Quantized Decentralized Consensus Optimization

Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, and Ramtin Pedarsani

***Abstract*— We consider the problem of decentralized consensus optimization, where the sum of n 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. 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 experimental results.**

I. Introduction

Distributed optimization of a sum of convex functions has a variety of applications in different areas including decentralized control systems [1], wireless systems [2], sensor networks [3], networked multiagent systems [4], multirobot networks [5], and large scale machine learning [6]. 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 $f_i(.)$ represents the local cost function of node i that is only known by this node.

Distributed optimization has been largely studied in the literature starting from seminal works in the 80s [7], [8]. 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 [9]–[12], distributed alternating direction method of multipliers (ADMM) [13]–[15], decentralized dual averaging [16], [17], and distributed Newton optimization method [18], [19]. Furthermore, the decentralized consensus optimization problem has been considered in online or dynamic settings, where the dynamic cost function becomes an online regret function [20], [21].

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 [22], [23].

Quantized communication for the agents is brought into the picture for bounded and stable control systems [24]. Furthermore, consensus distributed averaging algorithms are studied under discretized message passing [25]. Motivated by the energy and bandwidth-constrained wireless sensor networks, the work in [26] proposes distributed optimization algorithms under quantized variables and guarantees convergence within a non-vanishing error. Deterministic quantization has been considered in distributed averaging algorithms [27] 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 [28]. The work in [29] 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, [30] 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 [22] was introduced in which at each time step, the agents sequentially quantizie their local gradient vectors by entry-wise signs while contributing the quantization error induced in previous iteration. Moreover, in [31], 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 [32]–[35].

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 [31], we consider unbiased and variance bounded random quantizers $Q(\cdot)$, i.e. $\mathbb{E}[Q(\mathbf{x})] = \mathbf{x}$ and $\mathbb{E}[\|Q(\mathbf{x}) - \mathbf{x}\|^2] \leq \sigma^2$ for some fixed constant σ^2.

Our main contribution is to propose a Quantized Decen-

Amirhossein Reisizadeh and Ramtin Pedarsani are with the Department of Electrical and Computer Engineering at University of California, Santa Barbara reisizadeh@ucsb.edu, ramtin@ece.ucsb.edu

Aryan Mokhtari is with the Laboratory for Information and Decision Systems at Massachusetts Institute of Technology aryanm@mit.edu

Hamed Hassani is with the Department of Electrical and Systems Engineering at University of Pennsylvania hassani@seas.upenn.edu

tralized 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 $i = 1, \dots, n$ we obtain that for any arbitrary $\delta \in (0, 1)$, $\mathbb{E}[\|\mathbf{x}_{i,T} - \widetilde{\mathbf{x}}^*\|^2] \leq \mathcal{O}(\frac{1}{T^{(1-\delta)/2}})$, where $\mathbf{x}_{i,T}$ is the local decision variable of node i at iteration T and $\widetilde{\mathbf{x}}^*$ is the optimal solution. 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 provide simulation results that corroborate our theoretical results.

Notation. In this paper, we denote by $[n]$ the set $\{1, \cdots, n\}$ 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 $t \geq t_0$. we use $\lceil a \rceil$ to indicate the least integer greater than or equal to a.

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 theorem of the paper is stated and proved in Section IV. We provide numerical studies in Section V. Finally, we conclude the paper and discuss future directions in Section VI.

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 $\mathcal{V} = \{1, \cdots, n\}$ 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

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

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

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

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 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

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** $t = 0, \cdots, T-1$ **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** $\mathbf{x}_{i,T}$

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 decision variable $\mathbf{x}_{i,t+1}$ is evaluated by proper averaging over the local decision variable $\mathbf{x}_{i,t}$ and neighbors quantized vectors $\mathbf{z}_{j,t}$, and descending through the negative local gradient $\nabla f_i(\mathbf{x}_{i,t})$ with a proper stepsize. Note that quantized decision variables of the neighboring nodes contribute to the descent direction proportionally to step-size ε, unlike the noiseless local gradient which is scaled by $\alpha\varepsilon$. 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 [9], [10] 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 [9], [10], [26], [29]. QDGD improves the inexact convergence of 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.

