Distributed Average Consensus with Bounded Quantizer and Unbounded Input
Shengyu Zhu and Biao Chen

Abstract—This paper considers distributed average consensus using finite-bit bounded quantizer with possibly unbounded data. Under the framework of the alternating direction method of multipliers (ADMM), we develop distributed averaging algorithms where each node iteratively updates using only the local information and finitely quantized outputs from its neighbors. It is shown that all the agent variables either converge to the same quantization level or cycle around the data average after finite iterations. An error bound for the consensus value is established, which turns out to be the same as that of using the unbounded rounding quantizer provided that an algorithm parameter (i.e., ADMM step size) is small enough. We also analyze the effect of the algorithm parameter and propose an adaptive parameter selection strategy that only requires knowledge of the number of agents in order to accelerate the algorithm with certain consensus accuracy guarantee. Finally, simulations are performed to illustrate the effectiveness of the proposed algorithms.

Index Terms—Quantized consensus, distributed averaging algorithm, finite-bit bounded quantizer, alternating direction method of multipliers (ADMM).

I. INTRODUCTION

There have been extensive efforts devoted to applications where autonomous agents collaborate to accomplish a global objective. Of particular interest is distributed average consensus which computes the average of agents’ measurements using only local computation and communication. The problem originates from distributed computation and decision-making [2], [3], and arises in various applications including load balancing for parallel computers [4], multi-agent coordination [5], gossip algorithms [6], and distributed inference [7]. We refer the reader to [8] for an overview.

Distributed averaging algorithms are iterative algorithms that aim to achieve the average consensus for all agents. They are more energy efficient and are robust to link failures compared with fusion center based processing. Studied in [9]–[11] is a classical method where each agent updates itself with a weighted average of its own value and the values received from its neighbors. Another approach is a gossip based algorithm, initially introduced in [2] for consensus problems and further investigated in [6], [12], among others. In [13]–[15], the authors proposed a distributed averaging algorithm based on the alternating direction method of multipliers (ADMM), an iterative algorithm for solving convex problems and has received much attention recently (see [16]). The idea is to formulate the data average as the solution to a least-squares problem and then decentralize the ADMM update. While the above approaches converge to the desired average for each agent under fairly general conditions, it is important to note that agents are assumed to communicate real values of infinite precision with their neighbors. In practical applications such as those encountered in wireless sensor networks, data communications are usually subject to limited capacity and resource constraints. As such, it is commonly assumed that agents can reliably transmit only quantized data, resulting in quantized consensus [17].

Literature review on quantized consensus: A well studied approach to quantized consensus is to use dithered quantizers [18]. Utilizing the first- and second-order moments of dither quantizer output, many researchers have developed algorithms that yield consensuses at a random variable whose mean is the desired average, along with analyses on the mean squared error and convergence rate [14], [15], [19]–[22]. Deterministic algorithms for quantized consensus have also been studied. In [11], [22], the proposed algorithms ensure the convergence to a consensus in a neighborhood of the average within finite iterations and consensus accuracy is characterized by an error bound on the difference between the consensus result and the desired average. Recent work of [23], [24] reports a similar result for an ADMM based distributed averaging algorithm using rounding quantizer, which is referred to as deterministically quantized consensus ADMM (DQ-CADMM). An advantage of DQ-CADMM is that the error bound does not depend on the size of the network nor the agents’ data. The consensus algorithm in [25] is shown to either reach a quantized consensus close to the average in finite time or lead all agent variables to cycle in a small neighborhood around the average; in the latter case, consensus is not guaranteed. In [26], quantized consensus is formulated as a feedback control design problem for coding/decoding schemes. With an appropriate scaling function and carefully chosen control gain based on some spectral properties of the Laplacian matrix of the underlying fixed undirected graph, the proposed protocol with rounding quantizer can achieve the exact average consensus asymptotically.

We note that the quantizers in most existing works are still of infinite bits since the quantizer output has an unbounded range and infinite quantization levels. In order to achieve finite-bit communications at each iteration, these algorithms require the knowledge of a bound on agents’ data such that truncation can be employed. It is worth mentioning that there also exist algorithms that can work with predefined finite-bit quantizers; instead of directly truncating the communicated data, the
bound on agents’ data is used to pick appropriate updating weights to handle the finite-bit communication constraint. With the same bounded condition, the consensus algorithms in [21] and [20] can use finite-level quantizers to asymptotically achieve the average for each agent.

Summary of contributions: We study distributed average consensus using finite-bit bounded quantized communications with possibly unbounded inputs, i.e., without knowing a bound on agents’ data. This is primarily useful to statistical inference in sensor networks where the observations are intrinsically unbounded, e.g., when observations are corrupted by additive white Gaussian noise (AWGN). Naively truncating agents’ data to meet the bounded condition may lead to loss of inference performance. For example, considered in [27], [28] is distributed detection using consensus based approach where agents’ data for averaging are the log-likelihood ratios of local observations. Without properly handling the unbounded data, existing approaches fail to establish the optimal asymptotic detection performance using finite-bit information exchange per iteration. Another example is from parameter estimation. With AWGN, it is well known that the global average of local observations is a minimum variance unbiased estimator for the parameter. Thus, the data for averaging in consensus based approaches are the local observations at each node and are again unbounded. If truncation is used to make local data bounded, consensus based approach may result in biased estimator.

In this paper, we consider uniform deterministic quantization that is finite-bit and bounded. Our quantizer is constructed by projecting the input to a compact convex set and then applying the uniform rounding quantizer to the projected value. We propose an ADMM based quantized consensus algorithm by replacing the rounding quantizer in DQ-CADMM [23], [24] with this bounded quantizer. Since the projection operation may lead to unbounded quantization error with a bounded quantizer, the convergence analysis becomes more complicated. To proceed, we show that the proposed algorithm is equivalent to applying the usual rounding quantizer to the ADMM update of a constrained least-squares problem and that the agent variables are eventually bounded. We establish that the algorithm, within finite iterations, either converges to a common quantization point or cycles with a finite period around the average; in the latter case, every agent has the same sample average of quantized variable values over one period. Furthermore, using convexity properties of the constrained least-squares problem, we derive an upper bound on the consensus error which does not depend on the size of the network nor agents’ data. We also discuss the effect of the ADMM step size on the performance of the proposed algorithm and come up with a heuristic parameter selection scheme that requires the knowledge of the number of nodes and the number of edges (indeed, it suffices to know only the number of agents or an upper bound on it).

Preliminary results were reported in [11]. Besides providing thorough simulations on the proposed algorithm, the current paper characterizes the consensus error in cyclic cases and investigates the effect of the algorithm parameter. An adaptive strategy is then proposed for parameter selection, which is shown to significantly accelerate the algorithm with consensus accuracy guarantee.

Organization of the paper: The rest of this paper is organized as follows. Section III reviews consensus ADMM for distributed optimization and its application to distributed average consensus. In Section IV we define the bounded quantizer and propose a distributed averaging algorithm that uses this quantizer. We establish its convergence result and further come up with an improved algorithm that results in the same error bound as DQ-CADMM. Section V discusses the effect of the algorithm parameter. Simulations are provided in Section VI and Section VII concludes the paper.

Notations: We use $\mathbf{0}$ to denote the all-zero column vector with a suitably defined dimension. Notation $I_n$ denotes the $n$-dimensional all-one column vector; $\mathbf{0}_n$ and $\mathbf{I}_n$ are the $n \times n$ all-zero and identity matrices, respectively. For $\bar{x} \in \mathbb{R}$, $|\bar{x}|$ denotes the smallest integer that is greater than or equal to $x$. Notation $\|x\|_2$ denotes the Euclidean norm of a vector $x$. Given a positive semidefinite matrix $G$ with proper dimensions, the $G$-norm of $x$ is $\|x\|_G = \sqrt{x^T G x}$. For a symmetric matrix $L \in \mathbb{R}^{n \times n}$, denote its eigenvalues in the ascending order as $\lambda_1(L) \leq \lambda_2(L) \leq \cdots \leq \lambda_n(L)$. For any matrix $M$, $C(M)$ represents its column space.

II. CONSENSUS ADMM FOR DISTRIBUTED AVERAGING

This section briefly reviews consensus ADMM (CADMM) and its application to distributed average consensus. We start with the network model used in this paper.

A. Network Model

Consider a network of $n$ agents bi-directionally connected by $m$ edges (hence $2m$ arcs), and with a fixed topology. We describe this network as a symmetric directed graph $G_d = \{V, A\}$ or an undirected graph $G_u = \{V, E\}$, where $V = \{1, 2, \ldots, n\}$ is the set of vertices, $A = \{(i, j):$ the arc from agent $i$ to agent $j\}$ is the set of arcs, and $E$ is the set of edges with $|E| = m$. Define the oriented incidence matrix $M_{i} \in \mathbb{R}^{n \times 2m}$ with respect to $G_d$ as follows: $[M_{i}]_{l,i} = 1$ if the $l$th arc leaves agent $i$, $[M_{i}]_{l,i} = -1$ if the $l$th enters agent $i$, and $[M_{i}]_{l,i} = 0$ otherwise. The unoriented incidence matrix $M_{+} \in \mathbb{R}^{n \times 2m}$ is defined by setting $[M_{+}]_{l,i} = |[M_{i}]_{l,i}|$. Denote $N_i = \{j: (i, j) \in A\}$ as the set of neighbors of agent $i$. Further define $L_{-} = \frac{1}{2}M_{-}M_{-}^T$ and $L_{+} = \frac{1}{2}M_{+}M_{+}^T$ which are respectively the signed and signless Laplacian matrices with respect to $G_u$. Then $W = \frac{1}{2}(L_{-} + L_{+}) = \text{diag}\{\|N_1\|, \|N_2\|, \ldots, \|N_n\|\}$ is the degree matrix related to $G_u$, i.e., a diagonal matrix with $(i, i)$th entry being $\|N_i\|$ and other entries being $0$.

