Differentially Private Distributed Stochastic Optimization with Time-Varying Sample Sizes

Jimin Wang, *Member, IEEE*, and Ji-Feng Zhang, *Fellow, IEEE*

***Abstract*—Differentially private distributed stochastic optimization has become a hot topic due to the need for privacy protection in distributed stochastic optimization. In this article, two-time scale stochastic approximation-type algorithms for differentially private distributed stochastic optimization with time-varying sample sizes are proposed using gradient- and output-perturbation methods. For both gradient- and output-perturbation cases, the convergence of the algorithm and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations are simultaneously established, which is substantially different from the existing works. By a time-varying sample size method, the privacy level is enhanced, and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations is established. By properly choosing a Lyapunov function, the algorithm achieves almost sure and mean square convergence even when the added privacy noise has an increasing variance. Furthermore, we rigorously provide the mean square convergence rates of the algorithm and show how the added privacy noise affects the convergence rate of the algorithm. Finally, numerical examples, including distributed training on a benchmark machine learning dataset, are presented to demonstrate the efficiency and advantages of the algorithms.**

***Index Terms*—Convergence rate, differential privacy, distributed stochastic optimization, privacy-preserving, stochastic approximation.**

I. Introduction

In recent years, information and artificial intelligence technologies have been increasingly employed in emerging applications such as the Internet of Things, cloud-based control systems, smart buildings, and autonomous vehicles [1]. The ubiquitous employment of such technologies provides more ways for an adversary to access sensitive information in practical systems, for example, the locations of residence and work in traffic monitoring systems [2]; users' living habits and customs in the electric vehicle market [3]. As such, privacy has become a pivotal concern for modern control systems. So far, some privacy-preserving approaches have been recently proposed for control systems relying on homomorphic encryption [4], state decomposition [5], and adding artificial noise [6], [7]. Among others, differential privacy is a well-known privacy notion and provides strong privacy guarantees. Thanks to its powerful features, differential privacy has been widely used in stochastic optimization [8] and distributed consensus [9], [10].

Distributed (stochastic) optimization has been widely used in various fields, such as Big Data analytics, finance, and distributed learning [11], [12], [13], [15], [17]. At present, there are many important techniques to solve distributed stochastic optimization, such as stochastic approximation [12], [13], [14], [15] and time-varying sample-size. As a standard variance reduction technique, time-varying sample-size schemes have gained increasing research interests and have been widely used to solve various problems, such as large-scale machine learning [16], stochastic optimization [17], and stochastic generalized equations [18]. In the class of time-varying 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. In distributed stochastic optimization, sensitive personal information is frequently embedded in each agent's sampled gradient. The main reason is that the sampled gradient contains agent-specific data as input, and such data are often private in nature. For example, in smart grid applications, the power consumption data, contained in the sampled gradient, of each household should be protected from being revealed to others because it can demonstrate information regarding the householders (e.g., their activities and even their health conditions such as whether they are disabled or not). In machine learning applications, sampled gradients are directly calculated from and embed the information of sensitive training data. Hence, information regarding the sampled gradient is considered to be sensitive and should be protected from being revealed in the process of solving the distributed stochastic optimization problem.

Privacy-preserving distributed (stochastic) optimization method has recently been studied, including the inherent privacy protection method [19], quantization-enabled privacy protection method [20], and differential privacy method [21], [22], [23], [24], [25], [26], [27]. An important result that the convergence and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations hold simultaneously has been given for distributed optimization in [21], but this cannot be directly used for distributed stochastic optimization. Based on the gradient-perturbation mechanism [19] or a stochastic ternary quantization scheme [20], the privacy protection distributed stochastic optimization algorithm with only one iteration was proposed, respectively. Two common methods have been proposed for differential privacy distributed stochastic optimization, namely, gradient-perturbation [22], [23], [24], [25] and output-perturbation [22], [26], [27]. However, the existing method induces a tradeoff between privacy and accuracy. For the gradient-perturbation case, the mean square convergence of the proposed algorithm cannot be guaranteed, although a finite cumulative privacy budget ε for an infinite number of iterations

Manuscript received 20 October 2023; accepted 15 March 2024. Date of publication 20 March 2024; date of current version 29 August 2024. This work was supported in part by the National Key R&D Program of China under Grant 2018YFA0703800, and in part by the National Natural Science Foundation of China under Grant 62203045 and Grant T2293770. Recommended by Associate Editor Y. Shi. *(Corresponding author: Ji-Feng Zhang.)*

Jimin Wang is with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China, and also with the Key Laboratory of Knowledge Automation for Industrial Processes, Ministry of Education, University of Science and Technology Beijing, Beijing 100083, China (e-mail: jimwang@ustb.edu.cn).

Ji-Feng Zhang is with the Key Laboratory of Systems and Control, Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: jif@iss.ac.cn).

Digital Object Identifier 10.1109/TAC.2024.3379387

has been presented in [23], [24], and [25]. For the output-perturbation case, to guarantee the accuracy of the algorithm, ε-differential privacy was proven only for one iteration, leading to the cumulative privacy loss of $k\varepsilon$ after k iterations [22], [26], [27]. To the best of our knowledge, the convergence of the algorithm and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations has not been simultaneously established for distributed stochastic optimization. This observation naturally motivates the following interesting questions. 1) How to design the differentially private distributed stochastic optimization algorithm such that the algorithm protects each agent's sensitive information with a finite cumulative privacy budget ε and simultaneously guarantees convergence? 2) How does the added privacy noise affect the convergence rate of the algorithm? The current article mainly aims to answer these two questions.

Two differentially private distributed stochastic optimization algorithms with time-varying sample sizes are proposed in this article. Both the gradient- and output-perturbation methods are given. The main contributions of this article are summarized as follows.

1) A differentially private distributed stochastic optimization algorithm with time-varying sample sizes is presented for both output- and gradient-perturbation cases. By a time-varying sample sizes method, the convergence of the algorithm and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations can be simultaneously established even when the added privacy noises have an increasing variance. Compared with [22], [23], and [24], the mean square and almost sure convergence of the algorithm can be guaranteed for both gradient- and output-perturbation methods. Compared with [20], [22], [23], [24], [25], [26], and [27], a finite cumulative privacy budget ε for an infinite number of iterations is proven for both gradient- and output-perturbation methods.
2) The mean square convergence rate of the algorithm with a two-time scale stochastic approximation-type step size is provided by properly selecting a Lyapunov function. Compared with the existing privacy-preserving distributed stochastic optimization algorithms [19], [20], we present the mean square convergence rate of the algorithm. Furthermore, compared with [12] and [15], we give the convergence rate with more general noises.

The rest of this article is organized as follows. Section II introduces the problem formulation. In Sections III and IV, the privacy and convergence analyses for differentially private distributed stochastic optimization with time-varying sample sizes are presented for both output- and gradient-perturbation cases. Section V provides examples on distributed parameter estimation problems, and distributed training over "MNIST" datasets. Finally, Section VI concludes this article.

