrack` guarantees exact linear convergence to the global minimum under arbitrary objective heterogeneity. For the general non-convex setting, we established that `FedTrack` guarantees convergence to a first-order stationary point at a rate that matches the corresponding centralized rate. Importantly, our approach is the first to provide such guarantees in the finite-sum federated learning formulation.

References

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

[2] 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*. PMLR, 2020, pp. 4519–4529.

[3] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, and V. Smith, "On the convergence of federated optimization in heterogeneous networks," *arXiv preprint arXiv:1812.06127*, vol. 3, 2018.

[4] Z. Charles and J. Konečnỳ, "On the outsized importance of learning rates in local update methods," *arXiv preprint arXiv:2007.00878*, 2020.

[5] A. Mitra, R. Jaafar, G. J. Pappas, and H. Hassani, "Achieving linear convergence in federated learning under objective and systems heterogeneity," *arXiv preprint arXiv:2102.07053*, 2021.

[6] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 5132–5143.

[7] M. Schmidt, N. Le Roux, and F. Bach, "Minimizing finite sums with the stochastic average gradient," *Mathematical Programming*, vol. 162, no. 1-2, pp. 83–112, 2017.

[8] A. Defazio, F. Bach, and S. Lacoste-Julien, "Saga: A fast incremental gradient method with support for non-strongly convex composite objectives," in *Advances in neural information processing systems*, 2014, pp. 1646–1654.

[9] D. Blatt, A. O. Hero, and H. Gauchman, "A convergent incremental gradient method with a constant step size," *SIAM Journal on Optimization*, vol. 18, no. 1, pp. 29–51, 2007.

[10] M. Gurbuzbalaban, A. Ozdaglar, and P. A. Parrilo, "On the convergence rate of incremental aggregated gradient algorithms," *SIAM Journal on Optimization*, vol. 27, no. 2, pp. 1035–1048, 2017.

[11] R. Johnson and T. Zhang, "Accelerating stochastic gradient descent using predictive variance reduction," *Advances in neural information processing systems*, vol. 26, pp. 315–323, 2013.

[12] R. Pathak and M. J. Wainwright, "Fedsplit: An algorithmic framework for fast federated optimization," *arXiv:2005.05238*, 2020.

[13] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1245–1260, 2017.

[14] A. Nedic, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM Journal on Optimization*, vol. 27, no. 4, pp. 2597–2633, 2017.

[15] C. Xi, R. Xin, and U. A. Khan, "Add-opt: Accelerated distributed directed optimization," *IEEE Transactions on Automatic Control*, vol. 63, no. 5, pp. 1329–1339, 2017.

[16] S. Pu and A. Nedić, "Distributed stochastic gradient tracking methods," *Mathematical Programming*, pp. 1–49, 2020.

[17] R. Xin, A. K. Sahu, U. A. Khan, and S. Kar, "Distributed stochastic optimization with gradient tracking over strongly-connected networks," in *Proc. of the 58th IEEE Conference on Decision and Control (CDC)*, 2019, pp. 8353–8358.

[18] E. Gorbunov, F. Hanzely, and P. Richtárik, "Local sgd: Unified theory and new efficient methods," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 3556–3564.

[19] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2016, pp. 795–811.