Differentially Private Distributed Nonconvex Stochastic Optimization With Quantized Communication

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

***Abstract*—This article proposes a novel distributed nonconvex stochastic optimization algorithm that can achieve privacy protection and convergence simultaneously while improving communication efficiency. Specifically, each node adds general privacy noises to its local state to avoid information leakage, and then, quantizes its noise-perturbed state before transmitting to improve communication efficiency. By using a sampling parameter-controlled subsampling method, the proposed algorithm enhances the differential privacy level compared to the existing works. By using a new convergence analysis technique, the mean square convergence for nonconvex cost functions is given without assuming that gradients are bounded. Furthermore, when the nonconvex cost function satisfies the Polyak–Łojasiewicz condition, a convergence rate and the oracle complexity of the proposed algorithm are given. By using a two-time-scale step-sizes method and a probabilistic quantizer, the proposed algorithm achieves finite cumulative differential privacy budgets ϵ, δ and the mean square convergence simultaneously while improving communication efficiency as the sample-size goes to infinity. A numerical example of the distributed training on the "MNIST" dataset is given to show the effectiveness and advantages of the algorithm.**

***Index Terms*—Differential privacy, distributed stochastic optimization, probabilistic quantization.**

Received 23 April 2025; revised 30 June 2025; accepted 13 July 2025. Date of publication 22 July 2025; date of current version 30 December 2025. The work was supported by National Natural Science Foundation of China under Grant 62203045, Grant 62433020, and Grant T2293770. Recommended by Associate Editor Y. Wang. *(Corresponding author: Jimin Wang.)*

Jialong Chen is with the State Key Laboratory of Mathematical Sciences, 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: chenjialong23@mails.ucas.ac.cn).

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, Beijing 100083, China (e-mail: jimwang@ustb.edu.cn).

Ji-Feng Zhang is with the School of Automation and Electrical Engineering, Zhongyuan University of Technology, Zhengzhou 450007, China, and also with the State Key Laboratory of Mathematical Sciences, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China (e-mail: jif@iss.ac.cn).

Digital Object Identifier 10.1109/TAC.2025.3590872

I. Introduction

DISTRIBUTED optimization is gaining more and more attraction due to its fundamental role in cooperative control, smart grids, sensor networks, and large-scale machine learning [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]. As an important type of distributed optimization, distributed stochastic optimization has gained popularity due to its superior performance in handling stochastic cost functions [6], [7], [8], [9], [10]. So far, substantial efforts have been dedicated to the field of distributed stochastic optimization for both convex cost functions (e.g., [6] and [7]) and nonconvex cost functions (e.g., [8], [9], and [10]). In practice, nonconvex cost functions has wider applications than convex cost functions. For example, cost functions are often nonconvex in the training of recurrent neural networks [11] and the policy optimization of linear quadratic regulator [12]. It is worth mentioning that saddle points in nonconvex cost functions may cause sharp changes of gradients [11], [13], and thus, pose the difficulty in the convergence [7], [8], [9]. To prove the convergence, it usually requires the assumption that the gradients are bounded [10], which is hard to be satisfied or verified in many practical scenarios [11], [12].

When studying distributed stochastic optimization problems, there are two key issues worthy of attention. One is the network bandwidth limitation, and the other is the leakage of the sensitive information concerning cost functions. To solve the first issue, a common method is to transmit quantized information instead of the raw information. Generally, as a data compression technique to conserve network bandwidth, plenty of quantizers have been successfully applied to distributed stochastic optimization, such as the probabilistic quantizer [8], the cluster-aware sketch based quantizer [9], the uniform quantizer [14], the Lloyd-Max quantizer [15]. These works have made significant contributions to improving communication efficiency, neglecting the guarantee of convergence. Taking [14], [15] as examples, the amount of energy and bandwidth used for communication have been greatly reduced. However, the convergence cannot be guaranteed due to the biased quantization error [14], and the unbounded variance of the quantization error [15]. It is noteworthy that eliminating the effect of the quantization error on the convergence for distributed stochastic optimization is nontrivial. Fortunately, by a novel adaptive level quantizer, the convergence is achieved with requiring the increasing network bandwidth in [16]. This requirement restricts the applicability of this method. Recently, by using the probabilistic quantizer, Reisizadeh et al. [17] achieved the convergence and communication efficiency simultaneously.

1558-2523 © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

To solve the second issue, it needs to design privacy-preserving techniques to protect the sensitive information [18]. So far, various techniques have been employed to protect the sensitive information, such as homomorphic encryption [19], adding a constant uncertain parameter in step-sizes [20], state decomposition [21], adding deterministic perturbations in input and output [22], adding noises ([23], [24], [25], [26], [27], [28], [29]), and so on. Due to its simplicity in realization and immunity to postprocessing, differential privacy has attracted a lot of attention and been used to solve privacy issues in distributed stochastic optimization [30], [31], [32], [33], [34], [35], [36], [37], [38], [39]. One commonly used differential privacy in distributed stochastic optimization is (ϵ, δ)-differential privacy achieved by the Laplacian or (discrete) Gaussian noise [30], [31], [32], [33], [34], [35], [36], [37], the binomial-mechanism-aided quantizer [38]. Among others, the authors in [33], [35], and [38] achieved (ϵ, δ)-differential privacy while sacrificing the convergence accuracy, which is undesirable in accuracy-sensitive applications. To tackle this dilemma, some novel methods have been proposed in [30], [31], [32], [34], [36], and [37] under the assumption that gradients are bounded. For instance, by introducing a weakening factor to mitigate the impact of decaying privacy noises [30], [32] or constant privacy noises [31], [34], [36], the convergence and (ϵ, δ)-differential privacy are achieved simultaneously. By proposing an iteration maximum-based method, the convergence is achieved with enhanced differential privacy in [37]. Although the analysis is elegant in [30], [31], [32], [32], [33], [34], [35], [37], and [38], differential privacy budgets go to infinity over infinite iterations, and thus, the sensitive information therein cannot be protected over infinite iterations. By making interesting connections to the stochastic quantizer, $(0, \delta)$-differential privacy is proved in distributed stochastic optimization. A pioneering work in this direction is [39], where $(0, \delta)$-differential privacy is achieved by the ternary quantizer at each iteration. This implies that (0,1)-differential privacy is achieved over infinite iterations. Since (0,1)-differential privacy means the algorithm directly outputs the sensitive information, the sensitive information therein cannot be protected over infinite iterations.

Although some advancements have been made for considering network bandwidth limitation and privacy preserving simultaneously for distributed stochastic optimization [37], [38], [39], some open problems still persist. Yan et al. [38] provided new insights into the correlated nature of communication and privacy, but the convergence accuracy is sacrificed. The authors in [37] and [39] proposed a comprehensive solution that could simultaneously achieve privacy preserving, convergence, and improved communication efficiency under the assumption that gradients are bounded. Regarding the privacy preserving, the sensitive information in [37], [38], and [39] cannot be protected over infinite iterations.

Motivated by the aforementioned observations, the following questions may be raised: "how to design a privacy-preserving distributed nonconvex stochastic optimization algorithm that can enhance the differential privacy level while achieving convergence and improving communication efficiency simultaneously, especially avoiding the bounded gradients often occurred in existing results?" If there exists such an algorithm, then we are further concerned about "how do the added privacy noises affect the convergence rate of the algorithm?" In this article, we give analytical solutions to the above questions and propose a novel differentially private distributed nonconvex stochastic optimization algorithm with quantized communication. The main contributions are as follows.

1) A sampling parameter-controlled subsampling method is proposed to enhance the differential privacy level. By using this subsampling method, cumulative differential privacy budgets ϵ, δ are reduced with guaranteed mean square convergence for general privacy noises. Furthermore, finite cumulative differential privacy budgets ϵ and δ are achieved over infinite iterations.
2) In comparison to the existing results, the mean square convergence of the algorithm for nonconvex cost functions is achieved by removing the assumption that gradients are bounded. Furthermore, when the nonconvex cost function satisfies the Polyak–Łojasiewicz condition, a convergence rate of the algorithm for general privacy noises is provided, including decaying, constant and increasing privacy noises. This is nontrivial even without considering privacy protection problem.
3) A two-time-scale step-sizes method is employed to eliminate the effect of the quantization error on the convergence. By combining this method with a probabilistic quantizer, the mean square convergence of the algorithm is achieved while improving communication efficiency simultaneously. More interestingly, finite cumulative differential privacy budgets ϵ and δ over infinite iterations and the mean square convergence of the algorithm are achieved simultaneously while improving communication efficiency for the first time.
4) The effectiveness of our algorithm is evaluated by using the distributed training of a convolutional neural network (CNN) on the "MNIST" dataset. Our experimental results confirm that the proposed approach is superior to existing counterparts in terms of training/test accuracies, convergence rate, and differential privacy level.

The rest of this article is organized as follows. Section II formulates the problem to be investigated. Section III presents the main results including the privacy, convergence and oracle complexity analysis of the algorithm. Section IV provides a numerical example. Finally, Section V concludes this article.

Notations: $\mathbb{R}$ and $\mathbb{R}^r$ denote the set of all real numbers and r-dimensional Euclidean space, respectively. Range(F) denotes the range of a mapping F, and $F \circ G$ denotes the composition of mappings F and G. For sequences $\{a_k\}_{k=1}^{\infty}$ and $\{b_k\}_{k=1}^{\infty}$, $a_k = O(b_k)$ means there exists $A_1 \geq 0$ such that $\limsup_{k\to\infty} |\frac{a_k}{b_k}| \leq A_1$. $\mathbf{1}_n$ represents an n-dimensional vector whose elements are all 1. $A^\top$ stands for the transpose of the matrix A. We use the symbol $\|x\| = \sqrt{x^\top x}$ to denote the standard Euclidean norm of $x = [x_1, x_2, \ldots, x_m]^\top$, and $\|A\|$ to denote the 2-norm of the matrix A. $\mathbb{P}(\mathcal{B})$ and $\mathbb{E}(X)$ refer to the probability of an event $\mathcal{B}$ and the expectation of a random variable X, respectively. $\otimes$ denotes the Kronecker product of matrices. $\lfloor z \rfloor$ denotes the largest integer no larger than z. For a vector $v = [v_1, v_2, \ldots, v_n]^\top$, diag$(v)$ denotes the diagonal matrix with diagonal elements being $v_1, v_2, \ldots, v_n$. For a differentiable function $f(x)$, $\nabla f(x)$ denotes its gradient at the point x.

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

II. Preliminaries and Problem Formulation

A. Graph Theory

Consider a network of n nodes which exchange information on an undirected and connected communication graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$. $\mathcal{V}=\{1,2,\ldots,n\}$ is the set of all nodes, and $\mathcal{E}$ is the set of all edges. An edge $e_{ij}\in\mathcal{E}$ if and only if Node i can receive the information from j. Different nodes in $\mathcal{V}$ exchange information based on the weight matrix $\mathcal{A}=(a_{ij})_{1\le i,j\le n}$, whose entry a_{ij} is either positive if $e_{ij}\in\mathcal{E}$, or 0, otherwise. The neighbor set of Node i is defined as $\mathcal{N}_i=\{j\in\mathcal{V}:a_{ij}>0\}$, and the Laplacian matrix of $\mathcal{A}$ is defined as $\mathcal{L}=\mathrm{diag}(\mathcal{A}\mathbf{1}_n)-\mathcal{A}$. The assumption about the weight matrix $\mathcal{A}$ is given as follows.

Assumption 1: The weight matrix $\mathcal{A}$ is doubly stochastic, i.e., $\mathcal{A}\mathbf{1}_n=\mathbf{1}_n$, $\mathbf{1}_n^\top\mathcal{A}=\mathbf{1}_n^\top$.

Remark 1: Assumption 1 is standard and commonly used in undirected and connected communication graphs (see, e.g., [3], [4], [7], [23], [32], [33], [35], [36], [37], and [39]). There are many examples satisfying Assumption 1 in practice, such as, the dynamic load balancing of distributed memory processors [40], the distributed estimation of sensor networks [41] and the distributed machine learning [42].

B. Distributed Stochastic Optimization

In this article, the following distributed nonconvex stochastic optimization problem is considered:

$$\min_{x\in\mathbb{R}^r}F(x)=\min_{x\in\mathbb{R}^r}\frac{1}{n}\sum_{i=1}^{n}f_i(x),\ f_i(x)=\mathbb{E}_{\xi_i\sim\mathscr{D}_i}[\ell_i(x,\xi_i)] \quad (1)$$

where x is available to all nodes, $\ell_i(x,\xi_i)$ is a local cost function, which is private to Node i, and ξ_i is a random variable drawn from an unknown probability distribution $\mathscr{D}_i$. In practice, since the probability distribution $\mathscr{D}_i$ is difficult to obtain, it is replaced by the dataset $\mathcal{D}_i=\{\xi_{i,l}\in\mathbb{R}^p:l=1,\ldots,D\}$. Then, (1) is rewritten as the following empirical risk minimization problem:

$$\min_{x\in\mathbb{R}^r}F(x)=\min_{x\in\mathbb{R}^r}\frac{1}{n}\sum_{i=1}^{n}f_i(x),\ f_i(x)=\frac{1}{D}\sum_{l=1}^{D}\ell_i(x,\xi_{i,l}). \quad (2)$$

When solving the empirical risk minimization problem (2), a stochastic first-order oracle is often required [43], which returns a sampled gradient $\nabla\ell_i(x,\zeta_i)$ of $f_i(x)$ for any $i\in\mathcal{V},x\in\mathbb{R}^r$ and ζ_i uniformly sampled from $\mathcal{D}_i$. Then, the following standard assumption is given.

Assumption 2:

i) There exists $L_1,L_2>0$ such that for any $i\in\mathcal{V}$, $\ell_i(x,\zeta_i)$ is L_1- and L_2-smooth with respect to x and ζ_i, respectively, i.e., $\|\nabla\ell_i(x,\zeta_i)-\nabla\ell_i(y,\zeta_i)\|\le L_1\|x-y\|$, $\|\nabla\ell_i(x,\zeta_i)-\nabla\ell_i(x,\zeta_i')\|\le L_2\|\zeta_i-\zeta_i'\|$, $\forall x,y\in\mathbb{R}^r$, $\forall\zeta_i,\zeta_i'\in\mathbb{R}^p$.

ii) Each cost function is bounded from below, i.e., $\min_{x\in\mathbb{R}^r}f_i(x)=f_i^*>-\infty$.

iii) There exists $\sigma_\ell>0$ such that each sampled gradient $\nabla\ell_i(x,\zeta_i)$ satisfies $\mathbb{E}[\nabla\ell_i(x,\zeta_i)]=\nabla f_i(x)$, $\mathbb{E}[\|\nabla\ell_i(x,\zeta_i)-\nabla f_i(x)\|^2]\le\sigma_\ell^2$.

Remark 2: Assumption 2(i) is commonly used (see, e.g., [8], [11], [13], [30], [32], and [36]). Assumption 2(ii) ensures the existence of the optimal solution. Assumption 2(iii) requires that each sampled gradient $\nabla\ell_i(x,\zeta_i)$ is unbiased with a bounded variance σ_ℓ^2 (see, e.g., [6], [35], [37], [38], and [39]).

C. Quantized Communication

Due to the network bandwidth limitation, the exchange of the uncompressed information brings communication burden. To address this, the probabilistic quantizer is used to quantize the exchanged information in this article, which is a randomized mapping that maps an input to different values in a discrete set with some probability distribution, and satisfies the following assumption.

Assumption 3: The probabilistic quantizer $Q(x)$ is unbiased and its variance is bounded, which means there exists $\Delta>0$, such that $\mathbb{E}(Q(x)|x)=x$ and $\mathbb{E}(|Q(x)-x|^2|x)\le\Delta^2$.

Remark 3: Assumption 3 is standard and commonly used (see, e.g., [37]). Here, is an example: Given $\Delta>0$, the quantizer $Q(x)$ with the following probability distribution satisfies Assumption 3 by Lemma 1 of [44]:

$$\begin{cases}\mathbb{P}\left(Q(x)=\Delta\lfloor\frac{x}{\Delta}\rfloor\,\middle|\,x\right)=1-\frac{x}{\Delta}+\lfloor\frac{x}{\Delta}\rfloor\\ \mathbb{P}\left(Q(x)=\Delta\left(\lfloor\frac{x}{\Delta}\rfloor+1\right)\,\middle|\,x\right)=\frac{x}{\Delta}-\lfloor\frac{x}{\Delta}\rfloor.\end{cases} \quad (3)$$

D. Differential Privacy

As shown in [36] and [39], there are two kinds of adversary models widely used in the privacy-preserving issue for distributed stochastic optimization as follows.