Notations: Some standard notations are used throughout this article. $X \geq 0$ $(X > 0)$ means that the symmetric matrix X is semipositive definite (positive definite). $\mathbf{1}$ stands for the appropriate-dimensional column vector with all its elements equalling one. $\mathbb{R}^n$ and $\mathbb{R}^{m\times n}$ denote the n-dimensional Euclidean space and the set of all $m \times n$ real matrices, respectively. For any $w,\ v \in \mathbb{R}^n$, $\langle w, v\rangle$ denotes the standard inner product on $\mathbb{R}^n$. $\|x\|$ refers to the Euclidean norm of vector x. I, 0 are an identity matrix and a zero matrix with appropriate dimensions, respectively. For a differentiable function $f(\cdot)$, $\nabla f(w)$ denotes the gradient of $f(\cdot)$ at w. The expectation of a random variable X is represented by $\mathbb{E}[X]$. Given two real-valued functions $f(k)$ and $g(k)$ defined on $\mathbb{N}$ with $g(k)$ being strictly positive for sufficiently large k, denote $f(k) = O(g(k))$ if there exist $M > 0$ and $k_0 > 0$ such that $|f(k)| \leq Mg(k)$ for any $k \geq k_0$; $f(k) = o(g(k))$ if for any $\epsilon > 0$ there exists k_0 such that $|f(k)| \leq \epsilon g(k)$ for any $k > k_0$. $\lceil x \rceil$ denotes the smallest integer greater than x for $x \in \mathbb{R}$.

II. Problem Formulation

A. Distributed Stochastic Optimization

Consider the following optimization problems defined over a network, which needs to be distributedly solved by n agents:

$$\min_{x\in\mathbb{R}^d} \quad f(x) = \sum_{i=1}^{n} f_i(x), \quad f_i(x) \triangleq \mathbb{E}_{\xi_i\sim\mathcal{D}_i}[\ell_i(x,\xi_i)] \tag{1}$$

where x is common for any $i \in \mathcal{V}$, but ℓ_i is a local cost function private to Agent i, and ξ_i is a random variable. $\mathcal{D}_i$ is the local distribution of the random variable ξ_i, which usually denotes a data sample in machine learning. The following assumptions are presented to ensure the well-posedness of (1).

Assumption 1: For any $i \in \mathcal{V}$, each function ∇f_i is Lipschitz continuous, i.e., there exists $L_i > 0$ such that $\|\nabla f_i(x) - \nabla f_i(y)\| \leq L_i\|x - y\| \quad \forall x, y \in \mathbb{R}^d$. each function f_i is μ-strongly convex if and only if f_i satisfies $\langle \nabla f_i(x) - \nabla f_i(y), x - y\rangle \geq \mu\|x - y\|^2 \quad \forall x, y \in \mathbb{R}^d$.

B. Distributed Subgradient Methods

Distributed subgradient methods for solving the distributed (stochastic) optimization problem were first studied and rigorously analyzed by [11]. In these algorithms, each agent i iteratively updates its decision variables x_i by combining an average of the states of its neighbors with a gradient step as follows: $x_{i,k+1} = \sum_{j\in\mathcal{N}_i} a_{ij}x_{j,k} - \alpha_k g_i(x_{i,k})$, where α_k is the time-varying step size corresponding to the influence of the gradients on the state update rule at each time step. Considering the randomness in $\ell_i(x, \xi_i)$, the gradient $g_i(x_{i,k})$ that can be obtained by each agent i is subject to noises. To reduce the variance of the gradient observation noise, the time-varying sample sizes are used in [17]. In this case, the gradient that Agent i has for optimization at iteration k is denoted as $\frac{1}{\gamma_k}\sum_{l=1}^{\gamma_k} g_i(x_{i,k}, \xi_i^l)$, and γ_k is the number of the sampling gradients used at time k, and $\xi_i^l, l = 1, \ldots, \gamma_k$ represents the realizations of ξ_i. For the sake of notational simplicity, $\frac{1}{\gamma_k}\sum_{l=1}^{\gamma_k} g_i(x_{i,k}, \xi_i^l)$ is abbreviated as g_i^k. In this article, the following standard assumption was made about $g_i(x_{i,k}, \xi_i^l)$.

Assumption 2: For any fixed l and $x_{i,k} \in \mathbb{R}^d$, there exists a positive constant σ_g such that $g_i(x_{i,k}, \xi_i^l)$ satisfies $\mathbb{E}[g_i(x_{i,k}, \xi_i^l)] = \nabla f_i(x_{i,k})$ and $\mathbb{E}[\|g_i(x_{i,k}, \xi_i^l) - \nabla f_i(x_{i,k})\|^2] \leq \sigma_g^2$.

The communication topology $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a nonempty agent set $\mathcal{V} = \{1, 2, \ldots, n\}$ and an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. $A = [a_{ij}]$ is the adjacency matrix of $\mathcal{G}$, where $a_{ii} > 0$ and $a_{ij} > 0$ if $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$, otherwise. $\mathcal{N}_i = \{j \in \mathcal{V}, (j, i) \in \mathcal{E}\}$ denotes the neighborhood of Agent i including itself. $\mathcal{G}$ is called connected if, for any pair of agents (i_1, i_l), there exists a path from i_1 to i_l consisting of edges $(i_1, i_2), (i_2, i_3), \ldots, (i_{l-1}, i_l)$.

Assumption 3: The undirected communication topology $\mathcal{G}$ is connected, and the adjacency matrix A satisfies the following conditions. 1) There exists a positive constant η such that $a_{ij} > \eta$ for $j \in \mathcal{N}_i$, $a_{ij} = 0$ for $j \notin \mathcal{N}_i$. 2) A is doubly stochastic, namely, $\mathbf{1}^T A = \mathbf{1}^T$, $A\mathbf{1} = \mathbf{1}$.

It is considered that the following passive attackers exist in distributed stochastic optimization that have been widely used in the existing works [19], [20].

1) *Semihonest agents* are assumed to follow the specified protocol and perform the correct computations. However, they may collect all intermediate and input/output information in an attempt to learn sensitive information about the other agents.
2) *External eavesdroppers* are adversaries who steal information through wiretapping all communication channels and intercepting exchanged information between agents.

Due to the information exchange in the abovementioned algorithm, the potential passive attackers can always collect $x_{i,k}$ at each time k. Meanwhile, the attackers know the topology graph (A) and step-size (α_k). Combining all the information, it is easy for the potential passive attackers to infer the agents' sampled gradients. In this case, raw data directly computes the sampled gradients, further leaking the agents' sensitive information. Therefore, in this article, privacy is defined as preventing agents' sampled gradients from being inferable by potential passive attackers.

C. Differential Privacy

This section presents some preliminaries of differential privacy. In distributed stochastic optimization algorithms, preserving differential privacy is equivalent to hiding changes in the samples of the gradient information. Changes in the samples of the gradient information can be formally defined by a symmetric binary relation between two datasets called the adjacency relation. Inspired by [8], the following definition is given.

Definition 1: (Adjacent relation): Given a positive constant C, two different samples of the gradients $D_k = \{g_i(x_{i,k}, \xi_i^l), l = 1, 2, \cdots\}$, $D'_k = \{g_i(x_{i,k}, \xi_i^{l'}), l' = 1, 2, \cdots\}$ are said to be adjacent if they differ in exactly one data sample l_0, l'_0 such that $\|g_i(x_{i,k}, \xi_i^{l_0}) - g_i(x_{i,k}, \xi_i^{l'_0})\|_1 \le C$.

Remark 1: The adjacent relation indicates the specific sensitive information that needs to be protected in this article. From Definition 1, it follows that D_k and D'_k are adjacent if only one data sample l_0, l'_0 satisfies $\|g_i(x_{i,k}, \xi_i^{l_0}) - g_i(x_{i,k}, \xi_i^{l'_0})\|_1 \le C$ and the others satisfy $\|g_i(x_{i,k}, \xi_i^{l}) - g_i(x_{i,k}, \xi_i^{l'})\|_1 = 0$.

Definition 2 ([2] (Differential privacy)): Given $\varepsilon \ge 0$, a randomized algorithm $\mathcal{A}$ is ε-differentially private at the kth iteration if for all adjacent D_k and D'_k, and for any subsets of outputs $\Upsilon \subseteq \mathrm{Range}(\mathcal{A})$ such that $\mathbb{P}\{\mathcal{A}(D_k) \in \Upsilon\} \le e^{\varepsilon}\mathbb{P}\{\mathcal{A}(D'_k) \in \Upsilon\}$.

