Quantized Distributed Nonconvex Optimization Algorithms With Linear Convergence Under the Polyak–Łojasiewicz Condition

Lei Xu, *Member, IEEE*, Xinlei Yi, *Member, IEEE*, Jiayue Sun, *Member, IEEE*, Yang Shi, *Fellow, IEEE*, Karl H. Johansson, *Fellow, IEEE*, and Tao Yang, *Senior Member, IEEE*

***Abstract*—This article considers distributed optimization for minimizing the average of local nonconvex cost functions, by using local information exchange over undirected communication networks. To reduce the required communication capacity, we introduce an encoder–decoder scheme. By integrating it with distributed gradient tracking and proportional integral algorithms, respectively, we then propose two quantized distributed nonconvex optimization algorithms. Assuming the global cost function satisfies the Polyak–Łojasiewicz condition, which does not require the global cost function to be convex and the global minimizer is not necessarily unique, we show that our proposed algorithms linearly converge to a global optimal point. Moreover, we show that a low data rate is sufficient to guarantee linear convergence when the algorithm parameters are properly chosen. The theoretical results are illustrated by numerical examples.**

***Index Terms*—Distributed nonconvex optimization, linear convergence, Polyak–Łojasiewicz (P–L) condition, quantized communication.**

I. Introduction

Distributed optimization, which can be traced back to [1] and [2], has received a growing and renewed interest over the last decade due to its wide applications in resource allocation, machine learning, and sensor networks, just to name a few. Various distributed optimization algorithms have been developed. For an overview, see recent survey papers [3], [4]. The basic convergence results of distributed optimization algorithms guarantee *sublinear* convergence to an optimal point when the local cost functions are convex, e.g., [5], [6], [7], [8]. When the local cost functions are strongly convex and smooth, linear convergence results are established in [9], [10], [11], [12], and [13].

The aforementioned studies focus on the perfect communication. Since communication channels have limited bandwidth, distributed optimization algorithms with quantized communications have been developed. Early works on quantization for single-agent systems are given in [14] and [15]. Extensions have been proposed to deal with the distributed consensus problem with limited communication data rate, e.g., [16], [17], [18], and[19]. Recent research focuses on quantized distributed optimization. For the convex case, the authors in [20] and [21] proposed a quantized distributed incremental and subgradient algorithm, respectively. These algorithms *sublinearly* converge to a neighborhood around the optimal point. Pu et al. [22] developed a quantized distributed accelerated gradient algorithm and established linear convergence to a neighborhood around the optimal point. Yuan et al. [23] proposed a distributed dual averaging method with quantized communication by using a probabilistic quantizer, and demonstrated that the proposed algorithm *sublinearly* converges to the optimal point in expectation.

Recently, focusing on the strongly convex case, a few studies proposed quantized distributed algorithms that converge to the exact optimal point. For example, Yi and Hong [24] designed a quantized distributed algorithm by integrating the distributed subgradient algorithm and the uniform quantization, while the authors in [25] and [26] developed quantized distributed gradient algorithms by using the random quantizer and the sign of the relative state. Xiong et al. [27] proposed a quantized distributed mirror descent algorithm using time-varying quantizers. These algorithms, however, only have *sublinear* convergence rates. The authors in [28] and [29] proposed quantized distributed algorithms by equipping the distributed gradient tracking algorithm (DGTA) with the uniform quantizer, and established linear convergence to the exact optimal point for undirected and directed graphs, respectively.

Note that the aforementioned distributed algorithms with linear convergence to the exact optimal point only focus on strongly convex local cost functions. However, as demonstrated in [30, Lemma 2], the cost function of the LQR problem in reinforcement learning is quadratic and satisfies the Polyak–Łojasiewicz (P–Ł) condition under appropriate system dynamics and control laws. Moreover, in deep learning, [31, Th. 4] demonstrated that certain wide neural networks satisfy the P–Ł condition. This motivates us to consider the P–Ł condition case.

The main contributions are summarized as follows.

1) We propose quantized distributed algorithms by integrating the encoder–decoder scheme and the uniform quantizer with the DGTA and distributed proportional integral algorithm, respectively.

Received 20 August 2024; revised 27 February 2025; accepted 14 April 2025. Date of publication 22 April 2025; date of current version 29 September 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62133003, Grant 61991403, and Grant 61991400, in part by the Shanghai Municipal Science and Technology Major under Grant 2021SHZDZX0100, in part by the Knut and Alice Wallenberg Foundation, in part by the Swedish Foundation for Strategic Research, and in part by the Swedish Research Council. A preliminary version of this article has been accepted for the 61st IEEE Conference on Decision and Control, Cancun, Mexico, December 6–9, 2022 [DOI: 10.1109/CDC51059.2022.9992989]. Recommended by Associate Editor G. Notarstefano. *(Corresponding author: Tao Yang.)*

Lei Xu is with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, China, and also with the Department of Mechanical Engineering, University of Victoria, Victoria, BC V8W 2Y2, Canada (e-mail: 2010345@stu.neu.edu.cn).

Xinlei Yi is with the Shanghai Institute of Intelligent Science and Technology, Tongji University, Shanghai 201804, China (e-mail: xinleiyi@tongji.edu.cn).

Jiayue Sun and Tao Yang are with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, China (e-mail: sunjiayue@ise.neu.edu.cn; yangtao@mail.neu.edu.cn).

Yang Shi is with the Department of Mechanical Engineering, University of Victoria, Victoria, BC V8W 2Y2, Canada (e-mail: yshi@uvic.ca).

Karl H. Johansson is with the Division of Decision and Control Systems, School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, 100 44 Stockholm, Sweden (e-mail: kallej@kth.se).

Digital Object Identifier 10.1109/TAC.2025.3563431

TABLE I
COMPARISON OF DIFFERENT QUANTIZED DISTRIBUTED ALGORITHMS

Existing results in references	Exact solution	Linear convergence	Nonconvex (Local cost functions)
[20], [21], [32]	No	No	No
[23]–[27], [33]	Yes	No	No
[22], [34]	No	Yes	No
[28], [29], [35]	Yes	Yes	No
This paper	Yes	Yes	Yes

2) Assuming that the global cost function satisfies the P–Ł condition, Theorems 1 and 3 show that the proposed algorithms linearly converge to an exact global optimal point when the quantization level is larger than a certain threshold. This is more general than the existing results in [34], [28], [29], and [35], which require that each local cost function is strongly convex. Table I summarizes the comparison between this article and related studies.
3) Moreover, Theorems 2 and 4 show that the proposed algorithms with an arbitrary quantization level can still converge linearly to an exact global optimal point provided that the algorithm parameters are properly chosen. It is worth noting that [36] proposed the BEER algorithm for distributed nonconvex optimization under compressed communication. Unlike the randomized compression operator used in [36], our proposed algorithms employ a uniform quantizer, allowing even 1-bit data rate, which is most communication efficient.

Compared to the conference version [37], this article proposes not only the quantized distributed proportional integral algorithm (DPIA) but also the quantized DGTA. All the proofs are omitted due to the space limitation and can be found in [38].