1) *Semi-honest adversary:* This kind of adversary is defined as a node within the network, which has access to certain internal states (such as $x_{i,k}$ from Node i), follows the prescribed protocols and accurately computes iterative state correctly. However, it aims to infer the sensitive information of other nodes.
2) *Eavesdropper:* This kind of adversary refers to an external adversary who has capability to wiretap and monitor all communication channels, allowing them to capture distributed messages from any node. This enables the eavesdropper to infer the sensitive information of internal nodes.

When solving the empirical risk minimization problem (2), the stochastic first-order oracle needs data samples to return sampled gradients. Meanwhile, the adversaries above infer the sensitive information of data samples from sampled gradients [45]. Inspired by [27], [34], a symmetric binary relation called *adjacency relation* is defined as follows.

Definition 1 (Adjacency relation): Let $\mathcal{D}=\{\xi_{i,l}:i\in\mathcal{V},l=1,\ldots,D\}$, $\mathcal{D}'=\{\xi_{i,l}':i\in\mathcal{V},l=1,\ldots,D\}$ be two sets of data samples. If there exists $C>0$ and exactly one pair of data samples ξ_{i_0,l_0} and ξ_{i_0,l_0}' in $\mathcal{D}$ and $\mathcal{D}'$ such that for any $x\in\mathbb{R}^r$

$$\begin{cases}\|\nabla\ell_i(x,\xi_{i,l})-\nabla\ell_i(x,\xi_{i,l}')\|\le C, & \text{if } i=i_0 \text{ and } l=l_0\\ \|\nabla\ell_i(x,\xi_{i,l})-\nabla\ell_i(x,\xi_{i,l}')\|=0, & \text{if } i\ne i_0 \text{ or } l\ne l_0.\end{cases} \quad (4)$$

Then, $\mathcal{D}$ and $\mathcal{D}'$ are said to be adjacent, denoted by $\mathrm{Adj}(\mathcal{D},\mathcal{D}')$.

Remark 4: The constant C characterizes the "closeness" of a pair of data samples ξ_{i_0,l_0} and ξ_{i_0,l_0}'. By (4), the larger the constant C is, the larger the allowed magnitude of sampled gradients between adjacent datasets is, and thus, the better the privacy protection level is. Moreover, for any given constant C, as long as there exists a pair of sample sets $\mathcal{D}$ and $\mathcal{D}'$ satisfying the adjacency relation defined by this constant C, then the privacy analysis in Section III-B holds for $\mathrm{Adj}(\mathcal{D},\mathcal{D}')$.

Remark 5: Definition 1 allows us to avoid the assumption of bounded gradients required in [30], [31], [32], [33], [34],

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1: Differentially Private Distributed Nonconvex Stochastic Optimization Algorithm With Quantized Communication.

Initialization: $x_{i,0} \in \mathbb{R}^r$, weight matrix $(a_{ij})_{1\le i,j\le n}$, iteration maximum T, step-sizes $\alpha_T = \frac{a_1}{(T+1)^u}$, $\beta_T = \frac{a_2}{(T+1)^v}$ and sample-size $\gamma_T = \lfloor a_3 T^s \rfloor + 1$.

for $k = 0, \ldots, T$, **do**

1: Node i adds noise $d_{i,k}$ to $x_{i,k}$ and computes the quantized information $z_{i,k} = Q(x_{i,k} + d_{i,k}) = [Q(x_{i,k}^{(1)} + d_{i,k}^{(1)}), \ldots, Q(x_{i,k}^{(r)} + d_{i,k}^{(r)})]^\top$ with the probabilistic quantizer in the form of (3), where $d_{i,k} \sim N(0, \sigma_k^2 I_r)$.

2: Node i broadcasts $z_{i,k}$ to its neighbors $j \in \mathcal{N}_i$, receives $z_{j,k}$ from its neighbors $j \in \mathcal{N}_i$, and aggregates the received information by

$$\tilde{x}_{i,k} = (1-\beta_T)x_{i,k} + \beta_T \sum_{j\in\mathcal{N}_i} a_{ij} z_{j,k}. \tag{5}$$

3: Node i takes γ_T different data samples $\zeta_{i,k,1}, \ldots, \zeta_{i,k,\gamma_T}$ uniformly from $\mathcal{D}_i$ simultaneously (i.e., without replacement) to generate sampled gradients $\nabla\ell_i(x_{i,k}, \zeta_{i,k,1}), \ldots, \nabla\ell_i(x_{i,k}, \zeta_{i,k,\gamma_T})$. Then, Node i puts these data samples back into $\mathcal{D}_i$.

4: Node i computes the averaged sampled gradient by

$$\nabla\ell_{i,k} = \frac{1}{\gamma_T}\sum_{l=1}^{\gamma_T} \nabla\ell_i(x_{i,k}, \zeta_{i,k,l}). \tag{6}$$

5: Node i updates its state by

$$x_{i,k+1} = \tilde{x}_{i,k} - \alpha_T \nabla\ell_{i,k}. \tag{7}$$

end for

[35], [36], [37], and [39] to achieve differential privacy. Specifically, since $\mathcal{D}$ and $\mathcal{D}'$ have finite data samples, it follows that $\max_{\omega\in\mathcal{D}\cup\mathcal{D}'} \|\omega\| < \infty$. Then, for any $C \ge 2L_2 \max_{\omega\in\mathcal{D}\cup\mathcal{D}'} \|\omega\|$ and $x \in \mathbb{R}^r$, by Assumption 2(i), we have

$$\begin{cases} \|\nabla\ell_i(x,\xi_{i,l}) - \nabla\ell_i(x,\xi'_{i,l})\| \le L_2\|\xi_{i,l} - \xi'_{i,l}\| \\ \qquad \le 2L_2 \max_{\omega\in\mathcal{D}\cup\mathcal{D}'} \|\omega\| \le C, \\ \qquad\qquad\qquad\qquad \text{if } i = i_0 \text{ and } l = l_0 \\ \|\nabla\ell_i(x,\xi_{i,l}) - \nabla\ell_i(x,\xi'_{i,l})\| = 0, \qquad \text{if } i \ne i_0 \text{ or } l \ne l_0. \end{cases}$$

This shows (4) holds for any $x \in \mathbb{R}^r$. Thus, (4) holds no matter whether gradients are bounded or not.

To give the privacy-preserving level of the algorithm, we adopt the definition of (ϵ, δ)-differential privacy as follows.

Definition 2 ([34] (ϵ,δ)-differential privacy): Given $\epsilon > 0, 0 < \delta \le 1$, a mechanism $\mathcal{M}$ achieves (ϵ,δ)-differential privacy for $\mathrm{Adj}(\mathcal{D},\mathcal{D}')$ if $\mathbb{P}(\mathcal{M}(\mathcal{D}) \in \mathcal{T}) \le e^\epsilon \mathbb{P}(\mathcal{M}(\mathcal{D}') \in \mathcal{T}) + \delta$ for any Borel-measurable set $\mathcal{T} \subset \mathrm{Range}(\mathcal{M})$.

III. Main Result

A. Proposed Algorithm

In this section, we give a differentially private distributed nonconvex stochastic optimization algorithm with quantized communication. The detailed implementation steps are given in Algorithm 1.

Remark 6: By the subsampling method in Step 3 of Algorithm 1, there are sufficient data samples to run Algorithm 1 since data samples are put back into the dataset $\mathcal{D}_i$ at each iteration. Specially, when each node only has one data sample (i.e., $D = 1$), let $s = 0$, $a_3 = \frac{1}{2}$. Then, the sample-size $\gamma_T = 1 = D$. In this case, Algorithm 1 still works.

B. Privacy Analysis

In this section, we will show the differential privacy analysis of Algorithm 1. Inspired by [27], we first provide the sensitivity of the algorithm, which helps us to analyze the differential privacy of the algorithm.

Definition 3 (Sensitivity): Given $\mathrm{Adj}(\mathcal{D},\mathcal{D}')$, and a query q. For any $k = 0, \ldots, T$, let $\mathcal{D}_k = \{\zeta_{i,k,l} : i \in \mathcal{V}, l = 1, \ldots, \gamma_T\}$ and $\mathcal{D}'_k = \{\zeta'_{i,k,l} : i \in \mathcal{V}, l = 1, \ldots, \gamma_T\}$ be the data samples taken from $\mathcal{D}$ and $\mathcal{D}'$ at the kth iteration, respectively. Then, the sensitivity of Algorithm 1 at the kth iteration is defined as follows:

$$\Delta_k^q \triangleq \sup_{\mathcal{S}\in\mathbb{R}^{nr}} \sup_{y\in\mathcal{S}} \sup_{\mathrm{Adj}(\mathcal{D},\mathcal{D}')} \|q(\mathcal{D}_k|z_k = y) - q(\mathcal{D}'_k|z'_k = y)\|. \tag{8}$$

Remark 7: Definition 3 captures the magnitude by which one node's data sample can change the query q in the worst case. The sensitivity Δ_k^q is commonly used in [30], [31], [32], [37], [33], [35], [36], and [34], and determines how much noise should be added at the kth iteration to achieve (ϵ_k, δ_k)-differential privacy. In Algorithm 1, the query $q(\mathcal{D}_k|z_k{=}y)$ denotes the state x_{k+1} at the kth iteration under data samples $\mathcal{D}_k$ and the execution $z_k{=}[z_{1,k}^\top, \ldots, z_{n,k}^\top]^\top{=}[y_1^\top, \ldots, y_n^\top]^\top{=}y$, i.e., $q(\mathcal{D}_k|z_k{=}y) = x_{k+1}{=}[x_{1,k+1}^\top, \ldots, x_{n,k+1}^\top]^\top$. The mechanism $\mathcal{M}(\mathcal{D}_k)$ denotes the quantized noise-perturbed state at the kth iteration, i.e., $\mathcal{M}(\mathcal{D}_k){=}Q(q(\mathcal{D}_k|z_k{=}y){+}d_{k+1}){=}Q(x_{k+1}{+}d_{k+1}){=}[Q(x_{1,k+1}{+}d_{1,k+1})^\top, \ldots, Q(x_{n,k+1}{+}d_{n,k+1})^\top]^\top{=}z_{k+1}$.

The following lemma gives the sensitivity Δ_k of Algorithm 1 for any $k = 0, \ldots, T$.

Lemma 1: At the kth iteration, the sensitivity of Algorithm 1 satisfies $\Delta_k^q \le \frac{\alpha_T C}{\gamma_T}(\sum_{m=0}^k |1-\beta_T|^m)$.

Proof: When $k = 0$, (8) can be written as

$$\Delta_0^q = \sup_{\mathcal{S}\in\mathbb{R}^{nr}} \sup_{y\in\mathcal{S}} \sup_{\mathrm{Adj}(\mathcal{D},\mathcal{D}')} \|q(\mathcal{D}_0|z_0{=}y) - q(\mathcal{D}'_0|z'_0{=}y)\|$$
$$= \sup_{\mathcal{S}\in\mathbb{R}^{nr}} \sup_{y\in\mathcal{S}} \sup_{\mathrm{Adj}(\mathcal{D},\mathcal{D}')} \|x_1 - x'_1\|. \tag{9}$$

From (9), it can be seen that $z_{i,0}{=}y_i{=}z'_{i,0}$ holds for any $i{\in}\mathcal{V}$. Moreover, since $x_{i,0} = x'_{i,0}$ holds for any $i \in \mathcal{V}$, by (5), $\tilde{x}_{i,0} = \tilde{x}'_{i,0}$ holds for any $i \in \mathcal{V}$. Let $\nabla\ell_0 = [\nabla\ell_{1,0}^\top, \ldots, \nabla\ell_{n,0}^\top]^\top$. Then, substituting (7) into (9) implies

$$\Delta_0^q = \sup_{\mathcal{S}\in\mathbb{R}^{nr}} \sup_{y\in\mathcal{S}} \sup_{\mathrm{Adj}(\mathcal{D},\mathcal{D}')} \|\alpha_T(\nabla\ell_0 - \nabla\ell'_0)\|$$
$$= \sup_{\mathrm{Adj}(\mathcal{D},\mathcal{D}')} \|\alpha_T(\nabla\ell_0 - \nabla\ell'_0)\|. \tag{10}$$

By Definition 1, since $\mathcal{D}$ and $\mathcal{D}'$ are adjacent, there exists exactly one pair of data samples ξ_{i_0,l_0} and ξ'_{i_0,l_0} in $\mathcal{D}$ and $\mathcal{D}'$ such that (4) holds. This implies that $\nabla\ell_{j,0} = \nabla\ell'_{j,0}$ holds for any node $j \ne i_0$. Thus, (10) can be rewritten as

$$\Delta_0^q = \alpha_T \sup_{\mathrm{Adj}(\mathcal{D},\mathcal{D}')} \|\nabla\ell_{i_0,0} - \nabla\ell'_{i_0,0}\|. \tag{11}$$

Since γ_T different data samples are taken uniformly from $\mathcal{D}$ and $\mathcal{D}'$ simultaneously, there exists at most one pair of data samples $\zeta_{i_0,0,l_1}$ and $\zeta'_{i_0,0,l_1}$ such that $\zeta_{i_0,0,l_1} = \xi_{i_0,l_0}$ and $\zeta'_{i_0,0,l_1} = \xi'_{i_0,l_0}$. Thus, by (6), (11) can be rewritten

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

as

$$\Delta_0^q=\frac{\alpha_T}{\gamma_T}\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\left\|\sum_{l=1}^{\gamma_T}(\nabla\ell_{i_0}(x_{i_0,0},\zeta_{i_0,0,l})-\nabla\ell_{i_0}(x_{i_0,0},\zeta'_{i_0,0,l}))\right\|$$
$$=\frac{\alpha_T}{\gamma_T}\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\left\|\nabla\ell_{i_0}(x_{i_0,0},\zeta_{i_0,0,l_1})-\nabla\ell_{i_0}(x_{i_0,0},\zeta'_{i_0,0,l_1})\right\|$$
$$\le\frac{\alpha_T}{\gamma_T}\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\left\|\nabla\ell_{i_0}(x_{i_0,0},\xi_{i_0,l_0})-\nabla\ell_{i_0}(x_{i_0,0},\xi'_{i_0,l_0})\right\|$$
$$\le\frac{\alpha_T C}{\gamma_T}.$$

When $k=0,\dots,T$, by (8) we have

$$\Delta_k^q=\sup_{\mathcal{S}\in\mathbb{R}^{nr}}\sup_{y\in\mathcal{S}}\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\|q(\mathcal{D}_k|z_k=y)-q(\mathcal{D}'_k|z'_k=y)\|$$
$$=\sup_{\mathcal{S}\in\mathbb{R}^{nr}}\sup_{y\in\mathcal{S}}\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\left\|x_{k+1}-x'_{k+1}\right\|. \tag{12}$$

From (12), it can be seen that $z_{i,k}=z'_{i,k}$ holds for any $i\in\mathcal{V}$, $k=0,\dots,T$, and thus, $z_{i,T}=z'_{i,T}$ holds for any $i\in\mathcal{V}$. Moreover, note that $x_{i,0}=x'_{i,0}$ holds for any $i\in\mathcal{V}$ and $\nabla\ell_{j,m}=\nabla\ell'_{j,m}$ holds for any node $j\neq i_0$, $m=0,\dots,k$. Then, by (5), $\tilde{x}_{j,k}=\tilde{x}'_{j,k}$ holds for any node $j\neq i_0$. Thus, by (7), $x_{j,k+1}=x'_{j,k+1}$ holds for any node $j\neq i_0$. Hence, (12) can be rewritten as

$$\Delta_k^q=\sup_{\mathcal{S}\in\mathbb{R}^{nr}}\sup_{y\in\mathcal{S}}\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\|x_{i_0,k+1}-x'_{i_0,k+1}\|. \tag{13}$$

Note that $z_{i_0,k}=z'_{i_0,k}$. Then, substituting (5)–(7) into (13) implies

$$\Delta_k^q=\sup_{\mathcal{S}\in\mathbb{R}^{nr}}\sup_{y\in\mathcal{S}}\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\|(\tilde{x}_{i_0,k}-\tilde{x}'_{i_0,k})-\alpha_T(\nabla\ell_{i_0,k}-\nabla\ell'_{i_0,k})\|$$
$$=\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\|(\tilde{x}_{i_0,k}-\tilde{x}'_{i_0,k})-\alpha_T(\nabla\ell_{i_0,k}-\nabla\ell'_{i_0,k})\|$$
$$\le\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\|(1-\beta_T)(x_{i_0,k}-x'_{i_0,k})\|$$
$$+\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\|\frac{\alpha_T}{\gamma_T}\sum_{l=1}^{\gamma_T}(\nabla\ell_{i_0}(x_{i_0,k},\zeta_{i_0,k,l})-\nabla\ell_{i_0}(x_{i_0,k},\zeta'_{i_0,k,l}))\|. \tag{14}$$

Since $\mathcal{D}$ and $\mathcal{D}'$ are adjacent, there exists at most one pair of data samples $\zeta_{i_0,k,l_{k+1}}$ and $\zeta'_{i_0,k,l_{k+1}}$ such that $\zeta_{i_0,k,l_{k+1}}=\xi_{i_0,l_0}$, $\zeta'_{i_0,k,l_{k+1}}=\xi'_{i_0,l_0}$. Then, (14) can be rewritten as

$$\Delta_k^q\le\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\left\|(1-\beta_T)(x_{i_0,k}-x'_{i_0,k})\right\|$$
$$+\frac{\alpha_T}{\gamma_T}\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\left\|\nabla\ell_{i_0}(x_{i_0,k},\xi_{i_0,l_0})-\nabla\ell_{i_0}(x_{i_0,k},\xi'_{i_0,l_0})\right\|$$
$$\le|1-\beta_T|\sup_{\text{Adj}(\mathcal{D},\mathcal{D}')}\left\|x_{i_0,k}-x'_{i_0,k}\right\|+\frac{\alpha_T C}{\gamma_T}. \tag{15}$$

By iteratively computing (15), this lemma is proved. ■

Next, we show that Algorithm 1 achieves (ϵ_k,δ_k)-differential privacy at the kth iteration for any $k=0,\dots,T$.

