Distributed Variable Sample-Size Stochastic Optimization With Fixed Step-Sizes

Jinlong Lei, Member, IEEE, Peng Yi, Member, IEEE, Jie Chen, Fellow, IEEE, and Yiguang Hong, Fellow, IEEE

Abstract—In this article, we consider distributed stochastic optimization over randomly switching networks, where agents collaboratively minimize the average of all agents’ local expectation-valued convex cost functions. Due to the stochasticity in gradient observations, distributedness of local functions, and randomness of communication topologies, distributed algorithms with an exact convergence guarantee under fixed step-sizes have not been achieved yet. This work incorporates variance reduction scheme into the distributed stochastic gradient tracking algorithm, where local gradients are estimated by averaging across a variable number of sampled gradients. With an identically and independently distributed random network, we show that all agents’ iterates converge almost surely to the same optimal solution under fixed step-sizes. When the global cost function is strongly convex and the sample size increases at a geometric rate, we prove that the iterates geometrically converge to the unique optimal solution, and establish the iteration, oracle, and communication complexity. The algorithm performance, including rate and complexity analysis, are further investigated with constant step-sizes and a polynomially increasing sample size. Finally, the empirical algorithm performance are illustrated with numerical examples.

Index Terms—Distributed optimization, multiagent systems, stochastic optimization, variance reduction.

I. INTRODUCTION

Distributed optimization has wide applications in economic dispatch in power grids [2], [3], trajectory planning and control for multi-robots [4], as well as machine learning over Internet of Things [5], [6].

In distributed optimization, a group of agents connected over networks cooperatively minimizes the average of all agents’ local cost functions. Prominent first-order distributed optimization algorithms have been developed, including primal domain methods that combine classical (sub)gradient steps with local averaging, such as distributed subgradient methods [7], [8], first-order methods with historical gradients [9], distributed Nesterov gradient methods [10], and distributed gradient tracking methods [11], [12]; dual domain methods employing the Lagrangian dual, e.g., distributed dual decomposition [13] and distributed ADMM [14], [15]; and primal-dual domain methods [16]–[18]. In addition, there are some works on second-order methods for stochastic and distributed optimization [19]–[21]. Please refer to the survey [22], [23] for the recent progress.

Among various formulations in distributed optimization, stochastic optimization has particular research interests in multiagent networks due to its applications in distributed estimation, stochastic control, and machine learning [24]–[26], where the local cost function is the expectation of a stochastic function. In big data-driven applications, the expectation is a sum of sampling functions, while it might be prohibitive or cumbersome to compute the exact gradient. Stochastic gradient descent (SGD) becomes popular since it is relatively easy to implement and scales well in large datasets [27].

In distributed stochastic optimization, each agent utilizes locally available sampled gradients and neighboring information to cooperatively seek the optimal solution. The key idea in the algorithm design is to compensate the lack of global cost information with local communications and compensate noisy local gradient information with samplings. One method to handle gradient noises is to adopt diminishing step-sizes. For nonsmooth convex cost functions, Ram et al. [24] investigated a distributed stochastic subgradient projection algorithm and showed its mean convergence with both the gradient estimation error and the step-size diminishing to zero, whereas Srivastava and Nedich [28] further considered the asynchronous distributed SGD (D-SGD) over random networks and proved almost sure convergence with two diminishing step-size sequences. For nonconvex problems, Bianchi and Jakubowicz [25] showed that D-SGD methods with diminishing step-sizes can guarantee the almost sure convergence to Karush–Kuhn–Tucker points. Beyond the D-SGD, Lei et al. [29] proposed a primal-dual method for distributed stochastic convex optimization over random networks corrupted with stochastic communication noises, and showed the almost sure convergence with diminishing step-sizes. However, a convergent algorithm with non diminishing step-sizes is desirable in distributed stochastic optimization, since it can lead to a faster convergence rate, save the communication cost, and endow the multiagent network with adaptivity under model drifting [30]. As far as we know, distributed algorithms for stochastic convex optimization with an exact convergence guarantee under nondiminishing step-sizes have not been achieved yet.

There have been some distributed algorithms in investigating strongly convex stochastic optimization. For example, Nedich and Olshevsky [31] proposed a subgradient-push method over time-varying directed graphs with convergence rate $O(1/\sqrt{k})$, whereas Sayin et al. [32] designed a stochastic subgradient descent with time-dependent averaging and obtained a convergence rate $O(1/k)$. In addition, Yuan et al. [33] considered a distributed stochastic mirror descent method...
with rate $O(\ln(k)/k)$ for nonsmooth functions. While for random networks, Jakovetic et al. [34] established the mean-squared convergence rate $O(1/k)$ for D-SGD. Since the aforementioned works [31]–[34] adopted diminishing step-sizes, the derived convergence rates are not comparable with the geometric rate of deterministic strongly convex optimization with constant step-sizes. Recently, Pu and Nedić [35] and Xin et al. [36] proposed distributed stochastic gradient tracking methods with constant step-sizes, but only showed that the iterates are attracted to a neighborhood of the optimal solution in expectation at a geometric rate. With a different perspective, Alghunaim and Sayed[37] proposed a distributed penalty gradient method for constrained stochastic optimization with a fixed step-size, but also showed the geometric convergence to a neighborhood of the optimal solution. Thereby, how to distributedly achieve a linear convergence for strongly convex stochastic optimization needs further investigation.

Variance reduction schemes have gained increasing research interests in stochastic convex optimization [38]–[41]. In the class of variable sample-size schemes, the true gradient is estimated by the average of an increasing number of sampled gradients, which can progressively reduce the variance of the sample-averaged gradients. For example, Shanbhag and Blanchet [38] obtained the geometric rate for strongly convex problems, whereas Ghadimi and Lan [39] combined the accelerated method and proved the rate $O(1/k^2)$ for smooth convex problems. Alternative variance reduction schemes, such as SAGA [40] and SVRG [41], mainly applied to finite-sum optimization problems in machine learning, lead to the recovery of the convergence rates in deterministic cases. Such schemes were also investigated in distributed finite-sum optimization [42]–[44], but relying on periodically using exact gradients. However, distributed variance reduced schemes for general distributed stochastic optimization without using exact gradients remains open.

This article aims to provide a fast and communication-efficient algorithm for distributed stochastic optimization, where the communication is many times of the local computation cost. We incorporate the variable sample-size scheme into the distributed stochastic gradient tracking algorithm, and derive the following results.

1) We propose a distributed algorithm, where each agent estimates its local gradients by a variable number of sampled gradients, takes a weighted averaging of its neighbors’ iterates, and moves toward the negative direction of the weighted combination of its neighbors’ gradient estimations.

2) Assume that each sampled gradient is unbiased with a bounded variance, and each gradient function is Lipschitz continuous. For identically and independently distributed (i.i.d.) random networks with connected mean graph, we prove the almost sure convergence for merely convex functions.