Remark 2: The basic idea of differential privacy is to "perturb" the exact result before release. In this case, an adversary cannot tell from the output of D_k with a high probability that an agent's sensitive information has changed or not. The constant ε measures the privacy level of the randomized algorithm $\mathcal{A}$, i.e., a smaller ε implies a better privacy level.

Problem of interest: This article mainly seeks to develop two privacy-preserving distributed stochastic optimization algorithms such that each agent's sensitive information can be protected to a greater extent, and the convergence of the algorithm is guaranteed simultaneously.

III. Differentially Private Distributed Stochastic Optimization via Output-Perturbation

In this subsection, a differentially private distributed stochastic optimization algorithm with time-varying sample sizes is presented via output perturbation. Specifically, in each iteration of Algorithm 1, rather than its original state, each agent i sends its current noisy state $x_{i,k} + n_{i,k}$ to each of its neighbors $j \in \mathcal{N}_i$, where $x_{i,k}$ is the estimated state of Agent i at time k, $n_{i,k} \in \mathbb{R}^d$ is temporally and spatially independent, and each element is the zero-mean Laplace noise with a variance of $2\sigma_k^2$.

A. Privacy Analysis

This section demonstrates the ε-differential privacy of Algorithm 1. We first derive conditions on the noise variances under which Algorithm 1 satisfies ε-differential privacy for an infinite number of iterations. A critical quantity determines how much noise should be added to each iteration for achieving ε-differential privacy, referred to as sensitivity.

Algorithm 1: Differentially Private Distributed Stochastic Optimization With Time-Varying Sample Sizes via Output Perturbation.

Initialization: Set $k = 0$, $x_{i,0} \in \mathbb{R}^d$ is any arbitrary initial value for any $i \in \mathcal{V}$.

Iterate until convergence. Each agent $i \in \mathcal{V}$ updates its state as follows:

$$x_{i,k+1} = (1 - \beta_k)x_{i,k} + \beta_k \sum_{j \in \mathcal{N}_i} a_{ij}(x_{j,k} + n_{j,k}) - \alpha_k g_i^k \tag{2}$$

where $\alpha_k > 0$ is the step-size for the gradient step, a new step-size $0 < \beta_k < 1$ is introduced that determines the degree to which information from the neighbors should be weighed, and $n_{j,k}$ is the added privacy noises for Agent j at each time k.

Algorithm 2: Differentially Private Distributed Stochastic Optimization With Time-Varying Sample Sizes via Gradient Perturbation.

Initialization: Set $k = 0$, $x_{i,0} \in \mathbb{R}^d$ is any arbitrary initial value for any $i \in \mathcal{V}$.

Iterate until convergence. Each agent $i \in \mathcal{V}$ updates its state as follows:

$$x_{i,k+1} = (1 - \beta_k)x_{i,k} + \beta_k \sum_{j \in \mathcal{N}_i} a_{ij}x_{j,k} - \alpha_k(g_i^k + n_{i,k}).$$

Definition 3 ([3] (Sensitivity)): The sensitivity of an output map q at the kth iteration is defined as

$$\Delta_k = \sup_{D_k, D'_k : \mathrm{Adj}(D_k, D'_k)} \|q(D_k) - q(D'_k)\|_1.$$

Remark 3: The sensitivity of an output map q means that a single sampling gradient can change the magnitude of the output map q. It should be pointed out that q refers to $x_{i,k}$ for Algorithm 1, and g_i^k for Algorithm 2.

Lemma 1: The sensitivity of Algorithm 1 at the kth iteration satisfies

$$\Delta_k \le \begin{cases} \frac{C\alpha_0}{\gamma_0}, & k = 1 \\ \sum_{l=0}^{k-2} \prod_{t=l+1}^{k-1} (1 - \beta_t) \frac{C\alpha_l}{\gamma_l}, & k > 1. \end{cases} \tag{3}$$

Proof: The proof can be found in [30, Lemma 1]. □

Remark 4: Motivated by [22], the time-varying sample-size method is used to process multiple samples at the same iteration. Most importantly, the time-varying sample-size method has a great advantage in guaranteeing differential privacy for Algorithm 1. Observing the proof of Lemma 1, it is found that parameter $\frac{1}{\gamma_k}$ has reduced the sensitivity of Algorithm 1 and further enhances the privacy protection ability.

Theorem 1: Let C be any given positive number. If $\varepsilon = \sum_{k=1}^{\infty} \frac{\Delta_k}{\sigma_k}$, then Algorithm 1 is ε-differentially private for an infinite number of iterations.

Proof: The proof is similar to [9, Th. 3.5], and thus, is omitted here. □

Theorem 2: Let $\alpha_k = \frac{a_1}{(k+a_2)^{\alpha}}$, $\beta_k = \frac{a_1}{(k+a_2)^{\beta}}$, $\gamma_k = \lceil a_3(k + a_2)^{\gamma} \rceil$, and $\sigma_k = \underline{b}(k + a_2)^{\eta}$, $0 < \beta \le 1$, $0 < \alpha \le 1$, $\gamma \ge 0$, $\eta \ge 0$, $0 < a_1 < a_2^{\beta}, a_2, a_3, \underline{b} > 0$. If one of the following conditions holds:

1) $\beta = 1, \alpha + \gamma - a_1 < 1, \alpha + \gamma + \eta > 2$;
2) $\beta = 1, \alpha + \gamma - a_1 \ge 1, a_1 + \eta > 1$;

3) $0 < \beta < 1, \alpha + \gamma - \beta + \eta > 1$.

Then, Algorithm 1 is differentially private with a finite cumulative privacy budget ε for an infinite number of iterations.

Proof: We only need to prove that cumulative privacy budget ε is finite for all $k > 1$. When $\beta = 1$, note that $\alpha_k = \frac{a_1}{(k+a_2)^\alpha}, \beta_k = \frac{a_1}{k+a_2}$, $\gamma_k = \lceil a_3(k+a_2)^\gamma \rceil$, from (3), it follows that:

$$\Delta_k \leq \sum_{l=0}^{k-2} \prod_{t=l+1}^{k-1} \left(1 - \frac{a_1}{t+a_2}\right) \frac{Ca_1}{a_3(l+a_2)^{\alpha+\gamma}}, \quad k > 1.$$

For $k > 1$, from Lemma A.3, it follows that:

$$\Delta_k = \begin{cases} O\left((k+a_2)^{-\alpha-\gamma+1}\right), & \alpha+\gamma-a_1 < 1 \\ O\left((k+a_2)^{-a_1}\ln k\right), & \alpha+\gamma-a_1 = 1 \\ O\left((k+a_2)^{-a_1}\right), & \alpha+\gamma-a_1 > 1. \end{cases}$$

Furthermore, since $\sigma_k = \underline{b}(k+a_2)^\eta$, we have

$$\sum_{k=2}^{\infty} \frac{\Delta_k}{\sigma_k} = \begin{cases} O\left(\sum_{k=2}^{\infty}(k+a_2)^{-\alpha-\gamma-\eta+1}\right), & \alpha+\gamma-a_1 < 1 \\ O\left(\sum_{k=2}^{\infty}(k+a_2)^{-a_1-\eta}\ln k\right), & \alpha+\gamma-a_1 = 1 \\ O\left(\sum_{k=2}^{\infty}(k+a_2)^{-a_1-\eta}\right), & \alpha+\gamma-a_1 > 1. \end{cases}$$

From Lemma A.3, when $\alpha+\gamma-a_1 < 1, \alpha+\gamma+\eta > 2$ or $\alpha+\gamma-a_1 \geq 1, a_1+\eta > 1$, we have $\varepsilon = O(1)$.