Moreover, the proposed QDGD algorithm does not restrict the quantizer, except for few customary conditions. However, design of efficient quantizers has been taken into consideration. For instance as in [31], for any $\mathbf{x} \in \mathbb{R}^p$, a random quantization vector $Q(\mathbf{x}) = [Q_1(\mathbf{x}), \cdots, Q_p(\mathbf{x})]^\top$ is defined in its simplest variant as follows,

$$Q_i(\mathbf{x}) = \|\mathbf{x}\| \cdot \text{sgn}(\mathbf{x}_i) \cdot \xi_i(\mathbf{x}), \tag{4}$$

where $\xi_i(\mathbf{x})$'s are independent Bernoulli random variables with parameter $|\mathbf{x}_i|/\|\mathbf{x}\|$, i.e. each entry of the quantized vector takes on value in $\{0, \pm\|\mathbf{x}\|\}$. Generalized to more quatization levels, this quantizer is unbiased, variance bounded and sparse.

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 (1). The following assumptions hold through out the analysis of the algorithm.

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

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

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

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 $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$.

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

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

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

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

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 $\text{null}(I - W) = \text{span}(\mathbf{1})$.

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. Then, for each node i, the expected deviation of the output of Algorithm 1 from the solution to Problem (1) is upper bounded by

$$\mathbb{E}\left[\left\|\mathbf{x}_{i,T} - \widetilde{\mathbf{x}}^*\right\|^2\right] \leq \mathcal{O}\left(\frac{1}{T^{(1-\delta)/2}}\right), \tag{9}$$

for $\varepsilon = \frac{c_1}{T^{3(1-\delta)/4}}$, $\alpha = \frac{c_2}{T^{(1-\delta)/4}}$, any $\delta \in (0, 1)$ and $T \geq T_0$, where c_1, c_2 and T_0 are positive constants independent of T.

Remark 2. 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 by iteration.

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

$$\mathbf{x}_{i,t+1} = \mathbf{x}_{i,t} - \varepsilon_t \Big((1-w_{ii})\mathbf{z}_{i,t} - \sum_{j\neq i} w_{ij}\mathbf{z}_{j,t} + \alpha \nabla f_i(\mathbf{x}_{i,t}) \Big). \tag{10}$$

The next step is to write the update (10) 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 $\mathbf{x} = [\mathbf{x}_1; \cdots; \mathbf{x}_n] \in \mathbb{R}^{np}$. 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 (10) as follows,

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \varepsilon \Big(\big(\mathbf{W}_D - \mathbf{W}\big)\mathbf{z}_t + \big(\mathbf{I} - \mathbf{W}_D\big)\mathbf{x}_t + \alpha \nabla F(\mathbf{x}_t) \Big). \tag{11}$$

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,

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^{np}} \quad & F(\mathbf{x}) = \sum_{i=1}^{n} f_i(\mathbf{x}_i) \\ \text{subject to} \quad & (\mathbf{I} - \mathbf{W})^{1/2}\mathbf{x} = 0. \end{aligned} \tag{12}$$

We denote by $\mathbf{x}^* = [\widetilde{\mathbf{x}}^*; \ldots; \widetilde{\mathbf{x}}^*]$ the unique solution to (12).

Now, for given penalty parameter $\alpha > 0$, one can define the quadratic penalty function corresponding to the linearly constraint problem (12) as follows,

$$h_\alpha(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \big(\mathbf{I} - \mathbf{W}\big)\mathbf{x} + \alpha F(\mathbf{x}). \tag{13}$$

Since $\mathbf{I} - \mathbf{W}$ is a positive semi-definite matrix and F is L-smooth and μ-strongly convex, therefore 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.

$$\mathbf{x}^*_\alpha = \arg\min_{\mathbf{x}\in\mathbb{R}^{np}} h_\alpha(\mathbf{x}) = \arg\min_{\mathbf{x}\in\mathbb{R}^{np}} \frac{1}{2}\mathbf{x}^\top \big(\mathbf{I} - \mathbf{W}\big)\mathbf{x} + \alpha F(\mathbf{x}). \tag{14}$$

In the following, we link the solution of problem (14) 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.