3) If the global cost function is strongly convex, we prove the geometric convergence with a geometrically increasing sample size, and obtain the iteration, communication, and oracle complexity $O(\ln(1/\epsilon)), O(1/\epsilon^2)$, and $O(\epsilon^{-1}\ln(1/\epsilon))$ for achieving an $\epsilon$-optimal solution $\mathbb{E}[\|x - x^\ast\|] < \epsilon$. We show that with a constant sample size, the estimates geometrically converge to a neighborhood of the optimal solution, and further investigate the polynomial rate and complexity bounds with a polynomially increasing sample size. The previous results quantitatively characterize the tradeoff between communication complexity and computation complexity for distributed stochastic optimization.

The novel perspective of this article is that by progressively reducing the variance of gradient noises through increasing the sample size, we can adopt constant step-sizes to achieve an exact convergence in distributed stochastic optimization. Compared with algorithms using diminishing step-sizes [24], [25], [28], the proposed algorithm can achieve a faster convergence with constant step-sizes. Hence it can significantly reduce the communication costs. Moreover, for strongly convex stochastic optimization, the derived iteration complexity is of the same order as the centralized algorithm in deterministic cases [45]. The oracle complexity is also comparable with centralized SGD, for example, the bound of [46] is $O(1/\epsilon)$ for making the suboptimality gap $\mathbb{E}[F(x)] - F(x^\ast) < \epsilon$. Compared with existing methods [31]–[36] and [42]–[44], the proposed scheme saves the communication costs without increasing the overall sampling burden too much or using the exact gradient periodically. These findings provide the guide and insight for solving practical network problems, especially in wireless communication networks, where the communication costs are more expensive than sampling costs.

The rest of this article is organized as follows. A distributed variable sample-size stochastic gradient tracking algorithm is proposed in Section II. The almost sure convergence for convex functions is provided in Section III. Then, the geometric (respectively, polynomial) convergence rate along with complexity bounds are established in Section IV for strongly convex functions with geometrically (respectively, polynomially) increasing sample size. The numerical studies are presented in Section V. Finally, Section VI concludes this article.

**Notations:** Depending on the argument, $\cdot$ stands for the absolute value of a real number or the cardinality of a set. The Euclidean norm of a vector or a matrix is denoted as $\|\cdot\|_2$ or $\|\cdot\|$. The spectral radius of a matrix $A$ is denoted as $\rho(A)$. Let $\otimes$ denote the Kronecker product. The expectation of a random variable is denoted as $\mathbb{E}[\cdot]$. Let $1_x$ denote the $n$-dimensional column vectors with all entries equal to 1 and $I_i$ denote the $d \times d$ identity matrix.

A directed graph is denoted by $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, n\}$ is a finite set of nodes and an edge $(i, j) \in \mathcal{E}$ if node $j$ can receive information from agent $i$. A directed path in $G$ from $v_1$ to $v_p$ is a sequence of distinct nodes, $v_1, \ldots, v_p$, such that $(v_m, v_{m+1}) \in \mathcal{E}$ for all $m = 1, \ldots , p - 1$. The graph $G$ is termed strongly connected if for any two distinct nodes $i, j \in \mathcal{V}$, there is a directed path from node $i$ to node $j$. Given a nonnegative matrix $A = [a_{ij}] \in \mathbb{R}^{n \times n}$, denote by $G_A = \{V, E_A\}$ the corresponding digraph, where $V = \{1, \ldots, n\}$ and $(i, j) \in E_A$ if $a_{ij} > 0$.

**II. PROBLEM STATEMENT AND DISTRIBUTED ALGORITHM**

In this section, we formulate a distributed stochastic optimization problem and propose a fully distributed stochastic gradient tracking algorithm by using a variable number of sampled gradients to estimate the exact gradients.

**A. Problem Formulation**

Consider a network of $n$ agents indexed as $\mathcal{V} = \{1, \ldots, n\}$. Each agent $i \in \mathcal{V}$ has an expectation-valued cost function $f_i(x) \triangleq E_x[h_i(x, \xi)]$, where $x \in \mathbb{R}^m$, the random vector $\xi_i : \Omega_i \rightarrow \mathbb{R}^m$ is defined on the probability space $(\Omega_i, \mathcal{F}_i, P)$, and $h_i : \mathbb{R}^d \times \mathbb{R}^m \rightarrow \mathbb{R}$ is a proper scalar-valued function. The agents in the network need to cooperatively find an optimal solution that minimizes the average of all agents’ local cost functions, i.e.,

$$\min_{x \in \mathbb{R}^d} F(x) \triangleq \frac{1}{n} \sum_{i=1}^{n} f_i(x). \tag{1}$$

We require the cost functions to be convex and smooth.

**Assumption 1:** For each agent $i \in \mathcal{V}$, (i) the cost function $f_i$ is convex and (ii) the gradient function $\nabla f_i$ is $L$-Lipschitz continuous, i.e.,

$$\| \nabla f_i(x_1) - \nabla f_i(x_2) \| \leq L \| x_1 - x_2 \| \forall x_1, x_2 \in \mathbb{R}^d.$$ 

Denote by $X^* \triangleq \{x \in \mathbb{R}^d : \nabla F(x) = 0\}$ the optimal solution set and by $F^*$ the optimal function value. The first-order optimality condition, the optimal solution $x^* \in X^*$ satisfies $\nabla F(x^*) = 0$.

Suppose, in addition, that for agent $i \in \mathcal{V}$, there exists a stochastic first-order oracle that returns a sampled gradient $\nabla h_i(x, \xi)$ given $x, \xi$, which is an unbiased estimator of $\nabla f_i(x)$ with a bounded second-order moment.

**Assumption 2:** There exists a constant $\nu > 0$ such that for each $i \in \mathcal{V}$ and any given $x \in \mathbb{R}^d$, $E_{\xi_i}[\| \nabla h_i(x, \xi_i) - \nabla f_i(x) \|^2] \leq \nu^2$. 


