2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

On the Computation-Communication Trade-Off with A Flexible Gradient Tracking Approach

Yan Huang and Jinming Xu[†]

Abstract— **We propose a flexible gradient tracking approach with adjustable computation and communication steps for solving distributed stochastic optimization problems over networks. The proposed method allows each node to perform multiple local gradient updates and multiple inter-node communications in each round, aiming to strike a balance between computation and communication costs according to the properties of objective functions and network topology in non-i.i.d. settings. Leveraging a properly designed Lyapunov function, we derive both the computation and communication complexities for achieving arbitrary accuracy on smooth and strongly convex objective functions. Our analysis demonstrates sharp dependence of the convergence performance on graph topology and properties of objective functions, highlighting the trade-off between computation and communication. Numerical experiments are conducted to validate our theoretical findings.**

I. Introduction

With the proliferation of individual computing devices and local collected user data [1], distributed optimization methods have become increasingly popular in recent years due to their wide applications in various fields such as cooperative control [2], distributed sensing [3], large-scale machine learning [4], and just to name a few. In this paper, we consider the standard distributed stochastic optimization problem jointly solved by a number of n nodes over a network:

$$\min_{x\in\mathbb{R}^p} f(x) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{\mathbb{E}_{\xi_i\sim\mathcal{D}_i}\left[f_i(x;\xi_i)\right]}_{:=f_i(x)}, \quad (1)$$

where $x \in \mathbb{R}^p$ is the global decision variable and the objective function $f : \mathbb{R}^p \to \mathbb{R}$ is the sum-utility of n local objective function f_i conditioned on the local data sample ξ_i with distribution $\mathcal{D}_i$. As a promising approach, parallel/decentralized stochastic gradient decent [5], [6] is shown to be a simple yet efficient algorithm for solving the above distributed stochastic optimization problem (1) under certain scenarios. However, parallel/decentralized SGD may not ensure good performance in the presence of node-specific heterogeneity arising from imbalanced data sets, communication and computing resources [7].

To avoid high communication burden among nodes, plenty of communication-efficient methods have been studied in the optimization and machine learning community. Particularly, Federated Averaging (FedAvg) [5] (a.k.a. Local SGD [8]), as a variant of parallel SGD with parameter server (PS) architecture [9], has been widely used in federated learning, which executes multiple local updates between two consecutive communication steps with partial or full node participation to save communication cost. The effectiveness of FedAvg/Local SGD for independent and identically distributed (i.i.d.) datasets has been extensively studied in the literature [8], [10]–[12]. For instance, it has been shown in [10], [11] that Local SGD can outperform centralized mini-batch SGD [13] for quadratic objectives and certain convex cases. We note that these methods with PS architecture all require a central server for data aggregation, which may suffer from single-point failure and communication bottleneck [6]. To address this issue, many decentralized SGD methods have been proposed for solving Problem (1) over peer-to-peer networks [6], [14]. In general, gossip-based communication protocols [15], [16] are popular choices for distributed algorithms. For example, Lian *et al.* [6] proposed D-PSGD where each node communicates only with its neighbouring nodes for reaching consensus on optimization process, and, in [17], [18], only a subset of the nodes are activated in each round for exchanging information, thus reducing communication costs.

While communication cost is a key concern in distributed optimization, it is equally important to ensure that the accuracy of the algorithm is not significantly compromised in practical scenarios. In particular, when it comes to non-i.i.d. settings where data distribution of nodes are heterogeneous, the adoption of local updates, partial participation and gossip protocols in parallel/decentralized SGD methods will introduce more degrees of data heterogeneity yielding poor algorithmic performance [19], and thus many variants have been proposed to address this issue. For instance, gradient estimation techniques and primal-dual-like methods have been shown to be effective in tackling data heterogeneity among nodes [20]–[28]. In particular, Pu *et al.* [20] proposed a distributed stochastic gradient tracking (DSGT) method by introducing an auxiliary variable for each node to track the average gradient of local functions. To further reduce the communication cost, Nguyen *et al.* [24] proposed a variant of DSGT, termed LU-GT, employing multiple local updates, and they provided the communication complexity matching Local SGD for non-convex objectives. Building on this, Liu *et al.* [26] proposed another variant adopting gradient-sum-tracking that further reduces the communication complexity with reduced stochastic gradient variance. However, both the results in [24], [26] ignore the side effect of local updates that will amplify stochastic gradient noise on computation complexity. The readers are referred to the recent survey papers [29], [30]

This work has been supported in parts by National Natural Science Foundation of China under Grants 62003302, 62373323, 62088101 and in parts by the Key Laboratory of Collaborative Sensing and Autonomous Unmanned Systems (Key Lab of CS&AUS) of Zhejiang Province.

[†]The authors are with the College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China. Correspondence to jimmyxu@zju.edu.cn (Jinming Xu).

979-8-3503-0123-6/23/$31.00 ©2023 IEEE

for many other efforts devoted into improving communication efficiency, e.g., acceleration methods [31], [32].