The rest of this article is organized as follows. Section II presents the problem formulation. Section III introduces an encoder–decoder scheme for quantized communication. In Sections IV and V, we propose the distributed gradient tracking and DPIAs with finite data rates and analyze their convergence, respectively. Section VI presents numerical simulation examples. Finally, Section VII conclude this article.

Notation: Let $\mathbf{1}_n$ (or $\mathbf{0}_n$) be the $n \times 1$ vector with all ones (or zeros), and $\mathbf{I}_n$ be the n-dimensional identity matrix. $\|\cdot\|$ is the Euclidean vector norm or spectral matrix norm. For a column vector $x = (x_1, \dots, x_m)$, $\|x\|_\infty = \max_{1\le i\le m} |x_i|$. For a positive semidefinite matrix $\mathcal{M}$, $\rho(\mathcal{M})$ and $\underline{\rho}(\mathcal{M})$ are the spectral radius and the minimum positive eigenvalue of matrix $\mathcal{M}$, respectively. The minimum integer greater than or equal to c is denoted by $\lceil c \rceil$. Let $\mathrm{diag}[a_1, \dots, a_n]$ denote a diagonal matrix with the ith diagonal element being a_i. Given any differentiable function f, ∇f is the gradient of f. $A \otimes B$ denotes the Kronecker product of matrices A and B. $A \preceq B$ if all entries of matrix $A - B$ are not greater than zero, and $A \succ 0$ if all entries of matrix A are greater than zero.

II. PROBLEM FORMULATION

Consider a group of n agents distributed over an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, where $\mathcal{V} = \{1, 2, \dots, n\}$ is the vertex set and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. $(i, j) \in \mathcal{E}$ indicates that agents i and j can communicate with each other, and $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{n\times n}$ is the adjacency matrix, where $a_{ij} > 0$ if $(j, i) \in \mathcal{E}$, otherwise $a_{ij} = 0$. Let $\mathcal{N}_i = \{j \in \mathcal{V} : a_{ij} > 0\}$ and $d_i = \sum_{j=1}^n a_{ij}$ denote the neighbor set and weighted degree of agent i, respectively. The degree matrix is defined as $\mathcal{D} = \mathrm{diag}[d_1, \dots, d_n]$. The graph Laplacian matrix is $L = [L_{ij}] = \mathcal{D} - \mathcal{A}$. A path from agent i_1 to agent i_k is a sequence of agents $\{i_1, \dots, i_k\}$ such that $(i_j, i_{j+1}) \in \mathcal{E}$ for $j = 1, \dots, k-1$. An undirected graph is connected if there exists a path between any pair of distinct agents.

Assume that each agent has a local nonconvex cost function $f_i : \mathbb{R}^m \to \mathbb{R}$. All agents collaborate to solve the following optimization problem:

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

Throughout this article, we make the following assumptions.

Assumption 1: The undirected graph $\mathcal{G}$ is connected.

Assumption 2: Each local nonconvex cost function $f_i(x)$ is smooth with constant $L_f > 0$, i.e.,

$$\|\nabla f_i(x) - \nabla f_i(y)\| \le L_f\|x - y\|, \tilde{}\forall x, y \in \mathbb{R}^m. \tag{2}$$

Assumption 3: The optimal set $\mathbb{X}^\star = argmin_{x\in\mathbb{R}^m} f(x)$ is nonempty and $f^\star = \min_{x\in\mathbb{R}^m} f(x) > -\infty$.

Assumption 4: The global cost function $f(x)$ satisfies the P–L condition with constant $\nu > 0$, i.e.,

$$\frac{1}{2}\|\nabla f(x)\|^2 \ge \nu(f(x) - f^\star),\ \forall x \in \mathbb{R}^m. \tag{3}$$

Remark 1: Assumptions 1–3 are common in the literature, e.g., [3] and [4]. Assumption 4 is weaker than strong convexity, and the global minimizer is not necessarily unique, but every stationary point is a global minimizer. Every strongly convex function satisfies the P–Ł condition.

The objective of this article is to propose quantized distributed optimization algorithms with linear convergence under the P–Ł condition.

III. ENCODER–DECODER SCHEME FOR QUANTIZED COMMUNICATION

In this section, we introduce an encoder–decoder scheme.

To begin with, consider a uniform quantizer $q[a]$ with $2\mathcal{K} + 1$ quantization levels [39], i.e.,

$$q[a] = \begin{cases} j, & \frac{2j-1}{2} < a \le \frac{2j+1}{2},\ j = 0, \dots, \mathcal{K} \\ \mathcal{K}, & \frac{2\mathcal{K}+1}{2} < a \\ -q[-a], & a \le -\frac{1}{2}. \end{cases} \tag{4}$$

Next for $l = [l_1, l_2, \dots, l_m] \in \mathbb{R}^m$, define $Q[l] = (q[l_1], \dots, q[l_m])$. The quantizer $Q[l]$ is not saturated if $\|l\|_\infty \le \mathcal{K} + \frac{1}{2}$. In this case, the quantization error is bounded, i.e.,

$$\|l - Q[l]\|_\infty \le \frac{1}{2}. \tag{5}$$

Next, we introduce an encoder–decoder pair [28], [35] for agents to communicate with each other. First, the following encoder scheme is used to quantize the variable to be transmitted.

Then, the following decoder scheme is used to recover the variable to be received.

Encoder

For the vector $\mathcal{C}_j(k) \in \mathbb{R}^m$ that requires quantization, agent $j \in \mathcal{V}$ recursively generates the m-dimensional quantized output $z_j^{\mathcal{C}}(k)$, and internal state $b_j^{\mathcal{C}}(k)$ as follows: for any $k \geq 1$,

$$z_j^{\mathcal{C}}(k) = Q\left[\frac{1}{s(k-1)}(\mathcal{C}_j(k) - b_j^{\mathcal{C}}(k-1))\right], \quad (6a)$$

$$b_j^{\mathcal{C}}(k) = s(k-1)z_j^{\mathcal{C}}(k) + b_j^{\mathcal{C}}(k-1), \quad (6b)$$

where the initial value $b_j^{\mathcal{C}}(0) = \mathbf{0}_m$, scaling function $s(k) = s(0)\mu^k > 0$ is a decreasing sequence used to adaptively adjust the encoder, and $\mu \in (0,1)$ is a positive constant.

Decoder

When agent $i \in \mathcal{N}_j$ receives the quantized data $z_j^{\mathcal{C}}(k)$ from agent j, it recursively generates an estimate $\hat{\mathcal{C}}_j(k)$ of $\mathcal{C}_j(k)$ by the following rule: for any $k \geq 1$,

$$\hat{\mathcal{C}}_j(k) = s(k-1)z_j^{\mathcal{C}}(k) + \hat{\mathcal{C}}_j(k-1), \quad (7)$$

where the initial value $\hat{\mathcal{C}}_j(0) = \mathbf{0}_m$.