Let $t \geq \tilde{t} + 1$) $R N \in V_1(5)$ $R_{i+1} = \rho \tilde{\chi}_x$ generated by the expected adjacency matrix (3a) is the in the $F_i \in V_{\tilde{\chi}} \{ \tilde{j} \}$ denotes a diagonal matrix with $y_i \in \tilde{\chi}_y$ $i \approx j$ denotes a diagonal matrix with $\alpha_i \in \tilde{\chi}_y$ diagonal. Then, Algorithm 1 can be written in a compact form as $x(k + 1) = (A(k) \otimes I_d) x(k) - (\alpha \otimes I_d) y(k)$ $y(k + 1) = (A(k) \otimes I_d) y(k) + \nabla f(k + 1) + w(k + 1) - \nabla f(k) - w(k)$. Denote the averaged estimate of the optimal solution and the averaged gradient across the network as $\bar{x}(k) = \frac{1}{n} \sum_{i=1}^{n} x_i(k)$ and $\bar{y}(k) = \frac{1}{n} \sum_{i=1}^{n} y_i(k)$. (4) We further denote $D_{\perp} = I_n - \frac{1}{n} 1_n \otimes y$, and $\ddot{x}(k) = (D_{\perp} \otimes I_d) x(k) = (x(k) - (1_n \otimes I_d) \ddot{x}(k)$ $\ddot{y}(k) = (D_{\perp} \otimes I_d) y(k) = (y(k) - (1_n \otimes I_d) \ddot{y}(k)$ (6) Define $\mathcal{F}(k) = \{ x(0), A(0), \ldots, A(k-1), \{ \xi_i^h(t) \}_{j=1}^{N(k)} \}_{t=0}^{l \leq k, t = 1, \ldots, n}$. From Algorithm 1, it is seen that both $x(k)$ and $y(k)$ are adapted to $\mathcal{F}(k)$; hence, $\ddot{x}(k)$ and $\ddot{y}(k)$ are adapted to $\mathcal{F}(k)$. Recall that $\ddot{x}(k)$ is adapted to $\mathcal{F}(k)$, and $A(k)$ is independent of $\mathcal{F}(k)$. Then, by using $A(k) \frac{1}{n} \xi_i^h = \frac{1}{n} \xi_i$ and $A(k) \frac{1}{n} \xi_i^h = \frac{1}{n} \xi_i$, we derive $E \left[ \left| (A(k) - \frac{1}{n} \xi_i^h) \otimes I_d \ddot{x}(k) \right|^{2} / \mathcal{F}(k) \right] = E \left[ \ddot{x}(k)^T \left( A(k) - \frac{1}{n} \xi_i^h \right) A(k) \frac{1}{n} \xi_i^h \right] \otimes I_d \ddot{x}(k) \right] = \ddot{x}(k)^T E \left[ A(k)^T A(k) \frac{1}{n} \xi_i^h \right] \otimes I_d \ddot{x}(k) \leq \rho \left( E \left[ A(k)^T A(k) \right] - \frac{1}{n} \xi_i^h \right) \| \ddot{x}(k) \|^2$. Therefore, by applying the Jensen’s inequality for conditional expectations, we obtain that $E \left[ \left| (A(k) - \frac{1}{n} \xi_i^h) \otimes I_d \ddot{x}(k) \right|^{2} / \mathcal{F}(k) \right] \leq \rho \left( E \left[ A(k)^T A(k) \right] - \frac{1}{n} \xi_i^h \right) \| \ddot{x}(k) \|^2$. (7) By Assumption 3, we see that the graph generated by the matrix $E[A(k)^T A(k)]$ is undirected and connected. Thus, $\rho \left( E \left[ A(k)^T A(k) \right] - \frac{1}{n} \xi_i^h \right) \in (0, 1)$. The parameter $\rho_1$ depends on the network topology, where larger $\rho_1$ implies worse network connectivity. It was shown in [49, Prop. 5] that when the adjacency matrix follows the Lazy

### III. Almost Sure Convergence for Convex Functions

In this section, we provide the almost sure convergence of the algorithm for merely convex cost functions.

Define the gradient observation noise as follows:

\[
\begin{align*}
\omega_i(k) &\triangleq \tilde{g}_i(x_i(k)) - \nabla f_i(x_i(k)), \\
\omega(k) &\triangleq (\omega_1(k)^T, \ldots, \omega_n(k)^T)^T \in \mathbb{R}^{nd}.
\end{align*}
\]

Denote by

\[
\begin{align*}
x(k) &\triangleq (x_1(k)^T, \ldots, x_n(k)^T)^T \in \mathbb{R}^{nd}, \\
y(k) &\triangleq (y_1(k)^T, \ldots, y_n(k)^T)^T, \\
\nabla f(k) &\triangleq (\nabla f_1(x_1(k))^T, \ldots, \nabla f_n(x_n(k))^T)^T
\end{align*}
\]

and $\alpha \triangleq \text{diag} \{\alpha_1, \ldots, \alpha_n\} \in \mathbb{R}^{n \times n}$ where $\text{diag} \{\alpha_1, \ldots, \alpha_n\}$ denotes a diagonal matrix with $\alpha_i$ in the $i$th diagonal.

Due to the lack of global cost function information, the agents need to cooperatively solve the task by sharing information with local communications. Let the interaction among the agents at time $k = 0, 1, 2, \ldots$ be described by a digraph $G(k) = \{ V, E(k) \}$, where $(j, i) \in E(k)$ if node $i$ can receive information from agent $j$ at time $k$. Denote by $X_i(k) \triangleq \{ j \in V : (j, i) \in E(k) \}$ the neighboring set of node $i$ at time $k$. The corresponding adjacency matrix is $A(k) = [a_{ij}(k)]_{i,j=1}^{n}$, where $a_{ij}(k) > 0$ if $(j, i) \in E(k)$ and $a_{ij}(k) > 0$, otherwise. The following are the assumptions imposed on the communication graphs.

**Assumption 3:**

1. For all time $k \geq 0$, $A(k)$ is doubly stochastic, i.e., $1_n^T A(k) = 1_n^T$ and $A(k) 1_n = 1_n$.

2. $\{ A(k) \}_{k=0}^{\infty}$ is an i.i.d. matrix sequence.

3. The graph $G_A$ generated by the expected adjacency matrix $\bar{A} \triangleq E[A(k)]$ is strongly connected.

**Remark I:** Random graphs satisfying Assumption 3 can cover i.i.d. undirected graphs [47], random gossip and broadcast communications [28], [34], etc. Assumption 3 requires each digraph $G(k)$ to be weight-balanced was also used in existing works, such as [7], [12], and [48]. Specifically, for the gossip scheme in undirected and connected underlying graphs, the doubly stochastic adjacency matrix was designed [49]. Nevertheless, it is usually nontrivial to generate doubly stochastic weights for general digraphs, but there are distributed algorithms to fulfill the task (see, e.g., [50]). Assumption 3 implies that the graph sequence ($G(k)$) is i.i.d. over time $k$. Assumption 3 imposes a connectivity condition on the mean graph similarly to that of [12] and [34].