Lemma 2: Given Adj$(\mathcal{D},\mathcal{D}')$, the query q and $\epsilon_k>0$, $0<\delta_k\le1$, $x_{k+1},x'_{k+1}\in\mathbb{R}^{nr}$ for any $k=0,\dots,T$. Then, for any Borel-measurable set $\mathcal{S}\subset\mathbb{R}^{nr}$, the mechanism $\mathcal{M}(\mathcal{D}_k)=Q(x_{k+1}+d_{k+1})$ satisfies

$$\mathbb{P}(\mathcal{M}(\mathcal{D}_k)\in\mathcal{S})\le e^{\epsilon_k}\mathbb{P}(\mathcal{M}(\mathcal{D}'_k)\in\mathcal{S})+\delta_k$$

where $d_{k+1}=[d_{1,k+1}^\top,\dots,d_{n,k+1}^\top]^\top\sim N(0,\sigma_{k+1}^2I_{nr})$ is a Gaussian noise with the variance $\sigma_{k+1}^2=4\ln(\frac{1.25}{\delta_k})(\frac{\Delta_k^q}{\epsilon_k})^2$.

Proof: Note that Gaussian noises d_{k+1}, d'_{k+1} have the variance $\sigma_{k+1}^2=4\ln(\frac{1.25}{\delta_k})(\frac{\Delta_k^q}{\epsilon_k})^2$ for any $k=0,\dots,T$. Then, for any Borel-measurable $\mathcal{O}\subset\mathbb{R}^{nr}$, by the Gaussian mechanism [25, Thm. A.1] we have

$$\mathbb{P}(x_{k+1}+d_{k+1}\in\mathcal{O})\le e^{\epsilon_k}\mathbb{P}(x'_{k+1}+d'_{k+1}\in\mathcal{O})+\delta_k. \tag{16}$$

Thus, for the Borel-measurable set $\mathcal{S}=\mathcal{M}(\mathcal{O})$, by (16) and the postprocessing property [25, Prop. 2.1] we have $\mathbb{P}(\mathcal{M}(\mathcal{D}_k)\in\mathcal{S})\le e^{\epsilon_k}\mathbb{P}(\mathcal{M}(\mathcal{D}'_k)\in\mathcal{S})+\delta_k$. ■

Lemma 3 ([25, Corollary B.2]): Given Adj$(\mathcal{D},\mathcal{D}')$, if the mechanism $\mathcal{M}(\mathcal{D}_k)$ achieves (ϵ_k,δ_k)-differential privacy for any $k=0,\dots,T$, then the mechanism $\mathcal{M}(\mathcal{D})=(\mathcal{M}(\mathcal{D}_0),\dots,\mathcal{M}(\mathcal{D}_T))$ achieves $(\sum_{k=0}^T\epsilon_k,\sum_{k=0}^T\delta_k)$-differential privacy.

Theorem 1: For any $T=0,1,\dots,k=0,\dots,T$, let

$$\alpha_T=\frac{a_1}{(T+1)^u},\ \beta_T=\frac{a_2}{(T+1)^v},\ \gamma_T=\lfloor a_3T^s\rfloor+1$$
$$\sigma_k=(k+1)^w,\delta_k=\frac{1}{(k+2)^t},\ a_1,\ a_2,\ a_3>0.$$

If $0<a_2<1$ and $t>0$, then Algorithm 1 achieves (ϵ,δ)-differential privacy over finite iterations T, where

$$\epsilon=\sum_{k=0}^T\epsilon_k\le\sum_{k=0}^T\frac{2Ca_1\sqrt{\ln(1.25(k+2)^t)}}{a_2(T+1)^{u-v}(\lfloor a_3T^s\rfloor+1)(k+2)^w}$$
$$\delta=\sum_{k=0}^T\delta_k=\sum_{k=0}^T\frac{1}{(k+2)^t}. \tag{17}$$

Furthermore, if $u+s-v>\max\{1-w,0\}$, $t\ge2$, then Algorithm 1 achieves finite cumulative differential privacy budgets ϵ, δ over infinite iterations.

Proof: By Lemma 2, the mechanism $\mathcal{M}(\mathcal{D}_k)$ achieves (ϵ_k,δ_k)-differential privacy with $\epsilon_k=2\sqrt{\ln(\frac{1.25}{\delta_k})}\frac{\Delta_k^q}{\sigma_{k+1}}$ for any $k=0,\dots,T$. Then, using Lemma 3, it can be seen that the mechanism $\mathcal{M}(\mathcal{D})$ achieves the $(\sum_{k=0}^T\epsilon_k,\sum_{k=0}^T\delta_k)$-differential privacy, i.e., $\mathbb{P}(\mathcal{M}(\mathcal{D})\in\mathcal{T})\le e^{\sum_{k=0}^T\epsilon_k}\mathbb{P}(\mathcal{M}(\mathcal{D}')\in\mathcal{T})+\sum_{k=0}^T\delta_k$ for any Borel-measurable set $\mathcal{T}\subset\text{Range}(\mathcal{M})=\mathbb{R}^{n(T+1)r}$.

By Lemma 1, the cumulative differential privacy budget $\sum_{k=0}^T\epsilon_k$ can be rewritten as

$$\sum_{k=0}^T\epsilon_k=\sum_{k=0}^T\frac{2\alpha_TC\sqrt{\ln\left(\frac{1.25}{\delta_k}\right)}\Delta_k^q}{\sigma_{k+1}}$$
$$\le\sum_{k=0}^T\frac{2\alpha_TC\sqrt{\ln\left(\frac{1.25}{\delta_k}\right)}\left(\sum_{m=0}^k|1-\beta_T|^m\right)}{\gamma_T\sigma_{k+1}}. \tag{18}$$

Since $0<a_2<1$, it can be seen that $0<\beta_T<1$. Then, we have $\sum_{m=0}^k|1-\beta_T|^m=\frac{1-(1-\beta_T)^{k+1}}{\beta_T}\le\frac{1}{\beta_T}$. Substituting it into (18) implies

$$\sum_{k=0}^T\epsilon_k\le\sum_{k=0}^T\frac{2\alpha_TC\sqrt{\ln\left(\frac{1.25}{\delta_k}\right)}}{\beta_T\gamma_T\sigma_{k+1}}$$

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

$$= \sum_{k=0}^{T} \frac{2Ca_1\sqrt{\ln(1.25(k+2)^t)}}{a_2(T+1)^{u-v}(\lfloor a_3T^s\rfloor+1)(k+2)^w}$$

$$\leq \sum_{k=0}^{T} \frac{2Ca_1\sqrt{\ln(1.25(T+2)^t)}}{a_2(a_3T^{u+s-v}+1)(k+2)^w}$$

$$= \frac{2Ca_1\sqrt{\ln(1.25(T+2)^t)}}{a_2(a_3T^{u+s-v}+1)} \sum_{k=0}^{T} \frac{1}{(k+2)^w}$$

$$= O\left(\frac{(\ln(T+2))^{\frac{3}{2}}}{(T+1)^{u+s-v-\max\{1-w,0\}}}\right).$$

Thus, if $u+s-v>\max\{1-w,0\}$, then the cumulative differential privacy budget $\sum_{k=0}^{T}\epsilon_k$ is finite even over infinite iterations. In addition, if $t\geq 2$, then the cumulative differential privacy budget $\sum_{k=0}^{T}\delta_k$ is finite even over infinite iterations. Hence, this theorem is proved. ■

Remark 8: By Theorem 1, (ϵ,δ)-differential privacy is achieved for all nodes. When $\mathcal{T}=\mathbb{R}^{(i_0-1)(T+1)r}\times\mathcal{S}\times\mathbb{R}^{(n-i_0)(T+1)r}$ for any Borel-measurable set $\mathcal{S}\in\mathbb{R}^{(T+1)r}$, we have

$$\begin{aligned}
&\mathbb{P}((z_{i_0,0},\ldots,z_{i_0,T})\in\mathcal{S})\\
&=\mathbb{P}((z_{1,0},\ldots,z_{1,T},\ldots,z_{i_0,0},\ldots,z_{i_0,T},\ldots,z_{n,0},\ldots,z_{n,T})\\
&\quad\in\mathbb{R}^{(i_0-1)(T+1)r}\times\mathcal{S}\times\mathbb{R}^{(n-i_0)(T+1)r})\\
&\leq e^{\epsilon}\mathbb{P}((z'_{1,0},\ldots,z'_{1,T},\ldots,z'_{i_0,0},\ldots,z'_{i_0,T},\ldots,z'_{n,0},\ldots,z'_{n,T})\\
&\quad\in\mathbb{R}^{(i_0-1)(T+1)r}\times\mathcal{S}\times\mathbb{R}^{(n-i_0)(T+1)r})+\delta\\
&=e^{\epsilon}\mathbb{P}((z'_{i_0,0},\ldots,z'_{i_0,T})\in\mathcal{S})+\delta.
\end{aligned}$$

This implies (ϵ,δ)-differential privacy is achieved for a particular node i_0. In this case, the sensitive information of all nodes can be protected against both the eavesdropper and the semi-honest adversary. Thus, Theorem 1 provides a unified privacy analysis framework for both adversary models presented in Section II-D.

Remark 9: Theorem 1 shows how step-size parameters u and v, the sample-size parameter s, and the privacy noise parameter w affect cumulative differential privacy budgets ϵ and δ. As shown in (17), the larger the step-size parameter u, the sample-size parameter s and the privacy noise parameter w are, the smaller cumulative differential privacy budgets ϵ and δ are. In addition, the smaller the step-size parameter v is, the smaller cumulative differential privacy budgets ϵ and δ are.

Remark 10: By (17), the larger the sample-size γ_T is, the smaller cumulative differential privacy budgets ϵ,δ are. Then, the larger the sample-size γ_T is, the less privacy noises are required to achieve the same (ϵ,δ)-differential privacy, and thus, the effect of privacy noises $d_{i,k}$ is reduced.

Remark 11: The sample-size γ_T is not required to go to infinity to achieve finite cumulative differential privacy budgets ϵ and δ over infinite iterations. Specifically, let the sample-size parameter $s=0$. Then, the sample-size γ_T is constant. In this case, if $u-v>\max\{1-w,0\}$, $t\geq 2$, then Algorithm 1 can achieve finite cumulative differential privacy budgets ϵ and δ over infinite iterations. This result shows advantage over [30], [31], [32], [33], [34], [35], [36], [37], [38] that only achieve infinite cumulative differential privacy budgets ϵ and δ over infinite iterations and [39] that only achieves $(0,\delta)$-differential privacy at each iteration.

C. Convergence Analysis

In this section, we will give the convergence analysis of Algorithm 1. First, we introduce an assumption on step-sizes, the sample-size and the privacy noise parameter.

Assumption 4: For any $T=0,1,\ldots,k=0,\ldots,T$, step-sizes $\alpha_T=\frac{a_1}{(T+1)^u}$, $\beta_T=\frac{a_2}{(T+1)^v}$, the sample-size $\gamma_T=\lfloor a_3T^s\rfloor+1$ and the privacy noise parameter $\sigma_k=(k+1)^w$ satisfy a_1, $a_3>0$, $0<a_2<1$, $2u-v>1$, $\frac{1}{2}+\max\{w,0\}<v<u<1$.

Next, we first provide the mean square convergence of Algorithm 1, and then show a convergence rate of Algorithm 1 for cost functions satisfying the Polyak–Łojasiewicz condition.

1) Mean Square Convergence: Since saddle points make finding an optimal solution of the problem (2) non-deterministic polynomial-time (NP)-hard [46], finding a first-order stationary point rather than an optimal solution is actually the main goal for distributed nonconvex stochastic optimization algorithms (see, e.g., [7], [9], [10], [15], [17], [35], [37], and [39]). Inspired by [39], $\mathbb{E}\|\nabla F(x_{i,T+1})\|^2$ is used as an index to show the mean square convergence of Algorithm 1.

Theorem 2: If Assumptions 1–4 hold, then

$$\liminf_{T\to\infty}\mathbb{E}\|\nabla F(x_{i,T+1})\|^2=0\quad\forall i\in\mathcal{V}.$$

Proof: See Appendix B. ■

Remark 12: In Theorem 2, by constructing an auxiliary variable $Y_k=\frac{1}{n}((I_n-\frac{\mathbf{1}_n\mathbf{1}_n^\top}{n})\otimes I_r)x_k$, the convergence of Algorithm 1 is achieved without assuming that gradients are bounded. This result shows advantage over [8] that does not provide a convergence analysis, [7], [9], [14], [15], [33], [35], [38] that cannot achieve the mean square convergence, and [10], [31], [32], [34], [36], [37], [39] that assume the gradients are bounded. Thus, this new convergence technique has wider applicability than those in [10], [14], [15], [31], [32], [33], [34], [35], [36], [37], [38], and [39].

2) Convergence Rate Analysis:

Assumption 5 (Polyak–Łojasiewicz): The global cost function $F(x)$ satisfies the Polyak–Łojasiewicz condition, i.e., there exists $\mu>0$ such that $2\mu(F(x)-F^*)\leq\|\nabla F(x)\|^2$ for any $x\in\mathbb{R}^r$, where F^* is the global minimum of the problem (2).

Remark 13: Assumption 5 is commonly used (see, e.g., [10]), and means that the gradient $\nabla F(x)$ to grow faster than a quadratic function as the algorithm moves away from the optimal solution. Such functions exist, for example, $F(x)=x^2+3\sin^2 x$ is a nonconvex function satisfying Assumption 5 for any $0<\mu<0.3$. As shown in [47, Thm. 2], Assumption 5 is more general than the convex cost functions assumed in [6], [30], [31], [32], [33], [34], and [37].

Theorem 3: If Assumptions 1–5 hold, then

$$\mathbb{E}\|\nabla F(x_{i,T+1})\|^{\psi}=O\left(\frac{\Delta^2}{(T+1)^{\frac{\psi}{2}\min\{2v-2\max\{w,0\}-1,2u-v-1\}}}\right)$$

for any $i\in\mathcal{V}$, $T=0,1,\ldots$, and $\psi\in[1,2]$. Particularly, when $\psi=2$, we have

$$\mathbb{E}(F(x_{i,T+1})-F^*)=O\left(\frac{\Delta^2}{(T+1)^{\min\{2v-2\max\{w,0\}-1,2u-v-1\}}}\right) \tag{19}$$

where the constant in the big-O notation does not depend on Δ. Further, the mean square convergence of Algorithm 1 is achieved as T goes to infinity, i.e., $\lim_{T\to\infty}\mathbb{E}\|\nabla F(x_{i,T+1})\|^2=0$, $\forall i\in\mathcal{V}$.