Lemma 1. Consider the optimization Problem (14) and suppose Assumptions 1– 4 hold. Then, the expected deviation of the output of QDGD from the solution to Problem (14) is upper bounded by

$$\mathbb{E}\Big[\|\mathbf{x}_T - \mathbf{x}^*_\alpha\|^2\Big] \leq \mathcal{O}\left(\frac{c_1 n\sigma^2 \|W - W_D\|_F^2}{\mu c_2} \frac{1}{T^{(1-\delta)/2}} \right), \tag{15}$$

for $\varepsilon = \frac{c_1}{T^{3(1-\delta)/4}}$, $\alpha = \frac{c_2}{T^{(1-\delta)/4}}$, any $\delta \in (0,1)$ and $T \geq T_1$, where c_1 and c_2 are positive constants independent of T, and $T_1 \coloneqq \max\Big\{ \Big\lceil \Big(c_1 c_2 \mu\Big)^{1/(1-\delta)} \Big\rceil, \Big\lceil \Big(\frac{c_1(1+c_2 L)^2}{c_2\mu} \Big)^{2/(1-\delta)} \Big\rceil \Big\}$.

Lemma 1 guarantees convergence of the proposed iterations (3) to the solution of the later-defined Problem (14). 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. The following lemma guarantees such argument.

Lemma 2. Consider the distributed consensus optimization Problem (1) and the problem defined in (14). If Assumptions 1, 2 and 4 hold, then the deviation of the two solutions is bounded as

$$\|\mathbf{x}^*_\alpha - \mathbf{x}^*\| \leq \mathcal{O}\Big(\frac{c_2}{1-\beta} \cdot \frac{1}{T^{(1-\delta)/4}} \Big), \tag{16}$$

for $\alpha = \frac{c_2}{T^{(1-\delta)/4}}$, any $\delta \in (0,1)$ and $T \geq T_2$, where c_2 is a positive constant independent of T and $T_2 \coloneqq \max\Big\{ \Big\lceil \Big(\frac{c_2 L}{1+\lambda_n(W)} \Big)^{4/(1-\delta)} \Big\rceil, \Big\lceil c_2^4 (\mu + L)^{4/(1-\delta)} \Big\rceil \Big\}$.

Proofs of Lemmas 1 and 2 are skipped to the Appendix. Having set the main lemmas, now it is straightforward to prove Theorem 1.

Proof of Theorem 1. For the specified step-sizes ε and α and large enough iterations $T \geq \max\{T_1, T_2\}$, Lemmas 1 and 2 are applicable and we have

$$\begin{aligned} \mathbb{E}\Big[\|\mathbf{x}_T - \mathbf{x}^*\|^2\Big] &= \mathbb{E}\Big[\|\mathbf{x}_T - \mathbf{x}^*_\alpha + \mathbf{x}^*_\alpha - \mathbf{x}^*\|^2\Big] \\ &\leq 2\mathbb{E}\Big[\|\mathbf{x}_T - \mathbf{x}^*_\alpha\|^2\Big] + 2\|\mathbf{x}^*_\alpha - \mathbf{x}^*\|^2 \\ &\leq \mathcal{O}\Big(\frac{1}{T^{(1-\delta)/2}}\Big) + \mathcal{O}\Big(\frac{1}{T^{(1-\delta)/2}}\Big) \\ &= \mathcal{O}\Big(\frac{1}{T^{(1-\delta)/2}}\Big). \end{aligned} \tag{17}$$

Since $\mathbb{E}\Big[\big\|\mathbf{x}_{i,T} - \widetilde{\mathbf{x}}^*\big\|^2\Big] \leq \mathbb{E}\Big[\|\mathbf{x}_T - \mathbf{x}^*\|^2\Big]$ for any $i = 1, \ldots, n$, the inequality in (17) follows the claim of Theorem 1. □

V. Numerical Experiments

In this section, we evaluate the performance of the proposed QDGD Algorithm on a least squares problem. We

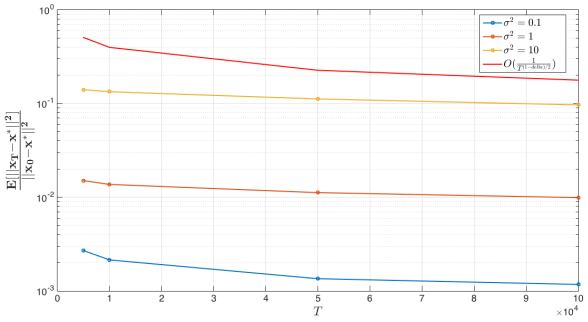

Fig. 1. Relative optimal squared error for three vales of quantization error: $\sigma^2 \in \{0.1, 1, 10\}$, compared with the order of upper bound.