The following lemma is stated to help establish our main results in Section III.

Lemma 1 ([29], [30]): Given a connected network, we have

a) $L_{-}$ is positive semidefinite and $0 = \lambda_1(L_{-}) \leq \lambda_2(L_{-}) \leq \cdots \leq \lambda_n(L_{-})$. Let $b = 0$ if and only if $b \in C(1_n)$.

b) $L_{+}$ is positive semidefinite and $\lambda_n(L_{+}) > 0$.

c) $C(M_{-}) = C(L_{-})$. For every $\alpha \in C(L_{-})$, there exists a unique $\beta \in C(M_{-}^T)$ such that $\alpha = M_{-}\beta$. 

B. CADMM for Distributed Average Consensus

Let \( r_i \in \mathbb{R} \) be the local data at node \( i, \ i = 1, 2, \ldots, n \). Distributed average consensus computes the global average

\[
\bar{r} = \frac{1}{n} \sum_{i=1}^{n} r_i,
\]

using only local information exchange among neighboring nodes. We assume that nodes do not know the global topology of the network, which is typical in large scale networks or when nodes are autonomous. As a result, the communicated information among linked nodes at each iteration is usually a real value or its quantized version in many distributed averaging algorithms.

The original CADMM is obtained by decentralizing the ADMM update of minimizing a sum of convex functions. Let \( f_i(\bar{x}) : \mathbb{R}^d \rightarrow \mathbb{R} \), where \( d \) is a positive integer, denote a convex local objective function that is only known to node \( i \).

It has been shown in [13], [31] that the CADMM update for minimizing \( \sum_{i=1}^{n} f_i(\bar{x}) \) is

\[
x^{k+1}_i = \arg \min_{\bar{x}} f_i(\bar{x}) + \rho |N_i| \bar{x}^T \bar{x} + \rho |N_i| x_i^k + \rho \sum_{j \in N_i} x_j^k - \alpha_i^k,
\]

\[
\alpha_i^{k+1} = \alpha_i^k + \rho \left( |N_i| x_i^{k+1} - \sum_{j \in N_i} x_j^{k+1} \right),
\]

where \( \rho > 0 \) is the ADMM parameter and \( \alpha_i^k \in \mathbb{R}^d \) is the local Lagrangian multiplier of node \( i \). The above update is fully decentralized as the update of \( x^{k+1}_i \) and \( \alpha^{k+1}_i \) only relies on local and neighboring information. We refer to (1) as the original CADMM.

The original CADMM converges under fairly mild conditions, which follows directly from global convergence of the ADMM [16], [32], [33]. To apply CADMM to distributed average consensus, it is noted the global average is the unique solution to a least-squares problem, i.e.,

\[
\bar{r} = \arg \min_{\bar{x}} \frac{1}{2} \sum_{i=1}^{n} (\bar{x} - r_i)^2.
\]

The corresponding CADMM update for distributed averaging is then given by

\[
x^{k+1}_i = \frac{1}{1 + 2 \rho |N_i|} \left( \rho |N_i| x_i^k + \rho \sum_{j \in N_i} x_j^k - \alpha_i^k + r_i \right),
\]

\[
\alpha_i^{k+1} = \alpha_i^k + \rho \left( |N_i| x_i^{k+1} - \sum_{j \in N_i} x_j^{k+1} \right).
\]

For ease of presentation, we further write (2) in a compact form. Define \( x^k \) and \( \alpha^k \) as the vectors concatenating all \( x_i^k \) and \( \alpha_i^k \), respectively. Then (2) is equivalent to

\[
x^{k+1} = (I_n + 2\rho W)^{-1} (\rho L_+ x^k - \alpha^k + r),
\]

\[
\alpha^{k+1} = \alpha^k + \rho L_- x^k.
\]

Letting \( s^k = [x^k; \alpha^k; r] \) and \( D_0 = (I_n + 2\rho W)^{-1} \), we have

\[
s^{k+1} = D s^k = D^{k+1} s_0,
\]

where \( D^{k+1} \) denotes the \((k+1)\)th power of the square matrix \( D \) defined by

\[
D \triangleq \begin{bmatrix}
-\rho D_0 L_+ & -D_0 & D_0 \\
\rho^2 L_- D_0 L_+ & I_n - \rho L_- D_0 & 0_n \\
0_n & 0_n & I_n
\end{bmatrix}.
\]

The convergence result of CADMM implies that

\[
\lim_{k \to \infty} s^k = s^* = \begin{bmatrix} x^* \\ \alpha^* \\ r \\ r \end{bmatrix},
\]

provided that \( \alpha^0 \) is initialized in \( C(L_-) \) (e.g., all 0). Moreover, \( s^k \) converges linearly to \( s^* \). Let \( z^k = \frac{1}{2} M_+ x^k \) and \( \beta^k \) be the unique vector in \( C(M_+) \) such that \( \alpha^k = M_- \beta^k \) (cf. Lemma I). Define also

\[
u^k = [z^k; \beta^k] \quad \text{and} \quad G = \begin{bmatrix} \rho I_{2m} & 0_{2m} \\ 0_{2m} & \frac{1}{\rho} I_{2m} \end{bmatrix}.
\]

The linear convergence of \( s^k \to s^* \) is formally stated below.

Theorem 1 (Linear convergence of CADMM for distributed average consensus [37]): Let \( x^0 \in \mathbb{R}^n \) and \( \alpha^0 \in C(L_-) \). Then \( u^k \) converges Q-linearly to \( u^* \) with \( z^* = \frac{1}{2} M_+ x^* \) and \( \beta^* \) being the unique vector in \( C(M_+) \) such that \( M_+ \beta^* = r - 1_n \bar{r} \), with respect to the \( G \)-norm:

\[
\|u^{k+1} - u^*\|_G^2 \leq \frac{1}{1 + \delta} \|u^k - u^*\|_G^2,
\]

where \( \mu > 1 \) can be any and

\[
\delta = \min \left\{ \frac{\mu - 1}{\lambda_2(L_-)}, \frac{2\rho \lambda_2(L_-) - \mu \lambda_2(L_-)}{\rho^2 \lambda_2(L_-) \lambda_2(L_-) + \mu} \right\}.
\]

Furthermore, \( s^k \) is R-linearly convergent to \( s^* \) as

\[
\|s^{k+1} - s^*\|_2 \leq \left( 1 + \sqrt{\frac{2\rho \lambda_2(L_-)}{1 + \delta}} \right) \|u^k - u^*\|_G.
\]

III. DISTRIBUTED AVERAGE CONSENSUS WITH FINITE-BIT BOUNDED QUANTIZATION

This section studies distributed average consensus subject to bounded quantization constraint without a known bound on agents’ data. Our finite-bit quantizer is defined as follows.

A. Finite-bit Bounded Quantizer

We assume that each agent can store and compute real values with infinite precision but can only send quantized data through the channel which are received by its neighbors without any error. With a predefined quantization resolution \( \Delta > 0 \), the quantization lattice in \( \mathbb{R} \) is

\[
\Lambda = \{ t\Delta : t \in \mathbb{Z} \}.
\]

Consider the usual rounding quantizer \( Q : \mathbb{R} \rightarrow \Lambda \) that maps a real value to the nearest point in \( \Lambda \):

\[
Q(\bar{x}) = t\Delta, \text{ if } \left( t - \frac{1}{2} \right) \Delta < \bar{x} \leq \left( t + \frac{1}{2} \right) \Delta.
\]
Let $e(\hat{x}) = Q(\hat{x}) - \hat{x}$ denote the quantization error. It is clear that $|e(\hat{x})| \leq \frac{\Delta}{2}$ for any $\hat{x} \in \mathbb{R}$. Noticing that the rounding quantizer has infinite quantization levels due to its unbounded range, we further define a bounded quantizer as follows.

Let $\mathcal{X}$ be a nonempty compact convex set in $\mathbb{R}$. We assume without loss of generality that $\mathcal{X} = [-L, L]$ for some $0 < L < \infty$, since we can always translate the set. Further assume that $L$ is a multiple of $\Delta$ and let $\Lambda_{\mathcal{X}} = [-L, L] \cap \Delta$. Define $T_{\mathcal{X}}(\cdot)$ as the projection operator that maps a real value to the nearest point in $\mathcal{X}$. A bounded quantizer $Q_b : \mathbb{R} \to \Lambda_{\mathcal{X}}$ is defined by first projecting its argument onto $\mathcal{X}$ and then applying the usual rounding quantizer $Q(\cdot)$ to the projected value, i.e.,

$$Q_b(\cdot) = Q \circ T_{\mathcal{X}}(\cdot).$$

It is straightforward to see that $|Q_b(\hat{x})| \leq L$ for any $\hat{x} \in \mathbb{R}$. Therefore, the bounded quantizer $Q_b(\cdot)$ has $2L/\Delta + 1$ quantization levels and its output can be represented by $\lceil \log_2(2L/\Delta + 1) \rceil$ bits. Define $e_b(\hat{x}) = Q_b(\hat{x}) - \hat{x}$ to be the quantization error of $Q_b(\cdot)$, then $|e_b(\hat{x})|$ can be unbounded for an unbounded $\hat{x}$. As a note, the above operators operate on each entry of the argument when they have a vector input.

