

DQC-ADMM: Decentralized Dynamic ADMM With Quantized and Censored Communications

Yaohua Liu[✉], Gang Wu, Zhi Tian[✉], *Fellow, IEEE*, and Qing Ling[✉], *Senior Member, IEEE*

Abstract—In distributed learning and optimization, a network of multiple computing units coordinates to solve a large-scale problem. This article focuses on dynamic optimization over a decentralized network. We develop a communication-efficient algorithm based on the alternating direction method of multipliers (ADMM) with quantized and censored communications, termed DQC-ADMM. At each time of the algorithm, the nodes collaborate to minimize the summation of their time-varying, local objective functions. Through local iterative computation and communication, DQC-ADMM is able to track the time-varying optimal solution. Different from traditional approaches requiring transmissions of the exact local iterates among the neighbors at every time, we propose to quantize the transmitted information, as well as adopt a communication-censoring strategy for the sake of reducing the communication cost in the optimization process. To be specific, a node transmits the quantized version of the local information to its neighbors, if and only if the value sufficiently deviates from the one previously transmitted. We theoretically justify that the proposed DQC-ADMM is capable of tracking the time-varying optimal solution, subject to a bounded error caused by the quantized and censored communications, as well as the system dynamics. Through numerical experiments, we evaluate the tracking performance and communication savings of the proposed DQC-ADMM.

Index Terms—Alternating direction method of multipliers (ADMM), communication censoring, decentralized network, dynamic optimization, quantization.

I. INTRODUCTION

THIS article considers the following decentralized dynamic consensus optimization problem over a bidirectionally connected network with n nodes, which is given by

$$\tilde{x}^*(k) = \arg \min_{\tilde{x}(k)} \sum_{i=1}^n f_i^k(\tilde{x}(k)) \quad (1)$$

Manuscript received February 5, 2020; revised July 19, 2020, September 28, 2020, and December 16, 2020; accepted January 8, 2021. The work of Zhi Tian was supported in part by the U.S. NSF Grant CCF-1527396, Grant IIS-1741338, and Grant CCF-1939553. The work of Qing Ling was supported in part by the NSF China under Grant 61973324, in part by the Guangdong Provincial Key Laboratory of Computational Science under Grant 2020B1212060032, and in part by the Fundamental Research Funds for the Central Universities. (*Corresponding author: Qing Ling.*)

Yaohua Liu is with the School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing 210044, China.

Gang Wu is with the Department of Automation, University of Science and Technology of China, Hefei 230027, China.

Zhi Tian is with the Department of Electrical and Computer Engineering, George Mason University, Fairfax, VA 22030 USA.

Qing Ling is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China, and also with the Guangdong Provincial Key Laboratory of Computational Science, Sun Yat-sen University, Guangzhou 510006, China (e-mail: lingqing556@mail.sysu.edu.cn).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TNNLS.2021.3051638>.

Digital Object Identifier 10.1109/TNNLS.2021.3051638

where $\tilde{x}(k) \in \mathcal{R}^p$ is the global optimization variable at time k and $f_i^k : \mathcal{R}^p \rightarrow \mathcal{R}$ is the time-varying local objective function private to node i at time k . The goal is to keep tracking the optimal and consensual solution trajectory $\tilde{x}^*(k) \in \mathcal{R}^p$ that minimizes the summation of the time-varying local objective functions $f_i^k(\tilde{x}(k))$. Such a problem arises in wireless sensor networks [2], control systems [3], power systems [4]–[6], and economics [7], [8]. It is also popular in distributed computing systems for large-scale machine learning and in the presence of streaming data [9], including federated learning for training deep neural networks [10].

For the case that the local objective functions on the nodes are time-invariant, (1) is known as the decentralized static consensus optimization problem, which has been intensively investigated in prior works. Many algorithms, such as (sub)gradient methods [11], [12], diffusion methods [13], [14], dual averaging [15], [16], second-order algorithms [17], [18], and alternating direction method of multipliers (ADMM) [19]–[22], have been proposed. In these algorithms, the local objective functions and local data are kept private. At each iteration, the nodes only exchange their local iterates (namely, estimates of the optimization variable) with their neighbors.

When the local objective functions on the nodes are time-varying, one approach to addressing (1) is to treat it as a sequence of static problems. Accordingly, these static problems can be solved one by one with the aforementioned iterative decentralized static consensus optimization algorithms. This approach, however, is two timescales. The nodes have to communicate multiple rounds for solving each static problem, leading to poor communication efficiency. In addition, many applications, such as real-time tracking, are delay-sensitive and cannot afford the time of solving all the static problems exactly. Hence, a natural way is approximately solving each static problem and then using its output as the initial value for the static problem at the next time. Intuitively speaking, if the local objective functions vary sufficiently slowly over time, achieving satisfactory tracking performance is still possible. Prior works have combined this idea with ADMM [23], [24] and the gradient/subgradient methods [25] to solve (1). The slow variation of the local objective functions also enables the prediction–correction method, which utilizes the second-order information to predict the trajectory of the optimal solution and corrects the prediction by incorporating descent steps [26], [27]. A decentralized implementation of the prediction–correction technique is introduced in [28], while Simonetto and Dall’Anese [29] extended the application to constrained optimization. It should be emphasized that the

decentralized dynamic consensus optimization problem can be solved in a continuous-time manner as well [30]–[32], but the focus of this article is on the discrete-time regime. One feature of the aforementioned algorithms is that, at every time, the accurate local estimates of the nodes must be exchanged with their neighbors. The resulting communication cost is large and undesirable for applications whose bottleneck is communication bandwidth and/or power (for example, drone networks and wireless sensor networks).

The aim of this article is to solve the decentralized dynamic consensus optimization problem (1) with a communication-efficient ADMM. We resort to two strategies, quantization and communication censoring, to achieve favorable communication efficiency. Quantization requires the nodes in the decentralized network to transmit quantized messages rather than raw ones when they decide to communicate with their neighbors. With communication censoring, every node estimates the states of itself and its neighbors and utilizes the state variables, rather than the exact local iterates, to perform local computation. The transmission between neighbors is allowed if and only if the new local iterate is sufficiently different from the current state variable. These are two complementary strategies—quantization reduces the number of bits of every transmission, whereas communication censoring reduces the total number of transmissions.

Quantization has demonstrated significant performance gain in reducing communication cost in decentralized static optimization. However, due to the inexact information exchange in the optimization process, existing quantized algorithms, including [33]–[35], cannot guarantee exact convergence to the optimal solution. The work of [36] derives an upper bound of the convergence error of a quantized decentralized ADMM. Recently, adaptive quantization approaches have been proposed in [37]–[40] to improve the convergence accuracy. However, the diminishing step sizes required in such adaptive quantization approaches compromise the tracking performance and are not suitable for tracking the time-varying optimal solution of (1). The adaptive quantizer proposed in [41] addresses this issue, but its focus is on limiting the amount of bits for each communication, not on reducing the total number of transmissions.

Meanwhile, communication censoring, as a useful tool to reduce unnecessary transmissions, has also attracted intensive interests in decentralized static consensus optimization [42]–[46]. Under mild conditions, Liu *et al.* [45] proved the linear convergence rate of the communication-censored ADMM for the decentralized static consensus optimization problem. The work of [46] combines the communication-censoring strategy with linearized ADMM to further reduce the computation cost. Nevertheless, to the best of our knowledge, there is no work adopting the communication-censoring strategy in the dynamic scenario of our interest. As we will discuss in Section III, the censoring threshold is a key parameter that needs to be set differently for the dynamic and static scenarios, and it is essential to do so with convergence guarantee in an autonomous network. The communication-censoring strategy is also related to event-triggered control of continuous-time systems [31], [47]–[52], which

differs from the discrete-time systems studied in this article. The recent work of [53] also combines quantization and communication-censoring strategies but focuses on decentralized stochastic consensus optimization, which is different from the discussed dynamic case.

Our main contributions are summarized as follows.

- 1) To solve the decentralized dynamic consensus optimization problem (1), we develop a quantized and communication-censored ADMM (DQC-ADMM), with emphasis on reducing the communication cost during the optimization process. To the best of our knowledge, our work is the first on designing communication-efficient strategies for decentralized dynamic consensus optimization with provable convergence. In DQC-ADMM, information exchange is not necessarily allowed at every iteration but subject to a communication-censoring strategy. In addition, for necessary transmissions, the nodes transmit quantized values, not the raw ones, which further reduces the communication cost.
- 2) Under mild conditions (to be specified in Section IV), we theoretically justify the capability of the proposed DQC-ADMM to track the dynamic optimal solution with bounded error. This bounded, steady-state error is affected by the quantization error, the communication-censoring error, and the variation of the local objective functions. Our proof demonstrates the fundamental difference in the choices of the censoring threshold for the dynamic and static scenarios, in order to guarantee convergence in an autonomous network. Extensive numerical experiments are conducted, illustrating the tradeoff between the tracking performance and the communication efficiency.