### B. Distributed Algorithm with Variable Sample-Sizes

Each agent $i$ at time $k$ maintains two variables $x_i(k)$ and $y_i(k)$ to estimate the optimal solution and to track the average gradient, respectively. Since the exact gradient of each expectation-valued cost function $f_i(x)$ is unavailable, we approximate it by averaging through a variable number of sampled gradients

\[
\tilde{g}_i(x_i(k)) = \frac{1}{N(k)} \sum_{p=1}^{N(k)} \nabla f_i(x_i(k), \xi_i^p), \quad \forall k \geq 0
\]

where $N(k)$ is the number of sampled gradients utilized at time $k$, and the samples $\{ \xi_i^p \}_{p=1}^{N(k)}$ are randomly and independently generated from the probability space $(\Omega, \mathcal{F}, P)$. The gradient estimate given by (2) is an unbiased estimate of the exact gradient $\nabla f_i(x_i(k))$, and the variance of the gradient observation noise $\tilde{g}_i(x_i(k)) - \nabla f_i(x_i(k))$ can be progressively reduced by increasing the sample size $N(k)$. By combining the distributed gradient tracking scheme with such a variance reduction scheme, we obtain Algorithm 1.

**Algorithm 1: Distributed Variable Sample-Size Stochastic Gradient Tracking Algorithm.**

**Initialization:** Set $k := 0$. For any $i = 1, \ldots, n$, let $y_i(0) = \tilde{g}_i(x_i(0))$ with an arbitrary initial $x_i(0) \in \mathbb{R}^d$.

**Iterate until convergence.**

Each agent $i = 1, \ldots, n$ updates its estimates as follows:

\[
\begin{align*}
x_i(k + 1) &= \sum_{j \in X_i(k)} a_{ij}(k) x_j(k) - \alpha_i y_i(k), \quad (3a) \\
y_i(k + 1) &= \sum_{j \in X_i(k)} a_{ij}(k) y_j(k) + \tilde{g}_i(x_i(k + 1)) - \tilde{g}_i(x_i(k)) \quad (3b)
\end{align*}
\]

where $\alpha_i > 0$ is the fixed step-size used by agent $i$, and $\tilde{g}_i(x_i(k))$ is defined in (2).
Suppose Assumptions 1(ii), 3, and 4 hold. Consider $\bar{x}$ a.s. can be replaced $\sum \otimes x - \sum \otimes x$ are defined in (8). Then the following satisfy the following with $\min \nabla = 0 \otimes \otimes$ to guarantee that $\max \frac{1}{n+1} \frac{1}{L}$.

Next, we give the almost sure convergence of Algorithm 1. We omit the proof of Theorem 1 due to the page limitation, while the detailed proof can be found in [52, Appendix A-C].

**Theorem 1:** Suppose Assumptions 1–3 hold. Let $\{x(k)\}$ and $\{y(k)\}$ be generated by Algorithm 1, where $\sum_{k=0}^{\infty} \frac{1}{\sqrt{\alpha}} < \infty$. Then, there exist $\alpha > 0$, $i \in V$ possibly depending on $\rho_1$, $L$, and $d_0$.

$$d_0 \triangleq \sqrt{\sum_{i=1}^{n} (\alpha_i - \bar{\alpha})^2} \quad (9)$$

such that

$$\lim_{k \to \infty} \|x(k) - x_i(k)\| = 0 \quad \forall i \in V, \text{ a.s.},$$

$$\lim_{k \to \infty} F(x(k)) = F^*, \text{ a.s.} \quad (10)$$

**Remark 2:** Theorem 1 shows that the exact convergence in an almost sure sense can be achieved for merely convex problems with uncoordinated constant step-sizes by suitably choosing the sample size. The proposed algorithm with constant step-sizes can achieve a faster convergence rate compared with the algorithms with diminishing step-sizes [31]–[34]. As similar discussions in [53], one major reason for considering the agent-specific step-size is due to the heterogeneity of agents and lack of coordination involved in distributed computation. Theorem 1 validates that the distributed variable sample-size stochastic gradient tracking algorithm with uncoordinated constant step-sizes can also achieve the exact convergence to an optimal solution in the almost sure sense.

Theorem 1 uses identical sample size for all agents just for the ease of proof presentation. Suppose that agents utilize different sample sizes, i.e., agent $i$ uses $N_i(t)$ at time $k$. Denote by $N_{\min}(k)$ the minimum of $N_i(k)$. Then, the condition $\sum_{k=0}^{\infty} \frac{1}{\sqrt{\alpha}} < \infty$ can be replaced with $\sum_{k=0}^{\infty} \frac{1}{\sqrt{\alpha}} < \infty$. There are many ways for choosing the batch-size $N(k)$, for example, $k \ln^d(k)$ or $k^{1+d}$ with $\delta > 0$.

The following corollary gives a sufficient condition on constant step-sizes when all agents take an identical step-size. Its proof can be found in [52, Appendix A-D]. It quantitatively characterizes the dependence of the step-size on the Lipschitz constant $L$ and the network connectivity parameter $\rho_1$. It can be seen that a larger $L$ leads to a smaller upper bound of the step-size, whereas a better network connectivity (i.e., smaller $\rho_1$) implies a larger step-size.

**Corollary 1:** Suppose Assumptions 1–3 hold. Consider Algorithm 1, where $\sum_{k=0}^{\infty} \frac{1}{\sqrt{\alpha}} < \infty$ and $\alpha_i \equiv \alpha$ with

$$\alpha \in \left(0, \frac{\alpha_i + 2\sqrt{\alpha_i} - \sqrt{\alpha_i + 2\sqrt{\alpha_i}^2 - 4\alpha_i^2}}{2L}\right).$$

with $c_0 \triangleq \frac{(1 - \rho_1)^2}{(2 - \rho_1)^2}$. Then the results established in (10) hold.

**IV. RATE ANALYSIS FOR STRONGLY CONVEX FUNCTIONS**

This section explores the convergence properties of Algorithm 1 when the global cost function is strongly convex. The geometric (respectively, polynomial) convergence rate is obtained if the number of the sampled gradients increases at a geometric (respectively, polynomial) rate. In addition, the complexity bounds for obtaining an $\epsilon$-optimal solution are established as well.

**A. Linear Convergence Rate Analysis**

**Assumption 4:** The global cost function $F(x)$ is $\eta$-strongly convex, i.e., for any $x_1, x_2 \in \mathbb{R}^d$

$$(\nabla F(x_1) - \nabla F(x_2))^T(x_1 - x_2) \geq \eta \|x_1 - x_2\|^2.$$  

With Assumption 4, the problem (1) has a unique optimal solution, denoted by $x^\star$, and $\nabla F(x^\star) = 0$. We analyze the algorithm performance by characterizing the interactions among three error sequences: 1) the distance from the average estimate to the optimal solution $\|\bar{x}(k) - x^\star\|$, 2) the consensus error $\|x_i(k) - (1_n \otimes I_d)(x_i(k))\|$, and 3) the consensus error of the gradient trackers $\|y_i(k) - (1_n \otimes I_d)(y_i(k))\|$.

We will bound the three error sequences in terms of linear combinations of their past values in the following lemma.