Remark 2: From the encoder–decoder scheme, we note that $b_j^{\mathcal{C}}(k)$ is a predictor, $s(k)$ is used to adjust the prediction error $\mathcal{C}_j(k) - b_j^{\mathcal{C}}(k-1)$. Moreover, the initial value $s(0)$ requires to be large enough to guarantee that the quantizer is not saturated, which implies that the quantization error is bounded. The positive constant $\mu \in (0,1)$ ensures that the agent gradually improves the accuracy of the estimate for the transmitted variables from neighbors.

IV. DGTA With Finite Data Rates

In this section, we propose a DGTA with quantized communication by integrating the DGTA [40], [41] with the encoder–decoder scheme. More specifically, the quantized DGTA is given in Algorithm 1.

Before stating the main convergence results, we provide the following preliminary results, whose proofs are omitted due to the space limitation but can be found in [38].

The following lemma provides a sufficient condition to ensure that a certain linear matrix inequality holds, which plays a crucial role in ensuring convergence of the consensus error, gradient tracking error, optimization error, and the nonsaturation of the uniform quantizer.

Lemma 1: Suppose that Assumptions 1–4 hold. Suppose that the parameters β and δ satisfy

$$\beta \in \left(0, \frac{\sqrt{2}}{2\rho(L)}\right) \quad (9)$$

$$\delta \in \left(0, \min\left\{\frac{\sqrt{c_1\Theta_1}}{2}, \frac{1}{4L_f}, \frac{2}{\nu}, \frac{1}{8+2L_f}, \frac{\sqrt{c_1}}{8L_f}\right\}\right) \quad (10)$$

where

$$c_1 = \frac{(1-\varrho^2-c_2)(1-\varrho^2)}{1+\varrho^2}, \ \varrho = \rho(\mathbf{I}_{nm} - \beta\boldsymbol{L} - \boldsymbol{H})$$

$$\boldsymbol{L} = L \otimes \mathbf{I}_m, \ \boldsymbol{H} = \frac{1}{n}(\mathbf{1}_n\mathbf{1}_n^T \otimes \mathbf{I}_m)$$

$$c_2 \in (0, 1-\varrho^2), \ \Theta_1 = \min\left\{\frac{c_1}{24L_f^2}, \frac{\nu\Theta_2}{2L_f^2}\right\}, \ \Theta_2 = \frac{c_1}{32L_f^2}.$$

Algorithm 1: Quantized Distributed Gradient Tracking Algorithm.

For each agent $i \in \mathcal{V}$.
Initialization:
$x_i(0) \in \mathbb{R}^m$, $u_i(0) = \nabla f_i(x_i(0))$, $\hat{x}_j(0) = \hat{u}_j(0) = \mathbf{0}_m$.
for $k \geq 0$:
Update:

$$x_i(k+1) = x_i(k) - \beta\sum_{j=1}^n L_{ij}\hat{x}_j(k) - \delta u_i(k), \quad (8a)$$

$$u_i(k+1) = u_i(k) - \beta\sum_{j=1}^n L_{ij}\hat{u}_j(k) + \nabla f_i(x_i(k+1)) - \nabla f_i(x_i(k)), \quad (8b)$$

where β and δ are gain parameters.
Send:
The quantized outputs $z_i^x(k+1)$ and $z_i^u(k+1)$ generated by encoder (6) to its neighbors.
Receive:
The quantized outputs $z_j^x(k+1)$ and $z_j^u(k+1)$ generated by encoder (6) from its neighbors.
Compute:
The variables $\hat{x}_j(k+1)$ and $\hat{u}_j(k+1)$ generated by decoder (7).

Then, the following linear matrix inequality holds:

$$\Phi\Theta \preceq (1-\iota)\Theta \quad (11)$$

where $\iota = \min\left\{\frac{c_2}{2}, \frac{\delta}{4}\nu\right\}$, $\Theta = [\Theta_1, 1, \Theta_2]^T$ and the nonnegative matrix Φ is given by

$$\Phi = \begin{bmatrix}\chi_1 & \chi_2 & 0\\ \chi_3 & \chi_4 & \chi_5\\ \chi_6 & 0 & \chi_7\end{bmatrix} \quad (12)$$

where

$$\chi_1 = (1+\sigma_1)\varrho^2, \ \chi_2 = 2\delta^2\left(1+\frac{1}{\sigma_1}\right)$$

$$\chi_3 = \left(1+\frac{1}{\sigma_1}\right)8L_f^2\left(\beta^2\rho^2(L) + \frac{2\delta^2L_f^2}{1-2\delta L_f}\right)$$

$$\chi_4 = (1+\sigma_1)\varrho^2 + \left(1+\frac{1}{\sigma_1}\right)8L_f^2\delta^2$$

$$\chi_5 = \left(1+\frac{1}{\sigma_1}\right)\frac{16L_f^2\delta(2-\delta\nu)}{1-2\delta L_f}$$

$$\chi_6 = \frac{\delta}{2}L_f^2, \ \chi_7 = 1-\frac{\delta}{2}\nu, \ \sigma_1 = \frac{1-\varrho^2}{2\varrho^2}.$$

The following lemma establishes an upper bound for $\|\Phi^k\|$.

Lemma 2: Suppose that Assumptions 1–4 hold. Suppose that the parameters β and δ are given in Lemma 1. Then, the following inequality holds:

$$\|\Phi^k\| \leq h\rho^k(\Phi) \quad (13)$$

where $h = \sqrt{3}\frac{\max_{1\leq i\leq 3}\zeta_i}{\min_{1\leq i\leq 3}\zeta_i}$, $\zeta = [\zeta_1, \zeta_2, \zeta_3]^T$ is an eigenvector of Φ corresponding to the spectral radius $\rho(\Phi)$.

Denote $F(\boldsymbol{x}) = \sum_{i=1}^n f_i(x_i)$, $\boldsymbol{x}(k) = [x_1^T(k), \ldots, x_n^T(k)]^T$, $\boldsymbol{u}(k) = [u_1^T(k), \ldots, u_n^T(k)]^T$, $\bar{x}(k) = \frac{1}{n}(\mathbf{1}_n^T \otimes \mathbf{I}_m)\boldsymbol{x}(k)$, $\bar{u}(k) = \frac{1}{n}(\mathbf{1}_n^T \otimes \mathbf{I}_m)\boldsymbol{u}(k)$, $\bar{\boldsymbol{x}}(k) = \mathbf{1}_n \otimes \bar{x}(k)$, and $\bar{\boldsymbol{u}}(k) = \mathbf{1}_n \otimes \bar{u}(k)$. The following proposition shows that the quantizer in (6a) is never saturated by appropriately choosing the proposed algorithm's parameters if the quantization level is larger than a certain threshold.