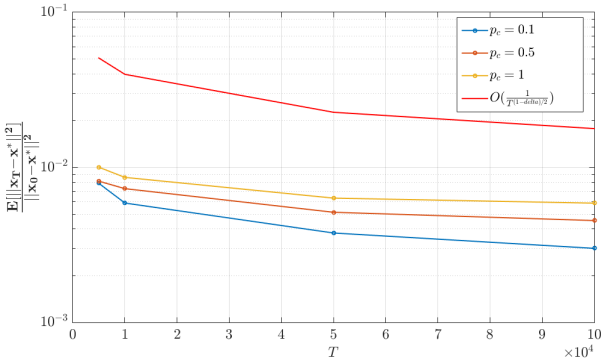

Fig. 2. Relative optimal squared error for three vales of graph connectivity ratio: $p_c \in \{0.1, 0.5, 1\}$, compared with the order of upper bound.

pictorially demonstrate the effect of quantization noise and graph topology on the relative expected error rate.

Consider the least squares optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x}) = \sum_{i=1}^{n} \frac{1}{2}\|\mathbf{A}_i\mathbf{x} - \mathbf{b}_i\|^2 , \tag{18}$$

where $f_i(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}_i\mathbf{x} - \mathbf{b}_i\|^2$ denotes the local objective function of node $i \in [n]$. We consider the linear model with $\mathbf{b}_i = \mathbf{A}_i\tilde{\mathbf{x}} + \boldsymbol{\eta}_i$ where the entries of $\mathbf{A}_i \in \mathbb{R}^{p\times p}$ and $\tilde{\mathbf{x}} \in \mathbb{R}^p$ are drawn randomly from standard normal distribution and $\boldsymbol{\eta}_i \in \mathbb{R}^p$ is additive white Gaussian noise. The unique solution to (18) is therefore $\mathbf{x}^* = (\sum_{i=1}^n \mathbf{A}_i^\top \mathbf{A}_i)^{-1}(\sum_{i=1}^n \mathbf{A}_i^\top \mathbf{b}_i)$.

We consider a connected Erdos-Renyi graph of $n = 100$ nodes and connectivity probability of $p_c = 0.1$ and dimension $p = 20$. Figure 1 shows the convergence rate corresponding to three values of quantization noise $\sigma^2 \in \{0.1, 1, 10\}$, compared to the theoretical upper bound derived in Theorem 1 in the logarithmic scale for $\delta = 0.3$. As expected, Figure 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, Figure 1 demonstrates that the convergence rate closely follows the upper bound derived in Theorem 1.

To observe the effect of graph topology, quantization noise variance is fixed to $\sigma^2 = 0.1$ and $\delta = 0.3$ and we varied the connectivity ratio by picking three different values, i.e. $p_c \in \{0.1, 0.5, 1\}$ where $p_c = 1$ corresponds to the complete graph case. As Figure 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.

VI. 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. We proved that the QGDG algorithm achieves a vanishing consensus error in mean-squared sense, and verified our theoretical results with numerical studies.

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.

VII. Appendix

Proof of 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 $\mathbf{z}_t = [\mathbf{z}_{1,t} \ldots, \mathbf{z}_n]$ 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 (11), i.e.,

$$\widetilde{\nabla} h_\alpha(\mathbf{x}_t) = \big(\mathbf{W}_D - \mathbf{W}\big)\mathbf{z}_t + \big(\mathbf{I} - \mathbf{W}_D\big)\mathbf{x}_t + \alpha\nabla F(\mathbf{x}_t), \tag{19}$$

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 (19).

It yields

$$\begin{aligned}\mathbb{E}\big[\widetilde{\nabla} h_\alpha(\mathbf{x}_t)|\mathcal{F}^t\big] &= \big(\mathbf{W}_D - \mathbf{W}\big)\mathbb{E}[\mathbf{z}_t|\mathcal{F}^t] + \big(\mathbf{I} - \mathbf{W}_D\big)\mathbf{x}_t + \alpha\nabla F(\mathbf{x}_t),\\ &= \big(\mathbf{I} - \mathbf{W}\big)\mathbf{x}_t + \alpha\nabla F(\mathbf{x}_t)\\ &= \nabla h_\alpha(\mathbf{x}_t),\end{aligned} \tag{20}$$

where we used the fact that $\mathbb{E}[\mathbf{z}_t|\mathcal{F}^t] = \mathbf{x}_t$ (Assumption 3). Hence, $\widetilde{\nabla} h_\alpha$ is an unbiased estimator for the true gradient ∇h_α. Now, we can rewrite the update rule (11) as