**Lemma 1:** Suppose Assumptions 1(ii), 3, and 4 hold. Consider Algorithm 1 with $0 < \alpha_i < \frac{2}{\eta + L}$.

Define

$$z(k) \triangleq \begin{pmatrix} \mathbb{E}[\|x(k) - x^\star\|] \\ \mathbb{E}[\|y(k) - (1_n \otimes I_d)(y(k))\|] \end{pmatrix} \text{ and }$$

$$J(\alpha) \triangleq \begin{pmatrix} 1 - \alpha \bar{\alpha} & \frac{\delta}{\sqrt{\alpha_0}} & \frac{\alpha}{\sqrt{\alpha_0}} \\ \frac{c_1 L}{\alpha} & \frac{\rho_1 + \frac{\alpha}{\sqrt{\alpha_0}}}{\sqrt{\alpha_0}} & \frac{\alpha_{\max}}{\sqrt{\alpha_0}} \\ c_2 L^2 & L + \frac{\alpha_{\max}}{\sqrt{\alpha_0}} & \rho_1 + \frac{\alpha_{\max} L}{\sqrt{\alpha_0}} \end{pmatrix} \quad (11)$$

where $\alpha, c_1, \alpha_{\max},$ and $c_2$ are defined in (8). Then the following componentwise linear matrix inequality holds for any $k \geq 0$:

$$z(k + 1) \leq J(\alpha) z(k)$$

and

$$+ \left( \frac{c_3}{\sqrt{\alpha_0}} \mathbb{E}[\|w(k + 1) - w(k)\|], \frac{c_4 L}{\sqrt{\alpha_0}} \mathbb{E}[\|w(k)\|] \right) .$$

**Proof:** The proof can be found in [52, Appendix B-A].

We now show the geometric convergence of Algorithm 1 with geometrically increasing sample size and suitably selected step-sizes. For nonidentical step-sizes $\alpha_i$, we have $d_0 > 0$ by (9). Define $\kappa \triangleq L/\eta$ and let $\alpha_i$ satisfy the following with $\rho_1$ defined by (7): $\beta > 0, \kappa > 0$.

$$\min \left\{ \beta^*, \frac{1}{\alpha_i} \frac{1}{\alpha_i} \right\} \quad \forall i \in V \quad (12)$$

with $\beta^* \triangleq \frac{\kappa^2 + 4\kappa(1 - \rho_1)\kappa}{2\kappa}$ and

$$c_3 = (\sqrt{d_0^2 + 1})(1 + \kappa d_0^2) + \kappa \sqrt{d_0^2 + 1}.$$

**Theorem 2:** Suppose Assumptions 1(ii)–4 hold. Let $\{x(k)\}$ and $\{y(k)\}$ be generated by Algorithm 1 with $N(k) = [q^{-2k}]$ for some $q \in (0, 1)$. If the step-size $\alpha_i$, $i \in V$ satisfies (12), then the spectral radius of $J(\alpha)$ in (11), denoted by $\rho(J(\alpha))$, is strictly smaller than 1. In addition, the error sequence $z(k)$ converges to zero at a linear rate $\mathcal{O}(\max(\rho(J(\alpha)), q^k))$.

**Proof:** The proof can be found in [52, Appendix B-B].

Equation (12) gives a sufficient condition for selecting step-sizes $\alpha_i, i \in V$ to guarantee that $\rho(J(\alpha)) < 1$. It shows how parameters $\eta, L, d_0$, and $\rho_1$ influence the selection of constant step-sizes. Theorem 2 implies that if the number of sampled gradients is increased at a geometric rate $[q^{-2k}]$ with $q \in (0, 1)$, the error sequences $\mathbb{E}[\|x(k) - x^\star\|$ and $\mathbb{E}[\|x(k) - (1_n \otimes I_d)(x(k))\|]$ converge to zero at a geometric rate.
The following corollary shows the convergence rate for the case with an identical step-size, i.e., $\alpha_i \equiv \alpha$. Define
\[
\bar{J}(\alpha) \triangleq \begin{pmatrix}
1 - \alpha \eta & \frac{1}{n} & 0 \\
0 & \sqrt{\alpha}x^2 & L + \alpha L^2 \\
0 & \rho_1 & \rho_1 + \alpha L
\end{pmatrix}.
\]
The condition (13), making $\rho(\bar{J}(\alpha)) < 1$, implies that a better network connectivity (namely, a smaller $\rho_1$) leads to a larger $\alpha$, while the ill-conditioned optimization problem with a large $\kappa$ narrows the possible selection of $\alpha$.

**Corollary 2:** Suppose that Assumptions 1(ii)--4 hold. Let $\{x(k)\}$ and $\{y(k)\}$ be generated by Algorithm 1, where
\[
\alpha_k \equiv \alpha < \frac{2 - \rho_1 + \sqrt{(2 - \rho_1)^2 + 4(1 + \kappa)(1 - \rho_1)^2}}{2L(1 + \kappa)}.
\]
Set $N(k) = [q^{-2 k}]$ for some $q \in (\rho(\bar{J}(\alpha)), 1)$. Then
\[
z(k) \approx \bar{J}(\alpha)^{\tau}z(0) + \nu q^{k-1} \left( I_3 - \bar{J}(\alpha)/q \right)^{-1} \left( \begin{array}{c}
\alpha \\
0 \\
(1 + \kappa + \alpha L)\sqrt{n}
\end{array} \right).
\]

**Proof:** The proof can be found in [52, Appendix B-C].

For strongly convex stochastic optimization, [35]--[37] also proved geometric convergence rates but only to a neighborhood of the optimal solution. By progressively reducing the gradient noises with geometrically increasing batch-sizes, we prove that the exact and geometric convergence in a mean-squared sense. The following corollary shows that when a constant sample size is used in Algorithm 1, the linear convergence to a neighborhood of the optimal solution can be obtained as well. It can be seen that the bounds depend on the network structure, batch-size, and step-size, as well as the problem parameters $\nu$, $L$, and $\nu$.

**Corollary 3:** Let Assumptions 1(ii)--4 hold. Consider Algorithm 1 with $N(k) \equiv B > 0$, where $\alpha_i \equiv \alpha$, $i \in V$, satisfies (13). Then $\sup_{k \geq 3} E[\|z(I) - x^*\|^2]$ and $\sup_{k \geq 3} E[\|x(I) - (1_n \otimes I_d) \bar{x}(k)\|]$ converge to $\lim_{k \to \infty} E[\|z(k) - x^*\|^2]$ and $\lim_{k \to \infty} E[\|z(k) - (1_n \otimes I_d) \bar{x}(k)\|]$ with a geometric rate $O(\rho(\bar{J}(\alpha))^k)$. Furthermore,
\[
\limsup_{k \to \infty} E[\|z(k) - x^*\|^2] \leq \frac{\nu(1 - \rho_1)^2 + \rho_1 + \alpha L}{\sqrt{B} \eta ((1 - \rho_1)^2 - (1 + \kappa)\alpha^2 L^2 - (2 - \rho_1)\alpha L)}
\]
and
\[
\limsup_{k \to \infty} E[\|z(k) - (1_n \otimes I_d) \bar{x}(k)\|^2] \leq \frac{\alpha \sqrt{n} \nu (\alpha L^2 + \nu (2 + \alpha L))}{\sqrt{B} \eta ((1 - \rho_1)^2 - (1 + \kappa)\alpha^2 L^2 - (2 - \rho_1)\alpha L)}.
\]

**B. Complexity Analysis**

Based on the geometric rate of convergence established in Theorem 2, we are able to establish the complexity bounds for obtaining an $\epsilon$-optimal solution satisfying $\|z(k)\| \leq \epsilon$. The iteration complexity is defined as $K(\epsilon) \triangleq \min_{k \geq 0} \{ k : \|z(k)\| \leq \epsilon \}$, while the oracle complexity, measured by the total number of sampled gradients for deriving an $\epsilon$-optimal solution, is $\sum_{k=1}^{K(\epsilon)} N(k)$.