Proposition 1 (Nonsaturation): Suppose that Assumptions 1–4 hold. Consider Algorithm 1 with parameters β and δ as given in Lemma 1. Then, for any

$$\mathcal{K} \geq \max\{\vartheta_1, \vartheta_2\} \tag{14}$$

where

$$\vartheta_1 = \sigma_3\sqrt{\frac{h\sigma_2}{4\mu^2(\mu^2-\bar{\rho})}} + \frac{(1+2\beta d)}{2\mu} - \frac{1}{2}$$

$$\vartheta_2 = \sigma_4\sqrt{\frac{h\sigma_2}{4\mu^2(\mu^2-\bar{\rho})}} + \frac{\sqrt{nm}L_f\beta\rho(L)}{2\mu} + \frac{(1+2\beta d)}{2\mu} - \frac{1}{2}$$

and

$$\mu \in (\sqrt{\bar{\rho}}, 1),\ \bar{\rho} = 1-\iota,\ d = \max\{d_1, \ldots, d_n\}$$

$$\sigma_2 = nm\sigma_5\sqrt{1+(1+4L_f^2)^2}$$

$$\sigma_3 = \sigma_6\sqrt{2+\frac{4\sigma_7}{\delta(1-2\delta L_f)}}$$

$$\sigma_4 = \beta\rho(L)(1+L_f) + L_f\delta\left(1+\sqrt{\frac{4\sigma_7}{\delta(1-2\delta L_f)}}\right)$$

$$\sigma_5 = \left(1+\frac{1}{\sigma_1}\right)2\beta^2\rho^2(L)$$

$$\sigma_6 = \sqrt{3}\max\{\beta\rho(L), \delta\}$$

$$\sigma_7 = \max\left\{1-\frac{\delta}{2}\nu, \frac{\delta}{2}L_f^2\right\}$$

the quantizer in (6a) is never saturated provided that

$$s(0) \geq \max\left\{\frac{2(C_x+\delta C_u)}{2\mathcal{K}+1}, \frac{2\|\nabla F(\boldsymbol{x}(0)-\delta\boldsymbol{u}(0))\|_\infty}{2\mathcal{K}+1}\right.$$
$$\left.\sqrt{\frac{4\|\Lambda(0)\|\mu^2(\mu^2-\bar{\rho})}{\sigma_2}}\right\} \tag{15}$$

where $C_x \geq \|\boldsymbol{x}(0)\|_\infty$, $C_u \geq \|\boldsymbol{u}(0)\|_\infty$, and $\Lambda(0) = [\|\boldsymbol{x}(0) - \bar{\boldsymbol{x}}(0)\|^2, \|\boldsymbol{u}(0) - \bar{\boldsymbol{u}}(0)\|^2, n(f(\bar{x}(0)) - f^\star)]^T$.

We are now ready to present the main convergence results.

Theorem 1 (High data rate): Suppose that Assumptions 1–4 hold. Let each agent $i \in \mathcal{V}$ run Algorithm 1 with the same β, δ, $\mathcal{K}$, μ, and $s(0)$ given in Proposition 1. Then

$$\|\boldsymbol{x}(k) - \bar{\boldsymbol{x}}(k)\|^2 + n(f(\bar{x}(k)) - f^\star) \leq \sigma_8\mu^{2k} \tag{16}$$

where $\sigma_8 = \frac{h\sigma_2 s^2(0)}{2\mu^2(\mu^2-\bar{\rho})}$.

Theorem 1 establishes linear convergence of the proposed algorithm provided that the quantization level is larger than a certain threshold given in (14). The following theorem establishes linear convergence for an arbitrarily low data rate, even one bit data rate, and thus is called low data rate theorem.

Theorem 2 (Low data rate): Suppose that Assumptions 1–4 hold. Let each agent $i \in \mathcal{V}$ run Algorithm 1 with $(\mu, \beta, \delta) \in \Pi$, where

$$\Pi = \left\{(\mu, \beta, \delta) : \mu \in (\sqrt{\bar{\rho}}, 1),\ \beta \in \left(0, \frac{\sqrt{2}}{2\rho(L)}\right)\right.$$
$$\delta \in \left(0, \min\left\{\frac{\sqrt{c_1\Theta_1}}{2}, \frac{1}{4L_f}, \frac{2}{\nu}, \frac{1}{8+2L_f}, \frac{\sqrt{c_1}}{8L_f}\right\}\right)$$
$$\left.\vartheta_1 \leq \mathcal{K},\ \vartheta_2 \leq \mathcal{K}\right\}.$$

Then, for any $\mathcal{K} \geq 1$ and $s(0)$ satisfying (15) in Proposition 1, Π is nonempty, and

$$\|\boldsymbol{x}(k) - \bar{\boldsymbol{x}}(k)\|^2 + n(f(\bar{x}(k)) - f^\star) \leq \sigma_8\mu^{2k}. \tag{17}$$

Remark 3: From (14), it can be observed that a smaller μ, i.e., a faster convergence speed, leads to a larger $\mathcal{K}$ and thus a larger quantization level, i.e., a larger communication bandwidth requirement. Moreover, a smaller μ requires fewer iterations to achieve a certain level of optimization accuracy. Note that in Theorem 2, linear convergence of Algorithm 1 is established even for 1-bit data rate, i.e., $\mathcal{K} = 1$, which is the most communication efficient.

Remark 4: For the strongly convex case, the authors in [28], [29], and [34] proposed quantized distributed algorithms with linear convergence. However, their analysis cannot be used for the P–Ł condition. For example, the linear system of inequalities in the above studies use $\|\bar{x}(k) - x^\star\|$, where $x^\star$ is the unique optimal solution which exists due to the strong convexity. In our case, the optimal solution is not unique due to the P–Ł condition. Therefore, we use $n(f(\bar{x}(k)) - f^\star)$, where $f^\star$ is the unique optimal value. Moreover, as shown in [28, Lemma 4.1] and [29, Lemma 8], the authors leveraged the strong convexity condition to directly apply [41, Lemma 10] for analyzing the upper bound of $\|\bar{x}(k) - x^\star\|$. However, this cannot be used for our case due to the lack of the strong convexity. Instead, we use the P–Ł condition to analyze the upper bound of $n(f(\bar{x}(k+1)) - f^\star)$ in [38, eq. (D.12)]. It turns out that the upper bound includes the term $\|x(k) - \bar{x}(k)\|^2$. We then use this term together with $\|u(k) - \bar{u}(k)\|^2$, and $n(f(\bar{x}(k+1)) - f^\star)$ as a state vector. By analyzing interrelationships of these terms (see [38, eqs. (D.4)–(D.14)]), we construct a novel linear system of inequalities as given by [38, eq. (D.25)]. Please refer to the proofs of Proposition 1 and Theorems 1 and 2 of [38] for the detailed analysis for linear convergence of Algorithm 1.

V. DPIA With Finite Data Rates

In this section, we propose a DPIA with quantized communication by integrating the DPIA [10], [11], [42] with the encoder–decoder scheme. More specifically, the quantized DPIA is given in Algorithm 2.

Remark 5: Algorithms 1 and 2 combine the DGTA [40], [41] and the DPIA [10], [42] with the quantization scheme [28], [35], respectively. The operational differences between Algorithms 1 and 2 stem from the differences between the distributed gradient tracking algorithm and the DPIA. It is known from [4], [10], and[40] that the distributed gradient tracking algorithm uses an auxiliary variable to track the average gradient and performs a distributed inexact gradient method, whereas the DPIA incorporates an integral feedback mechanism to correct errors caused by the distributed gradient descent method with a fixed step size.