The rest of this article is organized as follows. We first review the traditional ADMM to solve the decentralized dynamic consensus optimization problem in Section II. Then, we introduce the quantization and communication-censoring strategies to improve the communication efficiency of ADMM and propose DQC-ADMM in Section III. Next, we assess the performance of DQC-ADMM by theoretical analysis and numerical experiments in Sections IV and V, respectively. Section VI finally concludes this article.

Compared to the conference version [1], this article has two novel features. First, we quantize the differences of the state variables, rather than the iterates as in [1]. Consequently, the range of the transmitted messages can now be estimated, which significantly facilitates the implementation of the quantizer (see Lemma 4). New theoretical analysis on the tracking behavior is provided accordingly. Second, in the numerical experiments, we investigate how the variations of local objective functions influence the tracking accuracy (see Section V-C). We also demonstrate the performance of DQC-ADMM in handling streaming data (see Section V-D).

Notations: In this article, we use lowercases to denote scalars and vectors and uppercases to denote matrices unless specifically stated. $\|\cdot\|$ represents the Euclidean norm. For a vector v , $\|v\|_2$ represents its l_2 norm, and for a matrix G , $\|G\|_F$ represents its Frobenius norm. When matrix G is semidefinite,

the G -norm of vector v is $\|v\|_G = (v^T G v)^{1/2}$. The maximum singular value and the minimum nonzero singular value of the matrix G are $\sigma_{\max}(G)$ and $\tilde{\sigma}_{\min}(G)$, respectively. For any two matrices X and Y of the same dimension, $\langle X, Y \rangle$ represents their inner product.

II. PROBLEM STATEMENT

In this section, we describe the network and communication models. We also briefly review the classical ADMM, as a benchmark algorithm, for solving (1).

A. Network and Communication Models

1) *Network Model*: Suppose that throughout this article, the underlying communication graph with n nodes and r edges ($2r$ directed arcs) is undirected and connected. Define the underlying communication graph as $\mathcal{G} = \{\mathcal{V}, \mathcal{A}\}$, where $\mathcal{V}$ with cardinality $|\mathcal{V}| = n$ and $\mathcal{A}$ with cardinality $|\mathcal{A}| = 2r$ are sets of nodes and directed arcs, respectively. Two nodes i and j are neighbors if $(i, j) \in \mathcal{A}$, $\forall i, j \in \mathcal{V}$, and the neighborhood of node i is denoted as $\mathcal{N}_i$ with cardinality $|\mathcal{N}_i| = d_{ii}$. The symmetric adjacency matrix associated with the communication graph is denoted as $W \in \mathcal{R}^{n \times n}$, whose (i, j) th entry is 1 when i and j are neighbors or 0 otherwise. The communication graph has a degree matrix of $D \in \mathcal{R}^{n \times n}$ that is diagonal, and the i th diagonal element is exactly d_{ii} , the degree of node i . Let M_+ and $M_- \in \mathcal{R}^{n \times 2r}$ be the unsigned incidence matrix and the signed incidence matrix of the communication graph, respectively. For an arc l from i to j , the (i, l) th and (j, l) th entries of M_+ are both 1, whereas the (i, l) th entry of M_- is 1 and the (j, l) th entry of M_- is -1 . According to [54], we have the following equalities:

$$\begin{aligned} D + W &= \frac{1}{2} M_+ M_+^T \\ D - W &= \frac{1}{2} M_- M_-^T. \end{aligned}$$

2) *Communication Model*: Our focus in this article is on the synchronous algorithm, whose iterative process involves three distinct stages: communication, observation, and computation. In the communication stage, the nodes broadcast to and also receive from their neighbors according to a communication-censoring strategy, which shall be explained in Section III. It should be noted that one can also consider the unicast mode, in which every node communicates with its neighbors one by one. After communicating with their neighbors, each node observes its own time-varying local objective function. In the computation stage, the nodes estimate the state variables of itself and its neighbors based on the messages it has received. The state variables are different from the exact local iterates since the messages are quantized and possibly censored. The local update is then carried out by using the observed functions and the state variables. The state variables are also used in the next communication stage since the transmitted messages are the quantized differences between the local iterates and the state variables.

B. Decentralized Dynamic ADMM

Here, we introduce the classical ADMM developed in [23] to solve the decentralized dynamic consensus optimization. To solve (1) with ADMM in a decentralized manner, we define a local copy $x_i \in \mathcal{R}^p$ for the optimization variable $\tilde{x}$ at node i and an auxiliary variable z_{ij} associated with the arc $(i, j) \in \mathcal{A}$. When the network is connected, (1) is equivalent to

$$\begin{aligned} \min_{\{x_i\}, \{z_{ij}\}} \quad & \sum_{i=1}^n f_i^k(x_i) \\ \text{s.t.} \quad & x_i = z_{ij}, \quad x_j = z_{ij} \quad \forall (i, j) \in \mathcal{A}. \end{aligned} \quad (2)$$

For notational clarity, we collect the local copies x_i into a matrix $X = [x_1^T; \dots; x_n^T] \in \mathcal{R}^{n \times p}$ and also collect the auxiliary variables z_{ij} into a matrix $Z = [\dots; z_{ij}^T; \dots] \in \mathcal{R}^{2r \times p}$ in the order of arcs. Accordingly, we can rewrite (2) in the matrix form

$$\begin{aligned} \min_{X, Z} \quad & f^k(X) := \sum_{i=1}^n f_i^k(x_i) \\ \text{s.t.} \quad & \frac{1}{2} \begin{pmatrix} M_+^T + M_-^T \\ M_+^T - M_-^T \end{pmatrix} X = \begin{pmatrix} I_{2r} \\ I_{2r} \end{pmatrix} Z \end{aligned} \quad (3)$$

where $I_{2r} \in \mathcal{R}^{2r \times 2r}$ is an identity matrix. The unsigned and signed incidence matrices M_+ and M_- are defined in Section II-A. In the following, we define $A = 1/2[M_+^T + M_-^T; M_+^T - M_-^T]$ and $B = [I_{2r}; I_{2r}]$ to write the constraint in (3) as $AX = BZ$.

The augmented Lagrangian of (3) is

$$L^k(X, Z, \Omega) = f^k(X) + \langle \Omega, AX - BZ \rangle + \frac{c}{2} \|AX - BZ\|^2$$

where $c > 0$ is the penalty parameter. The Lagrange multiplier $\Omega \in \mathcal{R}^{4r \times p} := [\Theta; \Psi]$ consists of two matrices $\Theta, \Psi \in \mathcal{R}^{2r \times p}$ associated with the constraints $1/2(M_+^T + M_-^T)X = Z$ and $1/2(M_+^T - M_-^T)X = Z$, respectively. At time k , after observing the function f^k , the ADMM algorithm updates the primal and dual variables as

$$\begin{aligned} X(k) &= \arg \min_X L^k(X, Z(k-1), \Omega(k-1)) \\ Z(k) &= \arg \min_Z L^k(X(k), Z, \Omega(k-1)) \\ \Omega(k) &= \Omega(k-1) + c(AX(k) - BZ(k)). \end{aligned}$$

Note that given f^k , the updates are only carried out for one round in the dynamic setting.

It is proved in [23] that with initializations $\Theta(0) = -\Psi(0)$ and $Z(0) = (1/2)M_+^T X(0)$, the auxiliary variable Z can be eliminated and the Lagrange multiplier Ω can be replaced by a lower dimensional variable $\Lambda := [\lambda_1^T; \dots; \lambda_n^T] = M_- \Theta \in \mathcal{R}^{n \times p}$. The algorithm is given by

$$\begin{aligned} X(k) &= \arg \min_X f^k(X) + \langle X, cDX \rangle \\ &\quad + \langle X, \Lambda(k-1) - c(D+W)X(k-1) \rangle \end{aligned} \quad (4)$$

$$\Lambda(k) = \Lambda(k-1) + c(D-W)X(k). \quad (5)$$

Specifically, for every node i , it only needs to update a primal variable x_i and a dual variable $\lambda_i \in \mathcal{R}^p$ as

$$x_i(k) = \arg \min_{x_i} f_i^k(x_i) + cd_{ii}\|x_i\|_2^2 + \langle x_i, \lambda_i(k-1) - c \sum_{j \in \mathcal{N}_i} (x_i(k-1) + x_j(k-1)) \rangle \quad (6)$$

$$\lambda_i(k) = \lambda_i(k-1) + c \sum_{j \in \mathcal{N}_i} (x_i(k) - x_j(k)). \quad (7)$$

In summary, the ADMM algorithm for decentralized dynamic consensus optimization works as follows. At every time k , node i first observes its time-varying local objective function f_i^k and then computes the primal variable $x_i(k)$ by (6). Next, node i sends the updated primal variable $x_i(k)$ to and receives the updated primal variable $x_j(k)$ from its neighbors. After receiving $x_j(k)$ from all neighbors $j \in \mathcal{N}_i$, node i finally computes the dual variable $\lambda_i(k)$ by (7). Note that during the optimization process, the nodes in the network have to communicate with all neighbors every time. This is unfavorable given limited communication bandwidth and/or power. In Section III, we will improve the communication efficiency by designing a quantized and communication-censored ADMM algorithm.