**Theorem 3:** Let Assumptions 1(ii)--4 hold. Consider Algorithm 1 with $N(k) = [q^{-2 k}]$, $q \in (0, 1)$, and $\alpha_i$, $i \in V$ satisfying (12). i) When $\rho(J(\alpha)) < q < 1$, the iteration and oracle complexity required to obtain an $\epsilon$-optimal solution are $O(\ln(1/\epsilon))$ and $O(1/\epsilon^2)$, respectively.

ii) When $0 < q < \rho(J(\alpha))$, the iteration and oracle complexity required to obtain an $\epsilon$-optimal solution are $O(\ln(1/\epsilon))$ and $O(1/\epsilon^2)$, respectively.

**Proof:** By Theorem 2, there exists $C_1 > 0$ such that $\|z(k)\| \leq C_1 q^k$. Then, for any $k \geq K_1(\epsilon) = \ln(C_1(\epsilon) \ln(1/q^k))$, we have $\|z(k)\| \leq \epsilon$. This allows us to bound the oracle complexity by
\[
K_1(\epsilon) \leq \frac{\ln(C_1(\epsilon) \ln(1/q^k))}{\ln(1/q^k)} = \frac{1}{1 - q^k} \ln(1/q).
\]

**Remark 3:** Theorem 3 shows that, with geometrically increasing batch-size, the number of iterations required to obtain an $\epsilon$-optimal solution is $O(\ln(1/\epsilon))$, which matches the optimal iteration complexity for strongly convex optimization in the deterministic regime. The oracle complexity for achieving $\|z(k) - x^*\|^2 \leq \epsilon$ and $\|y(k) - x^*\|^2 \leq \epsilon$ is $O(1/\epsilon^2)$ when $q \in (\rho(J(\alpha)), 1)$. Recall that for the centralized SGD, the oracle complexity for making either the suboptimality gap $E[\|z(k) - F(x^*)\|] < \epsilon$ or the mean-squared error $E[\|z(k) - x^*\|^2] < \epsilon$ is $O(1/\epsilon^2)$ (see, e.g., [46]), which implies that the oracle complexity for obtaining $E[\|z(k) - x^*\|^2] < \epsilon$ is $O(1/\epsilon^2)$. Thus, the number of sampled gradient required by Algorithm 1 with $N(k) = [q^{-2 k}], q \in (\rho(J(\alpha)), 1)$, to achieve a given solution accuracy matches that of the centralized SGD.

Next, we investigate the communication complexity for obtaining an approximate solution. We consider a special case with a fixed graph and impose the following condition.

**Assumption 5:**

i) $\tilde{G}(k) \equiv G$, where $G$ is strongly connected.

ii) $A(\epsilon) \equiv A$, where the adjacency matrix $A$ associated with $G$ is doubly stochastic.

**Theorem 4:** Let Assumptions 1(ii)--5 hold. Consider Algorithm 1 with $N(k) = [q^{-2 k}], q \in (0, 1)$, where the step-size $\alpha_i$, $i \in V$ satisfies (12). Then, the number of communications required to obtain $\|z(k)\| \leq \epsilon$ is $O(n(\ln(1/\epsilon)))$.

**Proof:** In each iteration $k$, agent $i$ requires $2|N_i|$ rounds of communication to obtain its neighbors’ information $x_i(k)$ and $y_i(k)$. Thus, the number of communication rounds required across the network at time $k$ is $2|E|$. Since the number of iterations required to obtain $\|z(k)\| \leq \epsilon$ is $O(1/\epsilon^2)$, the number of total communication rounds required is $O(|E| \ln(1/\epsilon))$.

There could exist settings where a geometrically increasing batch-size is impractical. To this end, we consider the use of polynomially increasing batch-size that allows for more gentle growth of sampling batch, and proceed to investigate the convergence rate as well as the complexity bounds. The proof can be found in [52, Appendix B-E].
**Theorem 5:** Let Assumptions 1(ii)–5 hold. Consider Algorithm 1 with $N(k) = [(k + 1)^{2\theta}], \theta > 0$, and the step-size $\alpha_i, i \in \mathcal{V}$ satisfying (12). Then $z(k)$ converges to zero at a polynomial rate $O(k^{-\theta})$. In addition, the number of samples and communications required to make $\|z(k)\| \leq \epsilon$ is $O(1/(1/e)2^{1/\theta})$ and $O((1/e)1/\theta)$, respectively.

**Remark 4:** Although an increasing batch-size implies a higher sampling and computation burden than SGD with a single iteration, the proposed scheme can significantly reduce the communication burden compared with [31]–[35]. Thus, Algorithm 1 is superior in many practical networks especially in wireless networks, where the communication cost is usually much higher than gradient computations. Therefore, the proposed scheme can remarkably save the communication cost by fully exploiting the local computation resources. Theorems 4 and 5 quantitatively characterize the tradeoff between communication costs and sampling rate. \[\square\]

**V. NUMERICAL SIMULATIONS**

In this section, we examine the empirical performance of Algorithm 1 on the distributed parameter estimation problems.

**A. Distributed Parameter Estimation Problem**

Consider a network of $n$ spatially distributed sensors that aim to estimate an unknown $d$-dimensional parameter $x^*$. Each sensor $i$ collects a set of scalar measurements $\{d_{i,p}\}$ generated by the following linear regression model corrupted with noises:

$$d_{i,p} = u_{i,p}^T x^* + \nu_{i,p},$$

where $u_{i,p} \in \mathbb{R}^d$ is the regression vector accessible to agent $i$ and $\nu_{i,p} \in \mathbb{R}$ is a zero-mean Gaussian noise.

