An Exact Quantized Decentralized Gradient Descent Algorithm

Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, *Member, IEEE*, and Ramtin Pedarsani

***Abstract*—We consider the problem of decentralized consensus optimization, where the sum of n smooth and strongly convex functions are minimized over n distributed agents that form a connected network. In particular, we consider the case that the communicated local decision variables among nodes are quantized in order to alleviate the communication bottleneck in distributed optimization. We propose the Quantized Decentralized Gradient Descent (QDGD) algorithm, in which nodes update their local decision variables by combining the quantized information received from their neighbors with their local information. We prove that under standard strong convexity and smoothness assumptions for the objective function, QDGD achieves a vanishing mean solution error under customary conditions for quantizers. To the best of our knowledge, this is the first algorithm that achieves vanishing consensus error in the presence of quantization noise. Moreover, we provide simulation results that show tight agreement between our derived theoretical convergence rate and the numerical results.**

***Index Terms*—Communication-efficiency, decentralized optimization, gradient methods, quantization.**

I. Introduction

DISTRIBUTED optimization of a sum of convex functions has a variety of applications in different areas including decentralized control systems [2], wireless systems [3], sensor networks [4], networked multiagent systems [5], multirobot networks [6], and large scale machine learning [7]. In such problems, one aims to solve a consensus optimization problem to minimize $f(\mathbf{x}) = \sum_{i=1}^{n} f_i(\mathbf{x})$ cooperatively over n nodes or agents that form a connected network. The function $f_i(\cdot)$ represents the local cost function of node i that is only known by this node.

Manuscript received November 9, 2018; revised June 2, 2019; accepted July 12, 2019. Date of publication August 2, 2019; date of current version August 21, 2019. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Laura Cottatellucci. This work was supported in part by National Science Foundation (NSF) under Grant CCF-1755808 and in part by the UC Office of President under Grant LFR-18-548175. The work of H. Hassani was supported by the NSF under Grant 1755707 and Grant 1837253. This paper was presented in part at the 57th IEEE Conference on Decision and Control, Fontainebleau Miami Beach, Miami Beach, FL, USA, December 2018. *(Corresponding author: Amirhossein Reisizadeh.)*

A. Reisizadeh and R. Pedarsani are with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, Santa Barbara, CA 93106 USA (e-mail: reisizadeh@ucsb.edu; ramtin@ece.ucsb.edu).

A. Mokhtari is with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78705 USA (e-mail: mokhtari@austin.utexas.edu).

H. Hassani is with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: hassani@seas.upenn.edu).

Digital Object Identifier 10.1109/TSP.2019.2932876

Distributed optimization has been largely studied in the literature starting from seminal works in the 80s [8], [9]. Since then, various algorithms have been proposed to address decentralized consensus optimization in multiagent systems. The most commonly used algorithms are decentralized gradient descent or gradient projection method [10]–[13], distributed alternating direction method of multipliers (ADMM) [14]–[16], decentralized dual averaging [17], [18], and distributed Newton-type methods [19]–[21]. Furthermore, the decentralized consensus optimization problem has been considered in online or dynamic settings, where the dynamic cost function becomes an online regret function [22].

A major bottleneck in achieving fast convergence in decentralized consensus optimization is limited communication bandwidth among nodes. As the dimension of input data increases (which is the current trend in large-scale distributed machine learning), a considerable amount of information must be exchanged among nodes, over many iterations of the consensus algorithm. This causes a significant communication bottleneck that can substantially slow down the convergence time of the algorithm [23], [24].

Quantized communication for the agents is brought into the picture for bounded and stable control systems [25]. Furthermore, consensus distributed averaging algorithms are studied under discretized message passing [26]. Motivated by the energy and bandwidth-constrained wireless sensor networks, the work in [27] proposes distributed optimization algorithms under quantized variables and guarantees convergence within a nonvanishing error. Deterministic quantization has been considered in distributed averaging algorithms [28] where the iterations converge to a neighborhood of the average of initials. However, randomized quantization schemes are shown to achieve the average of initials, in expectation [29]. The work in [30] also considers a consensus distributed optimization problem over a cooperative network of agents restricted to quantized communication. The proposed algorithm guarantees convergence to the optima within an error which depends on the network size and the number of quantization levels. Aligned with the communication bottleneck described earlier, [31] provides a quantized distributed load balancing scheme that converges to a set of desired states while the nodes are constrained to remain under maximum load capacities.

More recently, 1-Bit SGD [23] was introduced in which at each time step, the agents sequentially quantize their local gradient vectors by entry-wise signs while contributing the quantization error induced in previous iteration. Moreover, in

[32], the authors propose the Quantized-SGD (QSGD), a class of compression scheme algorithms that is based on a stochastic and unbiased quantizer of the vector to be transmitted. QSGD provably provides convergence guarantees, as well a good practical performance. Recently, a different line of work has proposed the use of coding theoretic techniques to alleviate the communication bottleneck in distributed computation [33]–[36]. In particular, distributed computing algorithms such as MapReduce require shuffling of data or messages between different phases of computation that incur large communication overhead. The key idea to reducing this communication load is to exploit excess in storage and local computation so that *coded* messages can be sent in the phase of shuffling for reducing the communication load.

In this paper, our goal is to analyze the quantized decentralized consensus optimization problem, where node i transmits a quantized version of its local decision variable $Q(\mathbf{x}_i)$ to the neighboring nodes instead of the exact decision variable $\mathbf{x}_i$. Motivated by the stochastic quantizer proposed in [32], we consider two classes of unbiased random quantizers. While they both share the unbiasedness assumption, i.e. $\mathbb{E}\left[Q(\mathbf{x})|\mathbf{x}\right]=\mathbf{x}$, the corresponding variance differs for the two classes. We firstly consider variance bounded quantizers in which we have $\mathbb{E}\left[\|Q(\mathbf{x})-\mathbf{x}\|^2|\mathbf{x}\right]\leq\sigma^2$ for some fixed constant σ^2. Furthermore, we consider random quantizers for which the variance is bounded proportionally to the norm squared of the quatizer's input, that is $\mathbb{E}\left[\|Q(\mathbf{x})-\mathbf{x}\|^2|\mathbf{x}\right]\leq\eta^2\|\mathbf{x}\|^2$ for a constant η^2.

Our main contribution is to propose a Quantized Decentralized Gradient Descent (QDGD) method, which involves a novel way of updating the local decision variables by combining the quantized message received from the neighbors and the local information such that proper averaging is performed over the local decision variable and the neighbors' quantized vectors. We prove that under standard strong convexity and smoothness assumptions, for any unbiased and variance bounded quantizer, QDGD achieves a vanishing mean solution error: for all nodes $i=1,\ldots,n$ we obtain that for any arbitrary $\delta\in(0,1/2)$ and large enough T, $\mathbb{E}\left[\|\mathbf{x}_{i,T}-\widetilde{\mathbf{x}}^*\|^2\right]\leq\mathcal{O}\left(\frac{1}{T^\delta}\right)$, where $\mathbf{x}_{i,T}$ is the local decision variable of node i at iteration T and $\widetilde{\mathbf{x}}^*$ is the global optimum. To the best of our knowledge, this is the first decentralized gradient-based algorithm that achieves vanishing consensus error in the presence of non-vanishing quantization noise. We further generalize the convergence result to the second class of unbiased quantizers for which the variance is bounded proportionally to the norm squared of the quatizer's input and prove that the propsoed algorithm attains the same convergence rate. We also provide simulation results – for both synthetic and real data – that corroborate our theoretical results.

Notation: In this paper, we denote by $[n]$ the set $\{1,\ldots,n\}$ for any natural number $n\in\mathbb{N}$. The gradient of a function $f(\mathbf{x})$ is denoted by $\nabla f(\mathbf{x})$. For non-negative functions g and h of t, we denote $g(t)=\mathcal{O}(h(t))$ if there exist $t_0\in\mathbb{N}$ and constant c such that $g(t)\leq ch(t)$ for any $t\geq t_0$. We use $\lceil x\rceil$ to indicate the least integer greater than or equal to x.

Paper Organization: The rest of the paper is organized as follows. In Section II, we precisely formulate the quantized decentralized consensus optimization problem. We provide the description of the Quantized Decentralized Gradient Descent algorithm in Section III. The main theorems of the paper are stated and proved in Section IV. In Section V, we study the trade-off between communication cost and accuracy of the algorithm. We provide numerical studies in Section VI. Finally, we conclude the paper and discuss future directions in Section VII.

II. Problem Formulation

In this section, we formally define the consensus optimization problem that we aim to solve. Consider a set of n nodes that communicate over a connected and undirected graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$ where $\mathcal{V}=\{1,\ldots,n\}$ and $\mathcal{E}\subseteq\mathcal{V}\times\mathcal{V}$ denote the set of nodes and edges, respectively. We assume that nodes are only allowed to exchange information with their neighbors and use the notation $\mathcal{N}_i$ for the set of node i's neighbors. In our setting, we assume that each node i has access to a local convex function $f_i:\mathbb{R}^p\to\mathbb{R}$, and nodes in the network cooperate to minimize the aggregate objective function $f:\mathbb{R}^p\to\mathbb{R}$ taking values $f(\mathbf{x})=\sum_{i=1}^n f_i(\mathbf{x})$. In other words, nodes aim to solve the optimization problem

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

We assume the local objective functions f_i are strongly convex and smooth, and, therefore, the aggregate function f is also strongly convex and smooth. In the rest of the paper, we use $\widetilde{\mathbf{x}}^*$ to denote the unique minimizer of Problem (1).

In decentralized settings, nodes have access to a single summand of the global objective function f and to reach the optimal solution $\widetilde{\mathbf{x}}^*$, communication with neighboring nodes is inevitable. To be more precise, nodes need to minimize their local objective functions, while they ensure that their local decision variables are equal to their neighbors'. This interpretation leads to an equivalent formulation of Problem (1). If we define $\mathbf{x}_i$ as the decision variable of node i, the alternative formulation of Problem (1) can be written as

$$\begin{aligned}&\min_{\mathbf{x}_1,\ldots,\mathbf{x}_n\in\mathbb{R}^p}\ \sum_{i=1}^n f_i(\mathbf{x}_i)\\ &\text{subject to}\quad \mathbf{x}_i=\mathbf{x}_j,\qquad \text{for all } i,\ j\in\mathcal{N}_i.\end{aligned} \tag{2}$$

Since we assume that the underlying network is a connected graph, the constraint in (2) implies that any feasible solution should satisfy $\mathbf{x}_1=\cdots=\mathbf{x}_n$. Under this condition the objective function values in (1) and (2) are equivalent. Hence, it follows that the optimal solutions of Problem (2) are equal to the optimal solution of Problem (1), i.e., if we denote $\{\mathbf{x}_i^*\}_{i=1}^n$ as the optimal solutions of Problem (2) it holds that $\mathbf{x}_1^*=\cdots=\mathbf{x}_n^*=\widetilde{\mathbf{x}}^*$. Therefore, we proceed to solve Problem (2) which is naturally formulated for decentralized optimization in lieu of Problem (1).

The problem formulation in (2) suggests that each node i should minimize its local objective function f_i while keeping its decision variable $\mathbf{x}_i$ close to the decision variable $\mathbf{x}_j$ of its neighbors $j\in\mathcal{N}_i$. This goal can be achieved by exchanging local variables $\mathbf{x}_i$ among neighboring nodes to enforce consensus on the decision variables. Indeed, exchange of updated local vectors between the distributed nodes induces a potentially heavy communication load on the shared bus. To address this issue, we assume that each node provides a randomly quantized

Algorithm 1: QDGD at Node i.

Require: Weights $\{w_{ij}\}_{j=1}^{n}$, total iterations T
1: Set $\mathbf{x}_{i,0} = 0$ and compute $\mathbf{z}_{i,0} = Q(\mathbf{x}_{i,0})$
2: **for** $t = 0, \ldots, T-1$ **do**
3: Send $\mathbf{z}_{i,t} = Q(\mathbf{x}_{i,t})$ to $j \in \mathcal{N}_i$ and receive $\mathbf{z}_{j,t}$
4: Compute $\mathbf{x}_{i,t+1}$ according to the update in (3)
5: **end for**
6: **return** $\mathbf{x}_{i,T}$

variant of its local updated variable to the neighboring nodes. That is, if we denote by $\mathbf{x}_i$ the decision variable of node i, then the corresponding quantized variant $\mathbf{z}_i = Q(\mathbf{x}_i)$ is communicated to the neighboring nodes, $\mathcal{N}_i$. Exchanging quantized vectors $\mathbf{z}_i$ instead of the true vectors $\mathbf{x}_i$ indeed reduces the communication burden at the cost of injecting noise to the information received by the nodes in the network. The main challenge in this setting is to ensure that nodes can still converge to the optimal solution of Problem (2), while they only have access to a quantized variant of their neighbors' true decision variables.