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

Proof: Due to the space limitation, we leave the proof to the extended version [48] of this paper available at arXiv:2403.18254v3. ■

Remark 14: To eliminate the effect of the quantization error on the convergence of Algorithm 1, a two-time-scale step-sizes method is used. The fast step-size α_T is used in the stochastic gradient descent (SGD), and the slow step-size β_T is used to eliminate the effect of the quantization error on convergence. By Assumption 4, the slow step-size β_T satisfies $\lim_{T\to\infty}\beta_T^2\Delta^2 = 0$, which ensures the mean square convergence of Algorithm 1. Compared with [14], [15], [38], the mean square convergence of Algorithm 1 is achieved while improving communication efficiency simultaneously. Meanwhile, the problem of increasing network bandwidth in [16] is solved. Moreover, (19) in Theorem 3 shows the effect of the quantization error on the convergence rate, which is not considered in [37] and [39]. The larger the quantization error Δ is, the slower the convergence rate is. Therefore, the probabilistic quantization does slow down the convergence rate of Algorithm 1.

Remark 15: The mean square convergence of Algorithm 1 is guaranteed for general privacy noises, including increasing, constant (see, e.g., [31], [33], [34], [35], [36], and [37]) and decaying (see, e.g., [30] and [32]) privacy noises. This is nontrivial even without considering privacy protection problem. For example, let $\alpha_T = \frac{1}{T^{0.9}}$ and $\beta_T = \frac{1}{T^{0.75}}$. Then, the convergence of Algorithm 1 holds as long as the variance σ_k of the privacy noise has an increasing rate no more than $O(k^{0.25})$.

Remark 16: Note that by Theorem 2, the mean square convergence of Algorithm 1 holds for general cost functions, including convex and nonconvex cost functions. Then, when the global cost function is convex, Theorem 2 also holds. Furthermore, if the global cost function $F(x)$ is λ-strongly convex, i.e., there exists $\lambda > 0$ such that for any $x, y \in \mathbb{R}^r$, $F(y) \geq F(x) + \langle \nabla F(x), y - x\rangle + \frac{\lambda}{2}\|y - x\|^2$, then by [43, Lemma 6.9] we have $2\lambda(F(x) - F^*) \leq \|\nabla F(x)\|^2$, which means the global cost function $F(x)$ satisfies Assumption 5. Thus, Algorithm 1 achieves the same convergence rate as Theorem 3.

Remark 17: Note that distributed nonconvex stochastic optimization algorithms may converge to a saddle point instead of the desired global minimum. Then, the discussion of the avoidance of saddle points is necessary. Assumption 5 implies that each stationary point x^* of F satisfying $\nabla F(x^*) = 0$ is a global minimum of F, and thus, guarantees the avoidance of saddle points discussed in [36]. Furthermore, compared with [36], Assumption 5 helps us to give a convergence rate of Algorithm 1.

In practice, the time and number of running a distributed stochastic optimization algorithm are usually limited by various constraints, while selecting the best one from lots of running results is very time-consuming. To address this issue, the following low-probability convergence rate of Algorithm 1 is given based on Theorem 3.

Corollary 1: Under Assumptions 1–5

$$F(x_{i,T+1}) - F^* = O\left(\frac{1}{(T+1)^{\min\{2v-2\max\{w,0\}-1,2u-v-1\}}}\right)$$

with probability at least $1-\delta^*$, for any $i\in\mathcal{V}$, $T=0,1,\ldots,$ and $\delta^* \in (0,1)$.

Proof: Due to the space limitation, we leave the proof to the extended version [48] of this paper available at arXiv:2403.18254v3.

Remark 18: Corollary 1 guarantees the convergence of a single running result with probability at least $1-\delta^*$, and thus, avoids spending time on selecting the best one from lots of running results. Moreover, from Theorem 1, it follows that the low-probability convergence rate is affected by the failure probability δ^*. The larger the failure probability δ^* is, the faster the low-probability convergence rate is.

D. Tradeoff Between Privacy and Utility

Based on Theorems 1–3, the mean square convergence of Algorithm 1 as well as the differential privacy with finite cumulative differential privacy budgets ϵ, δ over infinite iterations can be established simultaneously, which is given in the following corollary.

Corollary 2: For any $T = 0, 1, \ldots, k = 0, \ldots, T$, let

$$\alpha_T = \frac{a_1}{(T+1)^u}, \beta_T = \frac{a_2}{(T+1)^v}, \gamma_T = \lfloor a_3 T^s \rfloor + 1$$

$$\sigma_k = (k+1)^w, \delta_k = \frac{1}{(k+2)^t}, a_1, a_3 > 0, 0 < a_2 < 1.$$

If Assumptions 1–3 and 5 hold, and $t \geq 2$, $\frac{1}{2} + \max\{w, 0\} < v < u < 1$, $u+s-v > \max\{1-w, 0\}$, $2u-v > 1$, then Algorithm 1 achieves the mean square convergence and finite cumulative differential privacy budgets ϵ, δ over infinite iterations simultaneously as the sample-size γ_T goes to infinity.

Proof: By Theorems 1–3, this corollary is proved. ■

Remark 19: Corollary 2 holds even when privacy noises have increasing variances. For example, when $u=0.98, v=0.8, w=0.2, s=0.7,$ and $t=2.5,$ or $u=0.9, v=0.6, w=0.05, s=0.8,$ and $t=2$, the conditions of Corollary 2 hold. In this case, the differential privacy with finite cumulative privacy budgets ϵ and δ over infinite iterations as well as the mean square convergence can be established simultaneously.

Remark 20: The result of Corollary 2 does not contradict the tradeoff between privacy and utility. In fact, to achieve differential privacy, Algorithm 1 incurs a compromise on the utility. However, different from [33] and [38], which compromise convergence accuracy to enable differential privacy, Algorithm 1 compromises the convergence rate and the sample-size (which are also utility metrics) instead. From Corollary 2, it follows that the larger the privacy noise parameter σ_k is, the slower the mean square convergence rate is. Besides, the sample-size γ_T is required to go to infinity when the mean square convergence of Algorithm 1 and finite cumulative privacy budgets ϵ and δ over infinite iterations are considered simultaneously. The ability to retain convergence accuracy makes our approach suitable for accuracy-critical scenarios.

E. Oracle Complexity

Since the sampling parameter-controlled subsampling method is employed in Algorithm 1, the total number of data samples to obtain an optimal solution is an issue worthy of attention. To show this, we give the definitions of η-optimal solutions and the oracle complexity as follows.

Definition 4 (η-optimal solution): Given $\eta > 0$, $x_T = [x_{1,T}^\top, \ldots, x_{n,T}^\top]^\top$ is an η-optimal solution if $\mathbb{E}|F(x_{i,T}) - F^*| < \eta, \forall i \in \mathcal{V}$.

Definition 5: Given $\eta > 0$, the oracle complexity is the total number of data samples to obtain an η-optimal solution $\sum_{k=0}^{N(\eta)} \gamma_T$, where $N(\eta) = \min\{T: x_T \text{ is an } \eta\text{-optimal solution}\}$.

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

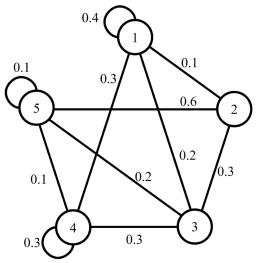


Fig. 1. Topology structure of the undirected graph.

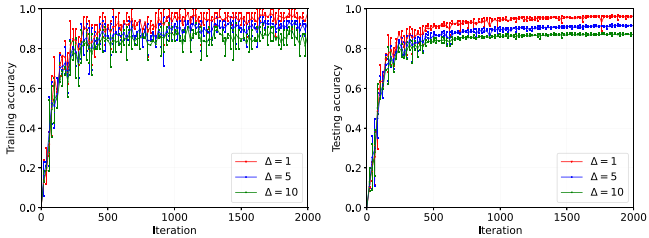


Fig. 2. Accuracy of Algorithm 1 with $\Delta = 1, 5, 10$. (a) Training accuracy. (b) Testing accuracy.

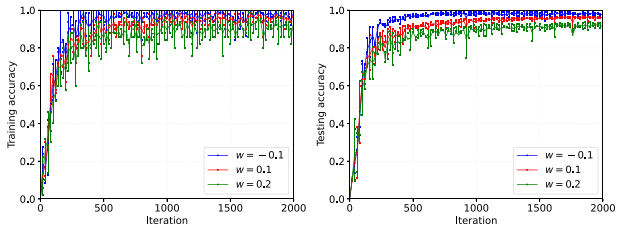


Fig. 3. Accuracy of Algorithm 1 with $w = -0.1, 0.1, 0.2$. (a) Training accuracy. (b) Testing accuracy.

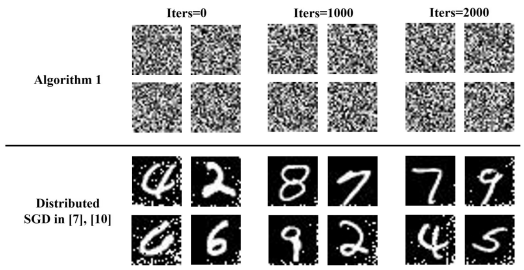


Fig. 4. Comparison of privacy protection between Algorithm 1 and distributed SGD in [7] and [10].

Based on Theorem 3, Definitions 4 and 5, the oracle complexity of Algorithm 1 for obtaining an η-optimal solution is given as follows.

Theorem 4: Given $0 < \eta < \frac{2}{5}$, let $u = 1 - \frac{\eta}{8}$, $v = \frac{2}{3} + \frac{7\eta}{12}$, $w = \eta$, and $s = \eta$. Then, under Assumptions 1–3 and 5, the oracle complexity of Algorithm 1 is $O(\eta^{-\frac{6+6\eta}{2-5\eta}})$.

Proof: Due to the space limitation, we leave the proof to the extended version [48] of this paper available at arXiv:2403.18254v3.

Remark 21: From Theorems 3 and 4, the faster the convergence rate is, the smaller the oracle complexity is. It is worth noting that as T becomes large, one might question how one deals with γ_T going to infinity. This issue does not arise in machine learning due to η-optimal solution is interested. For example, if $\eta = 0.02$, then the total number of data samples to obtain an η-optimal solution is $O(10^5)$, which does not go to infinity. This requirement for the total number of data samples is acceptable since the computational cost of centralized SGD is $O(10^5)$ to achieve the same accuracy as Algorithm 1.

IV. Numerical Examples

In this section, we verify the effectiveness and advantages of Algorithm 1 by the distributed training of a CNN on the "MNIST" dataset ([49]). Specifically, five nodes cooperatively train a CNN using the "MNIST" dataset over a topology depicted in Fig. 1, which satisfies Assumption 1. Then, the "MNIST" dataset is divided into two subdatasets for training and testing, respectively. The training dataset is uniformly divided into five subdatasets consisting of 12 000 binary images, and each of them can only be accessed by one agent to update its model parameters. The CNN model has two convolutional layers with 16 and 32 filters, respectively, followed by a fully connected layer. The activation function of each convolutional layer is the Sigmoid function $\varphi(x) = \frac{1}{1+e^{-x}}$. Then, the global cost function is nonconvex and satisfies Assumption 5. In the following, the effect of the noise and the quantization on convergence, the differential privacy level, and the comparison with methods in [31], [32], [33], [34], [35], [36], and [37] are presented for Algorithm 1, respectively.

A. Effect of the Noise and the Quantization on Convergence

Let step-sizes $\alpha_T = \frac{9.35}{2001^{0.9}} \approx 10^{-2}$, $\beta_T = \frac{0.2}{2001^{0.7}} \approx 10^{-3}$, the sample-size $\gamma_T = \lfloor 5.5 \cdot 10^{-4} \cdot 2000^{1.5} \rfloor + 1 = 50$, $\delta_k = \frac{1}{(k+2)^3}$, and the privacy noise parameter $\sigma_k = (k+1)^w$ with $w = -0.1, 0.1, 0.2$, respectively. The probabilistic quantizer is given in the form of (3) with $\Delta = 1, 5, 10$, respectively. Then, it can be seen that Assumptions 2–4 hold. The training and testing accuracy on the "MNIST" dataset are presented in Figs. 2 and 3, from which one can see that as iterations increase, the training and testing accuracy increase. More importantly, the smaller Δ and w are, the faster Algorithm 1 converges, which is consistent with Theorem 3.

B. Differential Privacy Level

Based on the model inversion attack given in [45], we compare Algorithm 1 and the algorithms without privacy protection in [7] and [10] to show that Algorithm 1 can protect the sensitive information from sampled gradients. A comparison of privacy protection between Algorithm 1 and distributed SGD on the "MNIST" dataset is presented in Fig. 4, from which one can see that adversaries cannot recover original handwritten digit images in Algorithm 1, while adversaries can completely recover original handwritten digit images in distributed SGD [7], [10].

Next, the relationship of the cumulative differential privacy budget ϵ over infinite iterations, the privacy noise parameter w and sample-size parameter s is presented in Fig. 5, from which one can see that as the privacy noise parameter w and the sample-size parameter s increase, the cumulative differential privacy budget ϵ decrease. This is consistent with the privacy

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

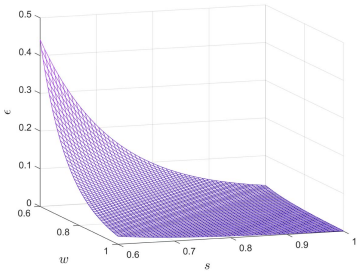


Fig. 5. Relationship of ϵ, w, and s.

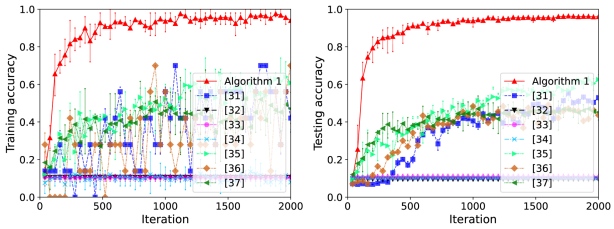


Fig. 6. Comparison of accuracy. (a) Training accuracy. (b) Testing accuracy.

analysis in Section III-B. Moreover, in the first 2000 iterations, the cumulative differential privacy budgets $\epsilon = 0.7594$ and $\delta = 0.2021$, which is consistent with Theorem 1.

C. Comparison With Methods in [31], [32], [33], [34], [35], [36], and [37]

Let $\Delta = 1, w = 0.1$ in Algorithm 1. Then, the comparison of accuracy between Algorithm 1 and methods in [31], [32], [33], [34], [35], [36], and [37] is presented in Fig. 6. To ensure a fair comparison, we set the same step-sizes in [31], [36], and [37] as this article, and the step-sizes in [32], [33], [34], and [35] as chosen therein. In addition, we set sample-sizes in [31], [32], [33], [34], [35], [36], and [37] as chosen therein. From Fig. 6(a) and (b), it can be seen that the convergence rate of Algorithm 1 is faster than [31], [32], [33], [34], [35], [36], [37].

Since the structure of the CNN model is known, the sampled gradient $\|\nabla \ell_i(x, \xi_{i,l})\|$ is bounded for any $x \in \mathbb{R}^{29034}$ and $\xi_{i,l} \in \mathcal{D}$. When running the CNN model on the "MNIST" dataset, the maximum magnitude of the sampled gradient $\|\nabla \ell_i(x, \xi_{i_0,l_0}) - \nabla \ell_i(x, \xi'_{i_0,l_0})\|$ is no more than 60 after changing one data sample ξ_{i_0,l_0} to any different data sample ξ'_{i_0,l_0}. Then, when the constant $C = 60$, Definition 1 contains the adjacency relation in [31], [32], [33], [34], [35], [36], and [37] and vice versa, which implies that Definition 1 is equivalent to the adjacency relation therein. Thus, cumulative differential privacy budgets ϵ, δ of Algorithm 1 can be compared with those of methods in [31], [32], [33], [34], [35], [36], and [37], and the comparison of cumulative differential privacy budgets ϵ, δ is presented in Fig. 7. From Fig. 7(a) and (b), one can see that cumulative differential privacy budgets ϵ and δ of Algorithm 1 are bounded by finite constants over infinite iterations, while cumulative differential privacy budgets ϵ, δ in [31], [32], [33], [34], [35], [36], and [37] go to infinity over infinite iterations.

In summary, the discussion above demonstrates Algorithm 1's superior performance over [31], [32], [33], [34], [35], [36], and [37] on the convergence rate and the differential privacy level.

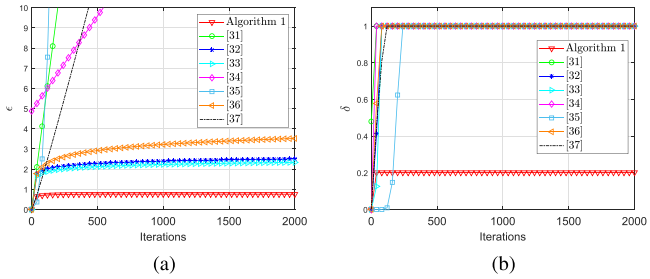


Fig. 7. Comparison of cumulative differential privacy budgets ϵ and δ. (a) ϵ under same σ_k and δ_k. (b) δ under same σ_k and ϵ_k.