### B. CADMM with Finite-bit Bounded Quantization

We now modify the original CADMM update in (2) using the bounded quantizer $Q_b(\cdot)$, as presented in Algorithm 1 which we refer to as bounded quantizer based CADMM (BQ-CADMM).

**Algorithm 1 BQ-CADMM for distributed average consensus**

**Require:** Initialize $x_i^0 = 0$ and $\alpha_i^0 = 0$ for each agent $i$, $i = 1, 2, \ldots, n$. Set $\rho > 0$ and $k = 0$.

1: repeat
2: \hspace{1em} every agent $i$
3: \hspace{2em} $x_i^{k+1} = \frac{1}{1 + 2\rho |N_i|} \left( \rho |N_i| Q_b(x_i^k) + \rho \sum_{j \in N_i} Q_b(x_j^k) - \alpha_i^k + r_i \right),$
4: \hspace{1em} $\alpha_i^{k+1} = \alpha_i^k + \rho \left( |N_i| Q_b(x_i^{k+1}) - \sum_{j \in N_i} Q_b(x_j^{k+1}) \right),$
5: \hspace{1em} set $k = k + 1$.
6: until a predefined stopping criterion (e.g., a maximum iteration number) is satisfied.

It is clear that BQ-CADMM is obtained from the CADMM update (2) by applying the bounded quantizer $Q_b(\cdot)$ to local variables $x_i^k$. Here we provide an alternative interpretation of this algorithm. Since $Q_b(\cdot) = Q \circ T_{\mathcal{X}}(\cdot)$, BQ-CADMM can be obtained by applying the rounding quantizer $Q(\cdot)$ to $T_{\mathcal{X}}(x_i^k)$ in the following update:

$$x_i^{k+1} = \frac{1}{1 + 2\rho |N_i|} \left( \rho |N_i| T_{\mathcal{X}}(x_i^k) + \rho \sum_{j \in N_i} T_{\mathcal{X}}(x_j^k) - \alpha_i^k + r_i \right),$$

$$\alpha_i^{k+1} = \alpha_i^k + \rho \left( |N_i| T_{\mathcal{X}}(x_i^{k+1}) - \sum_{j \in N_i} T_{\mathcal{X}}(x_j^{k+1}) \right).$$

That is, at the $k$th iteration, node $i$ has its local variable value $x_i^k$ and sends its projection $T_{\mathcal{X}}(x_i^k)$ to the rounding quantizer $Q(\cdot)$. At the $(k+1)$th iteration, $x_i^{k+1}$ and $\alpha_i^{k+1}$ are updated using $Q(T_{\mathcal{X}}(x_i^k))$ and $Q(T_{\mathcal{X}}(x_i^k))$. Since $T_{\mathcal{X}}(x_i^k)$ is the projection of $x_i^k$ onto $\mathcal{X}$, we can write (7) equivalently as

$$T_{\mathcal{X}}(x_i^{k+1}) = \arg\min_{\hat{x} \in \mathcal{X}} \frac{1}{2}(\hat{x} - r_i)^2 + \rho |N_i| |\hat{x} - \tilde{x}|^2 + \hat{x} \left( \rho |N_i| T_{\mathcal{X}}(x_i^k) + \rho \sum_{j \in N_i} T_{\mathcal{X}}(x_j^k) - \alpha_i^k \right).$$

where $I_{\mathcal{X}}(\tilde{x})$ denotes the indicator function

$$I_{\mathcal{X}}(\tilde{x}) = \begin{cases} 0, & \tilde{x} \in \mathcal{X}, \\ +\infty, & \tilde{x} \notin \mathcal{X}. \end{cases}$$

Comparing with the CADMM update in (1), we see that (9) (and hence (7)) together with (8) is the CADMM update with $T_{\mathcal{X}}(x_i^k)$ and $\alpha_i^k$ as local variables and $\frac{1}{2}(\hat{x} - r_i)^2 + I_{\mathcal{X}}(\tilde{x})$ as the local objective function at node $i$. As such, the convergence of the original CADMM implies that (7) and (8) yield

$$\lim_{k \to \infty} T_{\mathcal{X}}(x_i^k) = \arg\min_{\tilde{x}} \sum_{i=1}^n \frac{1}{2}(\tilde{x} - r_i)^2 + I_{\mathcal{X}}(\tilde{x}) = T_{\mathcal{X}}(\tilde{r}).$$

Thus, BQ-CADMM can also be viewed from applying the usual rounding quantizer to the original CADMM update of a constrained least-squares problem.

To recap, BQ-CADMM can be interpreted as applying bounded quantization to local variables of an unconstrained least-squares problem, or alternatively, applying unbounded quantization to updates of a constrained least-squares problem. If $L$ is chosen to be large enough such that $T_{\mathcal{X}}(x_i^k) = x_i^k$ for all $k$, then $Q_b(x_i^k) = Q(x_i^k)$ and BQ-CADMM becomes the same as our previous algorithm DQ-CADMM in (23), (24). In this sense, DQ-CADMM can be regarded as a special case of BQ-CADMM. The introduced projection operator or the additional indicator function, however, makes BQ-CADMM
much more complicated to analyze. In [14], [23], [24] where dithered quantizer is studied, convergence comes from the linearity of the CADMM update for average consensus (cf. Eq. (5)) as well as the fact that the expectation of the output of a dithered quantizer is equal to the input. This approach fails to apply to BQ-CADMM because $T_{X}(\cdot)$ is not a linear operator in general and one cannot simply change the order of projection and expectation operations. Additionally, [7] and [8] no longer possess the linear convergence rate due to the introduced bounded constraint. Thus, the idea of using the linear convergence rate, as in [23], [24], does not work for BQ-CADMM. On the other hand, if we view BQ-CADMM as a result from the CADMM update on the unconstrained least-squares problem modified by the bounded quantizer $Q_b(\cdot)$, it is still unclear how BQ-CADMM performs since the quantization error of $Q_b(\cdot)$ can be unbounded for a real input. Fortunately, local variables $\alpha^k_i$’s are inherently bounded by the BQ-CADMM update, as stated in the following lemma.

Lemma 2: Given local data $r_i \in \mathbb{R}$, consider BQ-CADMM with $x^0_i = 0$ and $\alpha^0_i = 0$ for $i = 1, 2, \ldots, n$. Then $\alpha^k_i$ is finitely bounded by

$$|\alpha^k_i| \leq (1 + 6\rho|N_i|)|r_i|.$$  

Proof: See Appendix.

To establish convergence result of BQ-CADMM, we define $s^k_Q = [Q_b(x^k_i); \alpha^k_i; r_i]$ and rewrite BQ-CADMM as the standard CADMM update on $s^k_Q$ plus an error term caused by bounded quantization. We first have

$$Q_b(x^{k+1}_i) = x^{k+1}_i + e_b(x^{k+1}_i),$$

$$\alpha^{k+1} = \alpha^k + \rho L_x x^{k+1} + \rho L_e e_b(x^{k+1}),$$

where $e_b(x^{k+1}) = Q_b(x^{k+1}) - x^{k+1}$ denotes the quantization error. By defining $s^{k+1}_e = [e_b(x^{k+1}); \rho L_e e_b(x^{k+1}); 0]$, we have the BQ-CADMM update equivalent to

$$s^{k+1}_Q = DS^k_Q + s^{k+1}_e \triangleq \Phi(s^k_Q).$$  

(10)

It is important to note that the above update $\Phi(\cdot)$ is deterministic, i.e., given $s_1 = s_2$, we must have $\Phi(s_1) = \Phi(s_2)$. We will use this fact to establish the main result given below.