To account for both computation and communication complexity, Berahas *et al.* [33] proposed a variant of deterministic gradient decent method with multiple communication steps at each iteration, named NEAR-DGD, and they evaluated the performance of the algorithm via a new metric accounting for both communication and computation complexity, and showed that employing multiple consensus steps is desirable when communication is relatively cheap. Building on this, a stochastic variant called S-NEAR-DGD is proposed in [34] to accelerate the computation process. More recently, Liu *et al.* [35] proposed a decentralized federated learning algorithm, named DFL, that employs multiple local updates and multiple inter-node communication at each round and analyzed the impact of communication and computation on the performance of the algorithm separately. Although these aforementioned algorithms offer theoretical guarantees in different scenarios, they tend to be effective only with i.i.d. datasets and may face challenges with data heterogeneity in non-i.i.d. settings. In a very recent parallel work [36], Berahas *et al.* adopted a similar protocol with that of [35], in conjunction with a gradient tracking method, to solve deterministic distributed optimization problems.

In this paper, we propose and analyze a flexible gradient tracking approach that employs adjustable communication and computation protocols for solving Problem (1) in non-i.i.d. settings. The main contributions are summarized as:

- We develop a flexible gradient tracking method (termed FlexGT) employing multiple local updates and multiple inter-node communication steps, which enables it to deal with data heterogeneity effectively and design customized protocols to balance communication and computation costs according to the properties of objective functions and network topology;
- We provide theoretical guarantees for the proposed FlexGT algorithm both in terms of communication and computation complexity for strongly convex and smooth problems. In particular, we show that the proposed FlexGT algorithm converges linearly to a neighborhood of the optimal solution regardless of data heterogeneity, which recovers the best known result of DSGT [37] as a special case under our settings. Moreover, the complexity results shed light on adjusting the communication and computation frequency according to the problem characteristics to achieve a better trade-off. This is in contrast to the existing works [34], [35] that merely focus on communication or computation.

Notations. Throughout this paper, we adopt the following notations: $\|\cdot\|$ represents the Frobenius norm, $\mathbb{E}[\cdot]$ denotes the expectation of a matrix or vector, $\mathbf{1}$ represents the all-ones vector, $\mathbf{I}$ denotes the identity matrix, and $\mathbf{J} = \mathbf{1}\mathbf{1}^T/n$ denotes the averaging matrix.

II. Problem Formulation and Algorithm Design

We consider solving the distributed stochastic optimization problem (1) where n agents are connected over a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, Here, $\mathcal{V} = \{1, 2, ..., n\}$ represents the set of agents, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the set of edges consisting of ordered pairs (i, j) representing the communication link from node j to node i. For node i, we define $\mathcal{N}_i = \{j, |, (i, j) \in \mathcal{E}\}$ as the set of its neighboring nodes. We then make the following blanket assumptions on the objective function and graph.

Assumption 1: (**Convexity and smoothness**) Each $f_i(x)$ is μ-strongly convex and L-smooth in x.

Then, we assume each agent i obtains an unbiased noisy gradient of the form $\nabla f_i(x; \xi_i)$ by querying a stochastic oracle ($\mathcal{SO}$), which satisfies the following assumption.

Assumption 2: (**Bounded variance**) For $\forall x, x' \in \mathbb{R}^p$, there exist $\sigma \geqslant 0$ such that

$$\mathbb{E}\left[\|\nabla f_i(x; \xi_i) - \nabla f_i(x)\|^2\right] \leqslant \sigma^2,$$

where the random sample ξ_i is generated from $\mathcal{SO}$.

Assumption 3: (**Graph connectivity**) The weight matrix W induced by graph $\mathcal{G}$ is doubly stochastic, i.e., $W\mathbf{1} = \mathbf{1}, \mathbf{1}^T W = \mathbf{1}^T$ and $\rho_W := \|W - \mathbf{J}\|_2^2 < 1$.

Now, we proceed to present the proposed FlexGT algorithm for solving problem (1), which is given in Algorithm 1.

Algorithm 1 FlexGT

```
Initialization: Initial points $x_{i,0} \in \mathbb{R}^p$ and $y_{i,0} = \nabla_x f_i(x_{i,0}; \xi_{i,0})$, communication and computation frequency $d_1, d_2 \geq 1$ and stepsize $\gamma > 0$.
for round $k = 0, 1, \cdots$, each node $i \in [n]$, do
    for $l = 0, 1, \cdots, d_2 - 1$ do
        Obtain an unbiased noisy gradient sample of $\xi_{i,d_2k+l+1}$ by querying the stochastic oracle $\mathcal{SO}$.
        Perform local update:
            $x_{i,d_2k+l+1} = x_{i,d_2k+l} - \gamma y_{i,d_2k+l},$
            $y_{i,d_2k+l+1} = \nabla f_i(x_{i,d_2k+l+1}; \xi_{i,d_2k+l+1}) + y_{i,d_2k+l} - \nabla f_i(x_{i,d_2k+l}; \xi_{i,d_2k+l}).$
    end for
    for $s = 0, 1, \cdots, d_1 - 1$ do
        Perform inter-node communication:
            $x_{i,d_2(k+1)} = \sum_{j \in \mathcal{N}_i} W_{i,j} x_{j,d_2(k+1)},$
            $y_{i,d_2(k+1)} = \sum_{j \in \mathcal{N}_i} W_{i,j} y_{j,d_2(k+1)}.$
    end for
end for
```