III. ALGORITHM DEVELOPMENT

In this section, we introduce two strategies: quantization and communication censoring, to make the ADMM algorithms (6) and (7) more communication-efficient. The quantization strategy limits the number of bits in every transmission, whereas the communication-censoring strategy reduces the total number of transmissions. The resulting algorithm is called decentralized dynamic quantized and communication-censored ADMM, abbreviated as DQC-ADMM and described as follows.

Quantization. In the quantization algorithms, messages are quantized before being transmitted. Here, we implement an elementwise rounding quantizer $Q(\cdot)$ defined as follows. Consider a scalar y within the range of $[l, u)$ that can be evenly divided into $q = 2^b$ intervals of equal length $\Delta = (u - l)/q$ and define τ_t 's as the midpoints of the intervals with $\tau_t = l + (t + 1/2)\Delta$, $t = 0, 1, \dots, q - 1$. Then, the output of the quantizer is given by

$$Q(y) = \tau_t, \quad \text{if } \tau_t - \frac{\Delta}{2} \leq y < \tau_t + \frac{\Delta}{2}. \quad (8)$$

In practice, it is not necessary to transmit $Q(y) = \tau_t$ since transmitting the integer t is more efficient. Because the number of intervals is $q = 2^b$, only b bits need to be transmitted.

With the rounding quantizer, one natural way to reduce the communication cost in ADMM is to transmit the quantized local iterates $Q(x_i(k))$, rather than the raw vectors $x_i(k)$. This effective quantization strategy has already been implemented in the static case [36]. However, different from the static case in which the upper and lower bounds of the iterates could be roughly estimated, the dynamic case is more complicated. The optimal solution evolves over time, not mentioning the iterates. Therefore, it is difficult to estimate the upper and

lower bounds u and l . To adapt to the large range of iterates and guarantee the quantization resolution, more bits are needed in the dynamic case.

To narrow the quantization range, we adopt a difference-based quantization scheme. For each node i , state variables $\hat{x}_i$ and $\hat{x}_j$, initialized as $\hat{x}_i(0) = Q(x_i(0))$ and $\hat{x}_j(0) = Q(x_j(0))$, are introduced to estimate iterates of itself and its neighbors j . At time k , after finishing the local update, node i calculates the difference between the exact local iterate $x_i(k)$ and the state variable $\hat{x}_i(k-1)$ as

$$h_i(k) = x_i(k) - \hat{x}_i(k-1). \quad (9)$$

If the transmission is not censored, node i transmits the quantized difference $Q(h_i(k))$ using only pb bits to its neighbors and updates its state variable as $\hat{x}_i(k) = \hat{x}_i(k-1) + Q(h_i(k))$. Similarly, upon receiving the quantized difference $Q(h_j(k))$ from neighbors j , node i also updates the state variables of its neighbors as $\hat{x}_j(k) = \hat{x}_j(k-1) + Q(h_j(k))$. The elementwise rounding quantizer requires the upper and the lower bounds of the difference, and Section IV provides an approach to roughly estimating the bounds.

Note that decentralized static optimization algorithms with quantization strategies have already been studied in prior works [33]–[36]. However, most of these works can only converge to a neighborhood of the optimal solution due to the quantization errors. To further reduce the quantization errors, adaptive approaches have been proposed recently in [37]–[40]. Nevertheless, the diminishing step sizes required in such adaptive approaches compromise the tracking performance and make these methods not suitable for tracking the dynamic optimal solution of (1). As we will show in the theoretical analysis in Section IV, for the decentralized dynamic optimization problem of our interest, a constant tracking error caused by quantization is often acceptable. Also, note that for simplicity, here we only consider using the above-mentioned rounding quantizer. Other quantization approaches [55] can be applied to the proposed DQC-ADMM as well.

1) Communication Censoring: To further improve the communication efficiency, DQC-ADMM censors unnecessary transmissions during the optimization process. The rationale is that if the calculated local iterate is not sufficiently different from the current state variable, then this local iterate makes little impact on state updating, and hence, its transmission is not allowed. In this way, the overall communication cost can be reduced. Specifically, only if the difference $\|h_i(k)\|_2$ exceeds a nonnegative threshold α , then node i transmits $Q(h_i(k))$ to its neighbors. Otherwise, the transmission is censored, and the state variable of node i remains the same, as $\hat{x}_i(k) = \hat{x}_i(k-1)$. Accordingly, the primal update in (6) and the dual update in (7) can be modified to

$$x_i(k) = \arg \min_{x_i} f_i^k(x_i) + \langle x_i, \lambda_i(k-1) - c \sum_{j \in \mathcal{N}_i} (\hat{x}_i(k-1) + \hat{x}_j(k-1)) \rangle + cd_{ii}\|x_i\|_2^2 \quad (10)$$

$$\lambda_i(k) = \lambda_i(k-1) + c \sum_{j \in \mathcal{N}_i} (\hat{x}_i(k) - \hat{x}_j(k)). \quad (11)$$

Algorithm 1 DQC-ADMM

Require: Predefine the threshold α . Each node i initializes variables $x_i(0) = 0$, $\lambda_i(0) = 0$, $\hat{x}_i(0) = 0$, and $\hat{x}_j(0) = 0$ for all $j \in \mathcal{N}_i$. Then, it observes the initialized local objective function f_i^0 .

1: **for** iterations $k = 1, 2, \dots$ **do**

2: Observe local objective function f_i^k .

3: Calculate local primal variable $x_i(k)$ via

$$x_i(k) = \arg \min_{x_i} f_i^k(x_i) + \langle x_i, \lambda_i(k-1) - c \sum_{j \in \mathcal{N}_i} (\hat{x}_i(k-1) + \hat{x}_j(k-1)) \rangle + cd_{ii} \|x_i\|_2^2.$$

4: Calculate the difference $h_i(k) = x_i(k) - \hat{x}_i(k-1)$, and quantize it as $Q(h_i(k))$.

5: If $\|h_i(k)\|_2 \geq \alpha$, set $\hat{x}_i(k) = \hat{x}_i(k-1) + Q(h_i(k))$ and transmit $Q(h_i(k))$ to neighbors. Else, set $\hat{x}_i(k) = \hat{x}_i(k-1)$ and do not transmit.

6: If receiving $Q(h_j(k))$ from neighboring node j , update $\hat{x}_j(k) = \hat{x}_j(k-1) + Q(h_j(k))$. Else, set $\hat{x}_j(k) = \hat{x}_j(k-1)$.

7: Calculate local dual variable $\lambda_i(k)$ via

$$\lambda_i(k) = \lambda_i(k-1) + c \sum_{j \in \mathcal{N}_i} (\hat{x}_i(k) - \hat{x}_j(k)).$$

8: **end for**

Communication censoring has been proved as a powerful strategy to improve communication efficiency in decentralized static consensus optimization [42]–[45] and continuous-time control [31], [47]–[52]. The proposed DQC-ADMM is the first algorithm that applies the communication-censoring strategy in decentralized dynamic consensus optimization. In this dynamic setting, the communication-censoring strategy requires to use a constant threshold α , instead of a diminishing one in the static setting. Otherwise, since $x_i(k)$ is tracking the dynamic optimal solution $\tilde{x}^*(k)$, the difference $\|h_i(k)\|_2$ could be eventually larger than the diminishing threshold such that no communication is censored. We shall analyze the impact of the threshold α in Section IV.