Theorem 2: For BQ-CADMM in Algorithm 1 there exists a finite iteration $k_0 > 0$ such that for $k \geq k_0$ all the quantized variable values

- either converge to the same quantization value:

$$Q_b(x^k_i) = \cdots = Q_b(x^n_i) \triangleq x^*_Q,$$

where $x^*_Q \in \Lambda_X$ and

$$|x^*_Q - T_{X}(\bar{r})| \leq \left(1 + 4\rho\frac{m}{n}\right) \frac{\Delta}{2},$$  

(11)

- or cycle around the average $\bar{r}$ with a finite period $T \geq 2$, i.e., $x^k_i = x^*_{i+T}$, $i = 1, 2, \ldots, n$. Furthermore, the sample average over one period reaches a consensus:

$$\frac{1}{T} \sum_{t=1}^{T} Q_b(x^k_{i+t}) = \cdots = \frac{1}{T} \sum_{t=1}^{T} Q_b(x^k_{n+t}) \triangleq \bar{x}_Q,$$  

(12)

and

$$|\bar{x}_Q - \bar{r}| \leq \left(1 + 4\rho\frac{m}{n}\right) \Gamma_0,$$  

(13)

where

$$\Gamma_0 \triangleq \max \left\{ \frac{\Delta}{2}, \frac{4\rho n L}{\Delta} \right\}.$$  

(14)

Proof: We first show that the sequence $s^{k+1}_Q$ is either convergent or cyclic. Note first that $Q_b(x^k_i)$ is bounded by $L$ and can only be a multiple of $\Delta$. Similarly, $\alpha^k_i$ is bounded as per Lemma 2 and is a multiple of $\rho \Delta$ as seen from the $\alpha_i$ update of BQ-CADMM. Thus, the number of states of local variables $[Q_b(x^k_i); \alpha^k_i]$ at node $i$ is bounded by

$$\left(\frac{2L}{\Delta} + 1\right) \left(\frac{L + |r_i|}{\rho \Delta} + \frac{6|N_i||L|}{\Delta} \right) \leq B.$$  

(15)

Therefore, the number of possible states of $s^{k+1}_Q$ is bounded by $B^n$ which is finite for given $r_i$’s because $\rho$ and $\Delta$ are both positive and fixed. From (10), the BQ-CADMM update $s^{k+1}_Q = \Phi(s^k_Q)$ is a deterministic function of only $s^k_Q$. Therefore, $s^{k+1}_Q$ must be either convergent or cyclic with a finite period $T \geq 2$ after a finite number of iterations denoted as $k_0$. The rest of the proof is to establish the respective error bounds.

Convergent case: In this case we know $\alpha^{k+1}_i = \alpha^k_i$ for $k \geq k_0$. Then the $\alpha$-update of BQ-CADMM together with Lemma 1 implies

$$Q_b(x^k_i) \in C(1).$$

Thus, $Q_b(x^k_i)$ reaches a consensus at a common quantization point in $\Lambda_X$. Denote $x^k_i = x^*_i$ and $\alpha^k_i = \alpha^*_i$ for $k \geq k_0$, i.e., when convergence is reached. Further letting $e^*_i = x^*_i - T_{X}(x^*_i)$, we have $|e^*_i| \leq \frac{\Delta}{4}$ since $Q_b(\cdot) = Q \circ T_{X}(\cdot)$ with $Q(\cdot)$ being the usual rounding quantization. Taking $k \to \infty$ on both sides of (9) and using the optimality condition for minimizing a convex function (see, e.g., [24]), we get

$$T_{X}(x^*_i) - r_i + 2\rho|N_i|T_{X}(x^*_i) + \partial I_{X}(T_{X}(x^*_i))$$

$$- \left(\rho|N_i|x^*_Q + \rho \sum_{j \in N_i} x^*_Q - \alpha^*_i \right) = 0,$$

where $\partial I_{X}(T_{X}(x^*_i))$ is a subgradient of $I_{X}(\cdot)$ at $T_{X}(x^*_i)$. After rearranging and plugging in $T_{X}(x^*_i) = x^*_Q - e^*_i$, we obtain

$$x^*_Q - e^*_i - r_i + \partial I_{X}(x^*_Q - e^*_i) + \alpha^*_i = 2 \sum_{j \in N_i} |N_i|e^*_j.$$  

(16)

Summing up both sides of (16) from $i = 1$ to $n$ yields

$$\sum_{i=1}^{n} (x^*_Q - e^*_i - r_i + \partial I_{X}(x^*_Q - e^*_i)) = 2 \sum_{i=1}^{n} |N_i|e^*_i.$$  

(17)
where $\sum_{i=1}^{n} \alpha_i^* = 0$ is due to $[\alpha_1^*; \alpha_2^*; \ldots; \alpha_n^*] \in C(L_\infty)$ and Lemma 1. Since $T_X(\bar{x}) = \arg\min_{\bar{x}} \sum_{i=1}^{n} \frac{1}{2} (\bar{x} - x_i)^2 + I_X(x_i))$, we also have

$$\sum_{i=1}^{n} (T_X(\bar{x}) - x_i) + \partial I_X(T_X(\bar{x})) = 0. \quad (18)$$

Subtracting (18) from (17), we get that

$$\sum_{i=1}^{n} (T_X(\bar{x}) - x^*_Q - \partial I_X(T_X(\bar{x})) - \partial I_X (x^*_Q - e^*_i)) = \sum_{i=1}^{n} (2|\mathcal{N}_i| + 1) e^*_i. \quad (19)$$

We only consider the case where $|T_X(\bar{x}) - x^*_Q| > \frac{\Delta}{2}$; otherwise (11) holds trivially. Recall that $|e^*_i| \leq \frac{\Delta}{2}$, we get

$$(T_X(\bar{x}) - x^*_Q)(T_X(\bar{x}) - (x^*_Q - e^*_i)) > 0.$$  

Note that the convexity of $I_X(\cdot)$ also implies

$$(\partial I_X(T_X(\bar{x})) - \partial I_X (x^*_Q - e^*_i)) (T_X(\bar{x}) - (x^*_Q - e^*_i)) \geq 0.$$  

The above two inequalities indicate that either $\partial I_X(T_X(\bar{x})) - \partial I_X (x^*_Q - e^*_i) = 0$, or $\partial I_X(T_X(\bar{x})) - \partial I_X (x^*_Q - e^*_i)$ and $T_X(\bar{x}) - x^*_Q$ have the same sign. Then the following is true:

$$|T_X(\bar{x}) - x^*_Q + \partial I_X(T_X(\bar{x})) - \partial I_X (x^*_Q - e^*_i)| \geq |T_X(\bar{x}) - x^*_Q|.$$  

Together with (19), we can establish the upper bound

$$|x^*_Q - T_X(\bar{x})| \leq \frac{1}{n} \sum_{i=1}^{n} (2|\mathcal{N}_i| + 1) e^*_i \leq \left(1 + \frac{4m}{n}\right) \frac{\Delta}{2},$$

where we use the fact that $\sum_{i=1}^{n} |\mathcal{N}_i| = 2m$ for an undirected connected graph.

**Cyclic case:** That $s^k_Q$ cycles with a period $T$ implies

$$\alpha^{kT} - \alpha^k = \rho L \sum_{l=1}^{T} Q_b(x^{k+l}) = 0,$$

which then leads to (12) by Lemma 1. While the bound on $\alpha^k$ in Lemma 2 imposes a bound on $x^k$ through the $x_i$-update, the cyclic behavior itself can be used to derive a tighter error bound.

Consider the local variable $x^k_i$ over one period. When $x^k_i \in \mathcal{X}$ for the entire period (and hence for all $k \geq k_0$), we simply have $|e_b(x^k_i)| \leq \frac{\Delta}{2}$. If there is some $x^k_{i+1} > L$ with $k \geq k_0 + T - 1$, there must exist a $k' \leq k$ which is the largest index such that $Q_b(x^k_{i+1}) < L$ and $Q_b(x^{k'+1}_{i+1}) = L$, for otherwise $Q_b(x^k_i)$ converges and hence BQ-CADMM must converge due to (12). Then we have $Q_b(x^{k'+1}_{i+1}) = L$ for $l = 1, 2, \ldots, k + 1 - k'$.

Recall the $\alpha_i$-update, we get $\alpha_i^{k+l} \leq \alpha_i^{k'}$ as $L$ is the largest quantization value. We further write the $x_i^{k+1}$-update as

$$x_i^{k+1} = \frac{1}{1 + 2\rho|\mathcal{N}_i|} \left( \rho |\mathcal{N}_i| Q_b(x^k_i) + \rho \sum_{j \in \mathcal{N}_i} Q_b(x^k_j) - \alpha_i^k + r_i \right)$$

$$(a) \leq \frac{1}{1 + 2\rho|\mathcal{N}_i|} \left( 2\rho |\mathcal{N}_i| L - \alpha_i^{k'} + r_i \right)$$

$$= x_i^{k'} + \frac{\rho}{1 + 2\rho|\mathcal{N}_i|} \left( 2|\mathcal{N}_i| L + \sum_{j \in \mathcal{N}_i} Q_b(x^k_j) - |\mathcal{N}_i| Q_b(x^k_i) \right)$$

$$- |\mathcal{N}_i| Q_b(x^{k'-1}_i) - \sum_{j \in \mathcal{N}_i} Q_b(x^{k'-1}_j) \right) \leq \frac{5\rho|\mathcal{N}_i| L}{1 + 2\rho|\mathcal{N}_i|} + x_i^{k'} - \frac{\rho |\mathcal{N}_i|}{1 + 2\rho|\mathcal{N}_i|} Q_b(x^k_i), \quad (20)$$

where $(a)$ and $(c)$ are due to the fact that $-L \leq Q_b(\bar{x}) \leq L$ for any $\bar{x} \in \mathbb{R}$, and $(b)$ from the BQ-CADMM update at the $k'$th iteration. Since $k'$ is the index such that $Q_b(x^{k'}_i) < L$, it is straightforward to see that (20) takes the largest value at $x_i^{k'} = L - \frac{\Delta}{2}$. This implies for $k \geq k_0$,

$$x_i^k \leq L - \frac{\Delta}{2} < \frac{4\rho|\mathcal{N}_i| L}{1 + 2\rho|\mathcal{N}_i|}.$$  

Similarly, it can be shown that for $k \geq k_0$,

$$x_i^k > -L - \frac{4\rho|\mathcal{N}_i| L}{1 + 2\rho|\mathcal{N}_i|}. $$

Therefore, when BQ-CADMM reaches a cyclic state, we have

$$|x_i^k| \leq L + \frac{4\rho|\mathcal{N}_i| L}{1 + 2\rho|\mathcal{N}_i|}, \quad (21)$$

and the quantization error satisfies

$$|e_b(x_i^k)| \leq \max \left\{ \frac{\Delta}{2} - \frac{4\rho|\mathcal{N}_i| L}{1 + 2\rho|\mathcal{N}_i|} \right\} \leq \Gamma_0, \quad (22)$$

where $\Gamma_0$ is defined in (14) and we use the fact that $1 \leq |\mathcal{N}_i| < n$.