When $0 < \beta < 1$, from (3), it follows that:

$$\Delta_k \leq \sum_{l=0}^{k-2} \prod_{t=l+1}^{k-1} \left(1 - \frac{a_1}{(t+a_2)^\beta}\right) \frac{Ca_1}{a_3(l+a_2)^{\alpha+\gamma}}, \quad k > 1.$$

By using Lemma A.2, we have

$$\Delta_k = O\left(\sum_{l=0}^{k-2} \exp\left(-\frac{a_1}{1-\beta}(k+a_2)^{1-\beta}\right) \right.$$
$$\left. \cdot \exp\left(\frac{a_1}{1-\beta}(l+a_2)^{1-\beta}\right)\frac{Ca_1}{a_3(l+a_2)^{\alpha+\gamma}}\right). \tag{4}$$

From (4) and Lemma A.1, it follows that:

$$\Delta_k = O\left(\exp\left(-\frac{a_1}{1-\beta}(k+a_2)^{1-\beta}\right)\right.$$
$$\left.\cdot\frac{1}{(k+a_2)^{\alpha+\gamma-\beta}}\exp\left(\frac{a_1}{1-\beta}(k+a_2)^{1-\beta}\right)\right)$$
$$= O\left((k+a_2)^{-\alpha-\gamma+\beta}\right).$$

Further, from Lemma A.3, it follows that when $0 < \beta < 1$, we have:

$$\sum_{k=2}^{\infty} \frac{\Delta_k}{\sigma_k} = O\left(\sum_{k=1}^{\infty}(k+a_2)^{-\alpha-\gamma+\beta-\eta}\right)$$
$$= \begin{cases} O\left((k+a_2)^{-\alpha-\gamma+\beta-\eta+1}\right), & \alpha+\gamma-\beta+\eta < 1 \\ O(\ln k), & \alpha+\gamma-\beta+\eta = 1 \\ O(1), & \alpha+\gamma-\beta+\eta > 1. \end{cases}$$

Based on the abovementioned discussion, when $\beta = 1, \alpha+\gamma-a_1 < 1, \alpha+\gamma+\eta > 2$, $\beta = 1, \alpha+\gamma-a_1 \geq 1, a_1+\eta > 1$, or $0 < \beta < 1, \alpha+\gamma-\beta+\eta > 1$ holds, cumulative privacy budget ε is finite for an infinite number of iterations. □

Remark 5: Theorem 2 gives a guidance for choosing α, β, γ, and η to achieve the differentially private with a finite cumulative privacy budget ε for an infinite number of iterations of Algorithm 1. ε-differential privacy was proven only for one iteration in [20], [22], and [26], leading to the cumulative privacy loss of $k\varepsilon$ after k iterations, and hence, the cumulative privacy budget growing to infinity with time. Therefore, ε for an infinite number of iterations is smaller in this article than the ones in [20], [22], and [26]. This implies that the algorithm achieves a better level of privacy protection than the ones therein.

B. Convergence Analysis

To facilitate convergence analysis of Algorithm 1, we define $x_k = [x_{1,k}, \ldots, x_{n,k}]^T$, $n_k = [n_{1,k}, \ldots, n_{n,k}]^T$, $G(x_k) = [(g_1^k), \ldots, (g_n^k)]^T$. Let $\overline{x}_k, \overline{n}_k \in \mathbb{R}^d$ be the average of $x_{i,k}$, $n_{i,k}$, respectively, i.e., $\overline{x}_k = \frac{1}{n}\sum_{i=1}^n x_{i,k} = \frac{1}{n}x_k^T\mathbf{1}$, $\overline{n}_k = \frac{1}{n}\sum_{i=1}^n n_{i,k}$. Additionally, we use the following notation $W = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T$, $U_k = \overline{x}_k - x^*$, $Y_k = x_k - \mathbf{1}\overline{x}_k^T = Wx_k$. Define σ-algebra $\mathcal{F}_k = \sigma\{G(x_t), n_t, 0 \leq t \leq k-1\}$. Then, the compact form of (2) can be rewritten as follows:

$$x_{k+1} = (1-\beta_k)x_k + \beta_k A(x_k + n_k) - \alpha_k G(x_k). \tag{5}$$

Since A is doubly stochastic, we have

$$\overline{x}_{k+1} = (1-\beta_k)\overline{x}_k + \beta_k(\overline{x}_k + \overline{n}_k) - \frac{\alpha_k}{n}\sum_{i=1}^n g_i^k. \tag{6}$$

Before discussing the convergence property of the algorithm, the following assumption is presented.

Assumption 4: The step sizes α_k, β_k, privacy noise parameters σ_k, and time-varying sample sizes γ_k satisfy the following conditions:

1) $\sup_k \frac{\alpha_k}{\beta_k} \leq \min\{\frac{2(1-\sigma_2)}{3\mu}, \frac{(1-\sigma_2)^2\mu^2\beta_0}{16(6L^2\alpha_0+n(1-\sigma_2)\mu\beta_0)(\beta_0+1)L^2}\}$ $\sum_{k=0}^\infty \frac{\alpha_k^2}{\beta_k} < \infty, \sum_{k=0}^\infty \beta_k^2\sigma_k^2 < \infty \sum_{k=0}^\infty \frac{\alpha_k^2}{\gamma_k\beta_k} < \infty, \sum_{k=0}^\infty \frac{\alpha_k^2}{\gamma_k} < \infty$.

Remark 6: Assumption 4 is satisfied for many kinds of step-sizes and noise parameters. For example, for sufficiently large a_2, Assumption 4 is satisfied in the form of $\alpha_k = (k+a_2)^{-1}$, $\beta_k = (k+a_2)^{-\beta}$, $\beta \in (1/2, 1)$, $\sigma_k = (k+a_2)^\eta, \eta < \beta - 1/2$, $\gamma_k = \lceil (k+a_2)^\gamma \rceil, \gamma \geq 0$. Especially, when σ_k and γ_k are constants, Assumption 4 becomes the commonly used two-time scale stochastic approximation step-size [12], [13].

Next, we provide the mean square and almost sure convergence of Algorithm 1.

Theorem 3: If Assumptions 1–4 hold, then Algorithm 1 converges in mean square and almost surely for any $i \in \mathcal{V}$, i.e., there exists an optimal solution x^* such that $\lim_{k\to\infty}\mathbb{E}[\|x_{i,k} - x^*\|^2] = 0$, and $\lim_{k\to\infty} x_{i,k} = x^*$, a.s. $\forall i \in \mathcal{V}$.

Proof: The proof can be found in [30, Th. 3]. □

Next, we show how the added privacy noise affects the convergence rate of the algorithm.

Theorem 4: If Assumptions 1–3 hold, and $\alpha_k = \frac{a_1}{(k+a_2)^\alpha}$, $\beta_k = \frac{a_1}{(k+a_2)^\beta}$,$\gamma_k = \lceil a_3(k+a_2)^\gamma \rceil$, and $\sigma_k = O((k+a_2)^\eta)$, $a_1, a_2, a_3 > 0$, $0 < \beta < \alpha \leq 1$, $0 \leq \gamma$, $0 \leq \eta \leq \frac{3\beta-2}{2}$, then the convergence rate of Algorithm 1 is given as follows. When $0 < \alpha < 1$, there holds $\mathbb{E}[\|x_{i,k} - x^*\|^2] = O(\frac{1}{(k+a_2)^{\min\{3\beta-2\alpha-2\eta,\alpha-\beta\}}})$. When $\alpha = 1$, there holds $\mathbb{E}[\|x_{i,k} - x^*\|^2] = O(\frac{\ln k}{(k+a_2)^{\min\{a_1\mu-1+\beta,3\beta-2\eta-2,1-\beta\}}})$, where μ is a positive constant in Assumption 1.