The proposed DQC-ADMM is presented in Algorithm 1. After observing the time-varying local objective function f_i^k at time k , node i computes the primal variable $x_i(k)$ via (10). Then, $h_i(k)$, the difference between the exact local iterate $x_i(k)$ and the current state variable $\hat{x}_i(k-1)$ is calculated by (9). When $\|h_i(k)\|_2 \geq \alpha$, where α is a predefined threshold, the difference $h_i(k)$ is quantized and $Q(h_i(k))$ is transmitted to neighbors, and the state variable of node i becomes $\hat{x}_i(k) = \hat{x}_i(k-1) + Q(h_i(k))$. Otherwise, when $\|h_i(k)\|_2 < \alpha$, no transmission occurs and the state variable of node i remains the same as $\hat{x}_i(k) = \hat{x}_i(k-1)$. Similarly, if node i receives $Q(h_j(k))$ from any neighbor j , it sets $\hat{x}_j(k) = \hat{x}_j(k-1) + Q(h_j(k))$. Otherwise, $\hat{x}_j(k) = \hat{x}_j(k-1)$. Finally, the dual variable $\lambda_i(k)$ is updated on node i by (11).

For the ease of presentation, in Algorithm 1, we assume that every node maintains a state variable for each of its neighbors. To reduce the storage cost, in practice, every node i can only

maintain the summation of the neighboring state variables, denoted by Σ_i . After receiving $Q(h_j(k))$ from all neighbors j , node i calculates $\Sigma_i(k) = \Sigma_i(k-1) + \sum_{j \in \mathcal{N}_i} Q(h_j(k))$ and uses this value in the updates of (10) and (11).

IV. THEORETICAL ANALYSIS

This section provides a theoretical analysis on the performance of DQC-ADMM for the dynamic decentralized consensus optimization problem (1). The analysis serves to justify the capability of DQC-ADMM in tracking the time-varying optimal solution of (1). The tracking error is bounded and affected by the network topology, the variation of local objective functions, the quantization resolution, as well as the censoring threshold.

Before starting the analysis, we make the following commonly used assumptions.

Assumption 1: The local objective functions f_i^k are supposed to be strongly convex with constants $m_{f_i^k} > 0$. Given any $\tilde{x}, \tilde{y} \in \mathcal{R}^p$, $\langle \nabla f_i^k(\tilde{x}) - \nabla f_i^k(\tilde{y}), \tilde{x} - \tilde{y} \rangle \geq m_{f_i^k} \|\tilde{x} - \tilde{y}\|_2^2$ for any i and k . The strong convexity constants are lower bounded by $m_f := \inf_{i,k} m_{f_i^k}$. The gradients of the local cost functions are Lipschitz continuous with constants $M_{f_i^k} > 0$. Given any $\tilde{x}, \tilde{y} \in \mathcal{R}^p$, $\|\nabla f_i^k(\tilde{x}) - \nabla f_i^k(\tilde{y})\|_2 \leq M_{f_i^k} \|\tilde{x} - \tilde{y}\|_2$ for any i and k . The Lipschitz constants are upper bounded by $M_f := \sup_{i,k} M_{f_i^k}$.

With Assumption 1, the aggregated objective functions f^k are strongly convex with constant m_f and have Lipschitz continuous gradients with constant M_f . With the assumption of strong convexity, we know that there is a unique optimal solution $\tilde{x}^*(k)$ at every time k . Define $X^*(k) := [\dots; (\tilde{x}^*(k))^T; \dots] \in \mathcal{R}^{n \times p}$ as the stack of n copies of $(\tilde{x}^*(k))^T$, in a row-by-row manner. To guarantee bounded tracking error, we further assume that the variation of the decentralized dynamic consensus optimization problem (1) is sufficiently slow, as in Assumption 2.

Assumption 2: The time-varying optimal solution $X^*(k)$ and the corresponding gradients $\nabla f^k(X^*(k))$ have bounded variations, that is, $\|X^*(k) - X^*(k-1)\|_F \leq \epsilon_1$ and $\|\nabla f^k(X^*(k)) - \nabla f^{k-1}(X^*(k-1))\|_F \leq \epsilon_2$ with finite positive constants ϵ_1 and ϵ_2 for all times k .

Next, we rewrite the DQC-ADMM updates (10) and (11) so as to facilitate the convergence analysis. Collecting all the local copies $X(k) = [x_1(k)^T; \dots; x_n(k)^T] \in \mathcal{R}^{n \times p}$ and the state variables $\hat{X}(k) = [\hat{x}_1(k)^T; \dots; \hat{x}_n(k)^T] \in \mathcal{R}^{n \times p}$, and letting

$$E(k) := \hat{X}(k) - X(k). \quad (12)$$

We show that the introduced error $\|E(k)\|_F$ in DQC-ADMM is upper bounded by the quantization error and the predefined censoring threshold.

Lemma 1: For the updates (10) and (11), if the quantized difference $Q(h_i(k))$ is allowed to be transmitted only when $\|h_i(k)\|_2 \geq \alpha$, then for any time $k > 0$, the overall error introduced in the DQC-ADMM is upper bounded by

$$\|E(k)\|_F \leq \varsigma := \max \left\{ \sqrt{na}, \sqrt{np} \frac{\Delta}{2} \right\} \quad (13)$$

where Δ is the length of the quantization interval.

Proof: Define $\Delta \hat{x}_i(k) = \hat{x}_i(k) - \hat{x}_i(k-1)$, and here, $\Delta \hat{x}_i(k) = Q(h_i(k))$ if transmission is allowed and $\Delta \hat{x}_i(k) = 0$ if no transmission occurs. The introduced error for each node i is

$$\begin{aligned} e_i(k) &= \hat{x}_i(k) - x_i(k) \\ &= \Delta \hat{x}_i(k) + \hat{x}_i(k-1) - x_i(k) \\ &= \Delta \hat{x}_i(k) - h_i(k). \end{aligned}$$

According to the censoring rule, if $\|h_i(k)\|_2 \geq \alpha$, we have $\Delta \hat{x}_i(k) = Q(h_i(k))$, meaning that $\|e_i(k)\|_2 = \|Q(h_i(k)) - h_i(k)\|_2 \leq \sqrt{p}\Delta/2$. Otherwise, if $\|h_i(k)\|_2 < \alpha$, we have $\Delta \hat{x}_i(k) = 0$, meaning that $\|e_i(k)\|_2 = \|h_i(k)\|_2 \leq \alpha$. In both cases, $\|e_i(k)\|_2 = \|\hat{x}_i(k) - x_i(k)\|_2 < \max\{\alpha, \sqrt{p}\Delta/2\}$. Consequently, $\|E(k)\|_F \leq \max\{\sqrt{n}\alpha, \sqrt{np}\Delta/2\}$. ■

Then, we rewrite the updates (10) and (11) in the matrix form

$$\begin{aligned} X(k) &= \arg \min_X f^k(X) + \langle X, cDX \rangle \\ &\quad + \langle X, \Lambda(k-1) - c(D+W)\hat{X}(k-1) \rangle \end{aligned} \quad (14)$$

$$\Lambda(k) = \Lambda(k-1) + c(D-W)\hat{X}(k). \quad (15)$$

Using the equalities $D+W = M_+M_+^T/2$ and $D-W = M_-M_-^T/2$ as well as the definition of $E(k)$ in (12), (14), and (15) are equivalent to

$$\begin{aligned} \nabla f^k(X(k)) + \frac{c}{2}(M_+M_+^T + M_-M_-^T)X(k) + \Lambda(k-1) \\ - \frac{c}{2}M_+M_+^TX(k-1) - \frac{c}{2}M_+M_+^TE(k-1) = 0 \end{aligned} \quad (16)$$

$$\Lambda(k) - \Lambda(k-1) - \frac{c}{2}M_-M_-^TX(k) - \frac{c}{2}M_-M_-^TE(k) = 0. \quad (17)$$

Observe from (17) that $\Lambda(k)$ stays in the column space of $M_-M_-^T$ if $\Lambda(0)$ is also initialized therein. Therefore, it is convenient to introduce variables $\Theta(k) \in \mathcal{R}^{2r \times p}$, which stay in the column space of M_-^T , and let $\Lambda(k) = M_- \Theta(k)$ for any $k \geq 0$. Thus, (17) is equivalent to

$$\Theta(k) - \Theta(k-1) - \frac{c}{2}M_-^TX(k) - \frac{c}{2}M_-^TE(k) = 0. \quad (18)$$

Using (18) and $\Lambda(k-1) = M_- \Theta(k-1)$ to eliminate $\Lambda(k-1)$, as well as introducing $Z := (1/2)M_+^TX \in \mathcal{R}^{2r \times p}$, we rewrite (16) as

$$\begin{aligned} \nabla f^k(X(k)) + M_- \Theta(k) + cM_+(Z(k) - Z(k-1)) \\ - \frac{c}{2}M_-M_-^TE(k) - \frac{c}{2}M_+M_+^TE(k-1) = 0. \end{aligned} \quad (19)$$

The following analysis is based on the equivalent form of the quantized and communication-censored algorithms given by (18) and (19).