III. QDGD Algorithm

In this section, we propose a quantized gradient based method to solve the decentralized optimization problem in (2) and consequently the original problem in (1) in a fully decentralized fashion. To do so, consider $\mathbf{x}_{i,t}$ as the decision variable of node i at step t and $\mathbf{z}_{i,t} = Q(\mathbf{x}_{i,t})$ as the quantized version of the vector $\mathbf{x}_{i,t}$. In the proposed Quantized Decentralized Gradient Descent (QDGD) method, nodes update their local decision variables by combining the quantized information received from their neighbors with their local information. To formally state the update of QDGD, we first define w_{ij} as the weight that node i assigns to node j. If nodes i and j are not neighbors then $w_{ij} = 0$, and if they are neighbors the weight $w_{ij} \geq 0$ is nonnegative. At each time step t, each node i sends its quantized $\mathbf{z}_{i,t}$ variant of its local vector $\mathbf{x}_{i,t}$ to its neighbors $j \in \mathcal{N}_i$ and receives their corresponding vectors $\mathbf{z}_{j,t}$. Then, using the received information it updates its local decision variable according to the update

$$\mathbf{x}_{i,t+1} = (1 - \varepsilon + \varepsilon w_{ii})\mathbf{x}_{i,t} + \varepsilon \sum_{j \in \mathcal{N}_i} w_{ij}\mathbf{z}_{j,t} - \alpha\varepsilon\nabla f_i(\mathbf{x}_{i,t}), \tag{3}$$

where ε and α are positive step-sizes.

The update of QDGD in (3) shows that the updated iterate is a linear combination of the weighted average of node i's neighbors' decision variable, i.e., $\varepsilon \sum_{j \in \mathcal{N}_i} w_{ij}\mathbf{z}_{j,t}$, and its local variable $\mathbf{x}_{i,t}$ and gradient $\nabla f_i(\mathbf{x}_{i,t})$. The parameter α behaves as the stepsize of the gradient descent step with respect to local objective function and the parameter ε behaves as an averaging parameter between performing the distributed gradient update $\varepsilon(w_{ii}\mathbf{x}_{i,t} + \sum_{j \in \mathcal{N}_i} w_{ij}\mathbf{z}_{j,t} - \alpha\nabla f_i(\mathbf{x}_{i,t}))$ and using the previous decision variable $(1-\varepsilon)\mathbf{x}_{i,t}$. By choosing a diminishing stepsize α and averaging using the parameter ε we control randomness induced by exchanging quantized variables. The steps of the proposed QDGD method are summarized in Algorithm 1.

Remark 1: The proposed QDGD algorithm can be interpreted as a variant of the decentralized (sub)gradient descent (DGD) method [10], [11] for quantized decentralized optimization (see Section IV). Note that the vanilla DGD method converges to a neighborhood of the optimal solution in the presence of quantization noise where the radius of convergence depends on the variance of quantization error [10], [11], [27], [30]. QDGD improves the inexact convergence of quantized DGD by modifying the contribution of quantized information received from neighboring noise as described in update (3). In particular, as we show in Theorem 1, the sequence of iterates generated by QDGD converges to the optimal solution of Problem (1) in expectation.

Note that the proposed QDGD algorithm does not restrict the quantizer, except for few customary conditions. However, design of efficient quantizers has been taken into consideration. Consider the following example as such quantizers.

Example 1: Consider a low-precision representation specified by $\gamma \in \mathbb{R}$ and $b \in \mathbb{N}$. The range representable by scale factor γ and b bits is $\{-\gamma \cdot 2^{b-1}, \ldots, -\gamma, 0, \gamma, \ldots, \gamma \cdot (2^b - 1)\}$. For any $k\gamma \leq x < (k+1)\gamma$ in the representable range, the low-precision quantizer outputs

$$Q_{(\gamma,b)}(x) = \begin{cases} k\gamma & \text{w.p. } 1 - \frac{x - k\gamma}{\gamma}, \\ (k+1)\gamma & \text{w.p. } \frac{x - k\gamma}{\gamma}. \end{cases} \tag{4}$$

For any x in the range, the quantizer is unbiased and variance bounded, i.e. $\mathbb{E}[Q_{(\gamma,b)}(x)] = x$ and $\mathbb{E}[\|Q_{(\gamma,b)}(x) - x\|^2] \leq \frac{\gamma^2}{4}$.

In Section IV, we formally state the required conditions for the quantization scheme used in QDGD and show that a large class of well-known quantizers satisfy the required conditions.

IV. Convergence Analysis

In this section, we prove that for sufficiently large number of iterations, the sequence of local iterates generated by QDGD converges to an arbitrarily precise approximation of the optimal solution of Problem (2) and consequently Problem (1). The following assumptions hold throughout the analysis of the algorithm.

Assumption 1: Local objective functions f_i are differentiable and smooth with parameter L, i.e.,

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \tag{5}$$

for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$. [1]

Assumption 2: Local objective functions f_i are strongly convex with parameter μ, i.e.,

$$\langle \nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \mu\|\mathbf{x} - \mathbf{y}\|^2, \tag{6}$$

for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$. [2]

[1] Local objectives may have different smoothness parameters, however, WLOG one can consider the largest smoothness parameter as the one for all the objectives.

[2] Local objectives may have different strong convexity parameters, however, WLOG one can consider the smallest strong convexity parameter as the one for all the objectives.

Assumption 3: The random quantizer $Q(\cdot)$ is unbiased and has a bounded variance, i.e.,

$$\mathbb{E}\left[Q(\mathbf{x})|\mathbf{x}\right]=\mathbf{x},\quad \text{and}\quad \mathbb{E}\left[\|Q(\mathbf{x})-\mathbf{x}\|^2|\mathbf{x}\right]\leq\sigma^2, \tag{7}$$

for any $\mathbf{x}\in\mathbb{R}^p$; and quantizations are carried out independently on distributed nodes.

Assumption 4: The weight matrix $W\in\mathbb{R}^{n\times n}$ with entries w_{ij} satisfies the following conditions

$$W=W^\top,\quad W\mathbf{1}=\mathbf{1},\quad \text{and}\quad \text{null}(I-W)=\text{span}(\mathbf{1}). \tag{8}$$

The conditions in Assumptions 1 and 2 imply that the global objective function f is strongly convex with parameter μ and its gradients are Lipschitz continuous with constant L. Assumption 3 poses two customary conditions on the quantizer, that are unbiasedness and variance boundedness. Assumption 4 implies that weight matrix W is symmetric and doubly stochastic. The largest eigenvalue of W is $\lambda_1(W)=1$ and all the eigenvalues belong to $(-1,1]$, i.e., the ordered sequence of eigenvalues of W are $1=\lambda_1(W)\geq\lambda_2(W)\geq\cdots\geq\lambda_n(W)>-1$. We denote by $1-\beta$ the spectral gap associated to the stochastic matrix W, where $\beta=\max\{|\lambda_2(W)|,|\lambda_n(W)|\}$ is the second largest magnitude of the eigenvalues of matrix W. It is also customary to assume $\text{rank}(I-W)=n-1$ such that $\text{null}(I-W)=\text{span}(\mathbf{1})$. We let W_D denote the diagonal matrix consisting of the diagonal entries of W, i.e. $\{w_{11},\ldots,w_{nn}\}$.

In the following theorem we show that the local iterations generated by QDGD converge to the global optima, as close as desired.

Theorem 1: Consider the distributed consensus optimization Problem (1) and suppose Assumptions 1–4 hold. Consider δ as an arbitrary scalar in $(0,1/2)$ and set $\varepsilon=\frac{c_1}{T^{3\delta/2}}$ and $\alpha=\frac{c_2}{T^{\delta/2}}$ where c_1 and c_2 are arbitrary positive constants (independent of T). Then, for each node i, the expected difference between the output of Algorithm 1 after T iterations and the solution of Problem (1), i.e. $\widetilde{\mathbf{x}}^*$ is upper bounded by

$$\mathbb{E}\Big[\|\mathbf{x}_{i,T}-\widetilde{\mathbf{x}}^*\|^2\Big]\leq\mathcal{O}\left(\left(\frac{4nc_2^2D^2\left(3+2L/\mu\right)^2}{(1-\beta)^2}+\frac{2c_1n\sigma^2\|W-W_D\|^2}{\mu c_2}\right)\frac{1}{T^\delta}\right), \tag{9}$$

if the total number of iterations satisfies $T\geq T_0$, where T_0 is a function of δ, c_1, c_2, μ, L, and $\lambda_n(W)$. Moreover,

$$D^2=2L\sum_{i=1}^n\left(f_i(0)-f_i^*\right),\quad f_i^*=\min_{\mathbf{x}\in\mathbb{R}^p}f_i(\mathbf{x}). \tag{10}$$

Theorem 1 demonstrates that the proposed QDGD provides an approximation solution with vanishing deviation from the optimal solution, despite the fact that the quantization noise does not vanish as the number of iterations progresses.

By the first glance at the expression in (9) one might suggest to set $\delta=1/2$ to obtain the best possible sublinear convergence rate which is $\mathcal{O}\left(\frac{1}{T^{1/2}}\right)$. However, T_0, which is a lower bound on the total number of iterations T, is an increasing function of $1/(1-2\delta)$, and by choosing δ very close to $1/2$, the total number of iterations T should be very large to obtain a fast convergence rate close to $\mathcal{O}\left(\frac{1}{T^{1/2}}\right)$. Therefore, there is a trade-off between the convergence rate and the minimum number of required iterations. By setting δ close to $1/2$ we obtain a fast convergence rate but at the cost of running the algorithm for a large number of iterations, and by selecting δ close to 0 the lower bound on the total number of iterations becomes smaller at the cost of having a slower convergence rate. We will illustrate this trade-off in the numerical experiments.

Moreover, note that the result in (9) shows a balance between the variance of quantization and the mixing matrix. To be more precise, if the variance of quantization σ^2 is small nodes should assign larger weights to their neighbors which decreases $(1-\beta)^{-2}$ and increases $\|W-W_D\|^2$. Conversely, when the variance σ^2 is large, to balance the terms in (9) nodes should assign larger weights to their local decision variables which decreases the term $\|W-W_D\|^2$ and increases $(1-\beta)^{-2}$.

A. Proof of Theorem 1

To analyze the proposed QDGD method, we start by rewriting the update rule (3) as follows

$$\mathbf{x}_{i,t+1}=\mathbf{x}_{i,t}-\varepsilon\left((1-w_{ii})\mathbf{x}_{i,t}-\sum_{j\neq i}w_{ij}\mathbf{z}_{j,t}+\alpha\nabla f_i(\mathbf{x}_{i,t})\right). \tag{11}$$

Note that to derive the expression in (11), we simply use the fact that $w_{ij}=0$ when $j\notin\mathcal{N}_i$.