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \varepsilon\widetilde{\nabla} h_\alpha(\mathbf{x}_t), \tag{21}$$

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 (21) converges to the unique minimizer of $h_\alpha(\mathbf{x})$. More precisely, we can write for $t \geq 1$,

$$\begin{aligned}\mathbb{E}\Big[\|\mathbf{x}_{t+1} - \mathbf{x}_\alpha^*\|^2 |\mathcal{F}^t\Big] &= \mathbb{E}\Big[\Big\|\mathbf{x}_t - \varepsilon\widetilde{\nabla} h_\alpha(\mathbf{x}_t) - \mathbf{x}_\alpha^*\Big\|^2 |\mathcal{F}^t\Big]\\ &= \|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 - 2\varepsilon\langle \mathbf{x}_t - \mathbf{x}_\alpha^*, \mathbb{E}[\widetilde{\nabla} h_\alpha(\mathbf{x}_t)|\mathcal{F}^t]\rangle\\ &+ \varepsilon^2\mathbb{E}\Big[\Big\|\widetilde{\nabla} h_\alpha(\mathbf{x}_t)\Big\|^2 |\mathcal{F}^t\Big]\\ &= \|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 - 2\varepsilon\langle \mathbf{x}_t - \mathbf{x}_\alpha^*, \nabla h_\alpha(\mathbf{x}_t)\rangle\\ &+ \varepsilon^2\mathbb{E}\Big[\Big\|\widetilde{\nabla} h_\alpha(\mathbf{x}_t)\Big\|^2 |\mathcal{F}^t\Big]\\ &\leq (1 - 2\mu_\alpha\varepsilon)\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 + \varepsilon^2\mathbb{E}\Big[\Big\|\widetilde{\nabla} h_\alpha(\mathbf{x}_t)\Big\|^2 |\mathcal{F}^t\Big].\end{aligned} \tag{22}$$

We have used the facts that $\widetilde{\nabla} h_\alpha$ is unbiased and h_α is strongly convex modulo μ_α. Next, we bound the second term in (22), that is

$$\begin{aligned}&\mathbb{E}\Big[\Big\|\widetilde{\nabla} h_\alpha(\mathbf{x}_t)\Big\|^2 |\mathcal{F}^t\Big]\\ &= \mathbb{E}\Big[\Big\|\big(\mathbf{W}_D - \mathbf{W}\big)\mathbf{z}_t + \big(\mathbf{I} - \mathbf{W}_D\big)\mathbf{x}_t + \alpha\nabla F(\mathbf{x}_t)\Big\|^2 |\mathcal{F}^t\Big]\\ &\leq \big\|\nabla h_\alpha(\mathbf{x}_t)\big\|^2 + \mathbb{E}\Big[\Big\|\big(\mathbf{W}_D - \mathbf{W}\big)(\mathbf{z}_t - \mathbf{x}_t)\Big\|^2 |\mathcal{F}^t\Big]\\ &\leq L_\alpha^2\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2 + n\sigma^2\|W - W_D\|^2,\end{aligned} \tag{23}$$

where we used the smoothness of h_α and boundedness of quantization noise. Plugging (23) into (22) yields

$$\begin{aligned}\mathbb{E}\Big[\|\mathbf{x}_{t+1} - \mathbf{x}_\alpha^*\|^2 |\mathcal{F}^t\Big] &\leq \big(1 - 2\mu_\alpha\varepsilon_t + \varepsilon^2 L_\alpha^2\big)\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2\\ &+ \varepsilon^2 n\sigma^2\|W - W_D\|^2.\end{aligned} \tag{24}$$

Let us $e_t = \mathbb{E}\Big[\|\mathbf{x}_t - \mathbf{x}_\alpha^*\|^2\Big]$ denote the expected squared deviation of the local variables of the optimal solution at time $t \geq 1$. By taking expectation over history $\mathcal{F}^t$ from (24), we have

$$\begin{aligned}e_{t+1} &\leq \big(1 - 2\mu_\alpha\varepsilon + \varepsilon^2 L_\alpha^2\big)e_t + \varepsilon^2 n\sigma^2\|W - W_D\|^2\\ &= \big(1 - \varepsilon(2\mu_\alpha - \varepsilon L_\alpha^2)\big)e_t + \varepsilon^2 n\sigma^2\|W - W_D\|^2.\end{aligned} \tag{25}$$