To derive the error bound, we now summarize the local variable values over one period and use (22), which leads to

$$\left| \frac{1}{T} \sum_{l=1}^{T} Q_b(x^{k+l}_i) - \frac{1}{T} \sum_{l=1}^{T} x^{k+l}_i \right| = |\bar{x}^Q - \frac{1}{T} \sum_{l=1}^{T} x^{k+l}_i | \leq \Gamma_0. \quad (23)$$

By the $x_i$-update, we can also get

$$(1 + 2\rho|\mathcal{N}_i|) \frac{1}{T} \sum_{l=1}^{T} x^{k+l}_i - \rho |\mathcal{N}_i| \frac{1}{T} \sum_{l=1}^{T} Q_b(x^{k+l}_i) - \rho \sum_{j \in \mathcal{N}_i} \left( \frac{1}{T} \sum_{l=1}^{T} Q_b(x^{k+l}_j) \right) \sum_{l=1}^{T} \alpha^{k+l}_i - r_i = 0. \quad (24)$$
Summing both sides of (24) over $i$ and using (12), we get

$$
\sum_{i=1}^{n} r_i + \sum_{i=1}^{n} \left( (1 + 2\rho |\mathcal{N}_i|) \frac{1}{T} \sum_{t=1}^{T} x_i^{t+1} \right) - \sum_{i=1}^{n} \left( 2\rho |\mathcal{N}_i| \frac{1}{T} \sum_{t=1}^{T} Q_b(x_i^{t+1}) \right) = 0. \tag{25}
$$

Finally, using (24) and dividing both sides of (25) by $n$, we can bound the consensus error by

$$
|\bar{x}_Q^* - \bar{r}| \leq \left( 1 + 4\rho \frac{m}{n} \right) \Gamma_0.
$$

This completes the proof.

**Remark 1:** We have mentioned that BQ-CADMM uses $Q_b(x_i^k)$ for the $(k+1)$th update at agent $i$ even though agents can compute and store real values with infinite precision. The reason is to guarantee that $\alpha^k$ is bounded and also lies in $\mathcal{C}(L_-)$ in order to achieve a consensus result. Theorem 2 may fail if $\alpha^k \notin \mathcal{C}(L_-)$.

**Remark 2:** While BQ-CADMM in Algorithm 1 initializes $x^0 = \alpha^0 = 0$, one can use similar arguments to show the same convergence result of Theorem 2 for any $x^0 \in \mathbb{R}^n$ and $\alpha^0 \in \mathcal{C}(L_-)$. As a result, we can pick $x^0$ and $\alpha^0$ that are respectively closer to the optimal $1_n\bar{r}$ and $r - 1_n\bar{r}$, which usually leads to better consensus performance. Since Theorem 2 holds for any $\rho > 0$, we can also use decreasing algorithm parameter $\rho$ such that the algorithm proceeds fast in early stages and guarantees certain consensus accuracy upon convergence or cycling (cf. Section V.D).

**Remark 3:** With convergence or small enough $\rho$ such that $\Gamma_0 \leq \frac{\Delta}{2}$, the error bounds (11) and (13) are exactly the same as those of DQ-CADMM when $\bar{r} \in \mathcal{X}$. Note that even with the prior knowledge that $\bar{r} \in \mathcal{X}$, the algorithms of (21), (24) fail to provide a guaranteed consensus result as agents’ data can still be unbounded. This not only saves energy for data communication but also broadens its applications; see consensus-based detection in [28] where we show that BQ-CADMM with one-bit information exchange between linked nodes at each iteration suffices to achieve the optimal exponential decay of error probabilities under the maximum a posteriori and the Neyman-Pearson criteria.

While the proposed algorithm is not guaranteed to converge for all cases, we can use history of the agent variable values (e.g., the running average technique) to ensure asymptotic convergence at a consensus. As such, we refer to $x_Q^*$ in the convergent case and $\bar{x}_Q^*$ in the cyclic case as the resulting consensus value. As a direct result from Theorem 2, the following corollary states that BQ-CADMM must converge when the data average resides outside the bounded set $\mathcal{X}$.

**Corollary 1:** BQ-CADMM must converge to a consensus at a quantization level in $\Lambda_\mathcal{X}$ when

$$
|\bar{r}| - L > \left( 1 + 4\rho \frac{m}{n} \right) \Gamma_0.
$$

If we further pick $\rho < \frac{\Delta}{4\frac{m}{n}}$, then $Q_b(x_i^k)$ must converge to $\text{sgn}(\bar{r})L$ where $\text{sgn}(\bar{r}) = 1$ if $\bar{r} > 0$ and $\text{sgn}(\bar{r}) = -1$ if $\bar{r} < 0$.

**C. Extended BQ-CADMM**

The error bounds in Theorem 2 indicate that BQ-CADMM eventually reaches a neighborhood of $\bar{r}$. When $\bar{r}$ is far from the bounded set $\mathcal{X}$, however, we may suffer from a large consensus error. Pick $\rho$ small enough such that $\Gamma_0 \leq \frac{\Delta}{2}$ and assume that $L \gg \left( 1 + 4\rho \frac{m}{n} \right) \frac{\Delta}{2}$ (e.g., $L \geq 5 \left( 1 + 4\rho \frac{m}{n} \right) \frac{\Delta}{2}$). Then Corollary 1 implies that $\bar{r}$ is either outside $\mathcal{X}$ or around its boundary when BQ-CADMM converges at $|x_Q^*| = L$. If each agent subtracts $\text{sgn}(x_Q^*)L$ from $r_i$, the new data average becomes $|\bar{r} - \text{sgn}(x_Q^*)L| = |\bar{r} - L| < |\bar{r}|$. Thus, we can run BQ-CADMM with updated data $r_i - \text{sgn}(x_Q^*)L$. This process is repeated until BQ-CADMM either converges to $|x_Q^*| \neq L$ or reaches a cyclic result. The consensus value at each agent is simply the sum of the final consensus value and the offset from each BQ-CADMM call. The above procedure is referred to as extended BQ-CADMM (EBQ-CADMM) presented in Algorithm 2.

**Algorithm 2** EBQ-CADMM for distributed average consensus

**Require:** Initialize $t_i = 0$ for each agent $i$, $i = 1, 2, \ldots, n$.

Pick $\rho > 0$ such that $\Gamma_0 \leq \frac{\Delta}{2}$.

1: **while** true **do**

2: Run BQ-CADMM with data $r_i$ at agent $i$ such that either convergence or cycling is achieved. Denote the consensus value as $x_{BQ}$.

3: **if** $|x_{BQ}| = L$ **then**

4: set $t_i = t_i + \text{sgn}(x_{BQ})L$ and $r_i = r_i - \text{sgn}(x_{BQ})L$

5: **else**

6: break

7: **end if**

8: **end while**

9: return $t_i$ and $x_{BQ}$

We remark that an additional BQ-CADMM is run only when a convergence is reached at either $-L$ or $L$. Notice that consensus is always guaranteed by Theorem 2 that is, if a node converges at $-L$ (or $L$), then every other node converges at $-L$ (or $L$). Hence, global synchronization is not needed at each BQ-CADMM call. Since EBQ-CADMM calls the BQ-CADMM at most $\lceil |\bar{r}|/L \rceil + 1$ times, we have the following theorem directly from Theorem 2 and Corollary 1.