The next step is to write the update (11) in a matrix form. To do so, we define the function $F:\mathbb{R}^{np}\to\mathbb{R}$ as $F(\mathbf{x})=\sum_{i=1}^n f_i(\mathbf{x}_i)$ where $\mathbf{x}_i\in\mathbb{R}^p$ and $\mathbf{x}=[\mathbf{x}_1;\cdots;\mathbf{x}_n]\in\mathbb{R}^{np}$ is the concatenation of the local variables $\mathbf{x}_i$. It is easy to verify that the gradient of the function F is the concatenation of local gradients evaluated at the local variable, that is $\nabla F(\mathbf{x}_t)=[\nabla f_1(\mathbf{x}_{1,t});\cdots;\nabla f_n(\mathbf{x}_{n,t})]$. We also define the matrix $\mathbf{W}=W\otimes I\in\mathbb{R}^{np\times np}$ as the Kronecker product of the weight matrix $W\in\mathbb{R}^{n\times n}$ and the identity matrix $I\in\mathbb{R}^{p\times p}$. Similarly, define $\mathbf{W}_D=W_D\otimes I\in\mathbb{R}^{np\times np}$, where $W_D=[w_{ii}]\in\mathbb{R}^{n\times n}$ denotes the diagonal matrix of the entries on the main diagonal of W. For the sake of consistency, we denote by the boldface $\mathbf{I}$ the identity matrix of size np. According to above definitions, we can write the concatenated version of (11) as follows,

$$\mathbf{x}_{t+1}=\mathbf{x}_t-\varepsilon\Big(\big(\mathbf{I}-\mathbf{W}_D\big)\mathbf{x}_t+\big(\mathbf{W}_D-\mathbf{W}\big)\mathbf{z}_t+\alpha\nabla F(\mathbf{x}_t)\Big). \tag{12}$$

As we discussed in Section II, the distributed consensus optimization Problem (1) can be equivalently written as Problem (2). The constraint in the latter restricts the feasible set to the consensus vectors, that is $\{\mathbf{x}=[\mathbf{x}_1;\cdots;\mathbf{x}_n]:\mathbf{x}_1=\cdots=\mathbf{x}_n\}$. According to the discussion on rank of the weight matrix W, the null space of the matrix $I-W$ is $\text{null}(I-W)=\text{span}(\mathbf{1})$. Hence, the null space of $\mathbf{I}-\mathbf{W}$ is the set of all consensus vectors, i.e., $\mathbf{x}\in\mathbb{R}^{np}$ is feasible for Problem (2) if and only if $(\mathbf{I}-\mathbf{W})\mathbf{x}=0$, or equivalently $(\mathbf{I}-\mathbf{W})^{1/2}\mathbf{x}=0$. Therefore, the alternative Problem (2) can be compactly represented as the

following linearly-constrained problem,

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^{np}} \quad & F(\mathbf{x})=\sum_{i=1}^{n} f_i(\mathbf{x}_i) \\ \text{subject to} \quad & (\mathbf{I}-\mathbf{W})^{1/2}\mathbf{x}=0. \end{aligned} \tag{13}$$

We denote by $\mathbf{x}^*=[\widetilde{\mathbf{x}}^*;\ldots;\widetilde{\mathbf{x}}^*]$ the unique solution to (13).

Now, for given penalty parameter $\alpha>0$, one can define the quadratic penalty function corresponding to the linearly constraint problem (13) as follows,

$$h_\alpha(\mathbf{x})=\frac{1}{2}\mathbf{x}^\top\big(\mathbf{I}-\mathbf{W}\big)\mathbf{x}+\alpha F(\mathbf{x}). \tag{14}$$

Since $\mathbf{I}-\mathbf{W}$ is a positive semi-definite matrix and F is L-smooth and μ-strongly convex, the function h_α is L_α-smooth and μ_α-strongly convex on $\mathbb{R}^{np}$ having $L_\alpha=1-\lambda_n(W)+\alpha L$ and $\mu_\alpha=\alpha\mu$. We denote by $\mathbf{x}_\alpha^*$ the unique minimizer of $h_\alpha(\mathbf{x})$, i.e.,

$$\mathbf{x}_\alpha^*=\arg\min_{\mathbf{x}\in\mathbb{R}^{np}} h_\alpha(\mathbf{x})=\arg\min_{\mathbf{x}\in\mathbb{R}^{np}} \frac{1}{2}\mathbf{x}^\top\big(\mathbf{I}-\mathbf{W}\big)\mathbf{x}+\alpha F(\mathbf{x}). \tag{15}$$

In the following, we link the solution of Problem (15) to the local variable iterations provided by Algorithm 1. Specifically, for sufficiently large number of iterations T, we demonstrate that for proper choice of step-sizes, the expected squared deviation of $\mathbf{x}_T$ from $\mathbf{x}_\alpha^*$ vanishes sub-linearly. This result follows from the fact that the expected value of the descent direction in (12) is an unbiased estimator of the gradient of the function $h_\alpha(\mathbf{x})$.

Lemma 1: Consider the optimization Problem (15) and suppose Assumptions 1–4 hold. Then, the expected deviation of the output of QDGD from the solution to Problem (15) is upper bounded by

$$\mathbb{E}\Big[\|\mathbf{x}_T-\mathbf{x}_\alpha^*\|^2\Big]\leq\mathcal{O}\left(\frac{c_1 n\sigma^2\|W-W_D\|^2}{\mu c_2}\frac{1}{T^\delta}\right), \tag{16}$$

for $\varepsilon=\frac{c_1}{T^{3\delta/2}}$, $\alpha=\frac{c_2}{T^{\delta/2}}$, any $\delta\in(0,1/2)$ and $T\geq T_1$, where c_1 and c_2 are positive constants independent of T, and

$$T_1:=\max\left\{e^{e^{\frac{1}{1-2\delta}}},\left\lceil(c_1c_2\mu)^{\frac{1}{2\delta}}\right\rceil,\left\lceil\left(\frac{c_1(2+c_2L)^2}{c_2\mu}\right)^{\frac{1}{\delta}}\right\rceil\right\}. \tag{17}$$

Proof: See Appendix A. ∎

Lemma 1 guarantees convergence of the proposed iterations according to the update in (3) to the solution of the later-defined Problem (15). Loosely speaking, Lemma 1 ensures that $\mathbf{x}_T$ is *close* to $\mathbf{x}_\alpha^*$ for large T. So, in order to capture the deviation of $\mathbf{x}_T$ from the global optima $\mathbf{x}^*$, it suffices to show that $\mathbf{x}_\alpha^*$ is *close* to $\mathbf{x}^*$, as well. As the problem in (15) is a penalized version of the original constrained program in (1), the solutions to these two problems should not be significantly different if the penalty coefficient α is small. We formalize this claim in the following lemma.

Lemma 2: Consider the distributed consensus optimization Problem (1) and the problem defined in (15). If Assumptions 1, 2 and 4 hold, then the difference between the optimal solutions to (13) and its penalized version (15) is bounded above by

$$\|\mathbf{x}_\alpha^*-\mathbf{x}^*\|\leq\mathcal{O}\left(\frac{\sqrt{2n}c_2D\,(3+2L/\mu)}{1-\beta}\frac{1}{T^{\delta/2}}\right), \tag{18}$$

for $\alpha=\frac{c_2}{T^{\delta/2}}$ and $T\geq T_2$, where c_2 is a positive constant independent of T, $\delta\in(0,1/2)$ is an arbitrary constant, and

$$T_2:=\max\left\{\left\lceil\left(\frac{c_2L}{1+\lambda_n(W)}\right)^{\frac{2}{\delta}}\right\rceil,\left\lceil c_2^4(\mu+L)^{\frac{2}{\delta}}\right\rceil\right\}. \tag{19}$$

Proof: See Appendix B. ∎

The result in Lemma 2 shows that if we set the penalty coefficient α small enough, i.e., $\alpha=\mathcal{O}(T^{-\delta/2})$, then the distance between the optimal solutions of the constrained problem in (1) and the penalized problem in (15) is of $\mathcal{O}(\frac{\alpha}{1-\beta})$.

Having set the main lemmas, now it is straightforward to prove the claim of Theorem 1. For the specified step-sizes ε and α and large enough iterations $T\geq T_0:=\max\{T_1,T_2\}$, Lemmas 1 and 2 are applicable and we have

$$\begin{aligned} \mathbb{E}\left[\|\mathbf{x}_T-\mathbf{x}^*\|^2\right] &= \mathbb{E}\left[\|\mathbf{x}_T-\mathbf{x}_\alpha^*+\mathbf{x}_\alpha^*-\mathbf{x}^*\|^2\right] \\ &\leq 2\mathbb{E}\left[\|\mathbf{x}_T-\mathbf{x}_\alpha^*\|^2\right]+2\|\mathbf{x}_\alpha^*-\mathbf{x}^*\|^2 \\ &\leq \mathcal{O}\left(\frac{1}{T^\delta}\right)+\mathcal{O}\left(\frac{1}{T^\delta}\right) \\ &= \mathcal{O}\left(\frac{1}{T^\delta}\right), \end{aligned} \tag{20}$$

where we used $\|\mathbf{a}+\mathbf{b}\|^2\leq 2\big(\|\mathbf{a}\|^2+\|\mathbf{b}\|^2\big)$ to derive the first inequality; and the constants can be found in the proofs of the two lemmas. Since $\mathbb{E}[\|\mathbf{x}_{i,T}-\widetilde{\mathbf{x}}^*\|^2]\leq\mathbb{E}[\|\mathbf{x}_T-\mathbf{x}^*\|^2]$ for any $i=1,\ldots,n$, the inequality in (20) implies the claim of Theorem 1.

B. Extension to More Quantizers

Based on the condition in Assumption 3, so far we have been considering only unbiased quantizers for which the variance of quantization is bounded by a constant scalar, i.e., $\mathbb{E}\left[\|Q(\mathbf{x})-\mathbf{x}\|^2|\mathbf{x}\right]\leq\sigma^2$. However, there are widely used representative quantizers where the quantization noise induced on the input is bounded proportionally to the input's magnitude, i.e., $\mathbb{E}\left[\|Q(\mathbf{x})-\mathbf{x}\|^2|\mathbf{x}\right]\leq\mathcal{O}\left(\|\mathbf{x}\|^2\right)$ [32].

Indeed, this condition is more challenging since the set of iterates norm $\|\mathbf{x}_t\|$ are not necessarily bounded, and we cannot uniformly bound the variance of the noise induced by quantization. In this subsection, we show that the proposed algorithm is converging with the same rate for quantizers satisfying this new assumption. Let us first formally state this assumption.

Assumption 5: The random quantizer $Q(\cdot)$ is unbiased and its variance is proportionally bounded by the input's squared norm, that is,

$$\mathbb{E}\left[Q(\mathbf{x})|\mathbf{x}\right]=\mathbf{x},\quad\text{and}\quad\mathbb{E}\left[\|Q(\mathbf{x})-\mathbf{x}\|^2|\mathbf{x}\right]\leq\eta^2\|\mathbf{x}\|^2, \tag{21}$$

for a constant η^2 and any $\mathbf{x}\in\mathbb{R}^p$; and quantizations are carried out independently on distributed nodes.

Before characterizing the convergence properties of the proposed QDGD method under the conditions in Assumption 5, let us review a subset of quantizers that satisfy this condition.

Example 2 (Low-precision quantizer): Consider the low precision quantizer $Q^{\text{LP}}:\mathbb{R}^p\to\mathbb{R}^p$ which is defined as

$$Q_i^{\text{LP}}(\mathbf{x})=\|\mathbf{x}\|\cdot\text{sign}(x_i)\cdot\xi_i(\mathbf{x},s), \tag{22}$$

where $\xi_i(\mathbf{x},s)$ is a random variable defined as

$$\xi_i(\mathbf{x},s)=\begin{cases}\frac{l}{s} & \text{w.p. } 1-q\left(\frac{|x_i|}{\|\mathbf{x}\|},s\right),\\ \frac{l+1}{s} & \text{w.p. } q\left(\frac{|x_i|}{\|\mathbf{x}\|},s\right),\end{cases} \tag{23}$$

and $q(a,s)=as-l$ for any $a\in[0,1]$. In above, the tuning parameter s corresponds to the number of quantization levels and $l\in[0,s)$ is an integer such that $|x_i|/\|\mathbf{x}\|\in[l/s,(l+1)/s]$. It is not hard to check that [32] the low precision quantizer Q^{LP} defined in (22) is an unbiased estimator of the vector $\mathbf{x}$ and the variance is bounded above by

$$\mathbb{E}\left[\|Q^{\text{LP}}(\mathbf{x})-\mathbf{x}\|^2\right]\le\min\left(\frac{p}{s^2},\frac{\sqrt{p}}{s}\right)\|\mathbf{x}\|^2. \tag{24}$$

The bound in (24) illustrates the trade-off between communication cost and quantization variance. Choosing a large s reduces the variance of quantization at the cost of increasing the levels of quantization and therefore increasing the communication cost.

The following example provides another quantizer which satisfies the conditions in Assumption 5.

Example 3 (Gradient sparsifier): The gradient sparsifier denoted by $Q^{\text{GS}}:\mathbb{R}^p\to\mathbb{R}^p$ is defined as

$$Q_i^{\text{GS}}(\mathbf{x})=\begin{cases}x_i/q_i & \text{w.p. } q_i,\\ 0 & \text{otherwise},\end{cases} \tag{25}$$

where q_i is probability that coordinate $i\in[p]$ is selected. It is easy to verify that this quantizer is unbiased, as for each i, $\mathbb{E}\left[Q_i^{\text{GS}}(\mathbf{x})\right]=x_i$. Moreover, one can show that the variance of this quantizer is bounded as follows,

$$\mathbb{E}\left[\|Q^{\text{GS}}(\mathbf{x})-\mathbf{x}\|^2\right]=\sum_{i=1}^{p}\left(\frac{1}{q_i}-1\right)x_i^2\le\left(\frac{1}{q_{\min}}-1\right)\|\mathbf{x}\|^2, \tag{26}$$

where $q_{\min}$ denotes the minimum of probabilities $\{q_1,\ldots,q_p\}$.