Compared with Algorithm 1, which requires two parameters β and δ, Algorithm 2 requires three parameters ξ, φ, and σ. However, in Algorithm 1, at each iteration each agent i needs to communicate one additional m-dimensional variable besides the communication of $z_i^x(k)$ with its neighbors, which makes the quantization scheme more involved.

The following proposition provides a sufficient condition for the nonsaturation of the designed quantizer. The proof is based on the following Lyapunov candidate function:

$$W(k) = V(k) + n(f(\bar{x}(k)) - f^\star)$$

Algorithm 2: Quantized Distributed Proportional Integral Algorithm.

For each agent $i \in \mathcal{V}$.
Initialization:
$x_i(0) \in \mathbb{R}^m$, $\sum_{j=1}^n u_j(0) = \mathbf{0}_m$, $\hat{x}_j(0) = \mathbf{0}_m$.
for $k \geq 0$**:**
Update:

$$x_i(k+1) = x_i(k) - \xi \sum_{j=1}^n L_{ij}\hat{x}_j(k) - \varphi u_i(k) - \sigma \nabla f_i(x_i(k)), \tag{18a}$$

$$u_i(k+1) = u_i(k) + \varphi \sum_{j=1}^n L_{ij}\hat{x}_j(k), \tag{18b}$$

where $\sigma > 0$ is the fixed step-size, ξ and φ are gain parameters.
Send:
The quantized output $z_i^x(k+1)$ generated by encoder (6) to its neighbors.
Receive:
The quantized output $z_j^x(k+1)$ generated by encoder (6) from its neighbors.
Compute:
The variable $\hat{x}_i(k)$ generated by decoder (7).

$$V(k) = \boldsymbol{x}^T(k)\boldsymbol{K}\boldsymbol{x}(k) + 2\boldsymbol{x}^T(k)\boldsymbol{K}\boldsymbol{P}\left(\boldsymbol{u}(k) + \frac{\sigma}{\varphi}\boldsymbol{g}(k)\right) + \left(\boldsymbol{u}(k) + \frac{\sigma}{\varphi}\boldsymbol{g}(k)\right)^T \left(\frac{\varphi+\xi}{\varphi}\boldsymbol{P}\right)\left(\boldsymbol{u}(k) + \frac{\sigma}{\varphi}\boldsymbol{g}(k)\right) \tag{19}$$

where $\boldsymbol{P} = P \otimes \mathbf{I}_m$, $\boldsymbol{K} = K_n \otimes \mathbf{I}_m$ with P and K_n are given in Lemma 3, and $\boldsymbol{g}(k) = \nabla F(\boldsymbol{x}(k))$. The detailed proof is omitted due to the space limitation and can be found in [38].

Proposition 2 (Nonsaturation): Suppose that Assumptions 1–4 hold. Let each agent $i \in \mathcal{V}$ run Algorithm 2, and the parameters are given as follows:

$$\xi \in \left[\frac{5}{\underline{\rho}(L)}\varphi, \kappa_1\varphi\right], \ \varphi \in [\sigma\kappa_2, \sigma\kappa_3]$$

$$\sigma \in \left(0, \min\left\{\frac{\varepsilon}{\eta_1}, \frac{\varepsilon}{\eta_2}, \frac{2}{\nu}, \frac{1}{4L_f}\right\}\right)$$

where the parameters $\varepsilon \in \left(0, \min\left\{\frac{\kappa_2}{2} - 2 - 3L_f^2\kappa_1^2 - \frac{L_f^2}{2}, \kappa_2 - 1 - \frac{3L_f^2+8}{\underline{\rho}(L)}\right\}\right)$, $\kappa_1 > \frac{5}{\underline{\rho}(L)}$, $\kappa_2 > \max\left\{6L_f^2(\kappa_1+1)^2\kappa_1^2\rho(L), 4 + 6L_f^2\kappa_1^2 + L_f^2, 6L_f^2(\kappa_1+1)^2, 1 + \frac{3L_f^2+8}{\underline{\rho}(L)}\right\}$ and $\kappa_3 > \kappa_2$ with

$$\eta_1 = \kappa_3^2\rho(L) + \frac{2}{\underline{\rho}(L)} + 2\kappa_3^2\rho(L) + 3\kappa_3^2L_f^2\left(\frac{\kappa_1+1}{\kappa_2^2} + \frac{3}{2}\rho(L)\right)$$

$$\eta_2 = 4\kappa_1^2\kappa_3^2\rho^2(L) + 2(\kappa_3^2(\kappa_1+1)\rho(L) + 1 + \kappa_3^2) + 3\kappa_1^2L_f^2\left((\kappa_1+1)\rho(L) + \frac{3}{2}\kappa_3^2\rho^2(L)\right).$$

Then, for any

$$\mathcal{K} \geq \Omega \tag{20}$$

where

$$\Omega = \epsilon_1\sqrt{\frac{\epsilon_2 nm}{4\mu^2(\mu^2-\epsilon_3)}} + \frac{(1+2\xi d)}{2\mu} - \frac{1}{2}$$

$\mu \in (\sqrt{\epsilon_3}, 1)$, and

$$\epsilon_1 = \max\left\{\xi^2\rho^2(L), \frac{\varphi^3\rho(L)}{\varphi+\xi}, \xi\varphi\rho^2(L)\right\}$$

$$\epsilon_2 = \xi\rho(L) + 2\varphi\rho(L) + 4\xi^2\rho^2(L) + 2(\varphi(\xi+\varphi)\rho(L) + \sigma^2 + \varphi^2 + 2\varphi)$$

$$\epsilon_3 = 1 - \frac{\epsilon_4}{\epsilon_5}$$

$$\epsilon_4 = \min\left\{\epsilon_6, \epsilon_7, \frac{\sigma}{2}\nu\right\}$$

$$\epsilon_5 = \max\left\{\frac{\xi\underline{\rho}(L)+\varphi}{\xi\underline{\rho}(L)}, 1 + \frac{2\xi}{\varphi}\right\}$$

$$\epsilon_6 = \varphi - \frac{8\sigma}{\underline{\rho}(L)} - \frac{6\sigma^2\varphi^2L_f^2(\xi+\varphi)^2}{\varphi^5} - \frac{3\sigma L_f^2}{\underline{\rho}(L)} - \left(\varphi^2\rho(L) + \frac{2\sigma^2}{\underline{\rho}(L)} + 2\varphi^2\rho(L) + 3\varphi^2L_f^2\left(\frac{\sigma^2(\xi+\varphi)}{\varphi^3} + \frac{3}{2}\rho(L)\right)\right)$$

$$\epsilon_7 = \epsilon_8 - \frac{\sigma}{2}L_f^2$$

$$\epsilon_8 = \xi\underline{\rho}(L) - \frac{9\varphi}{2} - \sigma - \frac{6\sigma^2\xi^2L_f^2(\xi+\varphi)^2}{\varphi^5}\rho(L) - \frac{3\sigma L_f^2\xi^2}{\varphi^2} - \left(4\xi^2\rho^2(L) + 2(\varphi(\xi+\varphi)\rho(L) + \sigma^2 + \varphi^2) + 3\xi^2L_f^2\left(\frac{\sigma^2(\xi+\varphi)}{\varphi^3}\rho(L) + \frac{3}{2}\rho^2(L)\right)\right)$$

the quantizer in (6a) is never saturated provided that

$$s(0) \geq \max\left\{\frac{C_x + \varphi C_u + \sigma C_g}{\mathcal{K} + \frac{1}{2}}, \sqrt{\frac{4\mu^2(\mu^2-\epsilon_3)W(0)}{\epsilon_2 nm}}\right\} \tag{21}$$

where $C_x \geq \|\boldsymbol{x}(0)\|_\infty$, $C_u \geq \|\boldsymbol{u}(0)\|_\infty$, $C_g \geq \|\boldsymbol{g}(0)\|_\infty$.