For simplicity, we introduce the following notations:

$$\begin{aligned} X_t &:= [x_{1,t}, x_{2,t}, \cdots, x_{n,t}]^T \in \mathbb{R}^{n \times p}, \\ Y_t &:= [y_{1,t}, y_{2,t}, \cdots, y_{n,t}]^T \in \mathbb{R}^{n \times p}, \\ \nabla G_t &:= [\cdots, \nabla f_i(x_{i,t}; \xi_{i,t}), \cdots]^T \in \mathbb{R}^{n \times p}, \\ \nabla F_t &:= \mathbb{E}[\nabla G_t] = [\cdots, \nabla f_i(x_{i,t}), \cdots]^T \in \mathbb{R}^{n \times p}. \end{aligned}$$

Then, Algorithm 1 can be rewritten in a compact form:

$$X_{d_2(k+1)} = W^{d_1}\left(X_{d_2k} - \gamma \sum_{j=0}^{d_2-1} Y_{d_2k+j}\right), \tag{2a}$$

$$Y_{d_2(k+1)} = W^{d_1}\left(Y_{d_2k} + \nabla G_{d_2(k+1)} - \nabla G_{d_2k}\right), \tag{2b}$$

where the integers $d_1, d_2 \in [1, \infty)$ denote the communication and computation steps in each round respectively.

Remark 1: The flexibility of the proposed FlexGT algorithm consists in the adjustable communication and computation protocol with respect to d_1 and d_2, which allows for customized multiple local updates and message exchange over the network in each round according to the properties of objective functions and network topology. This algorithm can also eliminate the effect of data heterogeneity among nodes thanks to the gradient tracking scheme. It should be also noted that FlexGT recovers the standard DSGT [20] algorithm ($d_1 = d_2 = 1$) and LU-GT [24] ($d_1 = 1, d_2 \geqslant 1$) as special cases, and has the potential to recover S-NEAR-DGD [34] and DFL [35] as well if independent weight matrix for the update of Y is employed.

III. Main Results

In this section, we present the main convergence results of the proposed FlexGT algorithm for strongly-convex and smooth objective functions. To this end, we first define the following Lyapunov function consisting of optimality gap, consensus error and gradient tracking error:

$$V_t := \left\|\bar{x}_t - x^*\right\|^2 + c_1 \left\|X_t - \mathbf{1}\bar{x}_t\right\|^2 + c_2 \left\|Y_t - \mathbf{1}\bar{y}_t\right\|^2, \tag{3}$$

where $t = d_2k$, c_1 and c_2 are coefficients to be properly designed later, and

$$\bar{x}_t := \frac{\mathbf{1}^T}{n}X_t, \quad \bar{y}_t := \frac{\mathbf{1}^T}{n}Y_t.$$

With these definitions, we are ready to present the main convergence results of the proposed FlexGT algorithm.

Theorem 1: Suppose Assumptions 1, 2 and 3 hold. Let the stepsize satisfy

$$\gamma \leqslant \min\left\{\frac{1}{10d_2L}, \frac{1-\rho_W^{d_1}}{37d_2L\sqrt[4]{\rho_W^{d_1}}}, \frac{\left(1-\rho_W^{d_1}\right)^2}{153d_2L\sqrt{\rho_W^{d_1}}}\right\}. \tag{4}$$

Then, we have for all $k \geqslant 0$,

$$\mathbb{E}\left[V_{d_2(k+1)}\right] \leqslant \left(1 - \min\left\{\frac{\mu d_2 \gamma}{4}, \frac{1-\rho_W^{d_1}}{8}\right\}\right)\mathbb{E}\left[V_{d_2k}\right] + \frac{d_2\gamma^2\sigma^2}{n} + \frac{d_2^3\gamma^3L}{\left(1-\rho_W^{d_1}\right)^3}M_\sigma, \tag{5}$$

where M_σ is a polynomial of σ as defined in (15).

Proof: See Section IV-B.1. ■

Corollary 1: Under the same setting of Theorem 1, the number of computation steps needed to achieve an accuracy of arbitrary small $\varepsilon \geqslant 0$ is

$$\tilde{\mathcal{O}}\left(\frac{d_2L}{\left(1-\rho_W^{d_1}\right)^2\mu} + \frac{\sigma^2}{\mu^2 n\varepsilon} + \frac{d_2\sqrt{L\rho_W^{d_1}\sigma^2}}{\sqrt{\mu^3\left(1-\rho_W^{d_1}\right)^3\varepsilon}}\right), \tag{6}$$

and the number of communication steps needed is

$$\tilde{\mathcal{O}}\left(\frac{d_1L}{\left(1-\rho_W^{d_1}\right)^2\mu} + \frac{d_1\sigma^2}{\mu^2 nd_2\varepsilon} + \frac{d_1\sqrt{L\rho_W^{d_1}\sigma^2}}{\sqrt{\mu^3\left(1-\rho_W^{d_1}\right)^3\varepsilon}}\right), \tag{7}$$

where the notation $\tilde{\mathcal{O}}(\cdot)$ hides the logarithmic factors.