In the following theorem, we extend our result in Theorem 1 to the case that variance of quantizer may not be uniformly bounded and is proportional to the squared norm of quantizer's input.

Theorem 2: Consider the distributed consensus optimization Problem (1) and suppose Assumptions 1, 2, 4, 5 hold. Then, for each node i, the expected squared difference between the output of the QDGD method outlined in Algorithm 1 and the optimal solution of Problem (1), i.e. $\widetilde{\mathbf{x}}^*$ is upper bounded by

$$\mathbb{E}\Big[\|\mathbf{x}_{i,T}-\widetilde{\mathbf{x}}^*\|^2\Big]\le\mathcal{O}\left(\left(\frac{4nc_2^2D^2\left(3+2L/\mu\right)^2}{(1-\beta)^2}+\frac{4c_1n\widetilde{B}^2\eta^2\|W-W_D\|^2}{\mu c_2}\right)\frac{1}{T^{\delta}}\right), \tag{27}$$

for $\varepsilon=\frac{c_1}{T^{3\delta/2}}$, $\alpha=\frac{c_2}{T^{\delta/2}}$, any $\delta\in(0,1/2)$ and $T\ge\widetilde{T}_0$, where c_1, c_2 and $\widetilde{T}_0$ are positive constants independent of T, and

$$\widetilde{B}^2=\frac{4c_2^2D^2\left(3+2L/\mu\right)^2}{(1-\beta)^2}+\frac{4(f_0-f^*)}{\mu}. \tag{28}$$

Proof: See Appendix C. ■

The result in Theorem 2 shows that under Assumption 5, the proposed QDGD method converges to the optimal solution at a sublinear rate of $\mathcal{O}\left(T^{-\delta}\right)$ which matches the result in Theorem 1. However, the lower bound on the total number of iterations $\widetilde{T}_0$ for the result in Theorem 2 is in general larger than T_0 for the result in Theorem 1. The exact expression of $\widetilde{T}_0$ could be found in Appendix C.

V. Optimal Quantization Level for Reducing Overall Communication Cost

In this section, we aim to study the trade-off between number of iterations until achieving a target accuracy and quantization levels. Indeed, by increasing quantization levels the variance of quantization reduces and the total number of iterations to reach a specific accuracy decreases, but the communication overhead of each round is higher as we have to transmit more bits. Conversely, if we use a quantization with a small number of levels the communication cost per iteration will be low; however, the total number of iterations could be very large. The fundamental question here is how to choose the quantization levels to optimize the overall communication cost which is the product of number of iterations and communication cost of each iteration.

In this section, we only focus on unbiased quantizers for which the variance is proportionally bounded with the squared norm of the quantizer's input vector, i.e., for any $\mathbf{x}\in\mathbb{R}^p$ it holds that $\mathbb{E}\left[Q(\mathbf{x})|\mathbf{x}\right]=\mathbf{x}$ and $\mathbb{E}\left[\|Q(\mathbf{x})-\mathbf{x}\|^2|\mathbf{x}\right]\le\eta^2\|\mathbf{x}\|^2$ for some fixed constant η. Theorem 2 characterizes the (order-wise) convergence of the proposed algorithm considering this assumption. More precisely, using the result in Theorem 2 and (27) we can write for each node i:

$$\mathbb{E}\left[\|\mathbf{x}_{i,T}-\widetilde{\mathbf{x}}^*\|^2\right]\le\left[\frac{4nc_2^2D^2\left(3+2L/\mu\right)^2}{(1-\beta)^2}+\frac{4c_1n\widetilde{B}^2\eta^2\|W-W_D\|^2}{\mu c_2}\right]\frac{1}{T^{\delta}}, \tag{29}$$

where the approximation is due to considering dominant terms in $B_1(T)$ and $B_2(T)$ (See Appendix B and C for notations and details of derivations). Therefore, given a target relative deviation error ρ and using (29) , the algorithm needs to iterate at least $T(\rho)$ where

$$T(\rho):=\left[\frac{4nc_2^2D^2\left(3+2L/\mu\right)^2}{(1-\beta)^2}+\frac{4c_1n\widetilde{B}^2\eta^2\|W-W_D\|^2}{\mu c_2}\right]^{1/\delta}\left(\frac{1}{\rho\|\widetilde{\mathbf{x}}^*\|^2}\right)^{1/\delta}. \tag{30}$$

It is shown in [32] that for the low-precision quantizer defined in (22) and (23) there exists an encoding scheme Code_s such that for any $\mathbf{x} \in \mathbb{R}^p$ and $s^2 + \sqrt{p} \leq p/2$, the communication cost of the quantized vector satisfies

$$\mathbb{E}\left[|\text{Code}_s(Q^{\text{LP}}(\mathbf{x}))|\right] \leq b + \left(3 + \frac{3}{2}\log^*\left(\frac{2(s^2+p)}{s^2+\sqrt{p}}\right)\right)(s^2+\sqrt{p}), \quad (31)$$

where $\log^*(x) = \log(x) + \log\log(x) + \cdots = (1+o(1))\log(x)$ and b denotes the number bits for representing one floating point number ($b \in \{32, 64\}$ are typical values). For large s, [32] also proposes a simple encoding scheme Code'_s which is proved to impose no more than the following communication cost on the quantized vector

$$\mathbb{E}\left[|\text{Code}'_s(Q^{\text{LP}}(\mathbf{x}))|\right] \leq b + \left(\frac{5}{2} + \frac{1}{2}\log^*\left(1 + \frac{s^2 + \min(d, s\sqrt{p})}{p}\right)\right)p. \quad (32)$$

Now we can easily derive the expected total communication cost (in bits) of a quantized decentralized consensus optimization in order for each agent to achieve a predefined target error. For instance, assume that the low-precision quantizer described above is employed for the quanization operations. Using this quantizer, the expected communication cost (in bits) for transmitting a single p-dimensional real vector is represented in (31) and (32) for two sparsity regimes of the tuning parameter s.

On the other hand, in order for each agent to obtain a relative error ρ, the proposed algorithm iterates $T(\rho)$ times as denoted in (30). Therefore, the total (expected) communication cost across all of the n agents is upper-bounded by $nT(\rho) \cdot \mathbb{E}\left[|\text{Code}_s(Q^{\text{LP}}(\mathbf{x}))|\right]$ and $nT(\rho) \cdot \mathbb{E}\left[|\text{Code}'_s(Q^{\text{LP}}(\mathbf{x}))|\right]$ for small and large s, respectively.

Remark 2: We can derive the total communication cost for the vanilla DGD method ([11]), as well. DGD method updates the iterations as follows:

$$\mathbf{x}_{i,t+1} = w_{ii}\mathbf{x}_{i,t} + \sum_{j \in \mathcal{N}_i} w_{ij}\mathbf{x}_{j,t} - \alpha \nabla f_i(\mathbf{x}_{i,t}), \quad (33)$$

where $\alpha = c/\sqrt{T}$ is the stepsize. DGD guarantees the following convergence rate for strongly convex objectives:

$$\|\mathbf{x}_{i,T} - \widetilde{\mathbf{x}}^*\|^2 \leq \frac{(3+2L/\mu)^2 D^2}{(1-\beta)^2}\alpha^2 = \frac{c^2(3+2L/\mu)^2 D^2}{(1-\beta)^2}\frac{1}{T}. \quad (34)$$

Hence, to reach the ρ approximation of the global optimal, DGD requires the total number of iterations

$$T_{\text{DGD}}(\rho) = \frac{c^2(3+2L/\mu)^2 D^2}{(1-\beta)^2}\frac{1}{\rho\|\widetilde{\mathbf{x}}^*\|^2}. \quad (35)$$

Given that each decision vector requires bp number of bits in an implementation of DGD (without quantization), the DGD method induces the communication cost of $nT_{\text{DGD}}(\rho)bp$.

In the following, we numerically evaluate the communication cost of the proposed QDGD method for the following least squares problem

$$\min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x}) = \sum_{i=1}^{n} \frac{1}{2}\|\mathbf{A}_i\mathbf{x} - \mathbf{b}_i\|^2. \quad (36)$$

We assume that the network contains $n = 50$ agents that collaboratively aim to solve problem (36) over the real field of size $p = 200$. The elements of the random matrices $\mathbf{A}_i \in \mathbb{R}^{p \times p}$ and the solution $\widetilde{\mathbf{x}}^*$ are picked from the normal distribution $\mathcal{N}(0,1)$. Moreover, we let $\mathbf{b}_i = \mathbf{A}_i\widetilde{\mathbf{x}}^* + \mathcal{N}(0, 0.1I_p)$. All nodes update their local variables with respect to the proposed algorithm and send the quantized updates to the neighbors using a low-precision quantizer with s quantization levels and $b = 64$ bits for representing one floating point number, until they satisfy the predefined relative error $\rho = 10^{-2}$. The underlying graph is an Erdős-Rényi with edge probability $p_c = 0.35$. The edge weight matrix is picked as $W = I - \frac{2}{3\lambda_{\max}(\mathbf{L})}\mathbf{L}$ where $\mathbf{L}$ is the Laplacian with $\lambda_{\max}(\mathbf{L})$ as its largest eigenvalue. We also set $\delta = 0.1$.

TABLE I
QUANTIZATION-COMMUNICATION TRADE-OFF FOR LEAST SQUARES PROBLEM

# quantization levels	# iterations ($\times 10^2$)	code length per vector (bits)	communication cost (bits) ($\times 10^7$)
$s = 1$	10800	216.9	1171
$s = 50$	11.6	949.8	5.5
$s^* = 77$	9.91	1062	5.27
$s = 10^3$	8.79	1793	7.88
$s = 10^5$	8.78	3122	13.71
$s = 10^{10}$	8.78	6443	28.3
$s = 10^{15}$	8.78	9765	42.9
$s = 10^{19}$	8.78	12420	54.56

Table I represents the total expected communication cost (in bits, as computed using (30), (31) and (32)) induced by the proposed algorithm to solve (36) using the low-precision quantizer –as described above– for four representative cases. As observed from this table and expected from the theoretical derivations, larger number of quantization levels translates to less noisy quantization and hence fewer iterations. Also, larger number of quantization levels induces more communication cost for each transmitted quantized data variable which results in larger code length per vector. However, the average total communication cost does not necessarily follow a monotonic trend. As Table I shows, the optimal $s^* = 77$ induces the smallest total communication cost among all levels $s \geq 1$. Moreover, Table I demonstrates the significant gain of picking the optimal levels s^* compared to the larger ones.

VI. NUMERICAL EXPERIMENTS

In this section, we evaluate the performance of the proposed QDGD Algorithm on decentralized quadratic minimization and ridge regression problems and demonstrate the effect of various parameters on the relative expected error rate. We carry out the simulations on artificial and real data sets corresponding to quadratic minimization and ridge regression problems, respectively. In both cases, the graph of agents is a connected Erdős-Rényi with edge probability p_c. We set the edge weight

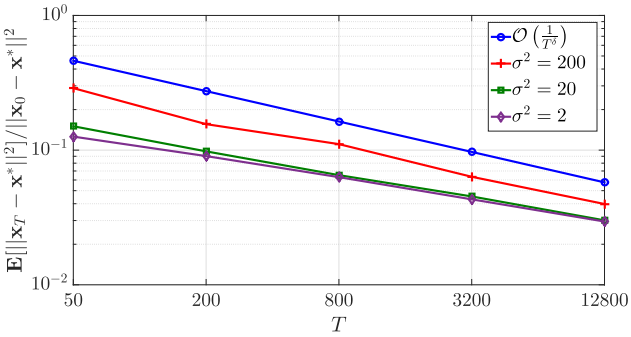

Fig. 1. Relative optimal squared error for three values of quantization noise variance: $\sigma^2 \in \{2, 20, 200\}$, compared with the order of upper bound.

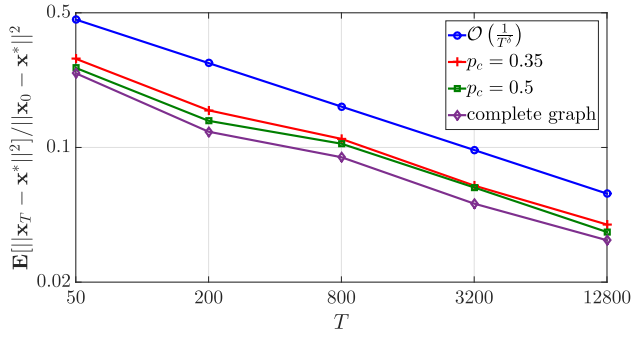

Fig. 2. Relative optimal squared error for three vales of graph connectivity ratio: $p_c \in \{0.35, 0.5, 1\}$, compared with the order of upper bound.

matrix to be $W = I - \frac{2}{3\lambda_{\max}(\mathbf{L})}\mathbf{L}$ where $\mathbf{L}$ is the Laplacian with $\lambda_{\max}(\mathbf{L})$ as its largest eigenvalue.