Notice that for the specified pick of ε and $T \geq T_1$, we have $T^{(1-\delta)/2} \geq T_1^{(1-\delta)/2} \geq \frac{c_1(1+c_2L)^2}{c_2\mu}$ and

$$\begin{aligned}\varepsilon &= \frac{c_1}{T^{3(1-\delta)/4}}\\ &\leq \frac{c_2\mu}{(1+c_2L)^2}\cdot\frac{1}{T^{(1-\delta)/4}}\\ &\leq \frac{\mu_\alpha}{\big(1 - \lambda_n(W) + \alpha L\big)^2}\\ &\leq \frac{\mu_\alpha}{L_\alpha^2}\end{aligned} \tag{26}$$

Therefore, (25) can be written as

$$\begin{aligned}e_{t+1} &\leq \big(1 - \varepsilon(2\mu_\alpha - \varepsilon L_\alpha^2)\big)e_t + \varepsilon^2 n\sigma^2\|W - W_D\|^2\\ &\leq (1 - \mu_\alpha\varepsilon)e_t + \varepsilon^2 n\sigma^2\|W - W_D\|^2\\ &= \Big(1 - \frac{c_1c_2\mu}{T^{1-\delta}}\Big)e_t + \frac{1}{T^{3(1-\delta)/2}}c_1^2 n\sigma^2\|W - W_D\|^2.\end{aligned} \tag{27}$$

Now we let $a = c_1c_2\mu$ and $b = c_1^2 n\sigma^2\|W - W_D\|^2$ and employ Lemma 3 to conclude that

$$\begin{aligned}\mathbb{E}\Big[\|\mathbf{x}_T - \mathbf{x}_\alpha^*\|^2\Big] &= e_T\\ &\leq \mathcal{O}\left(\frac{b}{aT^{(1-\delta)/2}}\right) = \mathcal{O}\Big(\frac{c_1 n\sigma^2\|W - W_D\|^2}{\mu c_2}\frac{1}{T^{(1-\delta)/2}}\Big),\end{aligned} \tag{28}$$

and the proof of Lemma 1 is complete. □

Proof of Lemma 2. First, recall the penalty function minimization in (14). 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 $\gamma = 1$,

$$\mathbf{u}_{t+1} = \mathbf{u}_t - \gamma\nabla h_\alpha(\mathbf{u}_t) = \mathbf{W}\mathbf{u}_t - \alpha\nabla F(\mathbf{u}_t). \tag{29}$$

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,

$$\begin{aligned}\|\mathbf{u}_t - \mathbf{x}_\alpha^*\|^2 &\leq (1 - \mu_\alpha)^t\|\mathbf{u}_0 - \mathbf{x}_\alpha^*\|^2\\ &= (1 - \alpha\mu)^t\|\mathbf{u}_0 - \mathbf{x}_\alpha^*\|^2.\end{aligned} \tag{30}$$

If we take $\alpha = \frac{c_2}{T^{(1-\delta)/4}}$, then (30) implies

$$\begin{aligned}\|\mathbf{u}_T - \mathbf{x}_\alpha^*\|^2 &\leq \mathcal{O}\Big((1 - \frac{c_2}{T^{(1-\delta)/4}})^T\Big)\\ &= \mathcal{O}\Big(e^{-c_2T^{(3+\delta)/4}}\Big) = o(1).\end{aligned} \tag{31}$$

On the other hand, it can be shown [10] 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 (30) converges to the $\mathcal{O}\Big(\frac{\alpha}{1-\beta}\Big)$-neighborhood of the optima $\mathbf{x}^*$, i.e.,

$$\|\mathbf{u}_t - \mathbf{x}^*\| \leq \mathcal{O}\Big(\frac{\alpha}{1-\beta}\Big). \tag{32}$$

If we take $\alpha = \frac{c_2}{T^{(1-\delta)/4}}$, the condition $T \geq T_2$ implies that $\alpha \leq \min\{\frac{1+\lambda_n(W)}{L}, \frac{1}{\mu+L}\}$. Therefore, (32) yields

$$\|\mathbf{u}_T - \mathbf{x}^*\| \leq \mathcal{O}\Big(\frac{\alpha}{1-\beta}\Big). \tag{33}$$