Proof: The proof can be found in [30, Th. 4]. □

Remark 7: Inspired by the linear two-time-scale stochastic approximation in [14], the almost sure and mean square convergence of the algorithm with $\sigma_k = O((k+a_2)^\eta)$, $0 \leq \eta \leq \frac{3\beta-2}{2}$, is studied by properly choosing a Lyapunov function. Based on this, the convergence rate of the algorithm is given in Theorem 4, and the related results are not provided for distributed stochastic optimization even when no privacy protection is considered. Note that the convergence rate for distributed optimization with nonvanishing noises is studied in [12] and [15], where $\sigma_k = O((k+a_2)^\eta)$, $\eta = 0$. Then, the convergence rate studied in this article is nontrivial and more general than the one in [12] and [15].

From Theorems 2 and 4, the mean square convergence of Algorithm 1 and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations can be simultaneously established, which will be shown in the following corollary.

Corollary 1: Let $\alpha_k = \frac{a_1}{(k+a_2)^\alpha}$, $\beta_k = \frac{a_1}{(k+a_2)^\beta}$, $\gamma_k = \lceil a_3(k+a_2)^\gamma \rceil$, and $\sigma_k = \underline{b}(k+a_2)^\eta, 0 < a_1 < a_2^\beta, a_2, a_3, \underline{b} > 0$. If $\alpha + \gamma - \beta + \eta > 1, 0 < \beta < \alpha \leq 1, 0 \leq \gamma, 0 \leq \eta \leq \frac{3\beta - 2}{2}$ hold, then the mean square convergence of Algorithm 1 and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations are established simultaneously.

Remark 8: Corollary 1 holds when the added privacy noises have an increasing variance. For example, when $\alpha = 1$, $\beta = 0.9$, $\gamma = 1.06$, $\eta = 0.35$, or $\alpha = 0.9$, $\beta = 0.8$, $\gamma = 1.8$, $\eta = 0.2$, the conditions of Corollary 1 hold. In this case, the mean square convergence of Algorithm 1 and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations can be established simultaneously. Note that ε-differential privacy is proven only for one iteration, leading to a cumulative privacy loss of $k\varepsilon$ after k iterations [20], [22], [24]. Then, Algorithm 1 is superior to the ones in [20], [22], and [24].

Remark 9: Our approach does not contradict the tradeoff between utility and privacy in the differential-privacy theory. In fact, to achieve differential privacy, our approach does incur a cost (compromise) on the utility. However, different from existing approaches that compromise convergence accuracy to enable differential privacy, our approach compromises the convergence rate (which is also a utility metric) instead. From Theorem 4, it follows that the convergence rate of the algorithm slows down with the increase of the privacy noise parameters. The ability to retain convergence accuracy makes our approach suitable for accuracy-critical scenarios.

IV. Differentially Private Distributed Stochastic Optimization via Gradient-Perturbation

This section presents a gradient perturbation method for privacy-preserving distributed stochastic optimization algorithms with time-varying sample sizes, i.e., Algorithm 2. Different from Algorithm 1, each agent i updates its state as follows: $x_{i,k+1} = (1-\beta_k)x_{i,k} + \beta_k \sum_{j \in \mathcal{N}_i} a_{ij}x_{j,k} - \alpha_k(g_i^k + n_{i,k})$, where $n_{i,k} \in \mathbb{R}^d$ is the added privacy noises for Agent i at each time k, and is temporally and spatially independent.

A. Privacy Analysis

In Algorithm 2, the privacy noise $n_{i,k}$ is added directly to the gradient. Then, the sensitivity of Algorithm 2 is $\Delta_k = \frac{1}{\gamma_k}\|g_i(x_{i,k}, \xi_i^{l_0}) - g_i(x_{i,k}, \xi_i^{l'_0})\|_1 \leq \frac{C}{\gamma_k}$.

Theorem 5: Let C be any given positive number. If $\varepsilon = \sum_{k=1}^{\infty} \frac{C}{\gamma_k \sigma_k}$, then Algorithm 2 is ε-differentially private for an infinite number of iterations. Furthermore, if $\sigma_k = \underline{b}(k+a_2)^\eta$, $\gamma_k = \lceil a_3(k+a_2)^\gamma \rceil$ with $\eta + \gamma > 1$, $a_2, a_3, \underline{b} > 0$, then Algorithm 2 is differentially private with a finite cumulative privacy budget ε for an infinite number of iterations.

Proof: The results can be obtained similar to the proof process of Theorem 1 with $\Delta_k \leq \frac{C}{\gamma_k}$, and differential privacy is robust to postprocessing as shown in [7, Proposition 2.1]. □

B. Convergence Analysis

For convergence analysis, we need the following assumptions about the step sizes α_k, β_k, privacy noise parameters σ_k, and time-varying sample sizes γ_k.

Assumption 5: The step sizes α_k, β_k, privacy noise parameters σ_k, and time-varying sample sizes γ_k satisfy the following conditions:

1) $\sup_k \frac{\alpha_k}{\beta_k} \leq \min\{\frac{2(1-\sigma_2)}{3\mu}, \frac{(1-\sigma_2)^2\mu^2\beta_0}{16(6L^2\alpha_0 + n(1-\sigma_2)\mu\beta_0)(\beta_0+1)L^2}\}$ $\sum_{k=0}^{\infty} \frac{\alpha_k^2}{\beta_k} < \infty, \sum_{k=0}^{\infty} \frac{\alpha_k^2}{\gamma_k \beta_k} < \infty, \sum_{k=0}^{\infty} \frac{\alpha_k^2 \sigma_k^2}{\beta_k} < \infty \sum_{k=0}^{\infty} \frac{\alpha_k^2}{\gamma_k} < \infty, \sum_{k=0}^{\infty} \alpha_k^2 \sigma_k^2 < \infty$.

Remark 10: For example, for sufficiently large a_2, Assumption 5 is satisfied in the form of $\alpha_k = (k+a_2)^{-1}$, $\beta_k = (k+a_2)^{-\beta}$, $\beta \in (1/2, 1)$, $\sigma_k = (k+a_2)^\eta, \eta < (1-\beta)/2, \gamma_k = \lceil (k+a_2)^\gamma \rceil, \gamma \geq 0$.

Next, we provide the mean square and almost sure convergence of Algorithm 2.

Theorem 6: If Assumptions 1–3 and 5 hold, then Algorithm 2 converges in mean square and almost surely for any $i \in \mathcal{V}$.

Proof: The proof can be found in [30, Th. 6]. □

Theorem 7: If Assumptions 1–3 hold, and $\alpha_k = \frac{a_1}{(k+a_2)^\alpha}$, $\beta_k = \frac{a_1}{(k+a_2)^\beta}$, $\gamma_k = \lceil a_3(k+a_2)^\gamma \rceil$ and $\sigma_k = O((k+a_2)^\eta)$, $a_1, a_2, a_3 > 0$, $0 < \beta < \alpha \leq 1$, $0 \leq \gamma$, $0 \leq \eta \leq \min\{\frac{\beta}{2}, \frac{\alpha-\beta}{2}\}$, then the convergence rate of Algorithm 2 is given as follows. When $0 < \alpha < 1$, there holds $\mathbb{E}[\|x_{i,k} - x^*\|^2] = O(\frac{1}{(k+a_2)^{\min\{\beta-2\eta, \alpha-\beta-2\eta\}}})$. When $\alpha = 1$, there holds $\mathbb{E}[\|x_{i,k} - x^*\|^2] = O(\frac{\ln k}{(k+a_2)^{\min\{a_1\mu-1+\beta, \beta-2\eta, 1-\beta-2\eta\}}})$, where μ is a positive constant in Assumption 1.