A. Decentralized Quadratic Minimization

In this section, we evaluate the performance of the proposed QDGD Algorithm on minimizing a distributed quadratic objective. We pictorially demonstrate the effect of quantization noise and graph topology on the relative expected error rate.

Consider the quadratic optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x}) = \sum_{i=1}^{n} \frac{1}{2}\mathbf{x}^\top \mathbf{A}_i \mathbf{x} + \mathbf{b}_i^\top \mathbf{x}, \tag{37}$$

where $f_i(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{A}_i \mathbf{x} + \mathbf{b}_i^\top \mathbf{x}$ denotes the local objective function of node $i \in [n]$. The unique solution to (37) is therefore $\widetilde{\mathbf{x}}^* = -\left(\sum_{i=1}^n \mathbf{A}_i\right)^{-1}\left(\sum_{i=1}^n \mathbf{b}_i\right)$. We pick diagonal matrices $\mathbf{A}_i$ such that $p/2$ of the diagonal entries of each $\mathbf{A}_i$ are drawn from the set $\{1, 2, 2^2\}$ and the other $p/2$ diagonal entries are drawn from the set $\{1, 2^{-1}, 2^{-2}\}$, all uniformly at random. Entries of vectors $\mathbf{b}_i$ are randomly picked from the interval (0,1). In our simulations, we let an additive noise model the quantization error, i.e. $Q(\mathbf{x}) = \mathbf{x} + \eta$ where $\eta \sim \mathcal{N}(0, \frac{\sigma^2}{p} I_p)$.

We first consider a connected Erdős-Rényi graph of $n = 50$ nodes and connectivity probability of $p_c = 0.35$ and dimension $p = 20$. Fig. 1 shows the convergence rate corresponding to three values of quantization noise $\sigma^2 \in \{2, 20, 200\}$ and $\delta = 3/8$, compared to the theoretical upper bound derived in Theorem 1 in the logarithmic scale. For each plot, stepsizes are pick as $\varepsilon = c_1/T^{3\delta/2}$ and $\alpha = c_2/T^{\delta/2}$ where the constants c_1, c_2 are finely tuned. As expected, Fig. 1 shows that the error rate linearly scales with the quantization noise; however, it does not saturate around a non-vanishing residual, regardless the variance. Moreover, Fig. 1 demonstrates that the convergence rate closely follows the upper bound derived in Theorem 1. For instance, for the plot corresponding to $\sigma^2 = 200$, the relative errors are evaluated as $e_{T_1}/e_0 = 0.1108$ and $e_{T_2}/e_0 = 0.0634$ for $T_1 = 800$ and $T_2 = 3200$, respectively. Therefore, $e_{T_2}/e_{T_1} \approx 0.57$ which is upper bounded by $(\frac{T_1}{T_2})^\delta \approx 0.59$.

To observe the effect of graph topology, quantization noise variance is fixed to $\sigma^2 = 200$ and we varied the connectivity ratio by picking three different values, i.e. $p_c \in \{0.35, 0.5, 1\}$ where $p_c = 1$ corresponds to the complete graph case. We also fix the parameter $\delta = 3/8$ and accordingly pick the stepsizes $\varepsilon = c_1/T^{3\delta/2}$ and $\alpha = c_2/T^{\delta/2}$ where the constants c_1, c_2 are finely tuned. As Fig. 2 depicts, for the same number of iterations, deviation from the optimal solution tends to increase as the graph is gets sparse. In other words, even noisy information of the neighbor nodes improves the gradient estimate for local nodes. It also highlights the fact that regardless of the sparsity of the graph, the proposed QDGD algorithm guarantees the consensus to the optimal solution for each local node, as long as the graph is connected.

B. Decentralized Ridge Regression

Consider the ridge regression problem:

$$\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x}) = \sum_{j=1}^{D} \|\mathbf{a}_j \mathbf{x} - b_j\|^2 + \frac{\lambda}{2}\|\mathbf{x}\|_2^2, \tag{38}$$

over the data set $\mathcal{D} = \{(\mathbf{a}_j, b_j) : j = 1, \ldots, D\}$ where each pair $(\mathbf{a}_j, b_j)$ denotes the predictors-response variables corresponding to data point $j \in [D]$ where $\mathbf{a}_j \in \mathbb{R}^{1\times p}$, $b_j \in \mathbb{R}$ and $\lambda > 0$ is the regularization parameter. To make this problem decentralized, we pick n agents and uniformly divide the data set $\mathcal{D}$ among the n agents, i.e., each agent is assigned with $d = D/n$ data points. Therefore, (38) can be decomposed as follows:

$$\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x}) = \sum_{i=1}^{n} f_i(\mathbf{x}), \tag{39}$$

where the local function corresponding to agent $i \in [n]$ is

$$f_i(\mathbf{x}) = \|\mathbf{A}_i \mathbf{x} - \mathbf{b}_i\|^2 + \frac{\lambda}{2n}\|\mathbf{x}\|^2, \tag{40}$$

and

$$\mathbf{A}_i = [\mathbf{a}_{(i-1)d+1}; \cdots; \mathbf{a}_{id}] \in \mathbb{R}^{d\times p}, \tag{41}$$

$$\mathbf{b}_i = [b_{(i-1)d+1}; \cdots; b_{id}] \in \mathbb{R}^{d}. \tag{42}$$

The unique solution to (39) is

$$\widetilde{\mathbf{x}}^* = \left(\sum_{i=1}^{n} \mathbf{A}_i^\top \mathbf{A}_i + \lambda I\right)^{-1}\left(\sum_{i=1}^{n} \mathbf{A}_i^\top \mathbf{b}_i\right). \tag{43}$$

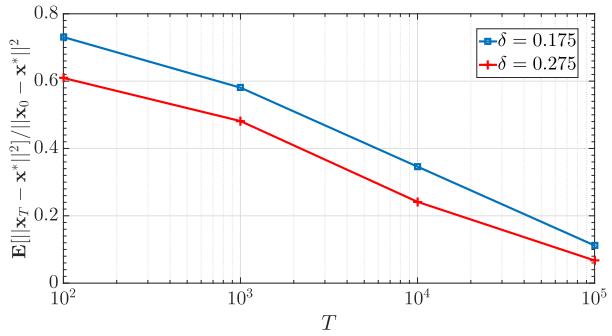

Fig. 3. Relative optimal squared error for two vales of δ: $\delta \in \{0.175, 0.275\}$.

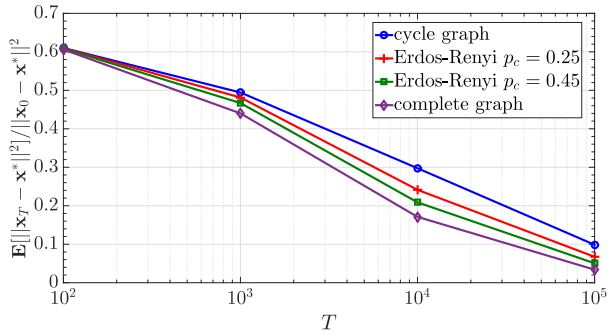

Fig. 4. Relative optimal squared error for Erdős-Rényi random graphs with two vales of graph connectivity ratio: $p_c \in \{0.25, 0.45\}$, complete graph and cycle graph.

To simulate the decentralized ridge regression (39), we pick "Pen-Based Recognition of Handwritten Digits Data Set" [37] and use $D = 5000$ training samples with $p = 16$ features and 10 possible labels corresponding to digits $\{\text{'0'}, \text{'1'}, \ldots, \text{'9'}\}$. We pick $\lambda = 2$ and consider a connected Erdős-Rényi graph with $n = 50$ agents and edge probability p_c, i.e. each assigned with $d = 100$ data points. The decision variables are quantized according to the low-precision quantizer with quantization level s, as described in Example 2.

Firstly, we fix $p_c = 0.25$ and $s = 1$ and vary the tuning parameter δ. Fig. 3 depicts the convergence trend corresponding to two values $\delta \in \{0.175, 0.275\}$. For each pick of δ, the stepsizes are set to $\varepsilon = c_1/T^{3\delta/2}$ and $\alpha = c_2/T^{\delta/2}$ with finely tuned constants c_1, c_2.

Secondly, to observe the effect of graph density, we let the quantization level be $s = 1$ and vary the graph configuration. For $\delta = 0.275$, Fig. 4 shows the resulting convergence rates for Erdős-Rényi random graphs with two vales of graph connectivity ratio $p_c \in \{0.25, 0.45\}$, complete graph and cycle graph.

C. Logistic Regression

To further evaluate the proposed method with other benchmarks, in this section we consider the logistic regression where the goal is to learn a classifier $\mathbf{x}$ to predict the labels $b_j \in \{+1, -1\}$. More specifically, consider the regularized logistic regression problem as follows:

$$\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x}) = \frac{1}{n}\sum_{j=1}^{D} \log\left(1 + \exp\left(-b_j \mathbf{a}_j \mathbf{x}\right)\right) + \frac{\lambda}{2}\|\mathbf{x}\|_2^2, \quad (44)$$

where $b_j \in \{+1, -1\}$ denotes the label of the jth data-point corresponding to the feature vector $\mathbf{a}_j \in \mathbb{R}^{1\times p}$. The total D data-points are distributed among the n nodes such that each node is assigned with $d = D/n$ samples. The underlying network is an Erdös-Rényi graph with $n = 50$ nodes and connectivity probability $p_c = 0.45$. We generate a data-set of $D = 5000$ samples as follows. Each sample with label $+1$ is associated with a feature vector of $p = 4$ random gaussian entries with mean μ and variance γ^2. Similarly, samples with labels -1 are associated with a feature vector of random gaussian entries with mean $-\mu$ and variance γ^2. We let $\mu = 3$ and $\gamma^2 = 1$.

In the implementation of the QDGD method, we pick the parameter $\delta = 0.45$ and accordingly pick the stepsizes $\varepsilon = c_1/T^{3\delta/2}$ and $\alpha = c_2/T^{\delta/2}$ where the constants c_1, c_2 are finely tuned.

As a benchmark, we compare our proposed QDGD method with the naive DGD algorithm [11] in which we let the nodes exchange quantized decision variables. That is, the update rule at node i and iteration t in this benchmark is

$$\mathbf{x}_{i,t+1} = w_{ii}\mathbf{x}_{i,t} + \sum_{j\in\mathcal{N}_i} w_{ij}\mathbf{z}_{j,t} - \alpha\nabla f_i(\mathbf{x}_{i,t}), \quad (45)$$

where we pick the stepsize $\alpha = c/T$ with finely tuned constant c.

In both methods, we use the low-precision quantizer in (22) with s levels of quantization. Note that unlike the proposed QDGD, the update rule in this benchmark employs only one stepsize α. In addition to this comparison, we illustrate the effect of the quantization level s on the convergence of the two methods. Fig. 5 demonstrates the loss values resulting from the two methods for five picks of $T \in \{750, 1000, 1250, 1500, 1750\}$. As we mentioned earlier, the proposed QDGD is an exact method, i.e. the local models converge to the global optimal model with any desired optimality gap. However, a naive generalization of the existing methods (e.g. DGD) with quantization (e.g. in (45)) will result in a convergence to a neighborhood of the global optimal.

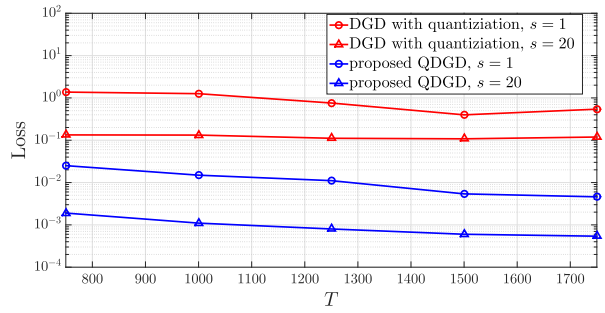

Fig. 5. Comparing the proposed QDGD method and the naive DGD with quantization (see (45)); and varying the quantizations levels $s \in \{1, 20\}$.