From (31) and (33), we have for $T \geq T_2$

$$\begin{aligned}\|\mathbf{x}^*_\alpha - \mathbf{x}^*\| &= \|\mathbf{x}^*_\alpha - \mathbf{u}_T + \mathbf{u}_T - \mathbf{x}^*\| \\ &\leq \|\mathbf{x}^*_\alpha - \mathbf{u}_T\| + \|\mathbf{u}_T - \mathbf{x}^*\| \\ &\leq \mathcal{O}\Big(\frac{c_2}{1-\beta}\cdot\frac{1}{T^{(1-\delta)/4}}\Big),\end{aligned} \tag{34}$$

and the claim in Lemma 2 follows. □

Lemma 3. Consider the sequence of iterates e_t satisfying the inequality

$$e_{t+1} \leq \left(1 - \frac{a}{T^{1-\delta}}\right) e_t + \frac{b}{T^{3(1-\delta)/2}}, \tag{35}$$

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

$$e_T \leq \mathcal{O}\left(\frac{1}{T^{(1-\delta)/2}}\right). \tag{36}$$

Proof. Use the expression in (35) for steps $t-1$ and t obtain

$$\begin{aligned}e_{t+1} &\leq \left(1 - \frac{a}{T^{1-\delta}}\right)^2 e_{t-1} \\ &\quad + \left(1 - \frac{a}{T^{1-\delta}}\right)\frac{b}{T^{3(1-\delta)/2}} + \frac{b}{T^{3(1-\delta)/2}} \\ &= \left(1 - \frac{a}{T^{1-\delta}}\right)^2 e_{t-1} \\ &\quad + \left[1 + \left(1 - \frac{a}{T^{1-\delta}}\right)\right]\frac{b}{T^{3(1-\delta)/2}}\end{aligned} \tag{37}$$

By recursively applying these inequalities for all steps $t \geq 0$ we obtain that

$$\begin{aligned}e_t &\leq \left(1 - \frac{a}{T^{1-\delta}}\right)^t e_0 \\ &\quad + \frac{b}{T^{3(1-\delta)/2}}\left[1 + \left(1 - \frac{a}{T^{1-\delta}}\right) + \cdots + \left(1 - \frac{a}{T^{1-\delta}}\right)^{t-1}\right] \\ &\leq \left(1 - \frac{a}{T^{1-\delta}}\right)^t e_0 + \frac{b}{T^{3(1-\delta)/2}}\left[\sum_{s=0}^{t-1}\left(1 - \frac{a}{T^{1-\delta}}\right)^s\right] \\ &\leq \left(1 - \frac{a}{T^{1-\delta}}\right)^t e_0 + \frac{b}{T^{3(1-\delta)/2}}\left[\sum_{s=0}^{\infty}\left(1 - \frac{a}{T^{1-\delta}}\right)^s\right] \\ &= \left(1 - \frac{a}{T^{1-\delta}}\right)^t e_0 + \frac{b}{T^{3(1-\delta)/2}}\left[\frac{1}{1-(1-\frac{a}{T^{1-\delta}})}\right] \\ &= \left(1 - \frac{a}{T^{1-\delta}}\right)^t e_0 + \frac{b}{aT^{(1-\delta)/2}}\end{aligned} \tag{38}$$

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

$$\begin{aligned}e_T &\leq \left(1 - \frac{a}{T^{1-\delta}}\right)^T e_0 + \frac{b}{aT^{(1-\delta)/2}} \\ &\leq (e^{-aT^\delta})e_0 + \frac{b}{aT^{(1-\delta)/2}} \\ &= \mathcal{O}\left(\frac{b}{aT^{(1-\delta)/2}}\right),\end{aligned} \tag{39}$$

and the claim in (36) follows. □

REFERENCES

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

[2] A. Ribeiro, "Ergodic stochastic optimization algorithms for wireless communication and networking," *IEEE Transactions on Signal Processing*, vol. 58, no. 12, pp. 6369–6386, 2010.

[3] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proceedings of the 3rd international symposium on Information processing in sensor networks*, pp. 20–27, ACM, 2004.

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

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

[6] K. I. Tsianos, S. Lawlor, and M. G. Rabbat, "Consensus-based distributed optimization: Practical issues and applications in large-scale machine learning," in *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*, pp. 1543–1550, IEEE, 2012.

[7] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE transactions on automatic control*, vol. 31, no. 9, pp. 803–812, 1986.