It has been proved in [23] that the Karush–Kuhn–Tucker (KKT) conditions of (3), which is equivalent to (1), are

$$\nabla f^k(X^*(k)) + M_- \Theta^*(k) = 0 \quad (20)$$

$$M_-^TX^*(k) = 0 \quad (21)$$

$$\frac{1}{2}M_+^TX^*(k) = Z^*(k) \quad (22)$$

where $(X^*(k), Z^*(k), \Theta^*(k))$ is the optimal primal–dual triplet at time k . Note that multiple optimal dual variables $\Theta^*(k)$ exist, but we are only interested in the one that lies in the

column space of M_-^T , whose existence as well as uniqueness have been proved in [23]. We will prove that the triplet $(X(k), Z(k), \Theta(k))$ generated by (18) and (19) always stays within a neighborhood of $(X^*(k), Z^*(k), \Theta^*(k))$. To do so, we define matrices

$$U := \begin{pmatrix} Z \\ \Theta \end{pmatrix} \quad \text{and} \quad G := \begin{pmatrix} cI_{2rn}, & 0_{2rn} \\ 0_{2rn}, & \frac{1}{c}I_{2rn} \end{pmatrix}$$

and analyze the dynamics of the Lyapunov function

$$\begin{aligned} \|U(k) - U^*(k)\|_G \\ = \sqrt{\frac{1}{c}\|\Theta(k) - \Theta^*(k)\|_F^2 + c\|Z(k) - Z^*(k)\|_F^2}. \end{aligned} \quad (23)$$

The following lemma characterizes the connection between $\|U(k-1) - U^*(k)\|_G$ and $\|U(k) - U^*(k)\|_G$.

Lemma 2: Suppose that Assumption 1 is satisfied and the dual variable $\Lambda(0)$ is initialized in the column space of $M_-M_-^T$. For the updates (10) and (11), if we set the penalty parameter as

$$\begin{aligned} c < \min \left\{ \frac{(\mu-1)\tilde{\sigma}_{\min}^2(M_-)}{\mu\eta_3\sigma_{\max}^2(M_+)} \right. \\ \left. \times \left(\frac{\eta_1}{4} + \frac{\eta_2\sigma_{\max}^2(M_+)}{8} \right)^{-1} \left(m_f - \frac{\eta_3\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)} \right) \right\} \end{aligned} \quad (24)$$

then it follows that:

$$\|U(k-1) - U^*(k)\|_G + \phi\varsigma \geq \sqrt{1+\delta}\|U(k) - U^*(k)\|_G. \quad (25)$$

Here

$$\phi := \sqrt{\left(\frac{\eta_3}{2} + \frac{\delta}{c} \right) \frac{c^2}{\tilde{\sigma}_{\min}^2(M_-)} (\sigma_{\max}^4(M_+) + \sigma_{\max}^4(M_-))} + s \quad (26)$$

$$s := \frac{c\sigma_{\max}^4(M_-)}{4\eta_1} + \frac{c\sigma_{\max}^2(M_+)}{2\eta_2} + \frac{\sigma_{\max}^2(M_-)}{2\eta_3} \quad (27)$$

$$\varsigma := \max \left\{ \sqrt{n}\alpha, \sqrt{np}\frac{\Delta}{2} \right\} \quad (28)$$

$$\begin{aligned} \delta \leq \min \left\{ \frac{(\mu-1)\tilde{\sigma}_{\min}^2(M_-)}{2\mu\sigma_{\max}^2(M_+)} - \frac{c\eta_3}{2} \right. \\ \times \left(\frac{c\sigma_{\max}^2(M_+)}{4} + \frac{2\mu M_f^2}{c\tilde{\sigma}_{\min}^2(M_-)} \right)^{-1} \\ \left. \times \left(m_f - \frac{c\eta_1}{4} - \frac{c\eta_2\sigma_{\max}^2(M_+)}{8} - \frac{\eta_3\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)} \right) \right\}. \end{aligned} \quad (29)$$

$\eta_1 > 0$, $\eta_2 > 0$, $\eta_3 > 0$, and $\mu > 1$ are arbitrary constants, m_f is the lower bound of the strong convexity constants for the local objective functions, M_f is the upper bound of the Lipschitz constants for the local gradients, $\sigma_{\max}(M_+)$ is the maximum singular value of M_+ (the unsigned incidence matrix), and $\tilde{\sigma}_{\min}(M_-)$ is the minimum nonzero singular value of M_- (the signed incidence matrix).

Proof: See Appendix A. ■

The following lemma further characterizes the connection between $\|U(k-1) - U^*(k)\|_G$ and $\|U(k-1) - U^*(k-1)\|_G$.

Note that in the static case, the gap between these two terms vanishes.

Lemma 3: Suppose that Assumption 2 is satisfied and the dual variable $\Lambda(0)$ is initialized in the column space of $M_-M_-^T$. For the updates (10) and (11), it follows that:

$$\|U(k-1) - U^*(k)\|_G \leq \|U(k-1) - U^*(k-1)\|_G + g \quad (30)$$

where

$$g := \frac{\sqrt{c}\epsilon_1}{2}\sigma_{\max}(M_+) + \frac{\epsilon_2}{\sqrt{c}\tilde{\sigma}_{\min}(M_-)}. \quad (31)$$

Here, ϵ_1 and ϵ_2 are the constants defined in Assumption 2.

Proof: See Appendix B. ■

Combining Lemmas 2 and 3, we are able to attain an upper bound for the asymptotic tracking error. We summarize the result in the following main theorem.

Theorem 1: Suppose that Assumptions 1 and 2 are satisfied and the dual variable $\Lambda(0)$ is initialized in the column space of $M_-M_-^T$. For the DQC-ADMM updates (10) and (11), let the penalty parameter c satisfy

$$c \in \left(0, \min \left\{ \frac{4m_f}{\eta_1}, \frac{(\mu-1)\tilde{\sigma}_{\min}^2(M_-)}{\mu\eta_3\sigma_{\max}^2(M_+)} \left(\frac{\eta_1}{4} + \frac{\eta_2\sigma_{\max}^2(M_+)}{8} \right)^{-1} \right. \right. \\ \left. \left. \times \left(m_f - \frac{\eta_3\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)} \right) \right\} \right) \quad (32)$$

then we have

$$\limsup_{k \rightarrow +\infty} \|X(k) - X^*(k)\|_F \\ \leq \left(\sqrt{m_f - \frac{c\eta_1}{4}} \right)^{-1} \\ \times \left(\left(1 + \max \left\{ \sqrt{\frac{\eta_2}{2}}, \sqrt{\frac{c\eta_3}{2}} \right\} \right) \frac{g + \phi\varsigma}{1 - (\sqrt{1+\delta})^{-1}} + g + \sqrt{s}\varsigma \right) \quad (33)$$

where ϕ , s , ς , and δ are defined in (26)–(29), respectively, while $\eta_1 > 0$, $\eta_2 > 0$, $\eta_3 > 0$, and $\mu > 1$ are arbitrary constants.

Proof: See Appendix C. ■

Note that to guarantee $c > 0$ and $\delta > 0$ in Theorem 1, we can choose μ to be slightly larger than 1, while η_1 , η_2 , and η_3 are to be positive constants of sufficiently small values. Furthermore, the dual variable $\Lambda(0)$ must be initialized in the column space of $M_-M_-^T$. This is easy to satisfy since one can simply let $\Lambda(0) = 0$.

Remark 1: Theorem 1 shows that the asymptotic tracking error is influenced by the network topology (characterized by $\sigma_{\max}(M_+)$, the maximum singular value of the unsigned incidence matrix, and $\tilde{\sigma}_{\min}(M_-)$, the minimum nonzero singular value of the signed incidence matrix), slopes of the objective functions (characterized by the Lipschitz continuous gradient constant M_f and the strong convexity constant m_f), the variation of the objective functions (characterized by ϵ_1 and ϵ_2), as well as the quantization and communication-censoring strategies implemented to improve the communication efficiency (characterized by α and Δ , respectively).

In particular, there is an approximately linear dependence of the asymptotic tracking error on g given by (31), which combines ϵ_1 (the variation of the optimal solutions) and ϵ_2 (the variation of the optimal gradients). This is reasonable—when the objective functions vary fast, tracking the optimal solution is difficult. There is also an approximately linear dependence of the asymptotic tracking error on $\varsigma := \max\{\sqrt{n}\alpha, \sqrt{n\rho}\Delta/2\}$ given by (28), which is determined by the communication-censoring error α and the quantization error Δ . Note that when $\alpha < \sqrt{\rho}\Delta/2$, the quantization error dominates. Otherwise, when $\alpha > \sqrt{\rho}\Delta/2$, the communication-censoring error is more important. In the analysis, the constants η_1 , η_2 , and η_3 are introduced to split the error $\|E(k)\|_F$ from the other terms, where $E(k) := \hat{X}(k) - X(k)$ is given in (12) and $\|E(k)\|_F \leq \varsigma$, $\forall k$. When neither communication censoring nor quantization is used, η_1 , η_2 , and η_3 disappear and ς equals 0 such that the upper bound of the asymptotic tracking error degenerates to that of the traditional decentralized dynamic ADMM proposed in [23].