Proof: The techniques used to adapt Theorem 1 to the complexity results in Collollary 1 are common and can be found in [12], [21], [38]. We thus omit the proof. ■

Remark 2: Theorem 1 shows that the proposed FlexGT algorithm converges linearly to a neighborhood of the optimal solution of Problem (1), depending on the problem characteristics as well as the frequency of communication and computation. Particularly, this result implies that increasing d_2 can result in a smaller stepsize and smaller steady-state error, while increasing d_1 can lead to a faster linear rate but also a significant increase in communication overhead. To further illustrate the insight, we derive the computation and communication complexity of FlexGT in Corollary 1. Notably, this result demonstrates the existence of a trade-off between the computation and communication complexity, that is, as the computation (resp. communication) frequency d_2 (resp. d_1) increases, the computation complexity will increase (resp. decrease), while the communication complexity will decrease (resp. increase or decrease depending on ρ_W), thereby calling for a careful design of d_1 and d_2 to balance communication and computation costs based on the problem setting. Moreover, compared to the existing results in [34], [35], the convergence of FlexGT is independent of data heterogeneity among nodes captured as $\zeta_f := \mathbf{sup}\left\{\frac{1}{n}\sum_{i=1}^{n}\left\|\nabla f_i\left(x\right) - \nabla f\left(x\right)\right\|^2\right\}, \forall x$ [6], without the assumption of uniformly bounded stochastic gradient, and is thus more robust in non-i.i.d. settings.

IV. Convergence Analysis

In this section, we carry out the convergence analysis for the main results. We begin by introducing the following key lemmas that are crucial for the proof of Theorem 1.

A. Key Lemmas

*Lemma 1 (**Bounding optimality gap**):* Suppose Assumptions 1, 2 and 3 hold. Let the setp-size satisfy $\gamma \leqslant \frac{1}{10d_2L}$.

Then, we have for $\forall k \geqslant 0$,

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\bar{x}_{d_2(k+1)} - x^*\right\|^2\right] \\
&\leqslant \left(1 - \frac{d_2 \mu \gamma}{2}\right) \mathbb{E}\left[\left\|\bar{x}_{d_2 k} - x^*\right\|^2\right] \\
&+ \frac{12 d_2 \gamma L}{n} \mathbb{E}\left[\left\|X_{d_2 k} - \mathbf{1}\bar{x}_{d_2 k}\right\|^2\right] \\
&+ \frac{12 d_2^3 \gamma^3 L}{n} \mathbb{E}\left[\left\|Y_{d_2 k} - \mathbf{1}\bar{y}_{d_2 k}\right\|^2\right] \\
&- d_2 \gamma \mathbb{E}\left[f\left(\bar{x}_{d_2 k}\right) - f\left(x^*\right)\right] + \frac{d_2 \gamma^2 \sigma^2}{n} + 36 d_2^3 \gamma^3 L \sigma^2.
\end{aligned}
\tag{8}
$$

Proof: See Appendix B.1 in [39] [1]. ∎

*Lemma 2 (**Bounding consensus error**):* Suppose Assumptions 1, 2 and 3 hold. Let the stepsize satisfy

$$
\gamma \leqslant \min \left\{ \frac{1}{8 d_2 L}, \frac{1 - \rho_W^{d_1}}{12 d_2 L \sqrt{\rho_W^{d_1}}} \right\}. \tag{9}
$$

Then, we have for $\forall k \geqslant 0$,

$$
\begin{aligned}
&\mathbb{E}\left[\left\|X_{d_2(k+1)} - \mathbf{1}\bar{x}_{d_2(k+1)}\right\|^2\right] \\
&\leqslant \frac{3 + \rho_W^{d_1}}{4} \mathbb{E}\left[\left\|X_{d_2 k} - \mathbf{1}\bar{x}_{d_2 k}\right\|^2\right] \\
&+ \frac{192 n d_2^4 \gamma^4 L^3 \rho_W^{d_1}}{1 - \rho_W^{d_1}} \mathbb{E}\left[f\left(\bar{x}_{d_2 k}\right) - f\left(x^*\right)\right] \\
&+ \frac{6 d_2^2 \gamma^2 \rho_W^{d_1}}{1 - \rho_W^{d_1}} \mathbb{E}\left[\left\|Y_{d_2 k} - \mathbf{1}\bar{y}_{d_2 k}\right\|^2\right] + \frac{18 n d_2^2 \gamma^2 \rho_W^{d_1}}{1 - \rho_W^{d_1}} \sigma^2.
\end{aligned}
\tag{10}
$$

Proof: See Appendix B.1 in [39]. ∎

*Lemma 3 (**Bounding tracking error**):* Suppose Assumptions 1, 2 and 3 hold. Let the stepsize satisfy

$$
\gamma \leqslant \left\{ \frac{1}{8 d_2 L}, \frac{1 - \rho_W^{d_1}}{10 d_2 L \sqrt{\rho_W^{d_1}}} \right\}. \tag{11}
$$

Then, we have for $\forall k \geqslant 0$,

$$
\begin{aligned}
&\mathbb{E}\left[\left\|Y_{d_2(k+1)} - \mathbf{1}\bar{y}_{d_2(k+1)}\right\|^2\right] \\
&\leqslant \frac{3 + \rho_W^{d_1}}{4} \mathbb{E}\left[\left\|Y_{d_2 k} - \mathbf{1}\bar{y}_{d_2 k}\right\|^2\right] \\
&+ \frac{30 \rho_W^{d_1} L^2}{1 - \rho_W^{d_1}} \mathbb{E}\left[\left\|X_{d_2 k} - \mathbf{1}\bar{x}_{d_2 k}\right\|^2\right] \\
&+ \frac{96 n \rho_W^{d_1} d_2^2 \gamma^2 L^3}{1 - \rho_W^{d_1}} \mathbb{E}\left[f\left(\bar{x}_{d_2 k}\right) - f\left(x^*\right)\right] + 6 n \rho_W^{d_1} \sigma^2.
\end{aligned}
\tag{12}
$$