Fig. 5 also shows that for less noisy quantizers (larger s), nodes receive more accurate models from the neighbors and hence they achieve a smaller loss within a fixed number of iterations.

VII. Conclusion

We proposed the QDGD algorithm to tackle the problem of quantized decentralized consensus optimization. The algorithm updates the local decision variables by combining the quantized messages received from the neighbors and the local information such that proper averaging is performed over the local decision variable and the neighbors' quantized vectors. Under customary conditions for quantizers, we proved that the QDGD algorithm achieves a vanishing consensus error in mean-squared sense, and verified our theoretical results with numerical studies. Following our preliminary work [1], there has been a growing interest in developing quantized decentralized optimization methods [38]–[40]. In particular, in [38] authors propose to use *adaptive* quantization which is kept tuned during the convergence. Authors in [40] relax the convexity assumption and develop another quantized method for a more general class of objective functions.

An interesting future direction is to establish a fundamental trade-off between the convergence rate of quantized consensus algorithms and the communication. More precisely, given a target convergence rate, what is the minimum number of bits that one should communicate in decentralized consensus? Another interesting line of research is to develop novel source coding (quantization) schemes that have low computation complexity and are information theoretically near-optimal in the sense that they have small communication load and fast convergence rate. Lastly, developing such communication-efficient decentralized optimization methods for convex or non-convex functions are highly critical given the rise of deep neural networks in the learning literature, which is another line in our future directions.

Appendix A
Proof of Lemma 1

To prove the claim in Lemma 1 we first prove the following intermediate lemma.

Lemma 3: Consider the non-negative sequence e_t satisfying the inequality

$$e_{t+1} \leq \left(1 - \frac{a}{T^{2\delta}}\right) e_t + \frac{b}{T^{3\delta}}, \tag{46}$$

for $t \geq 0$, where a and b are positive constants, $\delta \in [0, 1/2)$, and T is the total number of iterations. Then, after $T \geq \max\{a^{1/(2\delta)}, \exp(\exp(1/(1-2\delta)))\}$ iterations the iterate e_T satisfies

$$e_T \leq \mathcal{O}\left(\frac{b}{aT^{\delta}}\right). \tag{47}$$

Proof: Use the expression in (46) for steps $t-1$ and t to obtain

$$e_{t+1} \leq \left(1 - \frac{a}{T^{2\delta}}\right)^2 e_{t-1} + \left[1 + \left(1 - \frac{a}{T^{2\delta}}\right)\right] \frac{b}{T^{3\delta}}, \tag{48}$$

where $T \geq a^{1/(2\delta)}$. By recursively applying these inequalities for all steps $t \geq 0$ we obtain that

$$\begin{aligned} e_t &\leq \left(1 - \frac{a}{T^{2\delta}}\right)^t e_0 \\ &\quad + \frac{b}{T^{3\delta}} \left[1 + \left(1 - \frac{a}{T^{2\delta}}\right) + \cdots + \left(1 - \frac{a}{T^{2\delta}}\right)^{t-1}\right] \\ &\leq \left(1 - \frac{a}{T^{2\delta}}\right)^t e_0 + \frac{b}{T^{3\delta}} \left[\sum_{s=0}^{t-1} \left(1 - \frac{a}{T^{2\delta}}\right)^s\right] \\ &\leq \left(1 - \frac{a}{T^{2\delta}}\right)^t e_0 + \frac{b}{T^{3\delta}} \left[\sum_{s=0}^{\infty} \left(1 - \frac{a}{T^{2\delta}}\right)^s\right] \\ &= \left(1 - \frac{a}{T^{2\delta}}\right)^t e_0 + \frac{b}{T^{3\delta}} \left[\frac{1}{1 - \left(1 - \frac{a}{T^{2\delta}}\right)}\right] \\ &= \left(1 - \frac{a}{T^{2\delta}}\right)^t e_0 + \frac{b}{aT^{\delta}}. \end{aligned} \tag{49}$$

Therefore, for the iterate corresponding to step $t = T$ we can write

$$\begin{aligned} e_T &\leq \left(1 - \frac{a}{T^{2\delta}}\right)^T e_0 + \frac{b}{aT^{\delta}} \\ &\leq \exp\left(-aT^{(1-2\delta)}\right) e_0 + \frac{b}{aT^{\delta}} \qquad (50) \\ &= \mathcal{O}\left(\frac{b}{aT^{\delta}}\right), \qquad (51) \end{aligned}$$

and the claim in (47) follows. Note that for the last inequality we assumed that the exponential term in is negligible comparing to the sublinear term. It can be verified for instance if $1 - 2\delta$ is of $\mathcal{O}\left(1/\log(\log(T))\right)$ or greater than that, it satisfies this condition. Moreover, setting $\delta = 1/2$ results in a constant (and hence non-vanishing) term in (50). ■

Now we are at the right position to prove Lemma 1. We start by evaluating the gradient function of h_α at the concatenation of local variables at time $t \geq 1$, that is $\nabla h_\alpha(\mathbf{x}_t) = \big(\mathbf{I} - \mathbf{W}\big)\mathbf{x}_t + \alpha \nabla F(\mathbf{x}_t)$. Consider the vector $\mathbf{z}_t = [\mathbf{z}_{1,t}; \ldots; \mathbf{z}_{n,t}]$ as the concatenation of the quantized variant of the local updates $\mathbf{x}_t = [\mathbf{x}_{1,t}; \ldots; \mathbf{x}_{n,t}]$. Then, we obtain that the expression on the right hand side of (12), i.e.,

$$\widetilde{\nabla} h_\alpha(\mathbf{x}_t) = \big(\mathbf{W}_D - \mathbf{W}\big)\mathbf{z}_t + \big(\mathbf{I} - \mathbf{W}_D\big)\mathbf{x}_t + \alpha \nabla F(\mathbf{x}_t), \tag{52}$$

defines a stochastic estimate of the true gradient of h_α at time t, i.e., $\nabla h_\alpha(\mathbf{x}_t)$. We let $\mathcal{F}^t$ denote a sigma algebra that measures the history of the system up until time t and take the conditional expectation $\mathbb{E}[\cdot|\mathcal{F}^t]$ from both sides of (52). It yields

$$\begin{aligned} &\mathbb{E}\left[\widetilde{\nabla} h_\alpha(\mathbf{x}_t)|\mathcal{F}^t\right] \\ &\quad = \left(\mathbf{W}_D - \mathbf{W}\right) \mathbb{E}\left[\mathbf{z}_t|\mathcal{F}^t\right] + \left(\mathbf{I} - \mathbf{W}_D\right) \mathbf{x}_t + \alpha \nabla F(\mathbf{x}_t), \\ &\quad = \left(\mathbf{I} - \mathbf{W}\right) \mathbf{x}_t + \alpha \nabla F(\mathbf{x}_t) \\ &\quad = \nabla h_\alpha(\mathbf{x}_t), \end{aligned} \tag{53}$$

where we used the fact that $\mathbb{E}\left[\mathbf{z}_t|\mathcal{F}^t\right] = \mathbf{x}_t$ (Assumption 3). Hence, $\widetilde{\nabla} h_\alpha$ is an unbiased estimator for the true gradient ∇h_α.

Now, we can rewrite the update rule (12) as

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \varepsilon \widetilde{\nabla} h_\alpha(\mathbf{x}_t), \tag{54}$$

which resembles the stochastic gradient descent (SGD) update with step-size ε for minimizing the objective function $h_\alpha(\mathbf{x})$ over $\mathbf{x} \in \mathbb{R}^{np}$. Intuitively, one can expect that, for proper pick of step-size, the the sequence $\{\mathbf{x}_t; t = 1, 2, \dots\}$ produced by update rule (54) converges to the unique minimizer of $h_\alpha(\mathbf{x})$. More precisely, we can write for $t \geq 1$,

$$\begin{aligned}
&\mathbb{E}\left[\|\mathbf{x}_{t+1} - \mathbf{x}_\alpha^*\|^2 | \mathcal{F}^t\right] \\
&= \mathbb{E}\left[\|\mathbf{x}_t - \varepsilon \widetilde{\nabla} h_\alpha(\mathbf{x}_t) - \mathbf{x}_\alpha^*\|^2 | \mathcal{F}^t\right] \\
&= \|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 - 2\varepsilon \left\langle \mathbf{x}_t - \mathbf{x}_\alpha^*, \mathbb{E}\left[\widetilde{\nabla} h_\alpha(\mathbf{x}_t) | \mathcal{F}^t\right] \right\rangle \\
&\quad + \varepsilon^2 \mathbb{E}\left[\|\widetilde{\nabla} h_\alpha(\mathbf{x}_t)\|^2 | \mathcal{F}^t\right] \\
&= \|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 - 2\varepsilon \langle \mathbf{x}_t - \mathbf{x}_\alpha^*, \nabla h_\alpha(\mathbf{x}_t) \rangle \\
&\quad + \varepsilon^2 \mathbb{E}\left[\|\widetilde{\nabla} h_\alpha(\mathbf{x}_t)\|^2 | \mathcal{F}^t\right] \\
&\leq (1 - 2\mu_\alpha \varepsilon) \|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 + \varepsilon^2 \mathbb{E}\left[\|\widetilde{\nabla} h_\alpha(\mathbf{x}_t)\|^2 | \mathcal{F}^t\right].
\end{aligned} \tag{55}$$

We have used the facts that $\widetilde{\nabla} h_\alpha$ is unbiased and h_α is strongly convex with parameter μ_α. Next, we bound the second term in (55), that is

$$\begin{aligned}
&\mathbb{E}\left[\|\widetilde{\nabla} h_\alpha(\mathbf{x}_t)\|^2 | \mathcal{F}^t\right] \\
&= \mathbb{E}\left[\|(\mathbf{W}_D - \mathbf{W})\, \mathbf{z}_t + (\mathbf{I} - \mathbf{W}_D)\, \mathbf{x}_t + \alpha \nabla F(\mathbf{x}_t)\|^2 | \mathcal{F}^t\right] \\
&\leq \|\nabla h_\alpha(\mathbf{x}_t)\|^2 + \mathbb{E}\left[\|(\mathbf{W}_D - \mathbf{W})\, (\mathbf{z}_t - \mathbf{x}_t)\|^2 | \mathcal{F}^t\right] \\
&\leq L_\alpha^2 \|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 + n\sigma^2 \|W - W_D\|^2,
\end{aligned} \tag{56}$$

where we used the smoothness of h_α and boundedness of quantization noise. Plugging (56) into (55) yields

$$\begin{aligned}
\mathbb{E}\left[\|\mathbf{x}_{t+1} - \mathbf{x}_\alpha^*\|^2 | \mathcal{F}^t\right] \leq \left(1 - 2\mu_\alpha \varepsilon + \varepsilon^2 L_\alpha^2\right) \|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 \\
+ \varepsilon^2 n\sigma^2 \|W - W_D\|^2.
\end{aligned} \tag{57}$$

Let us define the sequence $e_t := \mathbb{E}\left[\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2\right]$ as the expected squared deviation of the local variables from the optimal solution $\mathbf{x}_\alpha^*$ at time $t \geq 1$. By taking the expectation of both sides of (57) with respect to all sources of randomness from $t = 0$ we obtain that

$$\begin{aligned}
e_{t+1} &\leq \left(1 - 2\mu_\alpha \varepsilon + \varepsilon^2 L_\alpha^2\right) e_t + \varepsilon^2 n\sigma^2 \|W - W_D\|^2 \\
&= \left(1 - \varepsilon(2\mu_\alpha - \varepsilon L_\alpha^2)\right) e_t + \varepsilon^2 n\sigma^2 \|W - W_D\|^2.
\end{aligned} \tag{58}$$

Notice that for the specified choice of ε and $T \geq T_1$, we have $T^\delta \geq T_1^\delta \geq \frac{c_1(1+c_2 L)^2}{c_2 \mu}$ and therefore

$$\begin{aligned}
\varepsilon &= \frac{c_1}{T^{3\delta/2}} \\
&\leq \frac{c_2 \mu}{(1 + c_2 L)^2} \cdot \frac{1}{T^{\delta/2}} \\
&\leq \frac{\mu_\alpha}{\left(1 - \lambda_n(W) + \alpha L\right)^2} \\
&\leq \frac{\mu_\alpha}{L_\alpha^2}.
\end{aligned} \tag{59}$$

Therefore, (58) can be written as

$$\begin{aligned}
e_{t+1} &\leq \left(1 - \varepsilon\left(2\mu_\alpha - \varepsilon L_\alpha^2\right)\right) e_t + \varepsilon^2 n\sigma^2 \|W - W_D\|^2 \\
&\leq (1 - \mu_\alpha \varepsilon)\, e_t + \varepsilon^2 n\sigma^2 \|W - W_D\|^2 \\
&= \left(1 - \frac{c_1 c_2 \mu}{T^{2\delta}}\right) e_t + \frac{c_1^2 n\sigma^2 \|W - W_D\|^2}{T^{3\delta}}.
\end{aligned} \tag{60}$$

Now we let $a = c_1 c_2 \mu$ and $b = c_1^2 n\sigma^2 \|W - W_D\|^2$ and employ Lemma 3 to conclude that

$$\begin{aligned}
e_T &= \mathbb{E}\left[\|\mathbf{x}_T - \mathbf{x}_\alpha^*\|^2\right] \\
&\leq \mathcal{O}\left(\frac{b}{aT^\delta}\right) \\
&= \mathcal{O}\left(\frac{c_1 n\sigma^2 \|W - W_D\|^2}{\mu c_2} \frac{1}{T^\delta}\right),
\end{aligned} \tag{61}$$

and the proof of Lemma 1 is complete.