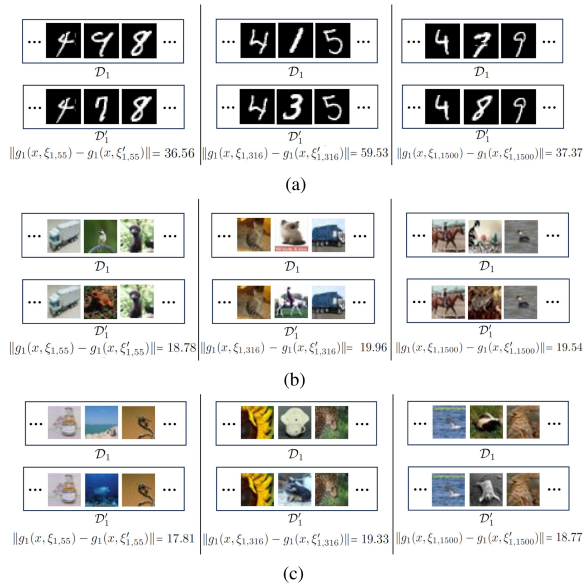


Fig. 8. Different constant C for different datasets. (a) "MNIST" dataset, $C = 60$. (b) "CIFAR-10" dataset, $C = 20$. (c) "CIFAR-100" dataset, $C = 19.5$.

Remark 22: It is noted that only when a comparison between the method of this article and the methods in [31], [32], [33], [34], [35], [36], and [37] is needed, the constant C can be different for different datasets. For example, Fig. 8 shows different constant C for the "MNIST," "CIFAR-10,"[50] and "CIFAR-100"[51], [52] dataset, respectively. For each dataset, we randomly change one data sample and compute the magnitude of sampled gradients. Due to the space limitation, only three examples are given for each dataset. Fig. 8(a) shows that for the "MNIST" dataset, the magnitude of sampled gradients after respectively changing the 55th, 316th, 1500th data sample is 36.56, 59.53, 37.37, which is no more than the constant $C = 60$. Similarly, Fig. 8(b) and (c) shows that the magnitude of sampled gradients is no more than the constant $C = 20$ and 19.5, respectively. This interesting finding is consistent with [31], [32], [33], [34], [35], [36], and [37], where the upper bound of bounded gradients is also different for different datasets. When a comparison is not needed, the constant C can be a fixed value for different datasets.

V. Conclusion

In this article, we have proposed a differentially private distributed nonconvex stochastic optimization algorithm with quantized communication. In the proposed algorithm, general privacy noises are added to each node's local states to protect the sensitive information, and then a probabilistic quantizer is

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

employed on noise-perturbed states to improve communication efficiency. By using the sampling parameter-controlled subsampling method, the differential privacy level of the algorithm is enhanced compared with the existing ones. By using a new convergence analysis technique and the two-time-scale step-sizes method, the effect of the quantization error on convergence is eliminated while improving communication efficiency, and thus, the mean square convergence for nonconvex cost functions is obtained. Then, under the Polyak–Łojasiewicz condition, the mean square convergence rate and the oracle complexity of the algorithm are given. Meanwhile, the tradeoff between the privacy and the utility is shown. Finally, a numerical example of the distributed training of CNN on the "MNIST" dataset is given to verify the effectiveness of the algorithm.

Appendix A
Useful Lemma

Lemma A.1: If a function $h(x)$ defined on $\mathbb{R}^r$ satisfies Assumption 2(i), and $\min_{x\in\mathbb{R}^r} h(x) = h^* > -\infty$, then the following results hold: (i) $h(y) \le h(x) + \langle \nabla h(x), y-x\rangle + \frac{L_1}{2}\|y-x\|^2, \forall x, y \in \mathbb{R}^r$; (ii) $\|\nabla h(x)\|^2 \le 2L_1(h(x)-h^*), \forall x \in \mathbb{R}^r$.

Proof: Lemma A.1(i) is directly from [43, Lemma 3.4]. By [43, eq. (3.5)], we have $\|\nabla h(x)\|^2 \le 2L_1(h(x) - h(x-\frac{1}{L_1}\nabla h(x))) \le 2L_1(h(x)-h^*)$, then Lemma A.1(ii) is proved. ■

Appendix B
Proof of Theorem 2

To provide an explanation of our results clearly, define

$$\nabla f(x_k) \triangleq [\nabla f_1(x_{1,k})^\top, \nabla f_2(x_{2,k})^\top, \ldots, \nabla f_n(x_{n,k})^\top]^\top$$
$$\nabla \ell(x_k) \triangleq [\nabla \ell_{1,k}^\top, \nabla \ell_{2,k}^\top, \ldots, \nabla \ell_{n,k}^\top]^\top$$
$$\nabla f(\bar{x}_k) \triangleq [\nabla f_1(\bar{x}_k)^\top, \nabla f_2(\bar{x}_k)^\top, \ldots, \nabla f_n(\bar{x}_k)^\top]^\top$$
$$W \triangleq I_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top,\ Y_k \triangleq (W\otimes I_r)x_k$$
$$e_k \triangleq z_k - x_k - d_k,\ w_k \triangleq \nabla\ell(x_k) - \nabla f(x_k)$$
$$\bar{x}_k \triangleq \frac{1}{n}(\mathbf{1}_n^\top \otimes I_r)x_k,\ \bar{w}_k \triangleq \frac{1}{n}(\mathbf{1}_n^\top \otimes I_r)w_k$$
$$\overline{\nabla f(x_k)} \triangleq \frac{1}{n}(\mathbf{1}_n^\top \otimes I_r)\nabla f(x_k) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(x_{i,k}).$$

Then, we can express (7) for all nodes in a compact form as follows:

$$x_{k+1} = ((I_n - \beta_T\mathcal{L})\otimes I_r)x_k - \alpha_T\nabla f(x_k) + \beta_T(\mathcal{A}\otimes I_r)(e_k+d_k) - \alpha_T w_k. \quad (20)$$

Next, the following six steps are given to prove Theorem 2.

Step 1: We first consider the term $\|Y_k\|^2$. Note that $W(I_n - \beta_T\mathcal{L}) = (I_n - \beta_T\mathcal{L})W$. Then, multiplying both sides of (20) by $W\otimes I_r$ gives

$$Y_{k+1} = ((I_n - \beta_T\mathcal{L})\otimes I_r)Y_k - \alpha_T(W\otimes I_r)\nabla f(x_k) + \beta_T(W\mathcal{A}\otimes I_r)(e_k+d_k) - \alpha_T(W\otimes I_r)w_k. \quad (21)$$

For any $k = 0, \ldots, T$, define σ-algebras $\mathcal{F}_k = \sigma(x_k, d_k)$, $\mathcal{H}_k = \sigma(x_k)$. Then, since $d_{i,k}$ is independent of $\mathcal{H}_k$ and follows the normal distribution $N(0, \sigma_k^2 I_r)$, we have

$$\mathbb{E}d_k = \mathbb{E}(d_k|\mathcal{H}_k) = 0 \quad (22)$$
$$\mathbb{E}\|d_k\|^2 = \mathbb{E}(\|d_k\|^2|\mathcal{H}_k) = nr\sigma_k^2. \quad (23)$$

Since w_k is independent of $\mathcal{F}_k$, by Assumption 2(iii), we have

$$\mathbb{E}w_k = \mathbb{E}(w_k|\mathcal{F}_k) = 0 \quad (24)$$
$$\mathbb{E}\|w_k\|^2 = \mathbb{E}(\|w_k\|^2|\mathcal{F}_k) \le \frac{n\sigma_\ell^2}{\gamma_T}. \quad (25)$$

Since e_k is independent of $\mathcal{F}_k$, by Assumption 3 we have

$$\mathbb{E}e_k = \mathbb{E}(e_k|\mathcal{F}_k) = 0 \quad (26)$$
$$\mathbb{E}\|e_k\|^2 = \mathbb{E}(\|e_k\|^2|\mathcal{F}_k) \le nr\Delta^2. \quad (27)$$

By (22), (24), and (26), taking mathematical expectation of $\|Y_{k+1}\|^2$ leads to

$$\begin{aligned}
&\mathbb{E}\|Y_{k+1}\|^2 \\
&= \mathbb{E}\|((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k - \alpha_T(W\otimes I_r)\nabla f(x_k) \\
&\quad + \beta_T(W\mathcal{A}\otimes I_r)(d_k+e_k) - \alpha_T(W\otimes I_r)w_k\|^2 \\
&= \mathbb{E}\|((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k - \alpha_T(W\otimes I_r)\nabla f(x_k)\|^2 \\
&\quad + \beta_T^2\mathbb{E}\|(W\mathcal{A}\otimes I_r)(d_k+e_k)\|^2 + \alpha_T^2\mathbb{E}\|(W\otimes I_r)w_k\|^2 \\
&\quad + 2\beta_T\mathbb{E}\langle((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k \\
&\quad - \alpha_T(W\otimes I_r)\nabla f(x_k), (\mathcal{A}W\otimes I_r)(d_k+e_k)\rangle \\
&\quad + 2\alpha_T\mathbb{E}\langle((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k \\
&\quad - \alpha_T(W\otimes I_r)\nabla f(x_k), (W\otimes I_r)w_k\rangle \\
&\quad + 2\alpha_T\beta_T\mathbb{E}\langle(W\mathcal{A}\otimes I_r)(d_k+e_k), (W\otimes I_r)w_k\rangle \\
&= \mathbb{E}\|((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k - \alpha_T(W\otimes I_r)\nabla f(x_k)\|^2 \\
&\quad + \beta_T^2\mathbb{E}\left(\|(W\mathcal{A}\otimes I_r)(d_k+e_k)\|^2\right) \\
&\quad + \alpha_T^2\mathbb{E}\|(W\otimes I_r)w_k\|^2. \quad (28)
\end{aligned}$$

Then, by the law of total expectation [48, Thm. 7.1.1], we have

$$\begin{aligned}
&\mathbb{E}\langle((W\mathcal{A})\otimes I_r)d_k, ((W\mathcal{A})\otimes I_r)e_k\rangle \\
&= \mathbb{E}(\mathbb{E}(\langle(((W\mathcal{A})\otimes I_r)d_k, ((W\mathcal{A})\otimes I_r)e_k\rangle|\mathcal{F}_k)) \\
&= \mathbb{E}\langle((W\mathcal{A})\otimes I_r)d_k, \mathbb{E}(((W\mathcal{A})\otimes I_r)e_k|\mathcal{F}_k)\rangle \\
&= \mathbb{E}\langle((W\mathcal{A})\otimes I_r)d_k, 0\rangle = 0. \quad (29)
\end{aligned}$$

Thus, substituting (29) into (28) implies

$$\begin{aligned}
&\mathbb{E}\|Y_{k+1}\|^2 \\
&= \mathbb{E}\|((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k - \alpha_T(W\otimes I_r)\nabla f(x_k)\|^2 \\
&\quad + \beta_T^2\mathbb{E}\left(\|(W\mathcal{A}\otimes I_r)d_k\|^2 + \|(W\mathcal{A}\otimes I_r)e_k\|^2\right) \\
&\quad + 2\mathbb{E}\langle((W\mathcal{A})\otimes I_r)d_k, ((W\mathcal{A})\otimes I_r)e_k\rangle + \alpha_T^2\mathbb{E}\|(W\otimes I_r)w_k\|^2 \\
&= \mathbb{E}\|((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k - \alpha_T(W\otimes I_r)\nabla f(x_k)\|^2 \\
&\quad + \beta_T^2\mathbb{E}\left(\|(W\mathcal{A}\otimes I_r)d_k\|^2 + \|(W\mathcal{A}\otimes I_r)e_k\|^2\right) \\
&\quad + \alpha_T^2\mathbb{E}\|(W\otimes I_r)w_k\|^2. \quad (30)
\end{aligned}$$

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

By Rayleigh Theorem [53, Thm. 4.2.2] and Assumption 1, $\|\mathcal{A}\| = 1$. Note that $\|A\mathbf{x}\| \le \|A\|\|\mathbf{x}\|$ for any $A \in \mathbb{R}^{n\times n}$, $\mathbf{x} \in \mathbb{R}^n$. Then, by $\|W\| = 1$, substituting (23), (25), and (27) into (30) implies

$$\mathbb{E}\,\|Y_{k+1}\|^2 \le \mathbb{E}\,\|((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k-\alpha_T(W\otimes I_r)\nabla f(x_k)\|^2 + nr\beta_T^2(\Delta^2+\sigma_k^2)+\frac{n\alpha_T^2\sigma_\ell^2}{\gamma_T}. \tag{31}$$

Furthermore, for any $\mathbf{a}, \mathbf{b} \in \mathbb{R}^r$, the following Cauchy–Schwarz inequality [54, Ex. 4(b)] holds: $\|\mathbf{a}+\mathbf{b}\|^2 \le (1+\rho_{\mathcal{L}}\beta_T)\|\mathbf{a}\|^2 + (1+\frac{1}{\rho_{\mathcal{L}}\beta_T})\|\mathbf{b}\|^2$, where $\rho_{\mathcal{L}} > 0$ is the second smallest eigenvalue of $\mathcal{L}$. This together with (31) gives

$$\begin{aligned}\mathbb{E}\|Y_{k+1}\|^2 \le\ & (1+\rho_{\mathcal{L}}\beta_T)\,\mathbb{E}\,\|((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k\|^2 \\ &+\left(1+\frac{1}{\rho_{\mathcal{L}}\beta_T}\right)\mathbb{E}\|\alpha_T(W\otimes I_r)\nabla f(x_k)\|^2 \\ &+\frac{n\alpha_T^2\sigma_\ell^2}{\gamma_T}+nr\beta_T^2(\Delta^2+\sigma_k^2).\end{aligned} \tag{32}$$

By Courant–Fischer's Theorem [53, Thm. 4.2.6] we have

$$\|((I_n-\beta_T\mathcal{L})\otimes I_r)Y_k\|^2 \le (1-\rho_{\mathcal{L}}\beta_T)^2\|Y_k\|^2. \tag{33}$$

Thus, substituting (33) into (32) and noticing $\|W\| = 1$, one can get

$$\begin{aligned}&\mathbb{E}\|Y_{k+1}\|^2 \\ &\le (1+\rho_{\mathcal{L}}\beta_T)(1-\rho_{\mathcal{L}}\beta_T)^2\mathbb{E}\|Y_k\|^2 + nr\beta_T^2(\Delta^2+\sigma_k^2) \\ &\quad+\frac{1+\rho_{\mathcal{L}}\beta_T}{\rho_{\mathcal{L}}\beta_T}\mathbb{E}\,\|\alpha_T(W\otimes I_r)\nabla f(x_k)\|^2+\frac{n\alpha_T^2\sigma_\ell^2}{\gamma_T} \\ &\le (1+\rho_{\mathcal{L}}\beta_T)(1-\rho_{\mathcal{L}}\beta_T)^2\mathbb{E}\|Y_k\|^2 + nr\beta_T^2(\Delta^2+\sigma_k^2) \\ &\quad+\frac{(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2}{\rho_{\mathcal{L}}\beta_T}\mathbb{E}\,\|\nabla f(x_k)\|^2+\frac{n\alpha_T^2\sigma_\ell^2}{\gamma_T} \\ &= (1+\rho_{\mathcal{L}}\beta_T)(1-\rho_{\mathcal{L}}\beta_T)^2\mathbb{E}\|Y_k\|^2 + nr\beta_T^2(\Delta^2+\sigma_k^2) \\ &\quad+\frac{(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2}{\rho_{\mathcal{L}}\beta_T}\mathbb{E}\|\nabla f(x_k)-\nabla f(\bar{x}_k)+\nabla f(\bar{x}_k)\|^2+\frac{n\alpha_T^2\sigma_\ell^2}{\gamma_T}.\end{aligned} \tag{34}$$

Note that for any $m \ge 1$ and $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m \in \mathbb{R}^r$, the following inequality holds:

$$\|\mathbf{a}_1+\mathbf{a}_2+\ldots+\mathbf{a}_m\|^2 \le m(\|\mathbf{a}_1\|^2+\|\mathbf{a}_2\|^2+\ldots+\|\mathbf{a}_m\|^2). \tag{35}$$