We are now ready to present the main convergence results.

Theorem 3 (High data rate): Suppose that Assumptions 1–4 hold. Let each agent $i \in \mathcal{V}$ run Algorithm 2 with the same ξ, φ, σ, μ, $\mathcal{K}$, and $s(0)$ given in Proposition 2. Then

$$\|\boldsymbol{x}(k) - \bar{\boldsymbol{x}}(k)\|^2 + n(f(\bar{x}(k)) - f^\star) \leq \epsilon_9\mu^{2k} \tag{22}$$

where $\epsilon_9 = \frac{nm\epsilon_2 s^2(0)}{4\epsilon_{10}\mu^2(\mu^2-\epsilon_3)}$, $\epsilon_{10} = \min\left\{\frac{\xi\underline{\rho}(L)-\varphi}{\xi\underline{\rho}(L)}, 1\right\}$.

Theorem 4 (Low data rate): Suppose that Assumptions 1–4 hold. Let each agent $i \in \mathcal{V}$ run Algorithm 2 with the same ξ, φ given in Proposition 2 and $(\mu, \sigma) \in \bar{\Pi}$, where

$$\bar{\Pi} = \{(\mu, \sigma) : \sigma \in \left(0, \min\left\{\frac{\varepsilon}{\eta_1}, \frac{\varepsilon}{\eta_2}, \frac{2}{\nu}, \frac{1}{4L_f}\right\}\right)$$

$$\mu \in (\sqrt{\epsilon_3}, 1), \ \Omega \leq \mathcal{K}\}.$$

Then, for any $\mathcal{K} \geq 1$ and $s(0)$ satisfying (21) in Proposition 2, $\bar{\Pi}$ is nonempty, and

$$\|\boldsymbol{x}(k) - \bar{\boldsymbol{x}}(k)\|^2 + n(f(\bar{x}(k)) - f^\star) \leq \epsilon_9\mu^{2k}. \tag{23}$$

Remark 6: Note that the quadratic Lyapunov used in [10] and[42] for convergence analysis relies on the strong convexity condition and the prefect communication. However, such analysis cannot be used for the P–Ł condition and quantized communication. To tackle this problem,

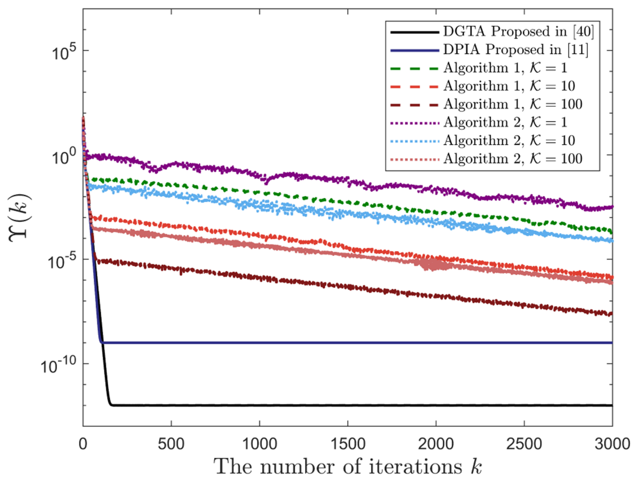

Fig. 1. Evolutions of $\Upsilon(k)$ with respect to the number of iterations for different algorithms.

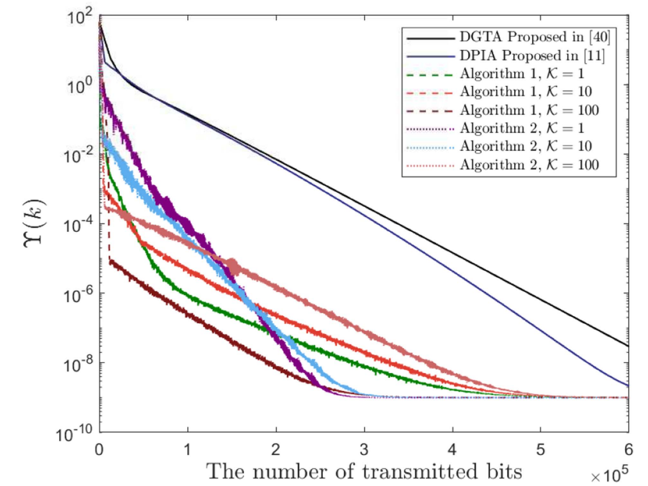

Fig. 2. Evolutions of $\Upsilon(k)$ with respect to the number of transmitted bits for different algorithms.

we design a novel Lyapunov function given in (19). Please refer to the proofs of Proposition 2 and Theorems 3 and 4 of [38] for the detailed analysis for linear convergence of Algorithm 2.

Remark 7: Note that Zhao et al. [36] proposed the BEER algorithm for distributed nonconvex optimization under compressed communication. The uniform quantizer used in our proposed algorithms differs from the randomized compression operator used in [36]. Theorems 2 and 4 establish linear convergence even for 1-bit data rate under the P–Ł condition, which is most communication efficient. By using the randomized compression operator, Zhao et al. [36] also established the sublinear convergence for the general nonconvex case.

VI. Numerical Examples

In this section, we demonstrate the effectiveness of the proposed quantized distributed algorithms through two simulation studies. In the first case, we compare the proposed algorithms with their unquantized counterparts. In the second case, we compare the proposed algorithms with existing quantized distributed optimization algorithms.