Proof: See Appendix B.2 in [39]. ∎

B. Proofs of Main Results

1) Proof of Theorem 1: Recalling the properly defined Lyapunov function (3) with

$$
c_1 = \frac{192 d_2 \gamma L}{n\left(1 - \rho_W^{d_1}\right)}, \quad c_2 = \frac{9312 d_2^3 \gamma^3 L}{n\left(1 - \rho_W^{d_1}\right)^3}. \tag{13}
$$

Then, with the help of Lemma 1, 2 and 3, we can bound the Lyapunov function as follows:

$$
\begin{aligned}
&\mathbb{E}\left[V_{d_2(k+1)}\right] \leqslant \left(1 - \min\left\{\frac{\gamma \mu d_2}{4}, \frac{1 - \rho_W^{d_1}}{8}\right\}\right) \mathbb{E}\left[V_{d_2 k}\right] \\
&\leqslant \frac{d_2 \gamma^2 \sigma^2}{n} + \frac{d_2^3 \gamma^3 L}{\left(1 - \rho_W^{d_1}\right)^3} M_\sigma + e_1 \mathbb{E}\left[f\left(\bar{x}_{d_2 k}\right) - f\left(x^*\right)\right] \\
&+ e_2 \mathbb{E}\left[\left\|X_{d_2 k} - \mathbf{1}\bar{x}_{d_2 k}\right\|^2\right] + e_3 \mathbb{E}\left[\left\|Y_{d_2 k} - \mathbf{1}\bar{y}_k\right\|^2\right],
\end{aligned}
\tag{14}
$$

where

$$
\begin{aligned}
e_1 &:= c_1 \frac{192 n d_2^4 \gamma^4 L^3 \rho_W^{d_1}}{1 - \rho_W^{d_1}} + c_2 \frac{96 n \rho_W^{d_1} d_2^2 \gamma^2 L^3}{1 - \rho_W^{d_1}} - d_2 \gamma, \\
e_2 &:= \frac{12 d_2 \gamma L}{n} + c_2 \frac{30 \rho_W^{d_1} L^2}{1 - \rho_W^{d_1}} - c_1 \frac{1 - \rho_W^{d_1}}{8}, \\
e_3 &:= \frac{12 d_2^3 \gamma^3 L}{n} + c_1 \frac{6 d_2^2 \gamma^2 \rho_W^{d_1}}{1 - \rho_W^{d_1}} - c_2 \frac{1 - \rho_W^{d_1}}{8},
\end{aligned}
$$

and

$$
\begin{aligned}
M_\sigma := {} & 36\left(1 - \rho_W^{d_1}\right)^3 \sigma^2 + 3456\left(1 - \rho_W^{d_1}\right) \rho_W^{d_1} \sigma^2 \\
& + 55872 \rho_W^{d_1} \sigma^2.
\end{aligned}
\tag{15}
$$

Letting $e_1, e_2, e_3 \leqslant 0$ with the stepsize γ satisfying (4), we obtain the result in (5), which completes the proof.

Remark 3: The proof of the main results relies on double-loop analysis and a properly designed Lyapunov function as depicted in (3). Specifically, we first derive the upper bounds of consensus error and gradient tracking error within a period (inner loop), respectively (c.f., Lemmas 7 and 6). These bounds are then used to establish the contraction properties of the above error terms at each round (outer loop) (c.f., Lemmas 2 and 3). Finally, we combine these obtained error terms to construct the Lyapunov function with properly designed coefficients (13), which allows us to obtain a rate result that shows the sharp dependence of convergence performance on the properties of objective functions, network topology as well as the frequency of communication and computation. It should be noted that this result also matches the best-known rate of the DSGT [37] (a special case with $d_1 = d_2 = 1$) under the same setting, i.e., $p = c = 1 - \rho_W$.

V. Numerical Experiments

In this section, we report a series of numerical experiments to verify the theoretical findings of the proposed FlexGT algorithm by means of a synthetic example. Specifically, we consider the following quadratic function:

$$
\min_x f(x) = \frac{1}{n} \sum_{i=1}^{n} \underbrace{\left(\mathbb{E}_{v_i}\left[\left(h_i^T x - \nu_i\right)^2 + \frac{\mu}{2}\|x\|^2\right]\right)}_{=: f_i}, \tag{16}
$$

where $\mu \geqslant 0$ is the regularization parameter, $h_i \in [0,1]^p$ denotes the feature parameters of node i with dimension $p = 10$, and $v_i \sim \mathcal{N}\left(\bar{v}_i, \sigma^2\right)$ with $\bar{v}_i \in [0,1]$. Therefore, the algorithm can obtain an unbiased noisy gradient $g_i\left(x_{i,t}\right) :=$

[1] The details of proof can be found in the arXiv version [39].