Then, by letting $m = 2$ in (35), $\|\nabla f(x_k)-\nabla f(\bar{x}_k)+\nabla f(\bar{x}_k)\|^2$ in (34) can be rewritten as

$$\begin{aligned}&\|\nabla f(x_k)\|^2 \\ &\le 2\,\|\nabla f(x_k)-\nabla f(\bar{x}_k)\|^2+2\,\|\nabla f(\bar{x}_k)\|^2 \\ &= 2\sum_{i=1}^n\|\nabla f_i(x_{i,k})-\nabla f_i(\bar{x}_k)\|^2+2\sum_{i=1}^n\|\nabla f_i(\bar{x}_k)\|^2.\end{aligned} \tag{36}$$

By Assumption 2(i), for any $x, y \in \mathbb{R}^r$, we have

$$\begin{aligned}&\|\nabla f_i(x)-\nabla f_i(y)\| = \|\mathbb{E}\nabla\ell(x,\xi_i)-\mathbb{E}\nabla\ell(y,\xi_i)\| \\ &\le \mathbb{E}\|\nabla\ell(x,\xi_i)-\nabla\ell(y,\xi_i)\| \le L_1\|x-y\|.\end{aligned}$$

Then, it can be seen that $\|\nabla f_i(x_{i,k})-\nabla f_i(\bar{x}_k)\| \le L_1\|x_{i,k}-\bar{x}_k\|$. Since $\|Y_k\|^2 = \|(W\otimes I_r)x_k\|^2 = \sum_{i=1}^n\|x_{i,k}-\bar{x}_k\|^2$, $\sum_{i=1}^n\|\nabla f_i(x_{i,k})-\nabla f_i(\bar{x}_k)\|^2$ in (36) can be rewritten as

$$\sum_{i=1}^n\|\nabla f_i(x_{i,k})-\nabla f_i(\bar{x}_k)\|^2 \le L_1^2\sum_{i=1}^n\|x_{i,k}-\bar{x}_k\|^2 = L_1^2\|Y_k\|^2. \tag{37}$$

By Assumption 2(ii) and Lemma A.1(ii), $\|\nabla f_i(\bar{x}_k)\|^2 \le 2L_1(f_i(\bar{x}_k)-f_i^*)$, we have

$$\sum_{i=1}^n\|\nabla f_i(\bar{x}_k)\|^2 \le 2L_1\sum_{i=1}^n(f_i(\bar{x}_k)-f_i^*). \tag{38}$$

Thus, substituting (37) and (38) into (36) gives

$$\begin{aligned}&\|\nabla f(x_k)-\nabla f(\bar{x}_k)+\nabla f(\bar{x}_k)\|^2 \\ &\le 2L_1^2\|Y_k\|^2+4L_1\left(\sum_{i=1}^n f_i(\bar{x}_k)-f_i^*\right).\end{aligned} \tag{39}$$

Note that by Assumption 2(ii), each cost function $f_i(x)$ has the minimum f_i^*. Then, the global cost function $F(x)$ has the global minimum $F^* = \min_{x\in\mathbb{R}^r}F(x)$. Let $M^* = F^* - \frac{1}{n}\sum_{i=1}^n f_i^*$. Then, (39) can be rewritten as $\|\nabla f(x_k)-\nabla f(\bar{x}_k)+\nabla f(\bar{x}_k)\|^2 \le 2L_1^2\|Y_k\|^2 + 4L_1(\sum_{i=1}^n f_i(\bar{x}_k)-f_i^*) = 2L_1^2\|Y_k\|^2 + 4nL_1(F(\bar{x}_k)-F^*)+4nL_1M^*$. This together with (34) implies

$$\begin{aligned}\mathbb{E}\|Y_{k+1}\|^2 \le\ & \left(1-\rho_{\mathcal{L}}\beta_T+\frac{2(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1^2}{\rho_{\mathcal{L}}\beta_T}\right)\mathbb{E}\|Y_k\|^2 \\ &+\frac{4n(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1}{\rho_{\mathcal{L}}\beta_T}\mathbb{E}(F(\bar{x}_k)-F^*)+\frac{n\alpha_T^2\sigma_\ell^2}{\gamma_T} \\ &+nr\beta_T^2(\Delta^2+\sigma_k^2)+\frac{4n(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1M^*}{\rho_{\mathcal{L}}\beta_T}.\end{aligned} \tag{40}$$

Step 2: We next focus on the term $F(\bar{x}_k)-F^*$. Multiplying both sides of (20) by $\frac{1}{n}(\mathbf{1}_n^\top\otimes I_r)$ implies

$$\bar{x}_{k+1} = \bar{x}_k-\alpha_T\overline{\nabla f(x_k)}-\alpha_T\bar{w}_k+\frac{\beta_T}{n}(\mathbf{1}_n^\top\otimes I_r)(e_k+d_k). \tag{41}$$

Then, by (41) and Lemma A.1(i), we can derive that

$$\begin{aligned}&F(\bar{x}_{k+1})-F^* \\ &\le (F(\bar{x}_k)-F^*)+\frac{L_1}{2}\|\bar{x}_{k+1}-\bar{x}_k\|^2+\langle\nabla F(\bar{x}_k),\bar{x}_{k+1}-\bar{x}_k\rangle \\ &= (F(\bar{x}_k)-F^*)+\frac{L_1}{2}\left\|\alpha_T\overline{\nabla f(x_k)}-\frac{\beta_T}{n}\left(\mathbf{1}_n^\top\otimes I_r\right)(e_k+d_k)\right. \\ &\quad\left.+\alpha_T\bar{w}_k\right\|^2-\left\langle\nabla F(\bar{x}_k),-\frac{\beta_T}{n}\left(\mathbf{1}_n^\top\otimes I_r\right)(e_k+d_k)\right. \\ &\quad\left.+\alpha_T\overline{\nabla f(x_k)}+\alpha_T\bar{w}_k\right\rangle.\end{aligned} \tag{42}$$

By (22), (24), and (26), taking mathematical expectation of (42) gives

$$\begin{aligned}&\mathbb{E}\,(F(\bar{x}_{k+1})-F^*) \\ &\le \mathbb{E}(F(\bar{x}_k)-F^*)-\alpha_T\mathbb{E}\left\langle\nabla F(\bar{x}_k),\overline{\nabla f(x_k)}\right\rangle \\ &\quad+\frac{L_1}{2}\mathbb{E}\left\|\alpha_T\overline{\nabla f(x_k)}-\frac{\beta_T}{n}\left(\mathbf{1}_n^\top\otimes I_r\right)(e_k+d_k)+\alpha_T\bar{w}_k\right\|^2\end{aligned}$$

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

$$= \mathbb{E}(F(\bar{x}_k) - F^*) - \alpha_T \mathbb{E}\left\langle \nabla F(\bar{x}_k), \overline{\nabla f(x_k)} \right\rangle + \frac{\beta_T^2 L_1}{2n^2} \mathbb{E}\left(\| \left(\mathbf{1}_n^\top \otimes I_r\right) e_k\|^2 + \| \left(\mathbf{1}_n^\top \otimes I_r\right) d_k\|^2 \right) + \frac{\alpha_T^2 L_1}{2} \mathbb{E}\left\| \overline{\nabla f(x_k)} \right\|^2 + \frac{\alpha_T^2 L_1}{2} \mathbb{E}\|\bar{w}_k\|^2. \tag{43}$$

Note that $\|(\mathbf{1}_n^\top \otimes I_r) d_k\|^2 = \| \sum_{i=1}^n d_{i,k}\|^2 \le n\|d_k\|^2$, $\|(\mathbf{1}_n^\top \otimes I_r) e_k\|^2 = \| \sum_{i=1}^n e_{i,k}\|^2 \le n\|e_k\|^2$, and $\|\bar{w}_k\|^2 = \|\frac{1}{n}\sum_{i=1}^n w_{i,k}\|^2 \le \frac{1}{n}\sum_{i=1}^n \|w_{i,k}\|^2$. Then, by (23), (25), and (27), (43) can be rewritten as

$$\begin{aligned} &\mathbb{E}\left(F(\bar{x}_{k+1}) - F^*\right) \\ &\le \mathbb{E}(F(\bar{x}_k) - F^*) - \alpha_T \mathbb{E}\left\langle \nabla F(\bar{x}_k), \overline{\nabla f(x_k)} \right\rangle \\ &\quad + \frac{\alpha_T^2 L_1}{2} \mathbb{E}\left\| \overline{\nabla f(x_k)} \right\|^2 + \frac{\beta_T^2 n r L_1}{2}(\Delta^2 + \sigma_k^2) + \frac{\alpha_T^2 \sigma_\ell^2 L_1}{2\gamma_T}. \end{aligned} \tag{44}$$

Note that $\langle \mathbf{a}, \mathbf{b}\rangle = \frac{1}{2}\|\mathbf{a}\|^2 + \frac{1}{2}\|\mathbf{b}\|^2 - \frac{1}{2}\|\mathbf{a}-\mathbf{b}\|^2$ for any $\mathbf{a}, \mathbf{b} \in \mathbb{R}^r$. Then, $-\alpha_T\langle \nabla F(\bar{x}_k), \overline{\nabla f(x_k)}\rangle$ in (44) can be rewritten as

$$\begin{aligned} &-\alpha_T \left\langle \nabla F(\bar{x}_k), \overline{\nabla f(x_k)} \right\rangle \\ &= -\frac{\alpha_T}{2}\|\nabla F(\bar{x}_k)\|^2 - \frac{\alpha_T}{2}\left\| \overline{\nabla f(x_k)} \right\|^2 + \frac{\alpha_T}{2}\left\| \nabla F(\bar{x}_k) - \overline{\nabla f(x_k)} \right\|^2 \\ &\le -\frac{\alpha_T}{2}\|\nabla F(\bar{x}_k)\|^2 + \frac{\alpha_T}{2}\left\| \nabla F(\bar{x}_k) - \overline{\nabla f(x_k)} \right\|^2. \end{aligned} \tag{45}$$

By letting $m = n$ in (35), $\|\nabla F(\bar{x}_k) - \overline{\nabla f(x_k)}\|^2$ in (45) can be rewritten as

$$\begin{aligned} \left\| \nabla F(\bar{x}_k) - \overline{\nabla f(x_k)} \right\|^2 &= \left\| \frac{1}{n}\sum_{i=1}^n (\nabla f_i(\bar{x}_k) - \nabla f_i(x_{i,k})) \right\|^2 \\ &\le \frac{1}{n}\sum_{i=1}^n \|\nabla f_i(\bar{x}_k) - \nabla f_i(x_{i,k})\|^2. \end{aligned} \tag{46}$$

Thus, by (37), (46) can be rewritten as

$$\left\| \nabla F(\bar{x}_k) - \overline{\nabla f(x_k)} \right\|^2 \le \frac{L_1^2}{n}\|Y_k\|^2. \tag{47}$$

Substituting (45) and (47) into (44) implies

$$\begin{aligned} &\mathbb{E}(F(\bar{x}_{k+1}) - F^*) \\ &\le \mathbb{E}(F(\bar{x}_k) - F^*) - \frac{\alpha_T}{2}\mathbb{E}\|\nabla F(\bar{x}_k)\|^2 + \frac{\alpha_T L_1^2}{2n}\mathbb{E}\|Y_k\|^2 \\ &\quad + \frac{\alpha_T^2 L_1}{2}\mathbb{E}\left\| \overline{\nabla f(x_k)} - \nabla F(\bar{x}_k) + \nabla F(\bar{x}_k) \right\|^2 \\ &\quad + \frac{\beta_T^2 n r L_1}{2}\left(\Delta^2 + \sigma_k^2\right) + \frac{\alpha_T^2 \sigma_\ell^2 L_1}{2\gamma_T}. \end{aligned} \tag{48}$$

Furthermore, by letting $m = 2$ in (35) and using (47), $\|\overline{\nabla f(x_k)} - \nabla F(\bar{x}_k) + \nabla F(\bar{x}_k)\|^2$ in (48) can be rewritten as

$$\begin{aligned} &\left\| \overline{\nabla f(x_k)} - \nabla F(\bar{x}_k) + \nabla F(\bar{x}_k) \right\|^2 \\ &\le 2\left\| \overline{\nabla f(x_k)} - \nabla F(\bar{x}_k) \right\|^2 + 2\|\nabla F(\bar{x}_k)\|^2 \\ &\le \frac{2L_1^2}{n}\|Y_k\|^2 + 2\|\nabla F(\bar{x}_k)\|^2. \end{aligned} \tag{49}$$

By letting $m = n$ in (35) and using (38), $\|\nabla F(\bar{x}_k)\|^2$ in (49) can be rewritten as

$$\begin{aligned} \|\nabla F(\bar{x}_k)\|^2 &\le \frac{1}{n}\sum_{i=1}^n \|\nabla f_i(\bar{x}_k)\|^2 \\ &\le \frac{2L_1}{n}\sum_{i=1}^n (f_i(\bar{x}_k) - f_i^*) \\ &= 2L_1\left(F(\bar{x}_k) - F^*\right) + 2L_1 M^*. \end{aligned} \tag{50}$$

Thus, substituting (49) and (50) into (48) implies

$$\begin{aligned} &\mathbb{E}\left(F(\bar{x}_{k+1}) - F^*\right) \\ &\le \left(1 + 2\alpha_T^2 L_1^2\right)\mathbb{E}(F(\bar{x}_k) - F^*) \\ &\quad - \frac{\alpha_T}{2}\mathbb{E}\|\nabla F(\bar{x}_k)\|^2 + \frac{\alpha_T L_1^2(1 + 2\alpha_T L_1)}{2n}\mathbb{E}\|Y_k\|^2 \\ &\quad + \frac{\alpha_T^2 \sigma_\ell^2 L_1}{2\gamma_T} + \frac{\beta_T^2 n r L_1}{2}(\Delta^2 + \sigma_k^2) + 2\alpha_T^2 L_1^2 M^* \\ &\le \left(1 + 2\alpha_T^2 L_1^2\right)\mathbb{E}(F(\bar{x}_k) - F^*) \\ &\quad + \frac{\alpha_T L_1^2(1 + 2\alpha_T L_1)}{2n}\mathbb{E}\|Y_k\|^2 + \frac{\alpha_T^2 \sigma_\ell^2 L_1}{2\gamma_T} \\ &\quad + \frac{\beta_T^2 n r L_1}{2}(\Delta^2 + \sigma_k^2) + 2\alpha_T^2 L_1^2 M^*. \end{aligned} \tag{51}$$

Let

$$\begin{aligned} \theta_1 = \max\Bigg\{ &1 + 2\alpha_T^2 L_1^2 + \frac{4n(1 + \rho_{\mathcal{L}}\beta_T)\alpha_T^2 L}{\rho_{\mathcal{L}}\beta_T}, \\ &1 - \rho_{\mathcal{L}}\beta_T + \frac{\alpha_T L_1^2(1 + 2\alpha_T L_1)}{2n} + \frac{2(1 + \rho_{\mathcal{L}}\beta_T)\alpha_T^2 L_1^2}{\rho_{\mathcal{L}}\beta_T} \Bigg\} \end{aligned} \tag{52}$$

$$\begin{aligned} \theta_{k,2} = &\frac{(L_1 + 2)nr\beta_T^2}{2}(\Delta^2 + \sigma_k^2) + \frac{\alpha_T^2 \sigma_\ell^2 (2n + L_1)}{2\gamma_T} \\ &+ 2\alpha_T^2 L_1^2 M^* + \frac{4n(1 + \rho_{\mathcal{L}}\beta_T)\alpha_T^2 L_1 M^*}{\rho_{\mathcal{L}}\beta_T}. \end{aligned} \tag{53}$$

Then, summing (40) and (51) implies

$$\begin{aligned} &\mathbb{E}(\|Y_{k+1}\|^2 + F(\bar{x}_{k+1}) - F^*) \\ &\le \theta_1 \mathbb{E}(\|Y_k\|^2 + F(\bar{x}_k) - F^*) + \theta_{k,2}. \end{aligned} \tag{54}$$

By iteratively computing (54), the following inequality holds:

$$\begin{aligned} &\mathbb{E}(\|Y_{T+1}\|^2 + F(\bar{x}_{T+1}) - F^*) \\ &\le \theta_1^{T+1}(\|Y_0\|^2 + F(\bar{x}_0) - F^*) + \sum_{k=0}^T \theta_1^{T-k}\theta_{k,2}. \end{aligned} \tag{55}$$

Step 3: At this step, we prove that there exists $G_1 \ge 0$ such that $\mathbb{E}(F(\bar{x}_T) - F^*) \le G_1$ for any $T = 0, 1, \ldots$. Note that $2\alpha_T^2 L_1^2 = O(\frac{1}{(T+1)^{2u}})$ and $\frac{4n(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2 L_1}{\rho_{\mathcal{L}}\beta_T} = O(\frac{1}{(T+1)^{2u-v}})$