Proof: The proof can be found in [30, Th. 7]. □

Corollary 2: Let $\alpha_k = \frac{a_1}{(k+a_2)^\alpha}$, $\beta_k = \frac{a_1}{(k+a_2)^\beta}$, $\gamma_k = \lceil a_3(k+a_2)^\gamma \rceil$, and $\sigma_k = \underline{b}(k+a_2)^\eta, a_1, a_2, a_3, \underline{b} > 0$. If $\gamma + \eta > 1, 0 < \beta < \alpha \leq 1, 0 \leq \gamma, 0 \leq \eta \leq \min\{\frac{\beta}{2}, \frac{\alpha-\beta}{2}\}$ hold, then the mean square convergence of Algorithm 2 and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations are established simultaneously.

Remark 11: For example, when we choose $\alpha = 1$, $\beta = 0.6$, $\gamma = 1.1$, and $\eta = 0.1$, Corollary 2 holds. Note that the mean square convergence of the proposed algorithm cannot be guaranteed [23], [24]. Then, Algorithm 2 is superior to the ones in [23] and [24].

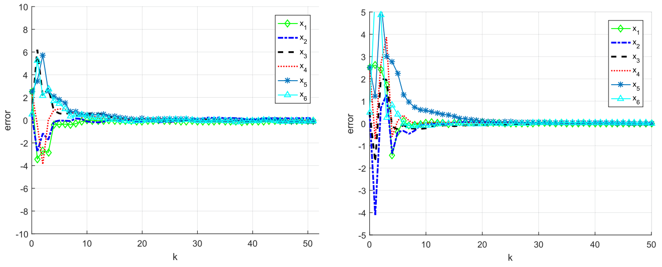
C. Oracle Complexity Analysis

Based on Theorem 7, we establish the oracle (sample) complexity for obtaining an ϵ-optimal solution satisfying $\mathbb{E}[\|x_{i,k} - x^*\|^2] \leq \epsilon$, where $\epsilon > 0$ is sufficiently small. The oracle complexity, measured by the total number of sampled gradients for deriving an ϵ-optimal solution, is $\sum_{k=0}^{K(\epsilon)} \gamma_k$, where $K(\epsilon) = \min_k\{k : \mathbb{E}[\|x_{i,k} - x^*\|^2] \leq \epsilon\}$.

Corollary 3: If Assumptions 1–3 hold, and $\alpha_k = \frac{a_1}{(k+1)^\alpha}$, $\beta_k = \frac{a_1}{(k+1)^\beta}$, $\gamma_k = \lceil a_3(k+1)^\gamma \rceil$, and $\sigma_k = \underline{b}(k+1)^\eta$, $0 < a_1 < 1$, $a_3, \underline{b} > 0$, $\beta = 0.5 + \epsilon$, $\alpha = 1 - \epsilon$, $\gamma = \epsilon$, $\eta = \epsilon$, then the oracle complexity of Algorithm 2 is $O(\epsilon^{-\frac{1+\epsilon}{0.5-4\epsilon}})$.

Proof: The proof can be found in [30, Corollary 3]. □

Remark 12: The increasing sample size schemes can generally be employed only when sampling is relatively cheap compared to the communication burden [17] or the main computational step, such as computing a projection or a prox [18]. As k becomes large, one might question how one deals with γ_k tending to $+\infty$. This issue does not arise in machine learning due to ϵ-optimal solution is interested; e.g., if $\epsilon = 10^{-3}$, then such a scheme requires approximately $O(10^6)$ samples in total from Corollary 3. Such requirements are not terribly onerous particularly since the computational cost of centralized stochastic gradient descent is $O(10^6)$ to achieve the same accuracy as our scheme. In addition, for the finite sample space, when the samples required by



(a) (b)

Fig. 1. Convergence. (a) Algorithm 1. (b) Algorithm 2.

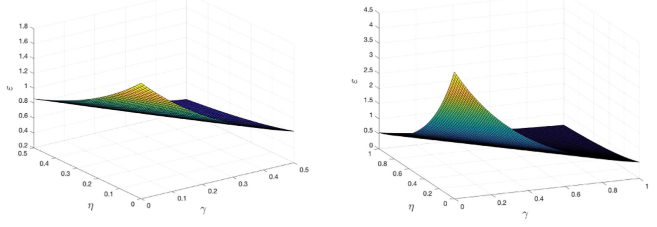
this scheme are larger than the total samples, the convergence can be guaranteed by setting the required samples equal to the total samples.

V. Example

This section shows the efficiency and advantages of Algorithms 1 and 2 on distributed parameter estimation problems and distributed training of a convolutional neural network over "MNIST" datasets.

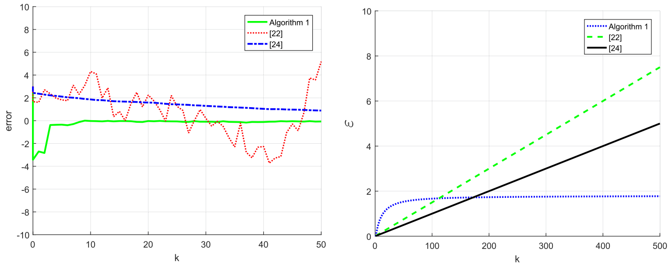
In distributed parameter estimation problems, we consider a network of n spatially distributed sensors that aim to estimate an unknown d-dimensional parameter x^*. Each sensor i collects a set of scalar measurements $d_{i,l}$ generated by the following linear regression model corrupted with noises, $d_{i,l} = u_{i,l}^T x^* + n_{i,l}$, where $u_{i,l} \in \mathbb{R}^d$ is the regression vector accessible to Agent i, and $n_{i,l} \in \mathbb{R}$ is a zero-mean Gaussian noise. Suppose that $u_{i,l}$ and $n_{i,l}$ are mutually independent Gaussian sequences with distributions $N(\mathbf{0}, R_{u,i})$ and $N(0, \sigma_{i,\nu}^2)$, respectively. Then, the distributed parameter estimation problem can be modeled as a distributed stochastic quadratic optimization problem, $\min \sum_{i=1}^n f_i(x)$, where $f_i(x) = \mathbb{E}[\|d_{i,l} - u_{i,l}^T x\|^2]$. Thus, $f_i(x) = (x - x^*)^T R_{u,i}(x - x^*) + \sigma_{i,v}^2$ is convex and $\nabla f_i(x) = R_{u,i}(x - x^*)$. By using the observed regressor $u_{i,l}$ and the corresponding measurement $d_{i,l}$, the sampled gradient $u_{i,l} u_{i,l}^T x - d_{i,l} u_{i,l}$ satisfies Assumption 2. Set the vector dimension $d = 6$ and the true parameter $x^* = \frac{\mathbf{1}}{2}$. Let $n = 6$; the adjacency matrix of the communication graph satisfies Assumption 3. In addition, the initial parameter estimates of these agents are chosen as $x_{i,0} = [3, 1, 1, 3, 3, 1]^T$, $i = 1, 2, 3, 4, 5, 6$. Let each covariance matrix $R_{u,i} = \begin{bmatrix} 2 & 1 & 0 & 1 & 0 & 0 \\ 1 & 2 & 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 1 & 1 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \end{bmatrix}$ be positive definite. Then, each $f_i(x)$ is strong convex.