First, consider an undirected connected network consisting of 100 agents and the communication graph is randomly generated. The local nonconvex cost functions are given by

$$f_i(x) = a_{i,1}x^2 + a_{i,2}\sin^2(x) + a_{i,3}\cos^2(x) + a_{i,4}\sin(x) + a_{i,5}\sqrt{x^4+3} + a_{i,6}(x^2+2)^{1/3} + \frac{a_{i,7}x^2}{(\sqrt{x^2+1})^{1/2}} - 1$$

where $\sum_{i=1}^n a_{i,1} = 1$, $\sum_{i=1}^n a_{i,2} = 4$, $\sum_{i=1}^n a_{i,3} = 1$, $\sum_{i=1}^n a_{i,4} = 0$, $\sum_{i=1}^n a_{i,5} = 0$, $\sum_{i=1}^n a_{i,6} = 0$, $\sum_{i=1}^n a_{i,7} = 0$. It is easy to check that Assumptions 1–3 are satisfied. Moreover, the global cost function is $\frac{1}{100}(x^2 + 3\sin^2(x))$, which satisfies Assumption 4, as shown in [43]. For different values of $\mathcal{K} = 1, 10, 100$. Based on the conditions (14) and (20), we set $s(0) = 0.3198, 0.0545, 0.0055$, respectively. Denote $\Upsilon(k) = \sum_{i=1}^n \|x_i(k) - \bar{x}(k)\|^2 + n(f(\bar{x}(k)) - f^\star)$. Figs. 1 and 2 illustrate its evolution with respect to the number of iterations k and the number of bits transmitted, respectively, for the DGTA in [40], DPIA in [11], Algorithms 1 and 2. The algorithm parameters used in the experiment are given in Table II.

Fig. 1 clearly shows that the proposed quantized distributed algorithms, even when $\mathcal{K} = 1$, have comparable convergence speeds as the corresponding algorithms with perfect communication. Moreover, larger quantization level leads to faster convergence. This is reasonable since a larger quantization level implies a smaller quantization error. From Fig. 2, we can see that our proposed algorithms converge significantly faster than the DPIA and DGTA when comparing their performances in terms of the number of bits transmitted, which shows the superiority of our proposed algorithms.

TABLE II
Parameter Settings for Our Proposed Algorithms

Algorithm	ξ	φ	σ	β	δ	μ
Algorithm 1	–	–	–	0.1	0.1	0.999
Algorithm 2	0.235	0.2	0.1	–	–	0.999

TABLE III
Parameter Settings for Different Quantized Distributed Algorithms

Algorithm	ξ	φ	σ	β	δ	h	η	μ
Algorithm 1	–	–	–	0.2	0.2	–	–	0.99
Algorithm 2	0.01	0.04	0.1	–	–	–	–	0.99
[24]	–	–	–	–	–	0.5	–	0.99
[34]	–	–	–	–	–	–	0.01	0.99

Next, consider an undirected connected network consisting of ten agents and the communication graph is randomly generated. The local cost functions associated with agents are

$$f_i(x) = \alpha_i \frac{(x^1 - \sin(x^2))^2}{2}$$

where $i \in \{1, \ldots, 10\}$, $\alpha_1 = \alpha_4 = \alpha_7 = 0.1$, $\alpha_2 = \alpha_6 = \alpha_8 = \alpha_9 = 0.05$, and $\alpha_3 = \alpha_{10} = 0.15$, $x = [x^1, x^2]^T$. This function is commonly used in deep learning applications and satisfies the P–Ł condition [44]. It is easy to check that Assumptions 1–4 are satisfied. Choose $\mathcal{K} = 300$ and $s(0) = 1$ such that the conditions (14) and (20) are satisfied. The parameters of the various algorithms used in the experiments are provided in Table III.

Fig. 3 plots the evolution of $\Upsilon(k)$ for different quantized distributed algorithms. It shows that Algorithms 1 and 2 are faster than the quantized subgradient descent algorithm in [24] and the quantized consensus-based algorithm in [34].

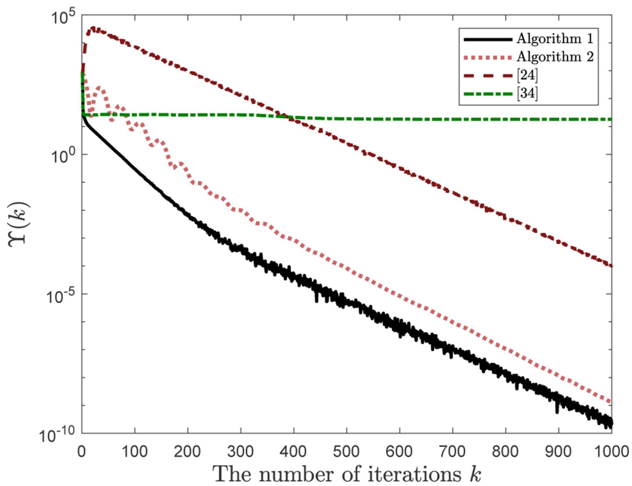

Fig. 3. Evolutions of $\Upsilon(k)$ with respect to the number of iterations for different quantized distributed algorithms.

VII. Conclusion

In this article, we introduced an encoder–decoder scheme to reduce the number of transmitted bits. By integrating it with distributed gradient tracking and DPIAs, respectively, we then proposed two quantized distributed algorithms for solving nonconvex optimization over an undirected connected network. For the case where local cost functions are smooth and the global cost function satisfies the P–Ł condition, we showed that the proposed algorithms linearly converge to an exact global optimal point provided that the quantization level is larger than a certain threshold. We also showed that, with appropriate algorithm parameters, the proposed algorithms with a low data rate, even one bit data rate, are sufficient to ensure linear convergence. One future direction is to consider directed graphs.

Appendix

The following lemma is used in the proofs.

Lemma 3 ([45, Lemma 3]): Let L be the Laplacian matrix of an undirected and connected graph $\mathcal{G}$ with n agents and $K_n = \mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T$. Then, L and K_n are positive semidefinite, $L \leq \rho(L)\mathbf{I}_n$, $\rho(K_n) = 1$

$$K_n L = L K_n = L \tag{24a}$$

$$0 \leq \underline{\rho}(L) K_n \leq L \leq \rho(L) K_n. \tag{24b}$$

Moreover, there exists an orthogonal matrix $[r \quad R] \in \mathbb{R}^{n\times n}$ with $r = \frac{1}{\sqrt{n}}\mathbf{1}_n$ and $R \in \mathbb{R}^{n\times(n-1)}$ such that

$$PL = LP = K_n \tag{25a}$$

$$\frac{1}{\rho(L)} I_n \leq P \leq \frac{1}{\underline{\rho}(L)} I_n \tag{25b}$$

where $\Lambda_1 = \mathrm{diag}([\lambda_2, \ldots, \lambda_n])$ with $0 < \lambda_2 \leq \cdots \leq \lambda_n$ being the nonzero eigenvalues of the Laplacian matrix L, and

$$P = \begin{bmatrix} r & R \end{bmatrix} \begin{bmatrix} \lambda_n^{-1} & 0 \\ 0 & \Lambda_1^{-1} \end{bmatrix} \begin{bmatrix} r^T \\ R^T \end{bmatrix}.$$

References

[1] J. Tsitsiklis, D. Bertsekas, and M. Athans, “Distributed asynchronous deterministic and stochastic gradient optimization algorithms,” *IEEE Trans. Autom. Control*, vol. AC-31, no. 9, pp. 803–812, Sep. 1986.

[2] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, vol. 23. Englewood Cliffs, NJ, USA: Prentice Hall, 1989.

[3] A. Nedić and J. Liu, “Distributed optimization for control,” *Annu. Rev. Control, Robot., Auton. Syst.*, vol. 1, pp. 77–103, 2018.