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

holds for any $T=0,1,\ldots$. Then, by $2u-v>1$ in Assumption 4, it can be seen that for any $T=0,1,\ldots$

$$\begin{aligned}&\left(1+2\alpha_T^2L_1^2+\frac{4n(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1}{\rho_{\mathcal{L}}\beta_T}\right)^{T+1}\\&=\left(1+O\left(\frac{1}{(T+1)^{2u-v}}\right)\right)^{T+1}\\&=\exp\left((T+1)\ln\left(1+O\left(\frac{1}{(T+1)^{2u-v}}\right)\right)\right)\\&=\exp\left(O\left(\frac{1}{(T+1)^{2u-v-1}}\right)\right).\end{aligned}\tag{56}$$

Note that $\ln(1+x)\leq x$ for any $x>-1$, and by $\frac{1}{2}+\max\{w,0\}<v<u$ in Assumption 4, there exists a positive integer T_0 such that $1-\rho_{\mathcal{L}}\beta_T+\frac{\alpha_TL_1^2(1+2\alpha_TL_1)}{2n}+\frac{2(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1^2}{\rho_{\mathcal{L}}\beta_T}\leq 1-\frac{\rho_{\mathcal{L}}\beta_T}{2}$ for any $T=T_0,T_0+1,\ldots$. Then, it can be seen that for any $T=T_0,T_0+1,\ldots$

$$\begin{aligned}&\left(1-\rho_{\mathcal{L}}\beta_T+\frac{\alpha_TL_1^2(1+2\alpha_TL_1)}{2n}+\frac{2(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1^2}{\rho_{\mathcal{L}}\beta_T}\right)^{T+1}\\&\leq\left(1-\frac{\rho_{\mathcal{L}}\beta_T}{2}\right)^{T+1}=\exp\left((T+1)\ln\left(1-\frac{\rho_{\mathcal{L}}a_2}{2(T+1)^v}\right)\right)\\&\leq\exp\left(-\frac{\rho_{\mathcal{L}}a_2}{2}(T+1)^{1-v}\right)\\&\leq\exp\left(-\frac{\rho_{\mathcal{L}}a_2}{2}T_0^{1-v}\right).\end{aligned}\tag{57}$$

Thus, for any $T=0,1,\ldots$, we have

$$\begin{aligned}&\left(1-\rho_{\mathcal{L}}\beta_T+\frac{\alpha_TL_1^2(1+2\alpha_TL_1)}{2n}+\frac{2(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L^2}{\rho_{\mathcal{L}}\beta_T}\right)^{T+1}\\&\leq\max\left\{\exp\left(-\frac{\rho_{\mathcal{L}}a_2}{2}T_0^{1-v}\right),\right.\\&\quad 1-\rho_{\mathcal{L}}a_2+\frac{a_1L_1^2(2a_1L_1+1)}{2n}+\frac{2(1+\rho_{\mathcal{L}}a_2)a_1^2L_1^2}{\rho_{\mathcal{L}}a_2},\ldots,\\&\quad\left.1-\frac{\rho_{\mathcal{L}}a_2}{T_0^v}+\frac{a_1L_1^2(2a_1L_1+T_0^u)}{2nT_0^{2u}}+\frac{2(T_0^v+\rho_{\mathcal{L}}a_2)a_1^2L^2}{\rho_{\mathcal{L}}a_2T_0^{2u}}\right\}.\end{aligned}\tag{58}$$

Hence, (56) together with (58) implies that there exists $G_0>1$ such that for any $T=0,1,\ldots$

$$1<\theta_1^{T+1}\leq G_0.\tag{59}$$

When $w\leq 0$, σ_k is decreasing, and then $\sigma_k\leq\sigma_0$ for any $k=0,\ldots,T$. When $w>0$, σ_k is increasing, and then $\sigma_k\leq\sigma_T$ for any $k=0,\ldots,T$. As a result, $\sigma_k\leq\max\{\sigma_0,\sigma_T\}$ for any $k=0,\ldots,T$. Hence, by the definition of $\theta_{k,2}$ in (53), $\theta_{k,2}\leq\max\{\theta_{0,2},\theta_{T,2}\}$ for any $k=0,\ldots,T$. This helps us to obtain that

$$\begin{aligned}\sum_{k=0}^{T}\theta_1^{T-k}\theta_{k,2}&\leq\sum_{k=0}^{T}\theta_1^{T+1}\theta_{k,2}\\&\leq(T+1)\max\{\theta_{0,2},\theta_{T,2}\}\theta_1^{T+1}.\end{aligned}\tag{60}$$

Note that

$$\begin{aligned}\max\{\theta_{0,2},\theta_{T,2}\}&=\frac{(L_1+2)nr\beta_T^2}{2}(\Delta^2+\max\{\sigma_0^2,\sigma_T^2\})\\&\quad+\frac{\alpha_T^2\sigma_\ell^2(2n+L_1)}{2\gamma_T}+2\alpha_T^2L_1^2M^*\\&\quad+\frac{4n(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1M^*}{\rho_{\mathcal{L}}\beta_T}\\&=O\left(\frac{1}{(T+1)^{2v-2\max\{w,0\}}}+\frac{1}{(T+1)^{2u-v}}\right).\end{aligned}\tag{61}$$

Then, by $2u-v>1$ and $\frac{1}{2}+\max\{w,0\}<v$ in Assumption 4, substituting (61) into (60) implies

$$\sum_{k=0}^{T}\theta_1^{T-k}\theta_{k,2}=O\left(\frac{1}{(T+1)^{2v-2\max\{w,0\}-1}}+\frac{1}{(T+1)^{2u-v-1}}\right).\tag{62}$$

Thus, by (62) there exists $G_0'>0$ such that for any $T=0,1,\ldots$

$$\sum_{k=0}^{T}\theta_1^{T-k}\theta_{k,2}\leq G_0'.\tag{63}$$

By (55), (59), and (63) we have

$$\begin{aligned}&\mathbb{E}(\|Y_{T+1}\|^2+F(\bar{x}_{T+1})-F^*)\\&\leq\theta_1^{T+1}(\|Y_0\|^2+F(\bar{x}_0)-F^*)+\sum_{k=0}^{T}\theta_1^{T-k}\theta_{k,2}\\&\leq G_0(\|Y_0\|^2+F(\bar{x}_0)-F^*)+G_0'.\end{aligned}$$

Let $G_1=G_0(\|Y_0\|^2+F(\bar{x}_0)-F^*)+G_0'$. Then, there exists $G_1\geq 0$ such that $\mathbb{E}(F(\bar{x}_T)-F^*)\leq G_1$ for any $T=0,1,\ldots$.

Step 4: At this step, we prove $\lim_{T\to\infty}\mathbb{E}\|Y_{T+1}\|^2=0$. By Step 3, since there exists $G_1\geq 0$ such that $\mathbb{E}(F(\bar{x}_T)-F^*)\leq G_1$ for any $T=0,1,\ldots$, by (40), we have

$$\begin{aligned}\mathbb{E}\|Y_{k+1}\|^2\leq&\left(1-\rho_{\mathcal{L}}\beta_T+\frac{2(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1^2}{\rho_{\mathcal{L}}\beta_T}\right)\mathbb{E}\|Y_k\|^2\\&+\frac{4n(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1(G_1+M^*)}{\rho_{\mathcal{L}}\beta_T}+\frac{n\alpha_T^2\sigma_\ell^2}{\gamma_T}\\&+nr\beta_T^2(\Delta^2+\sigma_k^2).\end{aligned}\tag{64}$$

Let

$$\theta_3=1-\rho_{\mathcal{L}}\beta_T+\frac{2(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1^2}{\rho_{\mathcal{L}}\beta_T}\tag{65}$$

$$\begin{aligned}\theta_{k,4}=&\frac{4n(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1(G_1+M^*)}{\rho_{\mathcal{L}}\beta_T}+\frac{n\alpha_T^2\sigma_\ell^2}{\gamma_T}\\&+nr\beta_T^2(\Delta^2+\sigma_k^2).\end{aligned}\tag{66}$$

Then, substituting (65) and (66) into (64) and iteratively computing (64) gives

$$\mathbb{E}\|Y_{k+1}\|^2\leq\theta_3^{k+1}\|Y_0\|^2+\sum_{m=0}^{k}\theta_3^{k-m}\theta_{m,4}.\tag{67}$$

Note that by the definition of θ_3 in (65) and $2u-v>1$, $\frac{1}{2}+\max\{w,0\}<v<u<1$ in Assumption 4, we have

$$\frac{1}{1-\theta_3}=\frac{1}{\rho_{\mathcal{L}}\beta_T-\frac{2(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1^2}{\rho_{\mathcal{L}}\beta_T}}=O((T+1)^v)\tag{68}$$

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

$$\max\{\theta_{0,4},\theta_{T,4}\} = \frac{4n(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2 L_1(G_1+M^*)}{\rho_{\mathcal{L}}\beta_T} + \frac{n\alpha_T^2\sigma_\ell^2}{\gamma_T} + nr\beta_T^2(\Delta^2+\max\{\sigma_T^2,\sigma_0^2\}) = O\left(\frac{1}{(T+1)^{2u-v}}+\frac{1}{(T+1)^{2v-2\max\{w,0\}}}\right). \tag{69}$$

Moreover, by the definition of $\theta_{k,4}$ in (66), $\theta_{k,4}\le\max\{\theta_{0,4},\theta_{T,4}\}$ for any $k=0,\ldots,T$. Then, it follows from (68) and (69) that

$$\sum_{k=0}^{T}\theta_3^{T-k}\theta_{k,4} \le \max\{\theta_{0,4},\theta_{T,4}\}\sum_{k=0}^{T}\theta_3^{T-k} = \max\{\theta_{0,4},\theta_{T,4}\}\frac{1-\theta_3^{T+1}}{1-\theta_3} = O\left(\frac{\max\{\theta_{0,4},\theta_{T,4}\}}{1-\theta_3}\right) = O\left(\frac{1}{(T+1)^{2u-2v}}+\frac{1}{(T+1)^{v-2\max\{w,0\}}}\right). \tag{70}$$

Meanwhile, by (57) we have

$$\theta_3^{T+1} \le \left(1-\rho_{\mathcal{L}}\beta_T+\frac{\alpha_T L_1^2(1+2\alpha_T L_1)}{2n}+\frac{2(1+\rho_{\mathcal{L}}\beta_T)\alpha_T^2L_1^2}{\rho_{\mathcal{L}}\beta_T}\right)^{T+1} = O\left(\left(1-\frac{\rho_{\mathcal{L}}\beta_T}{2}\right)^{T+1}\right) = O\left(\exp\left((T+1)\ln\left(1-\frac{\rho_{\mathcal{L}}\beta_T}{2}\right)\right)\right) = O\left(\exp\left(-\frac{\rho_{\mathcal{L}}a_2}{2}(T+1)^{1-v}\right)\right). \tag{71}$$

Let $k=T$ in (67). Then, substituting (70) and (71) into (67) implies $\mathbb{E}\|Y_{T+1}\|^2=O(\exp(-\frac{\rho_{\mathcal{L}}a_2}{2}(T+1)^{1-v}))+O(\frac{1}{(T+1)^{2u-2v}}+\frac{1}{(T+1)^{v-2\max\{w,0\}}})=O(\frac{1}{(T+1)^{2u-2v}}+\frac{1}{(T+1)^{v-2\max\{w,0\}}})$. Hence, we have $\lim_{T\to\infty}\mathbb{E}\|Y_{T+1}\|^2=0$.

Step 5: At this step, we give the estimation of $\sum_{k=0}^{T}\mathbb{E}\|Y_k\|^2$ for any $T=0,1,\ldots$. By defining $\sum_{k=1}^{0}\sum_{m=0}^{k}\theta_3^{k-m}\theta_{m,4}=0$, summing (67) from $k=0$ to T gives

$$\sum_{k=0}^{T}\mathbb{E}\|Y_k\|^2 \le \sum_{k=0}^{T}\theta_3^k\|Y_0\|^2+\sum_{k=1}^{T}\sum_{m=0}^{k}\theta_3^{k-m}\theta_{m,4}. \tag{72}$$

Then, it follows from (68) that

$$\sum_{k=0}^{T}\theta_3^k\|Y_0\|^2 = \frac{1-\theta_3^{T+1}}{1-\theta_3}\|Y_0\|^2 = O\left((T+1)^v\right). \tag{73}$$

Moreover, by (68)–(70), we have

$$\sum_{k=1}^{T}\sum_{m=0}^{k}\theta_3^{k-m}\theta_{m,4} \le \max\{\theta_{0,4},\theta_{T,4}\}\sum_{k=1}^{T}\sum_{m=0}^{k}\theta_3^{k-m} = \max\{\theta_{0,4},\theta_{T,4}\}\sum_{k=1}^{T}\frac{1-\theta_3^{k+1}}{1-\theta_3} = O\left(\frac{T\max\{\theta_{0,4},\theta_{T,4}\}}{1-\theta_3}\right) = O\left(\frac{1}{(T+1)^{2u-2v-1}}+\frac{1}{(T+1)^{v-2\max\{w,0\}-1}}\right). \tag{74}$$

Hence, substituting (73) and (74) into (72) implies

$$\sum_{k=0}^{T}\mathbb{E}\|Y_k\|^2 = O\left((T+1)^v+\frac{1}{(T+1)^{2u-2v-1}}+\frac{1}{(T+1)^{v-2\max\{w,0\}-1}}\right). \tag{75}$$

Step 6: Finally, we prove $\liminf_{T\to\infty}\mathbb{E}\|\nabla F(x_{i,T+1})\|^2=0$ for any $i\in\mathcal{V}$. From Step 3, since there exists $G_1\ge 0$ such that $\mathbb{E}(F(\bar{x}_T)-F^*)\le G_1$ for any $T=0,1,\ldots$, by Lemma A.1(ii) we have

$$\mathbb{E}\|\nabla F(\bar{x}_T)\|^2 \le 2L_1\mathbb{E}(F(\bar{x}_T)-F^*) \le 2L_1G_1. \tag{76}$$

Then, substituting (49) and (76) into (48) implies

$$\mathbb{E}(F(\bar{x}_{k+1})-F^*) \le \mathbb{E}(F(\bar{x}_k)-F^*)-\frac{\alpha_T}{2}\mathbb{E}\|\nabla F(\bar{x}_k)\|^2 + \frac{\alpha_T L_1^2(1+2\alpha_T L_1)}{2n}\mathbb{E}\|Y_k\|^2+\frac{\beta_T^2 nrL_1}{2}\left(\Delta^2+\sigma_k^2\right) + \frac{\alpha_T^2\sigma_\ell^2 L_1}{2\gamma_T}+2\alpha_T^2L_1^2G_1. \tag{77}$$

Note that (77) can be rewritten as

$$\frac{\alpha_T}{2}\mathbb{E}\|\nabla F(\bar{x}_k)\|^2 \le \mathbb{E}(F(\bar{x}_k)-F(\bar{x}_{k+1}))+\frac{\alpha_T L_1^2(1+2\alpha_T L_1)}{2n}\mathbb{E}\|Y_k\|^2 + \frac{\beta_T^2 nrL_1}{2}\left(\Delta^2+\sigma_k^2\right)+\frac{\alpha_T^2\sigma_\ell^2L_1}{2\gamma_T}+2\alpha_T^2L_1^2G_1. \tag{78}$$

Then, since $F^*\le F(x)$ holds for any $x\in\mathbb{R}^r$, summing (78) from $k=0$ to T gives

$$\frac{\alpha_T}{2}\sum_{k=0}^{T}\mathbb{E}\|\nabla F(\bar{x}_k)\|^2 \le \mathbb{E}(F(\bar{x}_0)-F(\bar{x}_{T+1}))+\frac{\alpha_T L_1^2(1+2\alpha_T L_1)}{2n}\sum_{k=0}^{T}\mathbb{E}\|Y_k\|^2 + \sum_{k=0}^{T}\left(\frac{\beta_T^2nrL_1}{2}\left(\Delta^2+\sigma_k^2\right)+\frac{\alpha_T^2\sigma_\ell^2L_1}{2\gamma_T}+2\alpha_T^2L_1^2G_1\right) \le \mathbb{E}(F(\bar{x}_0)-F^*)+\frac{\alpha_T L_1^2(1+2\alpha_T L_1)}{2n}\sum_{k=0}^{T}\mathbb{E}\|Y_k\|^2 + \sum_{k=0}^{T}\left(\frac{\beta_T^2nrL_1}{2}\left(\Delta^2+\sigma_k^2\right)+\frac{\alpha_T^2\sigma_\ell^2L_1}{2\gamma_T}+2\alpha_T^2L_1^2G_1\right). \tag{79}$$

By $\frac{1}{2}+\max\{w,0\}<v$ and $2u-v>1$ in Assumption 4, we have

$$\sum_{k=0}^{T}\left(\frac{\beta_T^2nrL_1}{2}\left(\Delta^2+\sigma_k^2\right)+\frac{\alpha_T^2\sigma_\ell^2L_1}{2\gamma_T}+2\alpha_T^2L_1^2G_1\right)$$

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

$$= O\left(\sum_{k=0}^{T}\left(\frac{1}{(T+1)^{2v-2\max\{w,0\}}}+\frac{1}{(T+1)^{2u}}\right)\right)$$
$$= O\left(\frac{1}{(T+1)^{2v-2\max\{w,0\}-1}}+\frac{1}{(T+1)^{2u-1}}\right). \quad (80)$$

Note that $2u-v>1$ and $\frac{1}{2}+\max\{w,0\}<v<u$ in Assumption 4. Then, we have $3u-2v-1=(2u-v-1)+(u-v)>0, u+v-2\max\{w,0\}-1>2v-2\max\{w,0\}-1>0$. For any $T=0,1,\ldots,$ substituting (75) and (80) into (79) implies

$$\alpha_T\sum_{k=0}^{T}\mathbb{E}\|\nabla F(\bar{x}_k)\|^2$$
$$= O\left(\frac{1}{(T+1)^{u-v}}+\frac{1}{(T+1)^{3u-2v-1}}+\frac{1}{(T+1)^{u+v-2\max\{w,0\}-1}}\right)$$
$$+O\left(\frac{1}{(T+1)^{2v-2\max\{w,0\}-1}}+\frac{1}{(T+1)^{2u-1}}\right)$$
$$+2(F(\bar{x}_0)-F(x^*)). \quad (81)$$

Thus, there exists $G_2>0$ such that $\alpha_T\sum_{k=0}^{T}\mathbb{E}\|\nabla F(\bar{x}_T)\|^2\le G_2$ for any $T=0,1,\ldots$.