**Theorem 3:** Assume that $L \gg \left( 1 + 4\rho \frac{m}{n} \right) \frac{\Delta}{2}$. For any agents’ data $r_i$’s, EBQ-CADMM in Algorithm 2 yields that

$$
t_1 = \cdots = t_n \equiv t^*,
$$

and

$$
|t^* + x_{BQ} - \bar{r}| \leq \left( 1 + 4\rho \frac{m}{n} \right) \frac{\Delta}{2}.
$$

It is noted that the above error bound in Theorem 3 has a non-vanishing term $\frac{\Delta}{2}$, which implies that the consensus result can not be made arbitrarily close to the true average in general. Nevertheless, by employing deterministic uniform quantization and a fixed step size, EBQ-CADMM possesses local variables which can be used to recover the original data average. Denote the final iteration index by $k^*$. At the end of EBQ-CADMM, the last BQ-CADMM run either converges at
some quantization point in \((-L, L)\), which indicates \(|x_i^n| < L\), or cycles with \(|x_i^{k+1} |\leq L + \frac{4\rho |N_i| L}{1+2\rho |N_i|}\) (cf. Eq. (21)) for any \(k \geq k^*\). Thus, at each node \(i\), we have

\[ \left| -\alpha_i^{k'} + r_i - t^* \right| = (1 + 2\rho |N_i|) x_i^{k'} + \rho L + \sum_{j \in N_i} Q_j(x_j^{k'}) \leq L + 8\rho |N_i| L. \]

Further picking \(\rho = \frac{1}{8m}\), we can get \(|-\alpha_i^{k'} + r_i - t^*| \leq 2L\). Also note that \(\sum_{n=1}^{n} \alpha_i^{k'} = 0\) according to Lemma [1] and that \(t^*\) is locally known to each node. Thus, setting local data to \(-\alpha_i^{k'} + r_i - t^*\), existing quantized consensus algorithms can use finite-bit quantization to asymptotically achieve the exact average consensus; more precisely, the variable value at each node is close to

\[ \frac{1}{n} \sum_{i=1}^{n} \left( -\alpha_i^{k'} + r_i - t^* \right) = \bar{r} - t^*, \]

and the true average \(\bar{r}\) is asymptotically achieved by adding \(t^*\).

IV. ALGORITHM PARAMETER AND STOPPING CRITERION

In this section, we discuss the effect of the algorithm parameter \(\rho\) on BQ-CADMM and EBQ-CADMM. Since \(L\) can be large enough such that \(Q_j(x_j) = Q_j(x_j^t)\), the discussion also covers DQ-CADMM using the usual rounding quantizer in [23], [24] where the effect of \(\rho\) is not explored.

A. Consensus Error

As seen from the consensus error bounds in (11) and (15), it is clear that they increase in the algorithm parameter \(\rho\). With the knowledge of the number of nodes or its upper bound, one can pick \(\rho\) small enough to achieve desired consensus accuracy. For example, if \(\rho \leq \min \{\frac{n}{4m}, \frac{\Delta}{8mL}\} = \frac{\Delta}{8mL}\) as \(2m \leq n(n-1)\) and \(L \geq \Delta\), the resulting consensus value of EBQ-CADMM is within one quantization resolution of the desired average. We also remark that practical consensus error is usually much smaller than the error bound (see simulations).

B. Cyclic Period

In the proof of Theorem 2 we show that the number of possible states of \(s^k_Q\) is bounded by \(B^n\) where \(B\) is given in (13). This bound works for both convergence time and cyclic period but is generally too loose to use in practice. Our simulations in Section IV-B show that BQ-CADMM converges in most cases, particularly with small enough \(\rho\). When the algorithm indeed cycles, the period depends on the network structure, agents’ data as well as the algorithm parameter, with a range between 2 and 15 for all simulated cases. We also observe that the cyclic period in all simulations consists of two consecutive quantization levels for each node. Indeed, we can derive tighter consensus error bounds if the cyclic period only has two consecutive quantization levels, e.g., we can replace \(L\) with \(\Delta\) in \(\Gamma_0\) that is defined by (14).

C. Convergence Time

We refer to the smallest \(k_0\) in convergent cases or smallest \(k_0 + T\) in cyclic cases as the convergence time for BQ-CADMM. First notice that it is meaningless to consider the convergence time without any constraint on the consensus error. To see this, recall that BQ-CADMM has initial variable values \(x_i^0 = \alpha_i^0 = 0\) at each node \(i\) and \(x_i^1 = \frac{1}{1+2\rho |N_i|} r_i\) at the first iteration. If \(\rho\) is chosen large enough, e.g., \(\rho > \max_i |r_i|\), such that \(|x_i^1| < \frac{\Delta}{2}\), then we have \(Q_j(x_j^1) = \alpha_j^k = 0, k = 0, 1, \ldots\), for all \(i\) and the convergence time is \(k_0 = 1\). Thus, we consider accelerating BQ-CADMM under some consensus accuracy constraint.

A direct approach is to apply similar techniques that accelerate the original ADMM. For example, reference [31] studies the optimal parameter selection of \(\rho\) that maximizes the convergence rate \(\delta\) given in Theorem 1 and reference [35] uses preconditioning to improve the convergence rate. However, these approaches require the knowledge of the network structure, which might be unrealistic for large scale networks. Furthermore, the resulting parameter selection may not meet the consensus accuracy requirement and indeed does not provide the best practical performance. Here we propose a heuristic selection \(\rho = \frac{n}{m}\) which only requires (an estimate of) the number of nodes and the number of edges. Our intuition is based on the fact that a larger \(\frac{n}{m}\) indicates that an agent on the average has more neighboring information for its update. Therefore, a smaller \(\rho\) provides adequate updates towards the consensus. Conversely, when agents have less information available, a larger \(\rho\) can help accelerate the speed. The good performance of this selection is validated by simulations.

When \(\rho = \frac{n}{m}\) fails to satisfy a given consensus accuracy constraint, a decreasing strategy can be used where the initial value of \(\rho\) is set at \(\frac{n}{m}\) and decreases as the number of iterations increases. In the case where only \(n\) is known but \(m\) is not, the decreasing strategy may simply start with \(\rho = 1\).

D. Stopping Criterion

A natural approach for stopping BQ-CADMM is to set the maximum number of iterations at each node, which then requires characterization of the convergence time. While our current analysis does not have a closed-form result, we alternatively provide a rough upper bound based on our observations. At the beginning of the algorithm where local agent variables are far from the optima, i.e., when \(\|u^k - u^*\|^2_G\) is large, the ADMM update in (11) and (12) will make much progress towards the optima and the effect of the quantization operation \(Q(\cdot)\) is negligible. Thus, BQ-CADMM and the original ADMM for the constrained least-squares problem have similar trajectories at early stages. When local variables becomes close to the optima, the original ADMM still has monotone decreasing \(\|u^k - u^*\|^2_G\) while BQ-CADMM either converges or cycles due to the quantization operation. This behavior is observed in all our simulations and an example is provided in Fig. 2. As such, a rough estimate can be obtained based on the convergence rate result of the original ADMM when applied to the constrained least-squares problem. Here we use the result from [36] where \(\|u^{k+1} - u^k\|^2_G\) is viewed
as a metric of how close local variables are to the optima. The convergence rate of the original CADMM is

$$\|u^{k+1} - u^k\|^2_G \leq \frac{1}{k+1} \|u^0 - u^*\|^2_G.$$  \hspace{1cm} (26)

We can bound the right-hand side of (26) by a small value and our simulations show that $4\rho^2\Delta^2$ is a good choice. Thus, a rough bound on convergence time, with $x_i^0 = 0$ for $i = 1, 2, \ldots, n$, is given by

$$\frac{1}{\Delta^2} \left( nT_X(\bar{r})^2 + \frac{1}{2\rho^2\lambda_2(L-\rho)} \|r - 1_n T_X(\bar{r})\|^2 \right).$$

We remark that this rough bound is quite loose in most of our simulations; see Fig. 4. An upper bound on the convergence time, which depends on the network topology and agents’ data, may still be insufficient as these quantities are locally unknown. In fully distributed settings, we can run additional algorithms to determine if a consensus has been reached; see, e.g., [37], [38]. These additional algorithms may take a long running time in large networks and will require extra data communications. In general, finding a practically meaningful and efficient stopping criterion in a fully distributed manner remains open.

As to determining the cyclic behavior, nodes may record a certain number of consecutive variable values and check if any cycle exists. Though no bound on cyclic period is provided in this paper, our simulations show that the cyclic period is between 2 and 15. In summary, we can set the maximum number of iterations or run the algorithm as long as permitted. If there is no convergence which can be determined by additional algorithms, we can then run other algorithms with agents’ data $-\alpha_k + r_i$ which are bounded with a known bound at each node.

V. SIMULATIONS

This section investigates the proposed algorithms via numerical examples. We construct a connected network with $n$ nodes and $m$ edges by first generating a complete graph of $n$ nodes and then randomly removing $\frac{n(n-1)}{2} - m$ edges while ensuring the graph stays connected. Let $\Delta = 1$ throughout this section.

A. Consensus Error

We first illustrate how BQ-CADMM and EBQ-CADMM proceed by showing the trajectories of agents’ variables. We simulate a connected network with $n = 50$ nodes and $m = 100$ edges. Let $L = \frac{n}{2} = 25$, $\rho = \frac{n}{m} = 0.5$, and $r_i \sim N(n, n^2)$. Set also the maximum iteration number of each BQ-CADMM to 50. By this setting, it is very likely to have $|\bar{r}| > L$ and we thus apply EBQ-CADMM for data averaging. The trajectories of $Q_0(x_i^k) + t_i, i = 1, 2, \ldots, n$ are plotted in Fig. 1. In this example, the desired average is 44.20 and the resulting consensus value of EBQ-CADMM is 44. The consensus error is 0.20, which is much smaller than the error bound $(1 + 4\rho^2\Delta^2)\frac{\Delta}{2} = 2.5$. The figure also indicates that EBQ-CADMM calls BQ-CADMM twice with the first and second calls converging at iterations 17 and 33, respectively. As a note, we run this setup for 10,000 times and no cyclic result is observed.