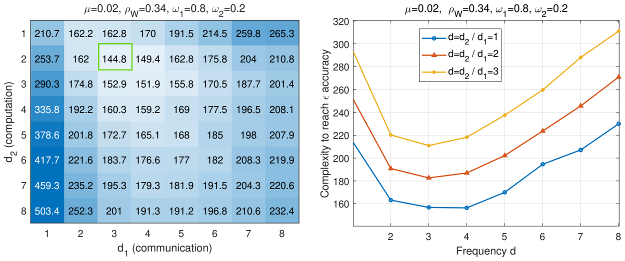

Fig. 1: Weighted-sum complexity of FlexGT algorithm to achieve $\varepsilon = 10^{-5}$ accuracy with different communication and computation frequency in each round over exponential graphs of $n = 20$ nodes with $N_i = 5$ neighbors.

$\nabla f_i\left(x_{i,t}\right) + \delta_{i,t}$ with $\delta_{i,t} \sim \mathcal{N}\left(0, \sigma^2\right)$ at each iteration t. Moreover, we set the stepsize according to the choice of d_1 and d_2, i.e., $\gamma = c\left(1 - \rho_W^{d_1}\right)^2 / \left(d_2 L\right)$, where $c = 10$ is a constant and Lipschitz constant is set to $L = 1$.

Communication and computation trade-off. To balance the communication and computation costs in practice, we attempt to minimize the weighted-sum of the obtained communication and computation complexity results:

$$\min_{d_1, d_2} \left(\omega_1 \mathcal{C}_1 + \omega_2 \mathcal{C}_2\right), \tag{17}$$

where $\mathcal{C}_1$ and $\mathcal{C}_2$ represent the obtained communication and computation complexity respectively, ω_1 and ω_2 are the corresponding weights. We note that $\mathcal{C}_2 = \mathcal{C}_1 d_1 / d_2$, and thus plot the heat-map of weighted complexity of FlexGT to achieve $\varepsilon = 10^{-5}$ accuracy with different d_1 and d_2 under specific settings in Fig. 1. It can be observed that the overal complexity of FlexGT varies with the increase of d_1 or d_2 and achieves the best performance at $d_1 = 3$ and $d_2 = 2$ (see green box on the left). Moreover, if we keep the radio of d_2 and d_1 fixed, i.e., $d_2/d_1 = d$, as shown on the right, there exists an optimal ratio to minimizing the weighted-sum complexity (17). These observations illustrate the trade-offs between communication and computation conditioned on the properties of objective functions and graph topology.

Eliminating the effects of node heterogeneity. In Fig. 2, we compare the convergence performance of the proposed FlexGT algorithm with DFL [35] and their special cases DSGT (FlexGT with $d_1 = d_2 = 1$) and D-PSGD (DFL with $d_1 = d_2 = 1$) in terms of computation and communication steps. We note that there is heterogeneity among the objectives f_i of nodes due to differences in $\{h_i\}_{i=1}^n$. In this scenario, it can be observed that FlexGT has a significant advantage over DFL in terms of steady-state error thanks to the gradient tracking method. Moreover, the choice of $d_1 = 3$ and $d_2 = 2$ makes FlexGT achieve better computation and communication complexity to reach an accuracy of $\varepsilon = 10^{-5}$.

VI. Conclusions

In this paper, we proposed a flexible and efficient distributed stochastic gradient tracking method FlexGT for solving distributed stochastic optimization problems under non-i.i.d settings. Our approach is able to handle data heterogeneity and design adjustable protocols to balance communication and computation costs according to the properties of the objective functions and network topology. We also provided theoretical guarantees for the proposed algorithm, including communication and computation complexity analysis for strongly convex and smooth objective functions, regardless of heterogeneity among nodes. These results provided an intuitive way to tune communication and computation protocols, highlighting their trade-offs. It will also be important to extend these results into more general settings, such as non-convex and time-varying cases, and consider acceleration methods both in computation and communication in future work.

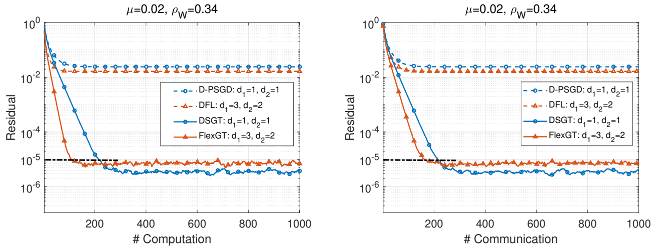

Fig. 2: Comparison for FlexGT, DFL, standard D-PSGD and DSGT algorithms over exponential graphs of $n = 20$ nodes with $N_i = 5$ neighbors.

Appendix

In this section, we provide the missing proofs for the lemmas and theorem in the main text. To this end, we first provide several supporting lemmas for the analysis.