Appendix B
Proof of Lemma 2

First, recall the penalty function minimization in (15). Following sequence is the update rule associated with this problem when the gradient descent method is applied to the objective function h_α with the unit step-size $\gamma = 1$,

$$\mathbf{u}_{t+1} = \mathbf{u}_t - \gamma \nabla h_\alpha(\mathbf{u}_t) = \mathbf{W}\mathbf{u}_t - \alpha \nabla F(\mathbf{u}_t). \tag{62}$$

From analysis of GD for strongly convex objectives, the sequence $\{\mathbf{u}_t : t = 0, 1, \cdots\}$ defined above exponentially converges to the minimizer of h_α, $\mathbf{x}_\alpha^*$, provided that $1 = \gamma \leq \frac{2}{L_\alpha}$. The latter condition is satisfied if we make $\alpha \leq \frac{1 + \lambda_n(W)}{L}$, implying $L_\alpha = 1 - \lambda_n(W) + \alpha L \leq 2$. Therefore,

$$\begin{aligned}
\|\mathbf{u}_t - \mathbf{x}_\alpha^*\|^2 &\leq (1 - \mu_\alpha)^t \|\mathbf{u}_0 - \mathbf{x}_\alpha^*\|^2 \\
&= (1 - \alpha\mu)^t \|\mathbf{u}_0 - \mathbf{x}_\alpha^*\|^2.
\end{aligned} \tag{63}$$

If we take $\mathbf{u}_0 = 0$, then (63) implies

$$\begin{aligned}
\|\mathbf{u}_T - \mathbf{x}_\alpha^*\|^2 &\leq (1 - \alpha\mu)^T \|\mathbf{x}_\alpha^*\|^2 \\
&\leq 2(1 - \alpha\mu)^T \left(\|\mathbf{x}^* - \mathbf{x}_\alpha^*\|^2 + \|\mathbf{x}^*\|^2\right) \\
&= 2(1 - \alpha\mu)^T \left(\|\mathbf{x}^* - \mathbf{x}_\alpha^*\|^2 + n\|\widetilde{\mathbf{x}}^*\|^2\right),
\end{aligned} \tag{64}$$

where $f_0 = f(0)$ and $f^* = \min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x}) = f(\widetilde{\mathbf{x}}^*)$. On the other hand, it can be shown [11] that if $\alpha \leq \min\{\frac{1 + \lambda_n(W)}{L}, \frac{1}{\mu + L}\}$, then the sequence $\{\mathbf{u}_t : t = 0, 1, \cdots\}$ defined in (63) converges to the $\mathcal{O}(\frac{\alpha}{1 - \beta})$-neighborhood of the optima $\mathbf{x}^*$, i.e.,

$$\|\mathbf{u}_t - \mathbf{x}^*\| \leq \mathcal{O}\left(\frac{\alpha}{1 - \beta}\right). \tag{65}$$

If we take $\alpha = \frac{c_2}{T^{\delta/2}}$, the condition $T \geq T_2$ implies that $\alpha \leq \min\{\frac{1+\lambda_n(W)}{L}, \frac{1}{\mu+L}\}$. Therefore, (65) yields

$$\|\mathbf{u}_T - \mathbf{x}^*\| \leq \mathcal{O}\left(\frac{\alpha}{1-\beta}\right). \tag{66}$$

More precisely, we have the following (See Corollary 9 in [11]):

$$\|\mathbf{u}_T - \mathbf{x}^*\| \leq \sqrt{n}\left(c_3^T\|\widetilde{\mathbf{x}}^*\| + \frac{c_4}{\sqrt{1-c_3^2}} + \frac{\alpha D}{1-\beta}\right), \tag{67}$$

where

$$c_3^2 = 1 - \frac{1}{2}\cdot\frac{\mu L}{\mu+L}\alpha, \tag{68}$$

$$\begin{aligned}\frac{c_4}{\sqrt{1-c_3^2}} &= \frac{\alpha L D}{1-\beta}\sqrt{4\left(\frac{\mu+L}{\mu L}\right)^2 - 2\cdot\frac{\mu+L}{\mu L}\alpha} \\ &\leq \frac{2\alpha D}{(1-\beta)}\left(1+L/\mu\right).\end{aligned} \tag{69}$$

From (67) and (66), we have for $T \geq T_2$

$$\begin{aligned}\|\mathbf{x}_\alpha^* - \mathbf{x}^*\|^2 &= \|\mathbf{x}_\alpha^* - \mathbf{u}_T + \mathbf{u}_T - \mathbf{x}^*\|^2 \\ &\leq 2\|\mathbf{x}_\alpha^* - \mathbf{u}_T\|^2 + 2\|\mathbf{u}_T - \mathbf{x}^*\|^2 \\ &\leq 4(1-\alpha\mu)^T\left(\|\mathbf{x}^* - \mathbf{x}_\alpha^*\|^2 + n\|\widetilde{\mathbf{x}}^*\|^2\right) \\ &\quad + 2n\Bigg(\left(1 - \frac{1}{2}\cdot\frac{\mu L}{\mu+L}\alpha\right)^{T/2}\|\widetilde{\mathbf{x}}^*\| \\ &\quad + \frac{\alpha D}{1-\beta}\left(3+2L/\mu\right)\Bigg)^2.\end{aligned} \tag{70}$$

Note that for our pick $\alpha = \frac{c_2}{T^{\delta/2}}$, we can write

$$\begin{aligned}(1-\alpha\mu)^T &\leq \exp\left(-c_2 T^{1-\delta/2}\right) =: e_1(T), \\ \left(1 - \frac{1}{2}\cdot\frac{\mu L}{\mu+L}\alpha\right)^{T/2} &\leq \exp\left(-\frac{1}{2}\cdot\frac{\mu L}{\mu+L}c_2 T^{1-\delta/2}\right) \\ &=: e_2(T).\end{aligned} \tag{71}$$

Therefore, from (70) we have

$$\begin{aligned}\|\mathbf{x}_\alpha^* - \mathbf{x}^*\|^2 &\leq \frac{1}{(1-4e_1(T))}\Bigg\{4e_1(T)n\|\widetilde{\mathbf{x}}^*\|^2 \\ &\quad + 2ne_2^2(T)\|\widetilde{\mathbf{x}}^*\|^2 \\ &\quad + 4ne_2(T)\|\widetilde{\mathbf{x}}^*\|\frac{\alpha D}{1-\beta}\left(3+2L/\mu\right) \\ &\quad + 2nD^2\left(3+2L/\mu\right)^2\left(\frac{\alpha}{1-\beta}\right)^2\Bigg\} \\ &\leq \frac{4n\left(2e_1(T)+e_2^2(T)\right)}{(1-4e_1(T))}\frac{f_0-f^*}{\mu} \\ &\quad + \frac{4\sqrt{2}ne_2(T)}{(1-4e_1(T))}\sqrt{\frac{f_0-f^*}{\mu}}\frac{\alpha D}{1-\beta}\left(3+2L/\mu\right) \\ &\quad + \frac{2nD^2\left(3+2L/\mu\right)^2}{(1-4e_1(T))}\left(\frac{\alpha}{1-\beta}\right)^2,\end{aligned} \tag{72}$$

where we used the fact that $\|\widetilde{\mathbf{x}}^*\|^2 \leq 2(f_0 - f^*)/\mu$. Let $B_1(T)$ denote the bound in RHS of (72). Given the fact that the terms $e_1(T)$ and $e_2(T)$ decay exponentially, i.e. $e_1(T) = o\left(\alpha^2\right)$ and $e_2(T) = o\left(\alpha^2\right)$, we have

$$\begin{aligned}\|\mathbf{x}_\alpha^* - \mathbf{x}^*\| &\leq \mathcal{O}\left(\sqrt{2n}D\left(3+2L/\mu\right)\left(\frac{\alpha}{1-\beta}\right)\right) \\ &= \mathcal{O}\left(\frac{\sqrt{2n}c_2 D\left(3+2L/\mu\right)}{1-\beta}\frac{1}{T^{\delta/2}}\right)\end{aligned} \tag{73}$$

which concludes the claim in Lemma 2. Moreover, due to the exponential decay of the two terms $e_1(T)$ and $e_2(T)$, we have

$$B_1(T) \approx 2nD^2\left(3+2L/\mu\right)^2\left(\frac{\alpha}{1-\beta}\right)^2 \tag{74}$$

$$= \frac{2nc_2^2 D^2\left(3+2L/\mu\right)^2}{(1-\beta)^2}\frac{1}{T^\delta}. \tag{75}$$

Appendix C
Proof of Theorem 2

Note that the steps of the proof are similar to the one for Theorem 1. There, we derived the convergence rate of each worker, i.e. $\mathbb{E}[\|\mathbf{x}_{i,T} - \widetilde{\mathbf{x}}^*\|^2]$ by bounding two quantities $\mathbb{E}[\|\mathbf{x}_T - \mathbf{x}_\alpha^*\|^2]$ and $\|\mathbf{x}_\alpha^* - \mathbf{x}^*\|$ as in Lemma 1 and 2 respectively. Here, replacing Assumption 3 by Assumption 5 acquires only the former quantity to revisit. From (55), we have that for $t \geq 1$,

$$\begin{aligned}\mathbb{E}\left[\|\mathbf{x}_{t+1} - \mathbf{x}_\alpha^*\|^2|\mathcal{F}^t\right] &\leq (1-2\mu_\alpha\varepsilon)\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 \\ &\quad + \varepsilon^2\mathbb{E}\left[\|\widetilde{\nabla}h_\alpha(\mathbf{x}_t)\|^2|\mathcal{F}^t\right].\end{aligned} \tag{76}$$

Considering Assumption 5, the second term in RHS of (56) can be bounded as follows,

$$\begin{aligned}&\mathbb{E}\left[\|\widetilde{\nabla}h_\alpha(\mathbf{x}_t)\|^2|\mathcal{F}^t\right] \\ &= \mathbb{E}\left[\|(\mathbf{W}_D - \mathbf{W})\,\mathbf{z}_t + (\mathbf{I} - \mathbf{W}_D)\,\mathbf{x}_t + \alpha\nabla F(\mathbf{x}_t)\|^2|\mathcal{F}^t\right] \\ &\leq \|\nabla h_\alpha(\mathbf{x}_t)\|^2 + \mathbb{E}\left[\|(\mathbf{W}_D - \mathbf{W})\,(\mathbf{z}_t - \mathbf{x}_t)\|^2|\mathcal{F}^t\right] \\ &\leq L_\alpha^2\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 + \eta^2\|W - W_D\|^2\|\mathbf{x}_t\|^2 \\ &= L_\alpha^2\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 + \eta^2\|W - W_D\|^2\|\mathbf{x}_t - \mathbf{x}_\alpha^* + \mathbf{x}_\alpha^*\|^2 \\ &\leq \left(L_\alpha^2 + 2\eta^2\|W - W_D\|\right)\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 \\ &\quad + 2\eta^2\|W - W_D\|^2\|\mathbf{x}_\alpha^*\|^2.\end{aligned} \tag{77}$$