We next compare the proposed algorithms with DQ-CADMM that uses rounding quantizer $Q(\cdot)$ to investigate the effect of the bounded constraint. Denote $t = [t_1; t_2; \cdots; t_n]$ for EBQ-CADMM and define the iterative error as

$$\begin{cases}
\|Q_0(x_i^k) - 1_n \bar{r}\|_2/\sqrt{n}, & \text{BQ-CADMM}, \\
\|Q_0(x_i^k) + t - 1_n \bar{r}\|_2/\sqrt{n}, & \text{EBQ-CADMM}, \\
\|Q(x_i^k) - 1_n \bar{r}\|_2/\sqrt{n}, & \text{DQ-CADMM}, \\
\|x_i^k - 1_n \bar{r}\|_2/\sqrt{n}, & \text{CADMM},
\end{cases}$$

which is equal to the consensus error when a consensus is reached. Set $n = 75$, $m = 200$ and $L = 30$. Pick $\rho = 0.5$ and set the maximum iteration number of each BQ-CADMM call in EBQ-CADMM to 50. We consider two cases: one with the average in the bounded set and the other with the average outside the set. Specifically, we generate $r_i \sim N(0, n^2)$ and run the distributed averaging algorithms twice with data $r_i$ and $r_i + 2n$, respectively. We use BQ-CADMM for the former case since the average lies in the bounded set with high probability. Simulation result is presented in Fig. 2.

From Fig. 2 all quantized consensus algorithms converge to a consensus after certain iterations with consensus errors much smaller than the upper bound, computed to be 1.83. When $|\bar{r}| > L$, BQ-CADMM and DQ-CADMM perform similarly and their iterative errors decrease exponentially until consensus is reached, while the original CADMM makes the iterative error decrease to 0. When $|\bar{r}| > L$, DQ-CADMM converges much faster than EBQ-CADMM. This is because each BQ-CADMM call has iterations running even when a consensus is reached.

B. Cyclic Period

Since the convergent case can be viewed as the cyclic case with period $T = 1$, we first investigate whether BQ-CADMM converges or cycles in various settings. Notice that the quantized consensus algorithm converges in all the above examples and indeed, BQ-CADMM and hence EBQ-CADMM tend to converge, particularly with small enough $\rho$. 

Fig. 1: Trajectories of EBQ-CADMM; $n = 50$, $m = 100$, $L = \frac{n}{2}$, $\Delta = 1$, $\rho = 0.5$, and $r_i \sim N(n, n^2)$. 

Fig. 2: Simulation result is presented in Fig. 2.
To illustrate this, we simulate star graphs which have the smallest number of edges for a connected network, randomly generated connected graphs with intermediate numbers of edges, and complete graphs which have the highest number of edges. As Corollary 1 indicates that BQ-CADMM must converge when \( \bar{r} \) is far from the bounded set \( \mathcal{X} \), we set \( L = 30 \) and generate \( r_1 \sim N(0, 100) + r_0 \) where \( r_0 \sim N(0, 25) \) such that most simulated cases have \( \bar{r} \in \mathcal{X} \). We then run BQ-CADMM with different \( \rho \) for the same data. The simulation result is plotted in Fig. 4 where each plotted value denotes the empirical probability of cyclic result in 10,000 runs and both data and graph are randomly generated at each run.

Fig. 3 indicates that BQ-CADMM always converges for large enough \( \rho \) with which the consensus result has a large error bound, as discussed in Section IV-C. When \( \rho \) decreases, the number of cyclic cases first increases and then decreases to 0. In particular, BQ-CADMM converges in all simulated examples with small enough \( \rho \). Another interesting observation is that peak occurrence of the cyclic case differs for different network structures. The star network has highest cyclic cases around \( \rho = 0.5 \frac{m}{n} \) and a larger network indicates less cyclic cases, while the intermediate and complete networks have cyclic cases centered around \( \rho = 5 \frac{m}{n} \) with larger networks having more cyclic cases.

In the same example, we also record the cyclic period when BQ-CADMM indeed cycles. We observe that the period of star networks is always 2 for all \( n \) and \( \rho \), and the intermediate and complete networks have periods between 8 and 15. While we cannot draw firm conclusions on cyclic period from the simulation result, we do find that the period in all cyclic cases consists of two consecutive quantization levels for each node, which can help derive better error bounds as discussed in Section IV-B.

**C. Convergence Time**

To study how \( \rho \) affects the convergence time of BQ-CADMM, we plot in Fig. 4 the average convergence time of the same example in the above section. We observe that BQ-CADMM converges immediately for large enough \( \rho \) but again a large consensus error may exist. The convergence time is about 10 when running BQ-CADMM with \( \rho = \frac{n}{m} \) for all network sizes and network structures. For small \( \rho \), the convergence time decreases polynomially as \( \rho \) increases. The network structure also plays an important role here. For the star network, the convergence time is almost the same for all \( n \) and a larger \( n \) results in a slightly longer convergence time.

On the other hand, the intermediate and complete networks have longer convergence time with smaller \( n \). Comparing the convergence time of the same network size, we also find that a denser network tends to converge faster for the same \( \rho \).

**D. Parameter Selection**

To achieve high consensus accuracy (e.g., the consensus error is within one quantization resolution), the consensus error bound of the cyclic case in Theorem 2 implies that one may need to pick a very small \( \rho \) which can make BQ-CADMM slow to reach convergence or cycling. Fortunately, simulations indicate that BQ-CADMM converges in most cases and \( \rho \) can be larger to meet the consensus error requirement in convergent cases. As such, we can start BQ-CADMM with a large \( \rho \) for a number of iterations. Even though the local variables may not converge, they become close to the optima. We then pick a small \( \rho \) and run BQ-CADMM with the current local variable values. We continue this process until either a convergence is reached and \( \rho \) is such that the consensus error bound (13) meets the required consensus accuracy, or \( \rho \) is small enough such that the cyclic error bound (11) satisfies the accuracy requirement. We suggest starting with \( \rho = \frac{n}{m} \) as it performs reasonably well in terms of both consensus accuracy and convergence time, as seen from Fig. 3 and Fig. 4.

If only \( n \) is known, we may simply start with \( \rho = 1 \) as \( n - 1 \leq m \leq \frac{n(n - 1)}{2} \) for connected networks.

To further illustrate the strategy with decreasing step size, we next apply it to the same example in Section IV-B. Starting with \( \rho = \frac{n}{m} \), we run BQ-CADMM for 50 iterations if \( \rho > 10^{-4} \) and then reduce it by a factor of 10. We repeat this process until \( \rho < 10^{-4} \) with which we run BQ-CADMM long enough such that either convergence or cycling is reached.

The average convergence time of this strategy is presented in Table 1. As one can see, this strategy reduces dramatically the convergence time. In addition, most iterations occur when \( \rho > 10^{-4} \) and BQ-CADMM only takes a few more iterations to reach the convergence result in all simulated cases when \( \rho \leq 10^{-4} \).

**VI. Conclusion and Discussion**

This paper proposes BQ-CADMM and EBQ-CADMM for distributed average consensus using finite-bit bounded quantization. It is shown that BQ-CADMM is equivalent to applying rounding quantizer to the CADMM update of a constrained least-squares problem. Based on this fact, we establish that BQ-CADMM either converges or cycles within finite iterations and further derive the consensus error bounds by utilizing convexity properties. An advantage of our algorithm is that the
agents’ data can be arbitrary; in contrast, existing quantized consensus algorithms that adopt finite-bit quantizers require the data to be bounded and a bound to be known. Noticing that the consensus error is large when the data average is much beyond the quantizer range, we proceed to develop EBQ-CADMM which is equivalent to repeated runs of BQ-CADMM. The resulting consensus error bound can be the same as that of using the unbounded rounding quantizer by picking a small enough algorithm parameter. Balancing fast convergence and small consensus error, an adaptive parameter selection scheme is devised that requires only the knowledge of the number of nodes.

While simulations show that BQ-CADMM converges fast in most cases, the current work lacks theoretical analysis (e.g., upper bound) on cyclic period and convergence time. A possible approach is to find a Lyapunov function with which one can characterize these two quantities under suitable assumptions. In addition, asynchronous protocols, link failures, and noisy communication channels can be studied under the BQ-CADMM framework.