A. Supporting Lemmas

*Lemma 4 (**Bernoulli's inequality**):* For constants $\beta \geqslant 1$ and $\lambda > 0$, we have

$$\left(1 + \frac{\lambda}{\beta}\right)^{\beta} \leqslant e^{\lambda}. \tag{18}$$

*Lemma 5 (**Bounding client divergence**):* Suppose Assumptions 1, 2 and 3 hold. Let the stepsize satisfy $\gamma \leqslant \frac{1}{8 d_2 L}$. Then, we have for $k \geqslant 0$ and integer $t \in [1, d_2 - 1]$, $d_2 \geqslant 2$,

$$\begin{aligned} &\mathbb{E}\left[\left\|X_{d_2 k+t} - \mathbf{1}\bar{x}_{d_2 k}\right\|^2\right] \\ &\leqslant 4\mathbb{E}\left[\left\|X_{d_2 k} - \mathbf{1}\bar{x}_{d_2 k}\right\|^2\right] + 4 d_2^2 \gamma^2 \mathbb{E}\left[\left\|Y_{d_2 k} - \mathbf{1}\bar{y}_k\right\|^2\right] \\ &+ 16 n d_2^2 \gamma^2 L \mathbb{E}\left[f\left(\bar{x}_{d_2 k}\right) - f\left(x^*\right)\right] + d_2^2 \gamma^2 \left(4\sigma^2 + 8n\sigma^2\right). \end{aligned} \tag{19}$$

Proof: See Appendix A in [39]. ■

*Lemma 6 (**Bounding tracking error within a period**):* Suppose Assumptions 1, 2 and 3 hold. Let the stepsize satisfy $\gamma \leqslant \frac{1}{8 d_2 L}$. Then, we have for $k \geqslant 0$ and integer $t \in [1, d_2 - 1]$, $d_2 \geqslant 2$,

$$\begin{aligned} &\mathbb{E}\left[\left\|Y_{d_2 k+t} - \mathbf{1}\bar{y}_{d_2 k+t}\right\|^2\right] \\ &\leqslant 3\mathbb{E}\left[\left\|Y_{d_2 k} - \mathbf{1}\bar{y}_{d_2 k}\right\|^2\right] + 16 L^2 \mathbb{E}\left[\left\|X_{d_2 k} - \mathbf{1}\bar{x}_{d_2 k}\right\|^2\right] \\ &+ 96 n d_2^2 \gamma^2 L^3 \mathbb{E}\left[f\left(\bar{x}_{d_2 k}\right) - f\left(x^*\right)\right] + 9n\sigma^2. \end{aligned} \tag{20}$$

Proof: See Appendix A in [39]. ■

*Lemma 7 (**Bounding consensus error within a period**):* Suppose Assumptions 1, 2 and 3 hold. Let the stepsize satisfy $\gamma \leqslant \frac{1}{8d_2L}$. Then, we have for $k \geqslant 0$ and integer $t \in [1, d_2 - 1]$, $d_2 \geqslant 2$,

$$\begin{aligned}
&\mathbb{E}\left[\left\|X_{d_2k+t} - \mathbf{1}\bar{x}_{d_2k+t}\right\|^2\right] \\
&\leqslant 3\mathbb{E}\left[\left\|X_{d_2k} - \mathbf{1}\bar{x}_{d_2k}\right\|^2\right] + 3d_2^2\gamma^2\mathbb{E}\left[\left\|Y_{d_2k} - \mathbf{1}\bar{y}_{d_2k}\right\|^2\right] \\
&+ 128nd_2^4\gamma^4L^3\mathbb{E}\left[f\left(\bar{x}_{d_2k}\right) - f\left(x^*\right)\right] + 9d_2^2\gamma^2n\sigma^2.
\end{aligned} \tag{21}$$

Proof: See Appendix A in [39]. ■

REFERENCES

[1] J. Konečný, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.

[2] A. Nedić and J. Liu, "Distributed optimization for control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 77–103, 2018.

[3] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," *Physical communication*, vol. 4, no. 1, pp. 40–62, 2011.

[4] K. I. Tsianos, S. Lawlor, and M. G. Rabbat, "Consensus-based distributed optimization: Practical issues and applications in large-scale machine learning," in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1543–1550, IEEE, 2012.

[5] 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*, pp. 1273–1282, PMLR, 2017.

[6] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, "Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent," *Advances in neural information processing systems*, vol. 30, 2017.

[7] J. Wang, Z. Charles, Z. Xu, G. Joshi, H. B. McMahan, M. Al-Shedivat, G. Andrew, S. Avestimehr, K. Daly, D. Data, *et al.*, "A field guide to federated optimization," *arXiv preprint arXiv:2107.06917*, 2021.

[8] S. U. Stich, "Local SGD converges fast and communicates little," in *International Conference on Learning Representations*, 2019.

[9] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang, *et al.*, "Large scale distributed deep networks," *Advances in neural information processing systems*, vol. 25, 2012.

[10] T. Lin, S. U. Stich, K. K. Patel, and M. Jaggi, "Don't use large mini-batches, use local sgd," in *International Conference on Learning Representations*, 2020.

[11] B. Woodworth, K. K. Patel, S. Stich, Z. Dai, B. Bullins, B. Mcmahan, O. Shamir, and N. Srebro, "Is local sgd better than minibatch sgd?," in *International Conference on Machine Learning*, pp. 10334–10343, PMLR, 2020.

[12] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich, "A unified theory of decentralized sgd with changing topology and local updates," in *International Conference on Machine Learning*, pp. 5381–5393, PMLR, 2020.

[13] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers*, pp. 177–186, Springer, 2010.