Suppose that $\{u_{i,p}\}$ and $\{\nu_{i,p}\}$ are mutually independent Gaussian sequences with distributions $\mathcal{N}(0, R_{u,i})$ and $\mathcal{N}(0, \sigma^2_{\nu,i})$, respectively. Then, the distributed parameter estimation problem can be modeled as a distributed stochastic quadratic optimization problem

$$\min_{x \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f_i(x),$$

$$\text{where } f_i(x) = \mathbb{E} \left[ \|d_{i,p} - u_{i,p}^T x\|^2 \right].$$

Thus, $f_i(x) = (x - x^*)^T R_{u,i} (x - x^*) + \sigma^2_{\nu,i}$ is convex and $\nabla f_i(x) = R_{u,i} (x - x^*)$. By using the observed regressor $u_{i,p}$ and the corresponding measurement $d_{i,p}$, the sampled gradient $u_{i,p}^T x^* - d_{i,p} u_{i,p}$ satisfies Assumption 2.

**B. Numerical Simulations**

Set $d = 10, n = 200$, and the true parameter $x^* = 1/\sqrt{d}$. We randomly generate ten undirected networks, where any two distinct agents are linked with probability 0.1. A graph is uniformly sampled at each iteration such that Assumption 3 holds. Weights of the adjacency matrix are constructed based on the Metropolis rule

$$a_{ij} = \begin{cases} \frac{\max(1, |d_{i}|, |d_{j}|)}{1 - \sum_{j \neq i} a_{ij}}, & \text{if } (i, j) \notin \mathcal{E}, \ i \neq j \\ 0, & \text{otherwise} \end{cases},$$

where $d_{i}$ denotes the number of neighbors of agent $i$.

**Validation of Theorem 1:** Let each covariance matrix $R_{u,i}$ be positive semidefinite with two eigenvalues equal to 0, that is, each $f_i(x)$ is merely convex. We run Algorithm 1 with $\alpha = 0.01$ and $N_k = \left[k^{1/3}\right]$, and display the estimation errors of a sample path in Fig. 1, which shows that the generated iterates will asymptotically converge to the true parameter $x^*$.

**Validation of Theorems 2 and 3:** Let each covariance matrix $R_{u,i}$ be positive definite. Then each $f_i(x)$ is strongly convex and $x^*$ is the unique optimal solution to (14). We run Algorithm 1 with $\alpha = 0.01$ and $N_k = \left[0.98^{-k}\right]$, and examine the empirical rate of convergence and oracle complexity, where the empirical mean-squared error is based on averaging across 50 sample trajectories. The convergence rate shown in Fig. 2 demonstrates that the iterates $\{x_k\}$ generated by Algorithm 1 converge to $x^*$ at a geometric rate. The oracle complexity is shown in Fig. 3, where $x$-axis is $1/e^2$ and $y$-axis denotes the number of sampled gradients required to ensure

$$e(k) \triangleq \mathbb{E} \left[ \left\|\frac{\bar{x}(k) - x^*}{x(k) - (1_n \otimes I_d) \bar{x}(k)}\right\| \right] < \epsilon.$$ 

The solid curve represents the empirical data, while the dashed curve denotes its linear fitting, which implies that the empirical oracle complexity fits well with the established theoretical bound $O(1/x^2)$.

**Comparison with [24] and [35]:** We compare Algorithm 1, abbreviated as D-VSS-SGT, with the D-SGD [24] and the distributed stochastic gradient tracking (D-SGT) [35] for strongly convex stochastic optimization.

First, we compare the algorithm performance of the three methods under fixed step-sizes. Set $\alpha = 0.005$ in all three schemes, and $N_k = [0.98^{-k}]$ in Algorithm 1. The empirical error $e(k) \triangleq \mathbb{E} \left[ \left\|\frac{\bar{x}(k) - x^*}{x(k) - (1_n \otimes I_d) \bar{x}(k)}\right\| \right]$ with respect to the number of sampled gradients is given in Fig. 4. It shows that the iterates of D-SGD and D-SGT ceased at a neighborhood of the true parameter $x^*$, while the iterates generated by Algorithm 1 will asymptotically converge to the true value $x^*$. It also shows that D-SGD and D-SGT are more efficient in utilizing the samples than Algorithm 1 at the first few samples, but with the increasing of gradient samples, Algorithm 1 is superior than D-SGD and D-SGT.

We further compare the iteration and oracle complexity of the three methods, where Algorithm 1 uses a constant step-size, while D-SGD
Fig. 4. Comparison of Algorithm 1 with D-SGD and D-SGT under the same step-size.

Fig. 5. Iteration complexity of Algorithm 1, D-SGD, and D-SGT.

Fig. 6. Oracle complexity of Algorithm 1, D-SGD and D-SGT.

Fig. 7. Rate of $e(k)$ with $N_k = \lceil \rho^{-k} \rceil.$

and D-SGT use decreasing step-sizes. The empirical number of iterations and sampled gradients required to obtain a solution with the same accuracy are demonstrated in Figs. 5 and 6. We see from Fig. 5 that Algorithm 1 can significantly reduce the iteration numbers, so do the communication rounds (costs). Meanwhile, Fig. 6 shows that Algorithm 1 requires more sampled gradients than D-SGD and D-SGT. In fact, in certain applications, such as wireless networks, high communication overhead may render a distributed scheme impractical. As such, the variance-reduced method proposed in this work is suitable for network problems when the communication costs are more expensive than sampling and local computations.

**Influence of the batch-sizes:** We run Algorithm 1 with $\alpha = 0.01$ and different geometric batch-sizes $N_k = \lceil \rho^{-k} \rceil.$ We set $\rho = 0.88, 0.9, 0.92,$ and display the empirical rate and oracle complexity in Figs. 7 and 8, respectively. We conclude from the figures that a faster increasing batch-size leads to a better convergence rate (namely, less rounds of communications) while at the cost of more sampled data and heavier computations. Hence, the parameter $\rho$ should be properly selected to balance communication costs, sampling costs, and computation costs in practice.

**Performance of Algorithm 1 with a constant batch-size:** Finally, we run the algorithm with $\alpha = 0.01$ and a constant batch-size $N_k \equiv 20,$ and display the empirical convergence in Fig. 9, which clearly shows that the estimates ceased at a neighborhood of the optimal solution.

**VI. CONCLUSION**

We proposed a distributed stochastic gradient tracking algorithm with variable sample-sizes for stochastic optimization over random networks. We proved that, with a suitably selected constant step-size and a properly increasing gradient sample-size, the iterates converge almost surely to the optimal solution for convex problems. For strongly convex problems, we further obtained the geometric convergence rate with geometrically increasing batch-sizes and established the complexity bounds for obtaining an $\epsilon$-optimal solution. Both the iteration complexity and the oracle complexity are comparable with the centralized SGD algorithm. It might be of interests to embed the push–pull method for resolving distributed stochastic optimization with general digraphs, and extend the optimal distributed algorithm [54] and its follow-up works to the stochastic setting. The extension of the current algorithm to non-convex/nonsmooth distributed stochastic optimization is a promising future research direction.