**APPENDIX**

**Proof of Lemma 2.** The proof is mainly based on the fact that $|Q_b(\tilde{x})| \leq L$ for any $\tilde{x} \in \mathbb{R}$. We first prove the following:

$$T_X(x_i^{k+1}) = \begin{cases} -L, & \text{if } \alpha_i^k > (1 + 4\rho|N_i|) L + |r_i|, \\ L, & \text{if } \alpha_i^k < -(1 + 4\rho|N_i|) L - |r_i|. \end{cases}$$

Assume that $T_X(x_i^{k+1}) > -L$ if $\alpha_i^k > (1 + 4\rho|N_i|) L + |r_i|$. Similar to (9), the $x_i$-update of BQ-CADMM implies that
Therefore, if \( k = 0 \) minimizes a constrained least-squares function:

\[
\mathcal{T}_k(x_{k+1}) = \arg \min_{\tilde{x} \in \mathcal{X}} \frac{1}{2} (\tilde{x} - r_i)^2 + \rho |\mathcal{N}_i| |\tilde{x}|^2 - \tilde{x} \left( \rho |\mathcal{N}_i| Q_0(x_{k+1}^i) \right)
\]

where, for ease of presentation, we define

\[
G_k^i(\tilde{x}) \triangleq \arg \min_{\tilde{x} \in \mathcal{X}} G_k^i(\tilde{x}) + \alpha_k^i \tilde{x},
\]

Since \( |Q_0(\tilde{x})| \leq L \) for any \( \tilde{x} \in \mathbb{R} \), it is straightforward to verify that \( G_k^i(\tilde{x}) \) is Lipschitz continuous over \( \mathcal{X} \): for any \( \tilde{x}, \tilde{y} \in \mathcal{X} \),

\[
|G_k^i(\tilde{x}) - G_k^i(\tilde{y})| < ((1 + 4\rho |\mathcal{N}_i|) L + |r_i|) |\tilde{x} - \tilde{y}|
\]

Now letting \( \tilde{x} = -L \in \mathcal{X} \), we have

\[
G_k^i(-L) = G_k^i(\mathcal{T}_k(x_{k+1})) - \alpha_k^i L - \alpha_k^i \mathcal{T}_k(x_{k+1}) < ((1 + 4\rho |\mathcal{N}_i|) L + |r_i| - \alpha_k^i) (\mathcal{T}_k(x_{k+1}) + L),
\]

where the last inequality is because \( \alpha_k^i < (1 + 4\rho |\mathcal{N}_i|) L - |r_i| \) can be shown analogously.

If \( \mathcal{T}_k(x_{k+1}) = -L \), the bounded quantization scheme implies \( Q_0(x_{k+1}^i) = -L \leq Q_0(x_{k+1}^j) \) for any \( j \in \mathcal{N}_i \). Therefore, if \( \alpha_k^i > (1 + 4\rho |\mathcal{N}_i|) L + |r_i| \), we get

\[
\alpha_k^{i+1} = \alpha_k^i + \rho |\mathcal{N}_i| Q_0(x_{k+1}^i) - \rho \sum_{j \in \mathcal{N}_i} Q_0(x_{k+1}^j) \geq \alpha_k^i
\]

Similarly, \( \alpha_k^{i+1} \geq \alpha_k^i \) if \( \alpha_k^i < -(1 + 4\rho |\mathcal{N}_i|) L - |r_i| \).

Next consider \( |\alpha_k^i| \leq (1 + 4\rho |\mathcal{N}_i|) L + |r_i| \). In this case, we can simply use the triangle inequality to conclude that

\[
|\alpha_k^{i+1}| \leq |\alpha_k^i| + |\mathcal{N}_i| |Q_0(x_{k+1}^i)| + \rho \sum_{j \in \mathcal{N}_i} |Q_0(x_{k+1}^j)|
\]

\[
\leq (1 + 6\rho |\mathcal{N}_i|) L + |r_i|.
\]

Since \( \alpha_0^i = 0 \), we finally have \( |\alpha_k^i| \leq (1 + 6\rho |\mathcal{N}_i|) L + |r_i| \) for all \( k \).

REFERENCES

[1] S. Zhu and B. Chen, “Distributed average consensus with bounded quantization,” in Proc. IEEE Int. Workshop on Signal Process. Advances in Wireless Commun., Edinburgh, UK, Jul. 2016.

[2] J. N. Tsitsiklis, “Problems in decentralized decision making and computation,” DTIC Document, Tech. Rep., 1984.

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

[4] C. Xu and F. C. Lau, Load Balancing in Parallel Computers: Theory and Practice. Dordrecht, Germany: Kluwer, 1997.

[5] W. Ren and R. W. Beard, “Consensus seeking in multiagent systems under dynamically changing interaction topologies,” IEEE Trans. Autom. Control, vol. 50, no. 5, pp. 655–661, May 2005.

[6] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, “Randomized gossip algorithms,” IEEE Trans. Inf. Theory, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.

[7] S. Kar, S. Aldosari, and J. M. F. Moura, “Topology for distributed inference on graphs,” IEEE Trans. Signal Process., vol. 56, no. 6, pp. 2609–2613, Jun. 2008.

[8] 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. Infor., vol. 9, no. 1, pp. 427–438, Feb. 2013.

[9] L. Elsner, I. Koltracht, and M. Neumann, “On the convergence of asynchronous paracontextualization with application to tomographic reconstruction from incomplete data,” Linear Algebra Appl., vol. 130, pp. 65–82, 1990.

[10] L. Xiao and S. Boyd, “Fast linear iterations for distributed averaging,” Syst. Contr. Lett., vol. 53, no. 1, pp. 65–75, 2004.

[11] A. Nedic, A. Olsheskyv, A. Ozdaglar, and J. N. Tsitsiklis, “On distributed averaging algorithms and quantization effects,” IEEE Trans. Autom. Control, vol. 54, no. 1, pp. 106–119, Jan. 2009.

[12] T. C. Aysal, M. E. Yildiz, A. D. Sarwate, and A. Scaglione, “Broadcast gossip algorithms for consensus,” IEEE Trans. Signal Process., vol. 57, no. 7, pp. 2748–2761, Jul. 2009.

[13] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, “Consensus in ad hoc WSNs with noisy links—Part I: Distributed estimation of deterministic signals,” IEEE Trans. Signal Process., vol. 56, no. 1, pp. 350–364, Jan. 2008.

[14] H. Zhu, G. B. Giannakis, and A. Cano, “Distributed in-network channel decoding,” IEEE Trans. Signal Process., vol. 57, no. 10, pp. 3970–3983, Oct. 2009.

[15] T. Erseghe, D. Zennaro, E. Dall’Anese, and L. Vangelista, “Fast consensus by the alternating direction multipliers method,” IEEE Trans. Signal Process., vol. 59, no. 11, pp. 5523–5537, Nov. 2011.

[16] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” Foundations and Trends in Machine Learning, vol. 3, no. 1–2, pp. 1–122, 2011.

[17] A. Kashyap, T. Başar, and R. Srikant, “Quantized consensus,” Automatica, vol. 43, no. 7, pp. 1192–1203, 2007.

[18] L. Schuchman, “Dither signals and their effect on quantization noise,” IEEE Trans. Commun. Technol., vol. 12, no. 4, pp. 162–165, Dec. 1964.

[19] T. C. Aysal, M. J. Costa, and M. G. Rabbat, “Distributed average consensus with dithered quantization,” IEEE Trans. Signal Process., vol. 58, no. 3, pp. 1383–1400, 2010.

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

[21] S. Zhu and B. Chen, “Quantized consensus by the ADMM: probabilistic versus deterministic quantizers,” IEEE Trans. Signal Process., vol. 64, no. 7, pp. 1700–1713, Apr. 2016.

[22] ——, “Corrections to “quantized consensus by the ADMM: probabilistic versus deterministic quantizers,”” IEEE Trans. Signal Process., vol. 65, no. 6, pp. 1638–1639, 2017.

[23] M. E. Chamie, J. Liu, and T. Başar, “Design and analysis of distributed averaging with quantized communication,” in Proc. 53rd IEEE Conf. Decision and Control, Los Angeles, CA, Dec. 2014.

[24] T. Li, M. Fu, L. Xie, and J. F. Zhang, “Distributed consensus with limited communication data rate,” IEEE Trans. Autom. Control, vol. 56, no. 2, pp. 279–292, Feb. 2011.

[25] S. Zhu and B. Chen, “Distributed detection over connected networks via one-bit quantizer,” in Proc. IEEE Int. Symp. Inf. Theory, Barcelona, Spain, Jul. 2016.

[26] ——, “Distributed detection in ad hoc networks through quantized consensus,” arXiv preprint arXiv:1612.01904, 2016.

[27] F. R. Chung, Spectral Graph Theory. American Mathematical Soc., 1997, vol. 92.

[28] M. Fiedler, “Algebraic connectivity of graphs,” Czechoslovak Mathematical Journal, vol. 23, no. 2, pp. 298–305, 1973.
[31] 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.

[32] B. He and X. Yuan, “On the $O(1/n)$ convergence rate of the Douglas-Rachford alternating direction method,” SIAM J. Numer. Anal., vol. 50, no. 2, pp. 700–709, 2012.

[33] W. Deng and W. Yin, “On the global and linear convergence of the generalized alternating direction method of multipliers,” J. Sci. Comput., vol. 66, no. 3, pp. 889–916, 2016.

[34] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge Univ. Press, 2004.

[35] P. Giselsson and S. Boyd, “Linear convergence and metric deection in Douglas-Rachford splitting and ADMM,” IEEE Trans. Autom. Control, to be published. doi:10.1109/TAC.2016.2564160.

[36] B. He and X. Yuan, “On non-ergodic convergence rate of douglas–rachford alternating direction method of multipliers,” Numerische Mathematik, vol. 130, no. 3, pp. 567–577, 2015.

[37] N. E. Manitara and C. N. Hadjicostis, “Distributed stopping for average consensus in undirected graphs via event-triggered strategies,” Automatica, vol. 70, pp. 121–127, 2016.

[38] V. Yadav and M. V. Salapaka, “Distributed protocol for determining when averaging consensus is reached,” in 45th Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, 2007.