First, we set $C = 0.2$, the step size $\alpha_k = 0.5/(k+1)^{0.9}$, $\beta_k = 0.5/(k+1)^{0.6}$, the sample size $\gamma_k = \lceil (k+1)^{1.1} \rceil$, and the privacy noise parameter $\sigma_k = (k+1)^{0.05}$. Then, the cumulative privacy budget for an infinite number of iterations is finite with $\varepsilon \approx 0.864$. The estimation error of Algorithm 1 is displayed in Fig. 1(a), showing that the generated iterations asymptotically converge to the true parameter x^*. Second, we set $C = 0.2$, the step size $\alpha_k = 0.5/(k+1)^{0.8}$ and $\beta_k = 0.5/(k+1)^{0.5}$, the sample size $\gamma_k = \lceil (k+1)^{1.2} \rceil$, and the privacy noise parameter $\sigma_k = (k+1)^{0.1}$. Then, the cumulative privacy budget for an infinite number of iterations is finite with $\varepsilon \approx 0.488$. The estimation error of Algorithm 2 is illustrated in Fig. 1(b), showing that the generated iterations asymptotically converge to the true parameter x^*. For both algorithms, we show the situation that ε is affected by η and γ in Fig. 2. As shown, ε decreases with the increase of η and γ, which is consistent with the theoretical analysis.



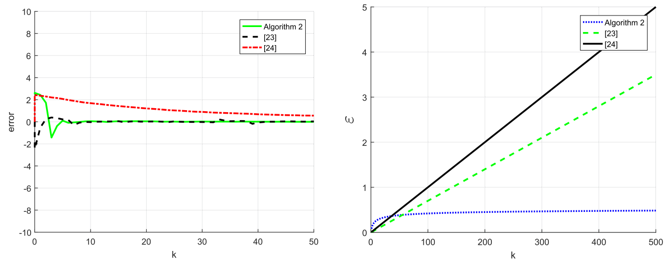
(a) (b)

Fig. 2. Relationship between ε, η, and γ. (a) Algorithm 1. (b) Algorithm 2.



(a) (b)

Fig. 3. Comparison between Algorithm 1 and the existing works. (a) Convergence accuracy. (b) Privacy level.



(a) (b)

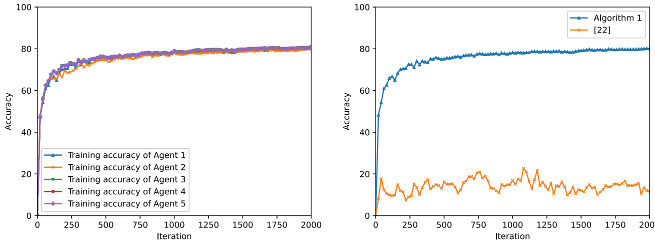
Fig. 4. Comparison between Algorithm 2 and the existing works. (a) Convergence accuracy. (b) Privacy level.

A. Comparison With the Existing Works

The comparison between Algorithm 1 and [22] and [24] is shown in Fig. 3; the comparison between Algorithm 2 and [23] and [24] is shown in Fig. 4, respectively. From Fig. 3, the mean square convergence of Algorithm 1 and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations are established simultaneously, but the algorithm in [22] and [24] cannot achieve the above results. From Fig. 4, the mean square convergence of Algorithm 2 and differential privacy with a finite cumulative privacy budget ε for an infinite number of iterations are established simultaneously, but the algorithm in [23] and [24] cannot achieve the above results. Based on the above discussions, Algorithms 1 and 2 achieve higher accuracy while keeping high-level privacy protection compared to [22], [23], and [24].

B. Distributed Training on a Benchmark Machine Learning Dataset

We evaluate the performance of Algorithm 1 through distributed training of a convolutional neural network (CNN) using the "MNIST" dataset. Specifically, five agents collaboratively train a CNN model on a communication graph, and the adjacency matrix satisfies Assumption 3.



(a)

(b)

Fig. 5. Training accuracy of Algorithm 1 using the “MNIST” dataset. (a) Training accuracy. (b) Comparison.

The “MNIST” dataset is uniformly divided into five pieces, each of which is sent to an agent. At each iteration, a time-varying batch of samples is drawn from each agent’s local dataset by the bootstrapping method. The CNN model has two convolutional layers, and each layer has 16 and 32 filters, respectively, followed by a max pooling layer. The Sigmoid function is used as the activation function, and hence Assumption 1 is satisfied. Then, the output is flattened and sent to a fully connected layer for ten classes. We set the noise parameters $\sigma_k = (k+2)^{0.01}$, step-sizes $\alpha_k = \frac{0.01}{(k+2)^{0.76}}$, $\beta_k = \frac{0.01}{(k+2)^{0.51}}$, and time-varying sample sizes $\gamma_k = \lceil (k+2)^{1.4} \rceil$. The validation accuracy of 5 agents after 2000 iterations is given in Fig. 5(a). Then, the comparison of Algorithm 1 and [22] is given in Fig. 5(b). To ensure the initial conditions, the same noise parameters and communication graph are used with the step-sizes $\alpha = 0.01$ and the batch size $B = 50$. From Fig. 5(b), it can be seen that the validation accuracy of Algorithm 1 is over 80% after 2000 iterations, but [22] cannot train the CNN model well.

VI. Conclusion

Two differentially private distributed stochastic optimization algorithms with time-varying sample sizes have been studied in this article. Both gradient- and output-perturbation methods are employed. By using two-time scale stochastic approximation-type conditions, the algorithm converges to the optimal point in an almost sure and mean square sense and is simultaneously differentially private with a finite cumulative privacy budget ε for an infinite number of iterations. Furthermore, it is shown how the added privacy noise affects the convergence rate of the algorithm. Finally, numerical examples, including distributed training over “MNIST” datasets, are provided to verify the efficiency of the algorithms. In the future, we will consider the privacy-preserving of other distributed stochastic optimization algorithms, including distributed alternating direction method of multipliers, distributed gradient tracking methods and distributed stochastic dual averaging.

Appendix A
Lemmas

Lemma A.1 ([10]): For any given c, $k_0 \geq 0$, $0 < p \leq 1$, and $q \in \mathbb{R}$, we have $\sum_{l=1}^{k} \frac{\exp(c(l+k_0)^p)}{(l+k_0)^q} = O(\frac{\exp(c(k+k_0)^p)}{(k+k_0)^{p+q-1}})$.

Lemma A.2: For $0 < \beta \leq 1$, $\alpha > 0$, $k_0 \geq 0$, sufficiently large l, we have

$$\prod_{i=l}^{k}\left(1 - \frac{\alpha}{(i+k_0)^\beta}\right) \leq \begin{cases} \left(\frac{l+k_0}{k+k_0}\right)^\alpha, & \beta = 1 \\ \exp\left(\frac{\alpha}{1-\beta}\left((l+k_0)^{1-\beta} - (k+k_0+1)^{1-\beta}\right)\right), & \beta \in (0,1). \end{cases} \tag{A.1}$$

If we further assume that $\rho > 0$, then for any $\gamma > 0$, we have

$$\prod_{i=l}^{k}\left(1 - \frac{\alpha}{i+k_0} + \frac{\gamma}{(i+k_0)^{1+\rho}}\right) = O\left(\left(\frac{l+k_0}{k+k_0}\right)^\alpha\right). \tag{A.2}$$

Proof: The proof can be found in [30, Lemma A.2]. □

Lemma A.3 ([28]): For the sequence h_k, assume that (i) h_k is positive and monotonically increasing; (ii) $\ln h_k = o(\ln k)$. Then, for real numbers a_1, a_2, χ, and any positive integer p

$$\sum_{l=1}^{k}\prod_{i=l+1}^{k}\left(1 - \frac{a_1}{i+a_2}\right)^p \frac{h_l}{l^{1+\chi}} = \begin{cases} O\left(\frac{1}{k^{pa_1}}\right), & pa_1 < \chi \\ O\left(\frac{h_k \ln k}{k^\chi}\right), & pa_1 = \chi \\ O\left(\frac{h_k}{k^\chi}\right), & pa_1 > \chi. \end{cases}$$

Lemma A.4 ([29]): Let V_k, u_k, β_k, ζ_k be nonnegative random variables. If $\sum_{k=0}^{\infty} u_k < \infty$, $\sum_{k=0}^{\infty} \beta_k < \infty$, and $\mathbb{E}[V_{k+1}|\mathcal{F}_k] \leq (1+u_k)V_k - \zeta_k + \beta_k$ for all $k \geq 0$, then V_k converges almost surely and $\sum_{k=0}^{\infty} \zeta_k < \infty$ almost surely. Here, $\mathbb{E}[V_{k+1}|\mathcal{F}_k]$ denotes the conditional mathematical expectation for the given $V_0, \ldots, V_k$, $u_0, \ldots, u_k$, $\beta_0, \ldots, \beta_k$, $\zeta_0, \ldots, \zeta_k$.