[8] J. N. Tsitsiklis, "Problems in decentralized decision making and computation.," tech. rep., MASSACHUSETTS INST OF TECH CAMBRIDGE LAB FOR INFORMATION AND DECISION SYSTEMS, 1984.

[9] A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "On distributed averaging algorithms and quantization effects," *IEEE Transactions on Automatic Control*, vol. 54, no. 11, pp. 2506–2517, 2009.

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

[11] D. Jakovetić, J. Xavier, and J. M. Moura, "Fast distributed gradient methods," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1131–1146, 2014.

[12] S. S. Ram, A. Nedić, and V. V. Veeravalli, "Distributed stochastic subgradient projection algorithms for convex optimization," *Journal of optimization theory and applications*, vol. 147, no. 3, pp. 516–545, 2010.

[13] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[14] 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 Processing*, vol. 62, no. 7, pp. 1750–1761, 2014.

[15] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, "Decentralized quadratically approximated alternating direction method of multipliers," in *Signal and Information Processing (GlobalSIP), 2015 IEEE Global Conference on*, pp. 795–799, IEEE, 2015.

[16] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Transactions on Automatic control*, vol. 57, no. 3, pp. 592–606, 2012.

[17] K. I. Tsianos, S. Lawlor, and M. G. Rabbat, "Push-sum distributed dual averaging for convex optimization," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pp. 5453–5458, IEEE, 2012.

[18] E. Wei, A. Ozdaglar, and A. Jadbabaie, "A distributed newton method for network utility maximization–i: Algorithm," *IEEE Transactions on Automatic Control*, vol. 58, no. 9, pp. 2162–2175, 2013.

[19] A. Mokhtari, Q. Ling, and A. Ribeiro, "Network newton distributed optimization methods," *IEEE Transactions on Signal Processing*, vol. 65, no. 1, pp. 146–161, 2017.

[20] F. Yan, S. Sundaram, S. Vishwanathan, and Y. Qi, "Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 11, pp. 2483–2493, 2013.

[21] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, "A decentralized second-order method for dynamic optimization," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pp. 6036–6043, IEEE, 2016.

[22] 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 *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

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

[24] S. Yuksel and T. Basar, "Quantization and coding for decentralized lti systems," in *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, vol. 3, pp. 2847–2852, IEEE, 2003.

[25] A. Kashyap, T. Basar, and R. Srikant, "Quantized consensus," *2006 IEEE International Symposium on Information Theory*, pp. 635–639, 2006.

[26] M. G. Rabbat and R. D. Nowak, "Quantized incremental algorithms for distributed optimization," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 798–808, 2005.

[27] M. El Chamie, J. Liu, and T. Başar, "Design and analysis of distributed averaging with quantized communication," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 3870–3884, 2016.

[28] T. C. Aysal, M. Coates, and M. Rabbat, "Distributed average consensus using probabilistic quantization," in *Statistical Signal Processing, 2007. SSP'07. IEEE/SP 14th Workshop on*, pp. 640–644, IEEE, 2007.

[29] A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "Distributed subgradient methods and quantization effects," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pp. 4177–4184, IEEE, 2008.

[30] E. Gravelle and S. Martínez, "Quantized distributed load balancing with capacity constraints," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pp. 3866–3871, IEEE, 2014.

[31] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "Qsgd: Communication-efficient sgd via gradient quantization and encoding," in *Advances in Neural Information Processing Systems*, pp. 1707–1718, 2017.

[32] S. Li, M. A. Maddah-Ali, Q. Yu, and A. S. Avestimehr, "A fundamental tradeoff between computation and communication in distributed computing," *arXiv preprint arXiv:1604.07086*, 2016.

[33] K. Lee, M. Lam, R. Pedarsani, D. Papailiopoulos, and K. Ramchandran, "Speeding up distributed machine learning using codes," in *Information Theory (ISIT), 2016 IEEE International Symposium on*, pp. 1143–1147, IEEE, 2016.

[34] Y. H. Ezzeldin, M. Karmoose, and C. Fragouli, "Communication vs distributed computation: an alternative trade-off curve," *arXiv preprint arXiv:1705.08966*, 2017.

[35] S. Prakash, A. Reisizadeh, R. Pedarsani, and S. Avestimehr, "Coded computing for distributed graph analytics," *arXiv preprint arXiv:1801.05522*, 2018.