Moreover, since the solution to Problem (1), i.e. $\|\widetilde{\mathbf{x}}^*\|$ (hence $\|\mathbf{x}^*\|$) is assumed to be bounded, the (unique) minimizer of $h_\alpha(\cdot)$, i.e. $\|\mathbf{x}_\alpha^*\|$ is also bounded as follows,

$$\begin{aligned}\|\mathbf{x}_\alpha^*\|^2 &= \|\mathbf{x}_\alpha^* - \mathbf{x}^* + \mathbf{x}^*\|^2 \\ &\leq 2\|\mathbf{x}_\alpha^* - \mathbf{x}^*\|^2 + 2\|\mathbf{x}^*\|^2\end{aligned}$$

$$\leq 2B_1(T) + \frac{4n(f_0 - f^*)}{\mu}$$
$$\leq 2B_1(1) + \frac{4n(f_0 - f^*)}{\mu} =: n\widetilde{B}^2. \quad (78)$$

Plugging (77) and (78) into (76) yields

$$\mathbb{E}\left[\|\mathbf{x}_{t+1} - \mathbf{x}_\alpha^*\|^2 | \mathcal{F}^t\right] \leq \left(1 - 2\mu_\alpha\varepsilon + \varepsilon^2\left(L_\alpha^2 + +2\eta^2\|W - W_D\|^2\right)\right)\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 + \varepsilon^2 n\widetilde{B}^2\|W - W_D\|^2. \quad (79)$$

Let us pick

$$\widetilde{T}_1 := \max\left\{e^{e^{\frac{1}{1-2\delta}}}, \left\lceil (c_1c_2\mu)^{1/(2\delta)}\right\rceil, \left\lceil\left(\frac{c_1((2+c_2L)^2 + 2\eta^2\|W - W_D\|^2)}{c_2\mu}\right)^{1/\delta}\right\rceil\right\}. \quad (80)$$

For $T \geq \widetilde{T}_1$, we have

$$\varepsilon = \frac{c_1}{T^{3\delta/2}} \leq \frac{c_2\mu}{(2+c_2L)^2 + 2\eta^2\|W - W_D\|^2} \cdot \frac{1}{T^{\delta/2}} \leq \frac{\mu_\alpha}{\left(1 - \lambda_n(W) + \alpha L\right)^2 + 2\eta^2\|W - W_D\|^2} = \frac{\mu_\alpha}{L_\alpha^2 + 2\eta^2\|W - W_D\|^2}, \quad (81)$$

which together with (79) yields

$$\mathbb{E}\left[\|\mathbf{x}_{t+1} - \mathbf{x}_\alpha^*\|^2\right] \leq (1 - \mu_\alpha\varepsilon)\mathbb{E}\left[\|\mathbf{x}_{t+1} - \mathbf{x}_\alpha^*\|^2\right] + 2\varepsilon^2 n\widetilde{B}^2\eta^2\|W - W_D\|^2. \quad (82)$$

Finally, from Lemma 3 with $a = c_1c_2\mu$ and $b = 2c_1^2 n\widetilde{B}^2\eta^2\|W - W_D\|^2$, we have that

$$\mathbb{E}\left[\|\mathbf{x}_T - \mathbf{x}_\alpha^*\|^2\right] \leq \frac{2c_1 n\widetilde{B}^2\eta^2\|W - W_D\|^2}{\mu c_2}\frac{1}{T^\delta} + \exp\left(-c_1c_2\mu T^\delta\right)\sqrt{n}\widetilde{B}. \quad (83)$$

Let $B_2(T)$ denote the bound in RHS of (83). Due to the exponential decay of the second term in $B_2(T)$, we have

$$\mathbb{E}\left[\|\mathbf{x}_T - \mathbf{x}_\alpha^*\|^2\right] \leq \mathcal{O}\left(\frac{2c_1 n\widetilde{B}^2\eta^2\|W - W_D\|^2}{\mu c_2}\frac{1}{T^\delta}\right), \quad (84)$$

and

$$B_2(T) \approx \frac{2c_1 n\widetilde{B}^2\eta^2\|W - W_D\|^2}{\mu c_2}\frac{1}{T^\delta}. \quad (85)$$

Hence, by putting (84) together with Lemma 2 we conclude the claim for any $T \geq \widetilde{T}_0 := \max\{\widetilde{T}_1, T_2\}$.

REFERENCES

[1] A. Reisizadeh, A. Mokhtari, H. Hassani, and R. Pedarsani, "Quantized decentralized consensus optimization," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 5838–5843.

[2] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Trans. Ind. Inform.*, vol. 9, no. 1, pp. 427–438, Feb. 2013.

[3] A. Ribeiro, "Ergodic stochastic optimization algorithms for wireless communication and networking," *IEEE Trans. Signal Process.*, vol. 58, no. 12, pp. 6369–6386, Dec. 2010.

[4] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proc. 3rd Int. Symp. Inf. Process. Sensor Netw.*, 2004, pp. 20–27.

[5] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[6] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control Syst. Mag.*, vol. 27, no. 2, pp. 71–82, Apr. 2007.

[7] K. I. Tsianos, S. Lawlor, and M. G. Rabbat, "Consensus-based distributed optimization: Practical issues and applications in large-scale machine learning," in *Proc. 50th Annu. Alleton Conf. Commun., Control, Comput.*, 2012, pp. 1543–1550.

[8] 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.

[9] J. N. Tsitsiklis, "Problems in decentralized decision making and computation," Ph.D. dissertation, Lab. Inf. Decis. Syst., Massachusetts Inst. Technol., Cambridge, MA, USA, 1984.

[10] A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "On distributed averaging algorithms and quantization effects," *IEEE Trans. Autom. Control*, vol. 54, no. 11, pp. 2506–2517, Nov. 2009.

[11] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," *SIAM J. Optim.*, vol. 26, no. 3, pp. 1835–1854, 2016.

[12] D. Jakovetić, J. Xavier, and J. M. Moura, "Fast distributed gradient methods," *IEEE Trans. Autom. Control*, vol. 59, no. 5, pp. 1131–1146, May 2014.

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

[14] S. Boyd *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

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

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

[17] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Trans. Autom. Control*, vol. 57, no. 3, pp. 592–606, Mar. 2012.

[18] K. I. Tsianos, S. Lawlor, and M. G. Rabbat, "Push-sum distributed dual averaging for convex optimization," in *Proc. IEEE 51st Annu. Conf. Decis. Control*, 2012, pp. 5453–5458.

[19] E. Wei, A. Ozdaglar, and A. Jadbabaie, "A distributed Newton method for network utility maximization–I: Algorithm," *IEEE Trans. Autom. Control*, vol. 58, no. 9, pp. 2162–2175, Sep. 2013.

[20] A. Mokhtari, Q. Ling, and A. Ribeiro, "Network Newton distributed optimization methods," *IEEE Trans. Signal Process.*, vol. 65, no. 1, pp. 146–161, Jan. 2017.

[21] M. Eisen, A. Mokhtari, and A. Ribeiro, "Decentralized quasi-newton methods," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2613–2628, May 2017.

[22] F. Yan, S. Sundaram, S. Vishwanathan, and Y. Qi, "Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 11, pp. 2483–2493, Nov. 2013.

[23] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu, "1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs," in *Proc. 15th Annu. Conf. Int. Speech Commun. Assoc.*, 2014, pp. 1058–1062.

[24] M. Chowdhury, M. Zaharia, J. Ma, M. I. Jordan, and I. Stoica, "Managing data transfers in computer clusters with orchestra," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 98–109, 2011.

[25] S. Yuksel and T. Basar, "Quantization and coding for decentralized LTI systems," in *Proc. 42nd IEEE Conf. Decis. Control*, 2003, vol. 3, pp. 2847–2852.
[26] A. Kashyap, T. Basar, and R. Srikant, "Quantized consensus," in *Proc. IEEE Int. Symp. Inf. Theory*, 2006, pp. 635–639.
[27] 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.
[28] M. El Chamie, J. Liu, and T. Başar, "Design and analysis of distributed averaging with quantized communication," *IEEE Trans. Autom. Control*, vol. 61, no. 12, pp. 3870–3884, Dec. 2016.
[29] T. C. Aysal, M. Coates, and M. Rabbat, "Distributed average consensus using probabilistic quantization," in *Proc. IEEE/SP Statist. Signal Process.*, 2007, pp. 640–644.
[30] A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "Distributed subgradient methods and quantization effects," in *Proc. 47th IEEE Conf. Decis. Control*, 2008, pp. 4177–4184.
[31] E. Gravelle and S. Martínez, "Quantized distributed load balancing with capacity constraints," in *Proc. IEEE 53rd Annu. Conf. Decis. Control*, 2014, pp. 3866–3871.
[32] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-efficient SGD via gradient quantization and encoding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1707–1718.
[33] S. Li, M. A. Maddah-Ali, Q. Yu, and A. S. Avestimehr, "A fundamental tradeoff between computation and communication in distributed computing," *IEEE Trans. Inf. Theory*, vol. 64, no. 1, pp. 109–128, 2017.
[34] K. Lee, M. Lam, R. Pedarsani, D. Papailiopoulos, and K. Ramchandran, "Speeding up distributed machine learning using codes," in *Proc. IEEE Int. Symp. Inf. Theory*, 2016, pp. 1143–1147.
[35] Y. H. Ezzeldin, M. Karmoose, and C. Fragouli, "Communication vs distributed computation: An alternative trade-off curve," in *Proc. IEEE Inf. Theory Workshop*, 2017, pp. 279–283.
[36] S. Prakash, A. Reisizadeh, R. Pedarsani, and S. Avestimehr, "Coded computing for distributed graph analytics," in *Proc. IEEE Int. Symp. Inf. Theory*, 2018, pp. 1221–1225.
[37] D. Dheeru and E. Karra Taniskidou, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml
[38] T. T. Doan, S. T. Maguluri, and J. Romberg, "Accelerating the convergence rates of distributed subgradient methods with adaptive quantization," 2018, arXiv:1810.13245.
[39] C.-S. Lee, N. Michelusi, and G. Scutari, "Finite rate quantized distributed optimization with geometric convergence," in *Proc. 52nd Asilomar Conf. Signals, Syst., Comput.*, 2018, pp. 1876–1880.
[40] X. Zhang, J. Liu, Z. Zhu, and E. S. Bentley, "Compressed distributed gradient descent: Communication-efficient consensus over networks," 2018, arXiv:1812.04048.

Amirhossein Reisizadeh received the B.S. degree in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 2014, and the M.S. degree in electrical engineering from the University of California, Los Angeles, Los Angeles, CA, USA, in 2016. He is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, Santa Barbara, CA, USA. He is interested in using information and coding-theoretic concepts to develop fast and efficient algorithms for large-scale machine learning, distributed computing and optimization. He was a finalist for the Qualcomm Innovation Fellowship.

Aryan Mokhtari received the B.Sc. degree in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 2011, the M.Sc. and Ph.D. degrees in electrical and systems engineering from the University of Pennsylvania (Penn), Philadelphia, PA, USA, in 2014 and 2017, respectively, and the A.M. degree in statistics from the Wharton School, Penn, in 2017. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, USA. Prior to that he was a Postdoctoral Associate with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, from January 2018 to July 2019. Before joining MIT, he was a Research Fellow with the Simons Institute for the Theory of Computing, University of California, Berkeley, for the program on Bridging Continuous and Discrete Optimization, from August to December 2017. His research interests include the areas of optimization, machine learning, and signal processing. His current research focuses on the theory and applications of convex and non-convex optimization in large-scale machine learning, and data science problems. He was the recipient of a number of awards and fellowships, including Penns Joseph and Rosaline Wolf Award for Best Doctoral Dissertation in electrical and systems engineering and the Simons-Berkeley Fellowship.

Hamed Hassani (M'10) received the B.Sc. degree in electrical engineering and mathematics from the Sharif University of Technology, Tehran, Iran, and the Ph.D. degree in computer and communication sciences from the Swiss Federal Institute of Technology, Lausanne, Switzerland. He is currently an Assistant Professor with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA, USA. Prior to that he was a Research Fellow with the Simons Institute, UC Berkeley, and a Postdoctoral Scholar with the Institute for Machine Learning, ETH Zurich. For his Ph.D. dissertation, he received the 2014 IEEE Information Theory Society Thomas M. Cover Dissertation Award. He also was the recipient of the Jack K. Wolf Student Paper Award from the 2015 IEEE International Symposium on Information Theory.

Ramtin Pedarsani received the B.Sc. degree in electrical engineering from the University of Tehran, Tehran, Iran, in 2009, the M.Sc. degree in communication systems from the Swiss Federal Institute of Technology, Lausanne, Switzerland, in 2011, and the Ph.D. degree from the University of California, Berkeley, Berkeley, CA, USA, in 2015. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, Santa Barbara, CA, USA. His research interests include machine learning, information and coding theory, networks, and transportation systems. He was a recipient of the IEEE International Conference on Communications Best Paper Award in 2014.