Lemma A.5: For a matrix $A \in \mathbb{R}^{n \times n}$ with eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n$ and corresponding nonzero mutually orthogonal eigenvectors $v_1, \ldots, v_n$. If a vector $u \in \mathbb{R}^n$ is orthogonal to $v_1, \ldots, v_{m-1}$ for some $m \leq n$, then $\|Au\| \leq \lambda_m \|u\|$.

Proof: The proof can be found in [30, Lemma A.5]. □

References

[1] J. F. Zhang, J. W. Tan, and J. M. Wang, “Privacy security in control systems,” *Sci. China Inf. Sci.*, vol. 64, pp. 176201:1–176201:3, 2021.

[2] J. L. Ny and G. J. Pappas, “Differentially private filtering,” *IEEE Trans. Autom. Control*, vol. 59, no. 2, pp. 341–354, Feb. 2014.

[3] S. Han, U. Topcu, and G. J. Pappas, “Differentially private distributed constrained optimization,” *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 50–64, Jan. 2017.

[4] Y. Lu and M. H. Zhu, “Privacy preserving distributed optimization using homomorphic encryption,” *Automatica*, vol. 96, pp. 314–325, 2018.

[5] Y. Q. Wang, “Privacy-preserving average consensus via state decomposition,” *IEEE Trans. Autom. Control*, vol. 64, no. 11, pp. 4711–4716, Nov. 2019.

[6] Y. L. Mo and R. M. Murray, “Privacy preserving average consensus,” *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 753–765, Feb. 2017.

[7] C. Dwork and A. Roth, “The algorithmic foundations of differential privacy,” *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3/4, pp. 211–407, 2014.

[8] R. Bassily, V. Feldman, and K. Talwar, “Private stochastic convex optimization with optimal rates,” in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 11282–11291.

[9] X. K. Liu, J. F. Zhang, and J. M. Wang, “Differentially private consensus algorithm for continuous-time heterogeneous multi-agent systems,” *Automatica*, vol. 12, 2020, Art. no. 109283.

[10] J. M. Wang, J. M. Ke, and J. F. Zhang, “Differentially private bipartite consensus over signed networks with time-varying noises,” *IEEE Trans. Autom. Control*, to be published, doi: 10.1109/TAC.2024.3351869.

[11] A. Nedic and A. Ozdaglar, “Distributed subgradient methods for multi-agent optimization,” *IEEE Trans. Autom. Control*, vol. 48, no. 1, pp. 48–61, Jan. 2009.

[12] T. T. Doan, S. T. Maguluri, and J. Romberg, “Convergence rates of distributed gradient methods under random quantization: A stochastic approximation approach,” *IEEE Trans. Autom. Control*, vol. 66, no. 10, pp. 4469–4484, Oct. 2021.

[13] T. T. Doan, S. T. Maguluri, and J. Romberg, “Fast convergence rates of distributed subgradient methods with adaptive quantization,” *IEEE Trans. Autom. Control*, vol. 66, no. 5, pp. 2191–2205, May 2021.

[14] T. T. Doan, “Finite-time analysis and restarting scheme for linear two-time-scale stochastic approximation,” *SIAM J. Control Optim.*, vol. 59, no. 4, pp. 2798–2819, 2021.

[15] H. Reisizadeh, B. Touri, and S. Mohajer, “Distributed optimization over time-varying graphs with imperfect sharing of information,” *IEEE Trans. Autom. Control*, vol. 68, no. 7, pp. 4420–4427, Jul. 2023.

[16] R. H. Byrd, G. M. Chin, J. Nocedal, and Y. Wu, "Sample size selection in optimization methods for machine learning," *Math. Program.*, vol. 134, pp. 127–155, 2012.

[17] J. L. Lei, P. Yi, J. Chen, and Y. G. Hong, "Distributed variable sample-size stochastic optimization with fixed step-sizes," *IEEE Trans. Autom. Control*, vol. 67, no. 10, pp. 5630–5637, Oct. 2022.

[18] S. Cui and U. V. Shanbhag, "Variance-reduced splitting schemes for monotone stochastic generalized equations," *IEEE Trans. Autom. Control*, vol. 68, no. 11, pp. 6636–6648, Nov. 2023.

[19] Y. Q. Wang and H. V. Poor, "Decentralized stochastic optimization with inherent privacy protection," *IEEE Trans. Autom. Control*, vol. 68, no. 4, pp. 2293–2308, Apr. 2023.

[20] Y. Q. Wang and T. Başar, "Quantization enabled privacy protection in decentralized stochastic optimization," *IEEE Trans. Autom. Control*, vol. 68, no. 7, pp. 4038–4052, Jul. 2023.

[21] Y. Q. Wang and A. Nedic, "Tailoring gradient methods for differentially-private distributed optimization," *IEEE Trans. Autom. Control*, vol. 69, no. 2, pp. 872–887, Feb. 2024.

[22] C. Li, P. Zhou, L. Xiong, Q. Wang, and T. Wang, "Differentially private distributed online learning," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 8, pp. 1440–1453, Aug. 2018.

[23] J. Xu, W. Zhang, and F. Wang, "A(DP)^{2}SGD: Asynchronous decentralized parallel stochastic gradient descent with differential privacy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 8036–8047, Nov. 2022.

[24] J. Ding, G. Liang, J. Bi, and M. Pan, "Differentially private and communication efficient collaborative learning," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 7219–7227.

[25] C. X. Liu, K. H. Johansson, and Y. Shi, "Private stochastic dual averaging for decentralized empirical risk minimization," *IFAC-PapersOnLine*, vol. 55, no. 13, pp. 43–48, 2022.

[26] Z. H. Huang, R. Hu, Y. X. Guo, E. Chan-Tin, and Y. N. Gong, "DP-ADMM: ADMM-based distributed learning with differential privacy," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 1002–1012, 2020.

[27] C. Gratton, N. K. D. Venkategowda, R. Arablouei, and S. Werner, "Privacy-preserved distributed learning with zeroth-order optimization," *IEEE Trans. Inf. Forensics Secur.*, vol. 17, pp. 265–279, 2022.

[28] J. M. Ke, Y. Wang, Y. L. Zhao, and J. F. Zhang, "Recursive identification of set-valued systems under uniform persistent excitations," 2023, *arXiv:2212.01777*.

[29] G. Goodwin and K. Sin, *Adaptive Filtering, Prediction and Control*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1984.

[30] J. M. Wang and J. F. Zhang, "Differentially private distributed stochastic optimization with time-varying sample sizes," 2023, *arXiv:2310.11892v1*.