[14] S. S. Ram, A. Nedić, and V. V. Veeravalli, "Asynchronous gossip algorithms for stochastic optimization," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pp. 3581–3586, IEEE, 2009.

[15] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE transactions on information theory*, vol. 52, no. 6, pp. 2508–2530, 2006.

[16] J. Lu, C. Y. Tang, P. R. Regier, and T. D. Bow, "Gossip algorithms for convex consensus optimization over networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 12, pp. 2917–2923, 2011.

[17] X. Lian, W. Zhang, C. Zhang, and J. Liu, "Asynchronous decentralized parallel stochastic gradient descent," in *International Conference on Machine Learning*, pp. 3043–3052, PMLR, 2018.

[18] B. Ying, K. Yuan, Y. Chen, H. Hu, P. Pan, and W. Yin, "Exponential graph is provably efficient for decentralized deep training," *Advances in Neural Information Processing Systems*, vol. 34, pp. 13975–13987, 2021.

[19] 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*, 2020.

[20] S. Pu and A. Nedić, "Distributed stochastic gradient tracking methods," *Mathematical Programming*, vol. 187, no. 1, pp. 409–457, 2021.

[21] Y. Huang, Y. Sun, Z. Zhu, C. Yan, and J. Xu, "Tackling data heterogeneity: A new unified framework for decentralized sgd with sample-induced topology," in *International Conference on Machine Learning*, pp. 9310–9345, PMLR, 2022.

[22] Y. Sun, G. Scutari, and A. Daneshmand, "Distributed optimization based on gradient tracking revisited: Enhancing convergence rate via surrogation," *SIAM Journal on Optimization*, vol. 32, no. 2, pp. 354–385, 2022.

[23] Y. Tian, G. Scutari, T. Cao, and A. Gasnikov, "Acceleration in distributed optimization under similarity," in *International Conference on Artificial Intelligence and Statistics*, pp. 5721–5756, PMLR, 2022.

[24] E. D. H. Nguyen, S. A. Alghunaim, K. Yuan, and C. A. Uribe, "On the performance of gradient tracking with local updates," *arXiv preprint arXiv:2210.04757*, 2022.

[25] Y. Huang, J. Xu, W. Meng, and H.-T. Wai, "Stochastic gradient tracking methods for distributed personalized optimization over networks," in *2022 IEEE 61st Conference on Decision and Control (CDC)*, pp. 4571–4578, IEEE, 2022.

[26] Y. Liu, T. Lin, A. Koloskova, and S. U. Stich, "Decentralized gradient tracking with local steps," *arXiv preprint arXiv:2301.01313*, 2023.

[27] K. Yuan, B. Ying, X. Zhao, and A. H. Sayed, "Exact diffusion for distributed optimization and learning—part i: Algorithm development," *IEEE Transactions on Signal Processing*, vol. 67, no. 3, pp. 708–723, 2018.

[28] K. Yuan, S. A. Alghunaim, B. Ying, and A. H. Sayed, "On the influence of bias-correction on distributed stochastic optimization," *IEEE Transactions on Signal Processing*, vol. 68, pp. 4352–4367, 2020.

[29] Z. Tang, S. Shi, X. Chu, W. Wang, and B. Li, "Communication-efficient distributed deep learning: A comprehensive survey," *arXiv preprint arXiv:2003.06307*, 2020.

[30] X. Cao, T. Başar, S. Diggavi, Y. C. Eldar, K. B. Letaief, H. V. Poor, and J. Zhang, "Communication-efficient distributed learning: An overview," *IEEE Journal on Selected Areas in Communications*, 2023.

[31] Y. Lu and C. De Sa, "Optimal complexity in decentralized training," in *International Conference on Machine Learning*, pp. 7111–7123, PMLR, 2021.

[32] K. Yuan, X. Huang, Y. Chen, X. Zhang, Y. Zhang, and P. Pan, "Revisiting optimal convergence rate for smooth and non-convex stochastic decentralized optimization," *Advances in Neural Information Processing Systems*, vol. 35, pp. 36382–36395, 2022.

[33] A. S. Berahas, R. Bollapragada, N. S. Keskar, and E. Wei, "Balancing communication and computation in distributed optimization," *IEEE Transactions on Automatic Control*, vol. 64, no. 8, pp. 3141–3155, 2018.

[34] C. Iakovidou and E. Wei, "S-NEAR-DGD: A flexible distributed stochastic gradient method for inexact communication," *IEEE Transactions on Automatic Control*, 2022.

[35] W. Liu, L. Chen, and W. Zhang, "Decentralized federated learning: Balancing communication and computing costs," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 8, pp. 131–143, 2022.

[36] A. S. Berahas, R. Bollapragada, and S. Gupta, "Balancing communication and computation in gradient tracking algorithms for decentralized optimization," *arXiv preprint arXiv:2303.14289*, 2023.

[37] A. Koloskova, T. Lin, and S. U. Stich, "An improved analysis of gradient tracking for decentralized machine learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 11422–11435, 2021.

[38] S. U. Stich, "Unified optimal analysis of the (stochastic) gradient method," *arXiv preprint arXiv:1907.04232*, 2019.

[39] Y. Huang and J. Xu, "On the computation-communication trade-off with a flexible gradient tracking approach," *arXiv preprint arXiv:2306.07159*, 2023.