Next, we prove $\liminf_{T\to\infty}\mathbb{E}\|\nabla F(\bar{x}_{T+1})\|^2=0$ by contradiction. Suppose there exists $G_3>0$ such that $\liminf_{T\to\infty}\mathbb{E}\|\nabla F(\bar{x}_{T+1})\|^2=G_3>0$. Then, there exists a positive integer T_1 such that $\mathbb{E}\|\nabla F(\bar{x}_T)\|^2\ge\frac{G_3}{2}$ for any $T=T_1,T_1+1,\ldots$. Thus, we have

$$\alpha_T\sum_{k=0}^{T}\mathbb{E}\|\nabla F(\bar{x}_k)\|^2\ge\alpha_T\sum_{k=T_1}^{T}\mathbb{E}\|\nabla F(\bar{x}_k)\|^2$$
$$\ge\frac{\alpha_T(T-T_1+1)G_3}{2}=O((T+1)^{1-u}). \quad (82)$$

Note that when T goes to infinity, $\alpha_T\sum_{k=0}^{T}\mathbb{E}\|\nabla F(\bar{x}_k)\|^2$ goes to infinity since the right-hand side of (82) goes to infinity, which contradicts (81). Then, we have $\liminf_{T\to\infty}\mathbb{E}\|\nabla F(\bar{x}_{T+1})\|^2=0$. Moreover, for any $i\in\mathcal{V}$, we have

$$\mathbb{E}\|\nabla F(x_{i,T+1})\|^2$$
$$=\mathbb{E}\|\nabla F(x_{i,T+1})-\nabla F(\bar{x}_{T+1})+\nabla F(\bar{x}_{T+1})\|^2$$
$$\le 2\mathbb{E}\|\nabla F(x_{i,T+1})-\nabla F(\bar{x}_{T+1})\|^2+2\mathbb{E}\|\nabla F(\bar{x}_{T+1})\|^2$$
$$\le 2L_1^2\mathbb{E}\|x_{i,T+1}-\bar{x}_{T+1}\|^2+2\mathbb{E}\|\nabla F(\bar{x}_{T+1})\|^2$$
$$\le 2L_1^2\mathbb{E}\|Y_{T+1}\|^2+2\mathbb{E}\|\nabla F(\bar{x}_{T+1})\|^2. \quad (83)$$

Therefore, by $\lim_{T\to\infty}\mathbb{E}\|Y_{T+1}\|^2=0$ in Step 4, $\liminf_{T\to\infty}\mathbb{E}\|\nabla F(x_{i,T+1})\|^2=0$ holds for any $i\in\mathcal{V}$. ■

REFERENCES

[1] M. Zhu and S. Martínez, "On distributed convex optimization under inequality and equality constraints," *IEEE Trans. Autom. Control*, vol. 57, no. 1, pp. 151–164, Jan. 2012.

[2] M. Zhu and S. Martínez, "An approximate dual subgradient algorithm for multi-agent non-convex optimization," *IEEE Trans. Autom. Control*, vol. 58, no. 6, pp. 1534–1539, Jun. 2013.

[3] 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.

[4] 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.

[5] R. Xin, U. A. Khan, and S. Kar, "A fast randomized incremental gradient method for decentralized nonconvex optimization," *IEEE Trans. Autom. Control*, vol. 67, no. 10, pp. 5150–5165, Oct. 2022.

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

[7] Z. Jiang, A. Balu, C. Hegde, and S. Sarkar, "Collaborative deep learning in fixed topology networks," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, Long Beach, CA, USA, 2017, pp. 5904–5914.

[8] Z. Zhang, Y. Zhang, D. Guo, S. Zhao, and X. Zhu, "Communication-efficient federated continual learning for distributed learning system with non-IID data," *Sci. China Inf. Sci.*, vol. 66, no. 2, 2023, Art. no. 122102.

[9] K. Ge, Y. Zhang, Y. Fu, Z. Lai, X. Deng, and D. Li, "Accelerate distributed deep learning with cluster-aware sketch quantization," *Sci. China Inf. Sci.*, vol. 66, no. 6, 2023, Art. no. 162102.

[10] K. Lu, H. Wang, H. Zhang, and L. Wang, "Convergence in high probability of distributed stochastic gradient descent algorithms," *IEEE Trans. Autom. Control*, vol. 69, no. 4, pp. 2189–2204, Apr. 2024.

[11] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. 30th Int. Conf. Mach. Learn.*, Atlanta, USA, 2013, pp. 1310–1318.

[12] M. Fazel, R. Ge, S. M. Kakade, and M. Mesbahi, "Global convergence of policy gradient methods for the linear quadratic regulator," in *Proc. 35th Int. Conf. Mach. Learn.*, Stockholm, Sweden, 2018, pp. 1467–1476.

[13] R. Ge, F. Huang, C. Jin, and Y. Yuan, "Escaping from saddle points—online stochastic gradient for tensor decomposition," in *Proc. 28th Conf. Learn. Theory*, Paris, France, 2015, pp. 797–842.

[14] M. G. Rabbat and R. D. Nowak, "Quantized incremental algorithms for distributed optimization," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 798–808, Apr. 2005.

[15] L. Chen, W. Liu, Y. Chen, and W. Wang, "Communication-efficient design for quantized decentralized federated learning," *IEEE Trans. Signal Process.*, vol. 72, pp. 1175–1188, 2024.

[16] F. Faghri, I. Tabrizian, I. Markov, D. Alistarh, D. M. Roy, and A. Ramezani-Kebrya, "Adaptive gradient quantization for data-parallel SGD," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2020, pp. 3174–3185.

[17] A. Reisizadeh, H. Taheri, A. Mokhtari, H. Hassani, and R. Pedarsani, "Robust and communication-efficient collaborative learning," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2019, pp. 8388–8399.

[18] J. F. Zhang, J. W. Tan, and J. Wang, "Privacy security in control systems," *Sci. China Inf. Sci.*, vol. 64, no. 7, 2021, Art. no. 176201.

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

[20] 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.

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

[22] Y. Lu and M. Zhu, "On privacy preserving data release of linear dynamic networks," *Automatica*, vol. 115, 2020, Art. no. 108839.

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

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

[25] 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.

[26] X. K. Liu, J. F. Zhang, and J. Wang, "Differentially private consensus algorithm for continuous-time heterogeneous multi-agent systems," *Automatica*, vol. 122, 2020, Art. no. 109283.

[27] J. Wang, J. F. Zhang, and X. He, "Differentially private distributed algorithms for stochastic aggregative games," *Automatica*, vol. 142, 2022, Art. no. 110440.

[28] X. Chen, C. Wang, Q. Yang, T. Hu, and C. Jiang, "Locally differentially private high-dimensional data synthesis," *Sci. China Inf. Sci.*, vol. 66, no. 1, 2023, Art. no. 112101.

[29] J. Wang, J. Ke, and J. F. Zhang, "Differentially private bipartite consensus over signed networks with time-varying noises," *IEEE Trans. Autom. Control*, vol. 69, no. 9, pp. 5788–5803, Sep. 2024.

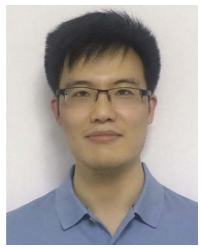
[30] X. Zhang, M. M. Khalili, and M. Liu, "Improving the privacy and accuracy of ADMM-based distributed algorithms," in *Proc. 35th Int. Conf. Mach. Learn.*, Stockholm, Sweden, 2018, pp. 5796–5805.

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.

[31] 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.
[32] Z. Huang, R. Hu, Y. Guo, E. Chan-Tin, and Y. Gong, "DP-ADMM: ADMM-based distributed learning with differential privacy," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 1002–1012, 2020.
[33] 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.
[34] C. Liu, K. H. Johansson, and Y. Shi, "Distributed empirical risk minimization with differential privacy," *Automatica*, vol. 162, 2024, Art. no. 111514.
[35] 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.
[36] Y. Wang and T. Başar, "Decentralized nonconvex optimization with guaranteed privacy and accuracy," *Automatica*, vol. 150, 2023, Art. no. 110858.
[37] J. Ding, G. Liang, J. Bi, and M. Pan, "Differentially private and communication efficient collaborative learning," in *Proc. AAAI Conf. Artif. Intell.*, Palo Alto, CA, USA, 2021, pp. 7219–7227.
[38] G. Yan, T. Li, K. Wu, and L. Song, "Killing two birds with one stone: Quantization achieves privacy in distributed learning," *Digit. Signal Process.*, vol. 146, 2024, Art. no. 104353.
[39] Y. Wang and T. Başar, "Quantization enabled privacy protection in decentralized stochastic optimization," *IEEE Trans. Autom. Control*, vol. 68, no. 7, pp. 4038–4052, 2023.
[40] G. Cybenko, "Dynamic load balancing for distributed memory multiprocessors," *J. Parallel Distrib. Comput.*, vol. 7, no. 2, pp. 279–301, 1989.
[41] D. Blatt and A. Hero, "Distributed maximum likelihood estimation for sensor networks," in *Proc. Int. Conf. Acoust. Speech Signal Process.*, 2004, pp. 929–932.
[42] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, "Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, Long Beach, CA, USA, 2017, pp. 5330–5340.
[43] S. Bubeck, "Convex optimization: Algorithms and complexity," *Found. Trends Theor. Comput. Sci.*, vol. 8, no. 3–4, pp. 231–357, 2015.
[44] T. C. Aysal, M. J. Coates, and M. G. Rabbat, "Distributed average consensus with dithered quantization," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 4905–4918, Oct. 2008.
[45] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2019, pp. 14774–14784.
[46] K. G. Murty and S. N. Kabadi, "Some NP-complete problems in quadratic and nonlinear programming," *Math. Prog.*, vol. 39, no. 2, pp. 117–129, 1987.
[47] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition," in *Proc. Mach. Learn. Knowl. Discov. Databases Eur. Conf.*, Riva del Garda, Italy, 2016, pp. 795–811.
[48] J. Chen, J. Wang, and J. F. Zhang, "Differentially private distributed nonconvex stochastic optimization with quantized communication," 2025, *arXiv:2403.18254v3*.
[49] Y. LeCun, C. Cortes, and C. J. C. Burges, "The MNIST database of handwritten digits," National Institute of Standards and Technology, 1998. [Online]. Available: http://yann.lecun.com/exdb/mnist/
[50] A. Krizhevsky, V. Nair, and G. Hinton, "Canadian institute for advanced research, 10 classes," Department of Computer Science, University of Toronto, Toronto, ON, Canada, 2009. [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html
[51] A. Krizhevsky, V. Nair, and G. Hinton, "Canadian institute for advanced research, 100 classes," Department of Computer Science, University of Toronto, Toronto, ON, Canada, 2009. [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html
[52] A. Krizhevsky, "Learning multiple layers of features from tiny images," M. S. thesis, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, CA, USA, 2009. [Online]. Available: http://www.cs.utoronto.ca/kriz/learning-features-2009-TR.pdf
[53] Y. S. Chow and H. Teicher, "ntegration in a probability space," in *Probability Theory: Independence, Interchangeability, Martingales*, New York, NY, USA: Springer, 2012, ch. 4, sec. 1, pp. 84–92.
[54] R. A. Horn and C. R. Johnson, "Hermitian matrices, symmetric matrices, and congruences," in *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2012, ch. 4, sec. 2, pp. 234–239.
[55] V. A. Zorich, "Integration," in *Mathematical Analysis I*. Berlin, German: Springer, 2015, ch. 6, sec. 2, pp. 349–360.

Jialong Chen received the B.S. degree in mathematics and applied mathematics from Beihang University, Beijing, China, in 2023. He is currently working toward the Ph.D. degree majoring in system theory with the Academy of Mathematics and Systems Science, Chinese Academy of Science, Beijing.

His research focuses on privacy and security in stochastic systems.



Jimin Wang (Member, IEEE) received the B.S. degree in mathematics from Shandong Normal University, Jinan, China, in 2012, and the Ph.D. degree in operational research and cybernetics from the School of Mathematics, Shandong University, Jinan, in 2018.

From May 2017 to May 2018, he was a joint Ph.D. student with the School of Electrical Engineering and Computing, The University of Newcastle, Callaghan, NSW, Australia. From July 2018 to December 2020, he was a Postdoctoral Researcher with the Institute of Systems Science, Chinese Academy of Sciences, Beijing, China. He is currently an Associate Professor with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing. His current research interests include privacy and security in cyber-physical systems, stochastic systems, and networked control systems.

Dr. Wang is a Member of IEEE CSS Technical Committee on Security and Privacy, IEEE CSS Technical Committee on Networks and Communication Systems, and the IFAC Technical Committee 1.5 on Networked Systems. He was the receipient of IEEE CSS Beijing Chapter Young Author Prize. He was an Associate Editor for *Systems and Control Letters* and *Journal of Automation and Intelligence*.

Ji-Feng Zhang (Fellow, IEEE) received the B.S. degree in mathematics from Shandong University, Jinan, China, in 1985, and the Ph.D. degree in operational research and cybernetics from the Institute of Systems Science, Chinese Academy of Sciences (CAS), Beijing, China, in 1991.

He is currently with the School of Automation and Electrical Engineering, Zhongyuan University of Technology, Zhengzhou, China, and the State Key Laboratory of Mathematical Sciences, Academy of Mathematics and Systems Science, CAS. His current research interests include system modeling, adaptive control, stochastic systems, and multiagent systems.

Dr. Zhang is an IFAC Fellow, CAA Fellow, and SIAM Fellow, Member of the European Academy of Sciences and Arts, and Academician of the International Academy for Systems and Cybernetic Sciences. He was the recipient of the Second Prize of the State Natural Science Award of China, in 2010 and 2015, respectively. He was a Vice-President of the Chinese Association of Automation, the Chinese Mathematical Society, and the Systems Engineering Society of China. He was a Vice-Chair of the IFAC Technical Board, Member of the Board of Governors, IEEE Control Systems Society, Convenor of Systems Science Discipline, Academic Degree Committee of the State Council of China. He was the Editor-in-Chief, Deputy Editor-in-Chief, or Associate Editor for more than 10 journals, including *Science China Information Sciences*, IEEE TRANSACTIONS ON AUTOMATIC CONTROL, *SIAM Journal on Control and Optimization,* etc.

Authorized licensed use limited to: CAS Academy of Mathematics & Systems Science. Downloaded on December 31,2025 at 00:20:51 UTC from IEEE Xplore. Restrictions apply.