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

[5] B. Johansson, T. Keviczky, M. Johansson, and K. H. Johansson, “Subgradient methods and consensus algorithms for solving convex optimization problems,” in *Proc. IEEE Conf. Decis. Control*, 2008, pp. 4185–4190.

[6] A. Nedić and A. Ozdaglar, “Distributed subgradient methods for multi-agent optimization,” *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.

[7] 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.

[8] A. Nedić and A. Olshevsky, “Distributed optimization over time-varying directed graphs,” *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 601–615, Mar. 2015.

[9] J. Lu and C. Y. Tang, “Zero-gradient-sum algorithms for distributed convex optimization: The continuous-time case,” *IEEE Trans. Autom. Control*, vol. 57, no. 9, pp. 2348–2354, Sep. 2012.

[10] S. S. Kia, J. Cortés, and S. Martínez, “Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication,” *Automatica*, vol. 55, pp. 254–264, 2015.

[11] J. Wang and N. Elia, “Control approach to distributed optimization,” in *Proc. Annu. Allerton Conf. Commun., Control, Comput.*, 2010, pp. 557–561.

[12] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, “Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes,” in *Proc. IEEE Conf. Decis. Control*, 2015, pp. 2055–2060.

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

[14] R. Larson, “Optimum quantization in dynamic systems,” *IEEE Trans. Autom. Control*, vol. AC-12, no. 2, pp. 162–168, Apr. 1967.

[15] R. Curry, “Separation theorem for nonlinear measurements,” *IEEE Trans. Autom. Control*, vol. AC-14, no. 5, pp. 561–564, Oct. 1969.

[16] K. Cai and H. Ishii, “Convergence time analysis of quantized gossip consensus on digraphs,” *Automatica*, vol. 48, no. 9, pp. 2344–2351, 2012.

[17] R. Carli, F. Fagnani, P. Frasca, and S. Zampieri, “Gossip consensus algorithms via quantized communication,” *Automatica*, vol. 46, no. 1, pp. 70–80, 2010.

[18] A. I. Rikos and C. N. Hadjicostis, “Event-triggered quantized average consensus via ratios of accumulated values,” *IEEE Trans. Autom. Control*, vol. 66, no. 3, pp. 1293–1300, Mar. 2021.

[19] C.-S. Lee, N. Michelusi, and G. Scutari, “Finite rate distributed weight-balancing and average consensus over digraphs,” *IEEE Trans. Autom. Control*, vol. 66, no. 10, pp. 4530–4545, Oct. 2021.

[20] 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.

[21] A. Nedić, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, “Distributed subgradient methods and quantization effects,” in *Proc. IEEE Conf. Decis. Control*, 2008, pp. 4177–4184.

[22] Y. Pu, M. N. Zeilinger, and C. N. Jones, “Quantization design for distributed optimization,” *IEEE Trans. Autom. Control*, vol. 62, no. 5, pp. 2107–2120, May 2017.

[23] D. Yuan, S. Xu, H. Zhao, and L. Rong, “Distributed dual averaging method for multi-agent optimization with quantized communication,” *Syst. Control Lett.*, vol. 61, no. 11, pp. 1053–1061, 2012.

[24] P. Yi and Y. Hong, “Quantized subgradient algorithm and data-rate analysis for distributed optimization,” *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 4, pp. 380–392, Dec. 2014.

[25] 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.

[26] J. Zhang, K. You, and T. Başar, "Distributed discrete-time optimization in multiagent networks using only sign of relative state," *IEEE Trans. Autom. Control*, vol. 64, no. 6, pp. 2352–2367, Jun. 2019.

[27] M. Xiong, B. Zhang, D. Yuan, and S. Xu, "Distributed quantized mirror descent for strongly convex optimization over time-varying directed graph," *Sci. China Inf. Sci.s*, vol. 65, no. 10, pp. 1–15, 2022.

[28] X. Ma, P. Yi, and J. Chen, "Distributed gradient tracking methods with finite data rates," *J. Syst. Sci. Complexity*, vol. 34, no. 5, pp. 1927–1952, 2021.

[29] Y. Xiong, L. Wu, K. You, and L. Xie, "Quantized distributed gradient tracking algorithm with linear convergence in directed networks," *IEEE Trans. Autom. Control*, vol. 68, no. 9, pp. 5638–5645, Sep. 2023.

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

[31] C. Liu, L. Zhu, and M. Belkin, "Loss landscapes and optimization in over-parameterized non-linear systems and neural networks," *Appl. Comput. Harmon. Anal.*, vol. 59, pp. 85–116, 2022.

[32] S. Liu, L. Xie, and D. E. Quevedo, "Event-triggered quantized communication-based distributed convex optimization," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 1, pp. 167–178, Mar. 2018.

[33] A. Reisizadeh, A. Mokhtari, H. Hassani, and R. Pedarsani, "An exact quantized decentralized gradient descent algorithm," *IEEE Trans. Signal Process.*, vol. 67, no. 19, pp. 4934–4947, Oct. 2019.

[34] Y. Kajiyama, N. Hayashi, and S. Takai, "Linear convergence of consensus-based quantized optimization for smooth and strongly convex cost functions," *IEEE Trans. Autom. Control*, vol. 66, no. 3, pp. 1254–1261, Mar. 2021.

[35] J. Lei, P. Yi, G. Shi, and B. D. Anderson, "Distributed algorithms with finite data rates that solve linear equations," *SIAM J. Optim.*, vol. 30, no. 2, pp. 1191–1222, 2020.

[36] H. Zhao, B. Li, Z. Li, P. Richtárik, and Y. Chi, "BEER: Fast $\mathcal{O}(1/T)$ rate for decentralized nonconvex optimization with communication compression," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2022, pp. 31653–31667.

[37] L. Xu, X. Yi, J. Sun, Y. Shi, K. H. Johansson, and T. Yang, "Quantized distributed nonconvex optimization with linear convergence," in *Proc. IEEE Conf. Decis. Control*, 2022, pp. 5837–5842.

[38] L. Xu, X. Yi, J. Sun, Y. Shi, K. H. Johansson, and T. Yang, "Quantized distributed nonconvex optimization algorithms with linear convergence under the Polyak–Łojasiewicz condition," 2022, *arXiv:2207.08106*.

[39] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2325–2383, Oct. 1998.

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

[41] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 3, pp. 1245–1260, Sep. 2018.

[42] B. Gharesifard and J. Cortés, "Distributed continuous-time convex optimization on weight–balanced digraphs," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 781–786, Mar. 2014.

[43] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the Polyak–Łojasiewicz condition," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, 2016, pp. 795–811.

[44] V. Apidopoulos, N. Ginatta, and S. Villa, "Convergence rates for the heavy-ball continuous dynamics for non-convex optimization, under Polyak–Łojasiewicz condition," *J. Glob. Optim.*, vol. 84, no. 3, pp. 563–589, 2022.

[45] X. Yi, S. Zhang, T. Yang, T. Chai, and K. H. Johansson, "Communication compression for distributed nonconvex optimization," *IEEE Trans. Autom. Control*, vol. 68, no. 9, pp. 5477–5492, Sep. 2023.