Note that the elementwise rounding quantizer requires nodes to estimate the upper and lower bounds. The following lemma provides a guideline for roughly estimating the range of the transmitted messages during the optimization process.

Lemma 4: Suppose that Assumptions 1 and 2 are satisfied and the dual variable $\Lambda(0)$ is initialized in the column space of $M_-M_-^T$. For the updates (10) and (11), set the penalty parameter the same as (32) and allow node i to transmit the quantized message only when $\|h_i(k)\|_2 \geq \alpha$. Then, for any time $k > 0$, the magnitude of the collected difference, $H(k) := [h_1(k)^T; \dots; h_n(k)^T]$, has an upper bound given by

$$\|H(k)\|_F \leq 2 \left(\sqrt{m_f - \frac{c\eta_1}{4}} \right)^{-1} \\ \times \left(\left(\max \left\{ \sqrt{\frac{\eta_2}{2}}, \sqrt{\frac{c\eta_3}{2}} \right\} + 1 \right) \right. \\ \times \left(\|U(0) - U^*(0)\|_G + \frac{g + \phi\varsigma}{1 - \sqrt{1+\delta}^{-1}} \right) \\ \left. + g + \sqrt{s}\varsigma \right) + \epsilon_1 + \varsigma \quad (34)$$

where ϕ , s , ς , and δ are defined in (26)–(29), respectively, while $\eta_1 > 0$, $\eta_2 > 0$, and $\eta_3 > 0$ are arbitrary constants, $\|U(k) - U^*(k)\|_G$ is defined in (23), and $\|U(0) - U^*(0)\|_G$ is determined by the initial values of the primal and dual variables.

Proof: See Appendix D. ■

The derived range depends on $U^*(0)$ that can be solved from the KKT conditions (20)–(22). This step needs global coordination of the entire network and can be done offline. An alternative is to construct a “virtual” optimization problem, which is easy to solve but satisfies Assumptions 1 and 2, at time $k = 0$. For example, if the objective functions $f_i^0(\tilde{x}(0)) = (\|\tilde{x}(0)\|_2^2)/2$ for all nodes i , then $U^*(0) = 0$.

V. NUMERICAL RESULTS

This section demonstrates the performance of the proposed DQC-ADMM in tracking the optimal solution trajectory with

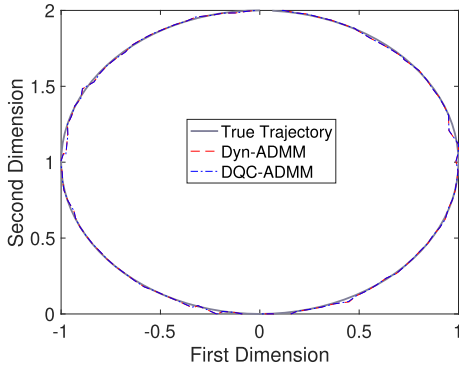


Fig. 1. Trajectory of circle.

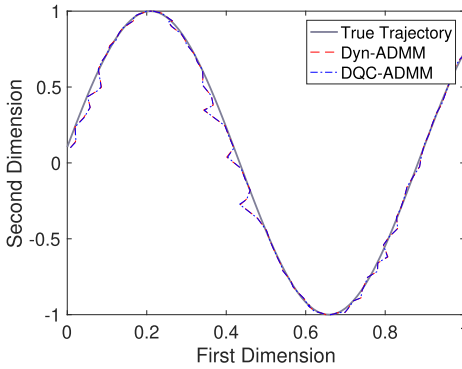


Fig. 2. Trajectory of sinusoid.

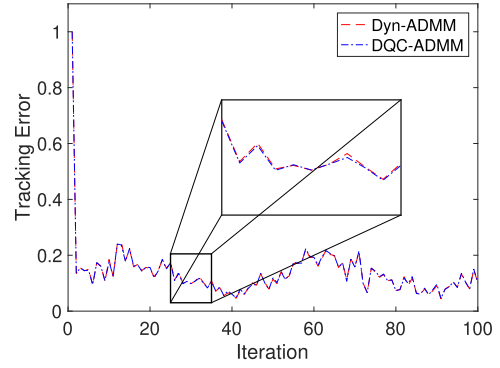


Fig. 3. Error of tracking the circle.

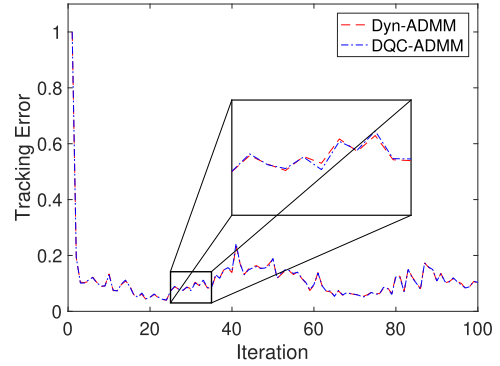


Fig. 4. Error of tracking the sinusoid.

extensive numerical experiments. A bidirectional network with $n = 50$ nodes and $r = 123$ arcs is randomly generated. Unless specifically pointed out, node i has the local objective function f_i^k given by

$$f_i^k(\tilde{x}(k)) = \frac{1}{2} \|A_i^k \tilde{x}(k) - b_i^k\|_2^2 \quad (35)$$

where $A_i^k \in \mathcal{R}^{p \times p}$ and $b_i^k \in \mathcal{R}^p$ vary with time and belong to node i . The elements in A_i^k are generated from the Gaussian distribution $\mathcal{N}(0, 1)$. Given $\tilde{x}(k)$, the true value of interest at time k , $b_i^k = A_i^k \tilde{x}(k) + \zeta_i^k$ with ζ_i^k following the Gaussian distribution $\mathcal{N}(0, 0.001)$. To clearly show the tracking trajectories in the figures, we choose the number of dimensions as $p = 2$. The resulting decentralized dynamic least-squares problem is

$$\tilde{x}^*(k) = \arg \min_{\tilde{x}(k)} \sum_{i=1}^n \frac{1}{2} \|A_i^k \tilde{x}(k) - b_i^k\|_2^2.$$

A. Effect of Quantization

Here, we exclude the communication-censoring strategy (namely, setting $\alpha = 0$) and study how quantization affects the performance of DQC-ADMM. Two different true trajectories, the circle and the sinusoid, as, respectively, shown in Figs. 1 and 2, are taken into consideration. Both trajectories have two dimensions and slowly evolve within 100 iterations. We divide each dimension into $q = 256$ quantization intervals, and hence, the corresponding number of bits is

$b = 8$. The estimated range of the circle trajectory is $[-2, 2]$ for the first dimension and $[-1, 3]$ for the second dimension, whereas the estimated range of the sinusoid trajectory is $[-0.5, 1.5]$ for the first dimension and $[-2, 2]$ for the second dimension. DQC-ADMM is compared with Dyn-ADMM, the decentralized dynamic ADMM algorithm without quantization proposed in [23]. In the two algorithms, the penalty parameter is set as $c = 0.001$. Figs. 1 and 2 show the capability of both DQC-ADMM and Dyn-ADMM in tracking the true trajectories, subject to slight biases. We define the tracking error as $\|X(k) - X^*(k)\| / \|X(0) - X^*(0)\|$ and quantify how it evolves with the optimization process in Figs. 3 and 4. For both trajectories, the tracking errors of DQC-ADMM and Dyn-ADMM are similar, indicating that the quantized transmissions do not obviously degrade the tracking performance. Next, we focus on the circle trajectory to study how different quantization resolutions affect the performance of DQC-ADMM. Divide each dimension into $q = 16$, $q = 64$, and $q = 256$ quantization intervals, respectively. The numbers of bits are hence $b = 4$, $b = 6$, and $b = 8$, and each node transmits 8, 12, and 16 bits at every time. The corresponding tracking performance and communication cost of DQC-ADMM is shown in Fig. 5. Tracking errors increase when the numbers of bits decrease. The linear correlation between the communication cost and the transmitted bits encourages a small number of transmitted bits, but the large tracking error led by the low-resolution quantization, such as $b = 4$, suggests the necessity in considering the tradeoff

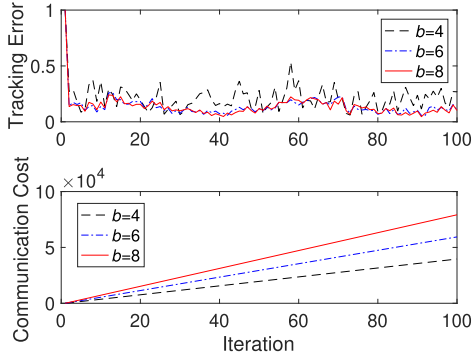


Fig. 5. Tracking error and communication cost of DQC-ADMM with different quantization resolutions.

between the tracking performance and communication cost. Note that when the number of bits is changed from $b = 8$ to $b = 6$, the tracking performance is almost the same, indicating the unnecessary of a high-resolution quantization. This result coincides with the analysis in Section IV that the tracking error is affected by the maximum of quantization and communication-censoring errors, as well as the variation of the local objective functions. When the quantization resolution is sufficiently high, the variation of the local objective functions dominates. In this case, choosing a proper quantization resolution helps attain reasonable tracking accuracy with low communication cost.