**REFERENCES**

[1] J. Lei, P. Yi, J. Chen, and Y. Hong, “A communication-efficient linearly convergent algorithm with variance reduction for distributed stochastic optimization,” in Proc. IEEE Eur. Control Conf. 2020, pp. 1250–1255.

[2] P. Yi, Y. Hong, and F. Liu, “ Initialization-free distributed algorithms for optimal resource allocation with feasibility constraints and application to economic dispatch of power systems,” *Automatica*, vol. 74, pp. 259–269, 2016.

[3] Q. Liu, X. Le, and K. Li, “A distributed optimization algorithm based on multiagent network for economic dispatch with region partitioning,” *IEEE Trans. Cybern.*, vol. 51, no. 5, pp. 2466–2475, May 2021.

[4] H. Fang, C. Shang, and J. Chen, “An optimization-based shared control framework with applications in multi-robot systems,” *Sci. China Inf. Sci.*, vol. 61, no. 1, 2018, Art. no. 014201.

[5] A. Nedić, A. Olshevsky, and C. A. Uribe, “Fast convergence rates for distributed non-Bayesian learning,” *IEEE Trans. Autom. Control*, vol. 62, no. 11, pp. 5538–5553, Nov. 2017.

[6] Y. Wang, P. Lin, and Y. Hong, “ Distributed regression estimation with incomplete data in multi-agent networks,” *Sci. China Inf. Sci.*, vol. 61, no. 9, 2018, Art. no. 092202.

[7] A. Nedić and A. Ozdaglar, “Distributed subgradient methods for multi-agent optimization,” *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.
[8] Y. Lou, L. Yu, S. Wang, and P. Yi, “Privacy preservation in distributed subgradient optimization algorithms,” IEEE Trans. Cybern., vol. 48, no. 7, pp. 2154–2165, Jul. 2018.

[9] W. Shi, Q. Ling, G. Wu, and W. Yin, “Extra: An exact first-order algorithm for decentralized consensus optimization,” SIAM J. Optim., vol. 25, no. 2, pp. 944–966, 2015.

[10] D. Jakovicet, J. Xavier, and J. M. Moura, “Fast distributed gradient methods,” IEEE Trans. Autom. Control, vol. 59, no. 5, pp. 1131–1146, May 2014.

[11] A. Nedić, A. Olshevsky, and W. Shi, “Achieving geometric convergence for distributed optimization over time-varying graphs,” SIAM J. Optim., vol. 27, no. 4, pp. 2597–2633, 2017.

[12] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, “Convergence of asynchronous distributed gradient methods over stochastic networks,” IEEE Trans. Autom. Control, vol. 63, no. 2, pp. 434–448, Feb. 2018.

[13] D. P. Palomar and M. Chiang, “A tutorial on decomposition methods for network utility maximization,” IEEE J. Sel. Areas Commun., vol. 24, no. 8, pp. 1439–1451, Aug. 2006.

[14] J. F. Motu, J. M. Xavier, P. M. Aguiar, and M. Püschel, “D-ADMM: A communication-efficient distributed algorithm for separable optimization,” IEEE Trans. Signal Process., vol. 61, no. 10, pp. 2718–2723, May 2013.

[15] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, “On the linear convergence of the ADMM in decentralized consensus optimization,” IEEE Trans. Signal Process., vol. 62, no. 7, pp. 1750–1761, Apr. 2014.

[16] T.-H. Chang, A. Nedić, and A. Scaglione, “Distributed constrained optimization by consensus-based primal-dual perturbation method,” IEEE Trans. Autom. Control, vol. 59, no. 6, pp. 1524–1538, Jun. 2014.

[17] P. Yi, Y. Hong, and F. Liu, “Distributed gradient algorithm for constrained optimization with application to load sharing in power systems,” Syst. Control Lett., vol. 83, pp. 45–52, 2015.

[18] J. Lei, H.-F. Chen, and H.-T. Fang, “Primal-dual algorithm for distributed constrained optimization,” Syst. Control Lett., vol. 96, pp. 110–117, 2016.

[19] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, “DQDM: Decentralized quadratically approximated alternating direction method of multipliers,” IEEE Trans. Signal Process., vol. 64, no. 19, pp. 5158–5173, Oct. 2016.

[20] H. Wai, N. M. Freris, A. Nedić, and A. Scaglione, “SUCAG: Stochastic unbiased curvature-aided gradient method for distributed optimization,” in Proc. IEEE 57th Conf. Decis. Control, 2018, pp. 1751–1756.

[21] Y. Li, N. M. Freris, P. G. Voulgaris, and D. M. Stipanovic, “D-SOP: Distributed second order proximal method for convex composite optimization,” in Proc. IEEE Amer. Control Conf., 2020, pp. 2844–2849.

[22] T. Yang et al., “A survey of distributed optimization,” Ann. Rev. Control, vol. 47, pp. 278–305, 2019.

[23] G. Notarstefano, I. Notarnicola, and A. Camisa, “Distributed optimization for smart cyber-physical networks,” Found. Trends Syst. Control, vol. 7, no. 3, pp. 253–283, 2019.

[24] S. S. Ram, A. Nedić, and V. V. Veeravalli, “Distributed stochastic subgradient projection algorithms for convex optimization,” J. Optim. Theory Appl., vol. 147, no. 3, pp. 516–545, 2010.

[25] P. Bianchi and J. Jakubowicz, “Convergence of a multi-agent projected stochastic gradient algorithm for non-convex optimization,” IEEE Trans. Autom. Control, vol. 58, no. 2, pp. 391–405, Feb. 2013.

[26] A. Agarwal and J. C. Duchi, “Distributed delayed stochastic optimization,” in Adv. Neural Inf. Process. Syst., 2011, pp. 873–881.

[27] L. Bottou, F. E. Curtis, and J. Nocedal, “Optimization methods for large-scale machine learning,” SIAM Rev., vol. 60, no. 2, pp. 223–311, 2018.

[28] K. Srivastava and A. Nedić, “Distributed asynchronous constrained stochastic optimization,” IEEE J. Sel. Topics Signal Process., vol. 5, no. 4, pp. 772–790, Aug. 2011.

[29] J. Lei, H.-F. Chen, and H.-T. Fang, “Asymptotic properties of primal-dual algorithm for distributed stochastic optimization over random networks with imperfect communications,” SIAM J. Control Optim., vol. 56, no. 3, pp. 2159–2188, 2018.

[30] A. H. Sayed et al., “Adaptation, learning, and optimization over networks,” Found. Trends Mach. Learn., vol. 7, no. 4–5, pp. 311–801, 2014.

[31] A. Nedić and A. Olshevsky, “Stochastic gradient-push for strongly convex functions on time-varying directed graphs,” IEEE Trans. Autom. Control, vol. 61, no. 12, pp. 3936–3947, Dec. 2016.