B. Effect of Communication Censoring

To evaluate how the censoring threshold α affects the tracking performance, we compare DQC-ADMM with different censoring thresholds. In the circle trajectory, we divide each dimension into $q = 64$ ($b = 6$) quantization intervals and set the communication-censoring thresholds as $\alpha = 0$, $\alpha = 0.044$, and $\alpha = 0.1$. Fig. 6 shows the tradeoff between tracking error and communication cost of DQC-ADMM with different communication-censoring thresholds. When the communication-censoring threshold is changed from $\alpha = 0$ to $\alpha = 0.044$, the communication cost is reduced, but the tracking errors are almost the same. This result corroborates the analysis in Section IV that the quantization error dominates the communication-censoring error when $\alpha < \sqrt{p}\Delta/2 \simeq 0.044$. If we keep increasing the communication-censoring threshold to $\alpha = 0.1$, the slight degradation of the tracking performance can be compensated by the reduced communication cost.

C. Effect of Variation

Another factor influencing the tracking accuracy is the variation of local objective functions. To investigate how the variation affects the performance of DQC-ADMM, we still consider the circle trajectory, but let the true values evolve at different speeds. Within 100 iterations, the true values traverse 100%, 50%, and 20% of the circle. In DQC-ADMM, we divide each dimension into $q = 64$ ($b = 6$) quantization intervals and let the communication-censoring threshold $\alpha = 0.044$. As shown in Fig. 7, the tracking error is the largest

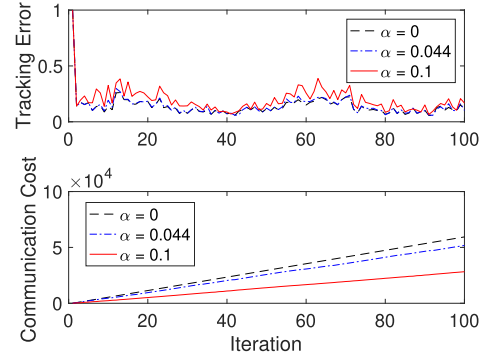


Fig. 6. Tracking error and communication cost of DQC-ADMM with different censoring thresholds.

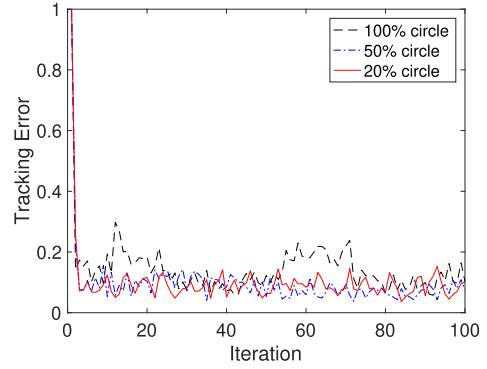


Fig. 7. Tracking error of DQC-ADMM with different time-varying objective functions.

when tracking the 100% circle (namely, the variation is the largest). When tracking the 50% or 20% circle, the variation becomes smaller and is no longer dominating comparing to the errors caused by quantization and communication censoring. Therefore, the tracking errors are much smaller than that of tracking the 100% circle.

D. Performance in Handling Streaming Data

Now, we evaluate the performance of DQC-ADMM for decentralized online consensus optimization, in which online data streams are separately collected by the nodes. The decentralized online consensus optimization problem [56], [57] is given by

$$\tilde{x}^* = \arg \min_{\tilde{x}} \sum_{i=1}^n \sum_{t=1}^T g_i^t(\tilde{x}) \quad (36)$$

which can be looked as a special case of the decentralized dynamic consensus optimization problem (1). For each node i , the dynamic local objective function at time k is a summation of instantaneous cost functions g_i^t , namely, $f_i^k(x) = \sum_{t=1}^k g_i^t(x)$. When $k \rightarrow T$, the dynamic optimal solution $\tilde{x}^*(k)$ eventually approaches $\tilde{x}^*$. The corresponding decentralized least squares problem is

$$\tilde{x}^*(k) = \arg \min_{\tilde{x}(k)} \sum_{i=1}^n \sum_{t=1}^k \frac{1}{2} \|A_i^t \tilde{x}(k) - b_i^t\|_2^2.$$

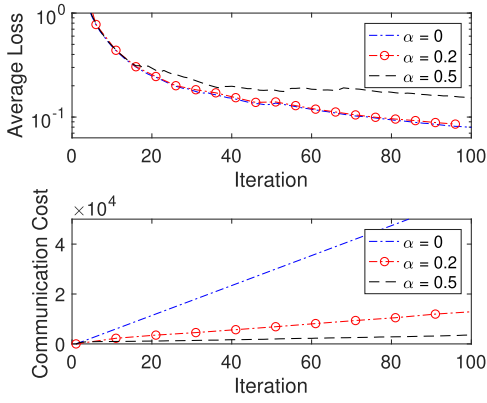


Fig. 8. Average loss and communication cost of DQC-ADMM for streaming data.

The average loss $\mathcal{L}$ is introduced to evaluate the performance of DQC-ADMM for the decentralized online optimization as

$$\mathcal{L} = \max_j \frac{1}{nT} \sum_{i=1}^n \sum_{t=1}^T (g_i^t(x_j^t) - g_i^t(\tilde{x}^*)). \quad (37)$$

It calculates the difference between the estimate of an arbitrary agent j and the final optimal solution $\tilde{x}^*$ on all instantaneous cost functions over time 1 to T . We divide each dimension into $q = 64$ ($b = 6$) quantization intervals and choose $p = 2$. We compare three values of the threshold α , from 0 and 0.2 to 0.5. Fig. 8 shows that with a proper threshold $\alpha = 0.2$, the average losses of DQC-ADMM with and without censoring are almost the same, but DQC-ADMM with censoring significantly reduces the communication cost. When the threshold α is set larger, e.g., $\alpha = 0.5$, the communication cost could further decrease, but the average loss of DQC-ADMM deteriorates.

VI. CONCLUSION

This article proposes the DQC-ADMM algorithm, aiming at tracking the optimal solution trajectory of a dynamic optimization problem over a decentralized network. DQC-ADMM is a communication-efficient algorithm, due to the applications of the quantization and communication-censoring strategies. Specifically, unnecessary information exchanges are censored in DQC-ADMM. Furthermore, when communication happens, DQC-ADMM only transmits a limited number of bits by invoking quantization. We theoretically justify the capability of DQC-ADMM in tracking the optimal solution trajectory. The tracking error is bounded and affected by the quantization error, the communication-censoring error, and the variation of the local objective functions. With extensive numerical experiments, we validate the communication efficiency and tracking performance, for both batch and streaming data. In the future work, we will extend the communication-saving strategies and the analytical tools to the training of neural networks, where the objective functions are nonconvex.

APPENDIX A PROOF OF LEMMA 2

Proof: The proof is organized as four steps.

Step 1: We first characterize the relationship between the iterates and the dynamic optimal solutions. Subtracting (19) by the KKT condition (20) gives

$$\begin{aligned} & \nabla f^k(X(k)) - \nabla f^k(X^*(k)) \\ &= \frac{c}{2} M_- M_-^T E(k) + \frac{c}{2} M_+ M_+^T E(k-1) \\ & \quad - M_- (\Theta(k) - \Theta^*(k)) - c M_+ (Z(k) - Z(k-1)). \end{aligned} \quad (38)$$

For both sides of (38), taking inner product with $X(k) - X^*(k)$ yields

$$\begin{aligned} & \langle \nabla f^k(X(k)) - \nabla f^k(X^*(k)), X(k) - X^*(k) \rangle \\ & \stackrel{(a)}{=} -\langle \Theta(k) - \Theta^*(k), M_-^T (X(k) - X^*(k)) \rangle \\ & \quad - 2c \langle Z(k) - Z(k-1), Z(k) - Z^*(k) \rangle \\ & \quad + \frac{c}{2} \langle M_-^T E(k), M_-^T (X(k) - X^*(k)) \rangle \\ & \quad + c \langle M_+^T E(k-1), Z(k) - Z^*(k) \rangle \\ & \stackrel{(b)}{=} -\frac{2}{c} \langle \Theta(k) - \Theta^*(k), \Theta(k) - \Theta(k-1) \rangle \\ & \quad + \langle \Theta(k) - \Theta^*(k), M_-^T E(k) \rangle \\ & \quad - 2c \langle Z(k) - Z(k-1), Z(k) - Z^*(k) \rangle \\ & \quad + \frac{c}{2} \langle M_-^T E(k), M_-^T (X(k) - X^*(k)) \rangle \\ & \quad + c \langle M_+^T E(k-1), Z(k) - Z^*(k) \rangle. \end{aligned} \quad (39)$$

In (a), we use the fact that $Z(k) - Z^*(k) = (1/2) M_+^T (X(k) - X^*(k))$, which comes from the definition of $Z(k) = (1/2) M_+^T X(k)$ and the KKT condition $Z^*(k) = (1/2) M_+^T X^*(k)$ given by (22). In (b), we use the dual update $\Theta(k) - \Theta(k-1) - (c/2) M_-^T X(k) - (c/2) M_-^T E(k) = 0$ given by (18) and the KKT condition $M_-^T X^*(k) = 0$ given by (21) to split the term $-\langle \Theta(k) - \Theta^*(k), M_-^T (X(k) - X^*(k)) \rangle$.

Step 2: The inner products in (39) couple the variables. In this step, we manipulate the inner products to characterize the connection between $\|U(k) - U^*(k)\|_G^2$ and $\|U(k-1) - U^*(k)\|_G^2$.

By Assumption 1, f^k is strongly convex with constant m_f . Thus, the left-hand side of (39) has a lower bound

$$\begin{aligned} & \langle \nabla f^k(X(k)) - \nabla f^k(X^*(k)), X(k) - X^*(k) \rangle \\ & \geq m_f \|X(k) - X^*(k)\|_F^2. \end{aligned} \quad (40)$$

In the following, we consider the right-hand side of (39) and derive an upper bound. From the equality $2\langle a, b \rangle = \|a\|^2 + \|b\|^2 - \|a - b\|^2$ that hold for any vectors a and b of the same size, we know that the terms without the introduced error $E(k)$ equal to

$$\begin{aligned} & -\frac{2}{c} \langle \Theta(k) - \Theta^*(k), \Theta(k) - \Theta(k-1) \rangle \\ & \quad - 2c \langle Z(k) - Z(k-1), Z(k) - Z^*(k) \rangle \\ &= \frac{1}{c} \|\Theta(k-1) - \Theta^*(k)\|_F^2 - \frac{1}{c} \|\Theta(k) - \Theta(k-1)\|_F^2 \\ & \quad - \frac{1}{c} \|\Theta(k) - \Theta^*(k)\|_F^2 + c \|Z(k-1) - Z^*(k)\|_F^2 \\ & \quad - c \|Z(k) - Z(k-1)\|_F^2 - c \|Z(k) - Z^*(k)\|_F^2. \end{aligned} \quad (41)$$

Notice that for any matrices A and B of the same size, $\langle A, B \rangle \leq (\eta/2) \|A\|_F^2 + (1/2\eta) \|B\|_F^2$ for any $\eta > 0$, as well

as $\|AB\|_F \leq \sigma_{\max}(A)\|B\|_F$, where $\sigma_{\max}(A)$ denotes the maximum singular value of the matrix A . The rest of the right-hand side of (39) hence has an upper bound

$$\begin{aligned} & \frac{c}{2} \langle M_-^T E(k), M_-^T (X(k) - X^*(k)) \rangle \\ & + c \langle M_+^T E(k-1), Z(k) - Z^*(k) \rangle \\ & + \langle \Theta(k) - \Theta^*(k), M_-^T E^k \rangle \\ & \leq \frac{c\eta_1}{4} \|X(k) - X^*(k)\|_F^2 + \frac{c\sigma_{\max}^4(M_-)}{4\eta_1} \|E(k)\|_F^2 \\ & + \frac{c\eta_2}{2} \|Z(k) - Z^*(k)\|_F^2 + \frac{c\sigma_{\max}^2(M_+)}{2\eta_2} \|E(k-1)\|_F^2 \\ & + \frac{\eta_3}{2} \|\Theta(k) - \Theta^*(k)\|_F^2 + \frac{\sigma_{\max}^2(M_-)}{2\eta_3} \|E(k)\|_F^2 \quad (42) \end{aligned}$$

where η_1 , η_2 , and η_3 are positive constants. According to the censoring rule of DQC-ADMM, we know that the quantized difference $Q(h_i(k))$ is allowed to transmit only when $\|h_i(k)\|_2 \geq \alpha$, and Lemma 1 shows the introduced error $\|E(k)\|_F \leq \varsigma := \max\{\sqrt{n}\alpha, \sqrt{n\rho}\Delta/2\}$, and so is $\|E(k-1)\|_F$. Then, (42) can be further bounded by

$$\begin{aligned} & \frac{c}{2} \langle M_-^T E(k), M_-^T (X(k) - X^*(k)) \rangle \\ & + c \langle M_+^T E(k-1), Z(k) - Z^*(k) \rangle \\ & + \langle \Theta(k) - \Theta^*(k), M_-^T E^k \rangle \\ & \leq \frac{c\eta_1}{4} \|X(k) - X^*(k)\|_F^2 + \frac{c\eta_2}{2} \|Z(k) - Z^*(k)\|_F^2 \\ & + \frac{\eta_3}{2} \|\Theta(k) - \Theta^*(k)\|_F^2 + s\varsigma^2 \quad (43) \end{aligned}$$

where

$$s := \frac{c\sigma_{\max}^4(M_-)}{4\eta_1} + \frac{c\sigma_{\max}^2(M_+)}{2\eta_2} + \frac{\sigma_{\max}^2(M_-)}{2\eta_3} > 0.$$

Substituting (40), (41), and (43) into (39) yields

$$\begin{aligned} & c \|Z(k) - Z^*(k)\|_F^2 + \frac{1}{c} \|\Theta(k) - \Theta^*(k)\|_F^2 \\ & \leq c \|Z(k-1) - Z^*(k)\|_F^2 + \frac{1}{c} \|\Theta(k-1) - \Theta^*(k)\|_F^2 \\ & - c \|Z(k) - Z(k-1)\|_F^2 - \frac{1}{c} \|\Theta(k) - \Theta(k-1)\|_F^2 \\ & + \left(\frac{c\eta_1}{4} - m_f\right) \|X(k) - X^*(k)\|_F^2 + \frac{c\eta_2}{2} \|Z(k) - Z^*(k)\|_F^2 \\ & + \frac{\eta_3}{2} \|\Theta(k) - \Theta^*(k)\|_F^2 + s\varsigma^2 \quad (44) \end{aligned}$$

or equivalently

$$\begin{aligned} & (1+\delta)c \|Z(k) - Z^*(k)\|_F^2 + (1+\delta)\frac{1}{c} \|\Theta(k) - \Theta^*(k)\|_F^2 \\ & \leq c \|Z(k-1) - Z^*(k)\|_F^2 + \frac{1}{c} \|\Theta(k-1) - \Theta^*(k)\|_F^2 \\ & - c \|Z(k) - Z(k-1)\|_F^2 - \frac{1}{c} \|\Theta(k) - \Theta(k-1)\|_F^2 \\ & + \left(\frac{c\eta_1}{4} - m_f\right) \|X(k) - X^*(k)\|_F^2 \\ & + \left(\frac{c\eta_2}{2} + c\delta\right) \|Z(k) - Z^*(k)\|_F^2 \\ & + \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right) \|\Theta(k) - \Theta^*(k)\|_F^2 + s\varsigma^2 \quad (45) \end{aligned}$$

for any constant δ . According to the definition of $\|\cdot\|_G$, (45) translates to the connection between $\|U(k) - U^*(k)\|_G^2$ and $\|U(k-1) - U^*(k)\|_G^2$ as

$$\begin{aligned} & (1+\delta) \|U(k) - U^*(k)\|_G^2 \\ & \leq \|U(k-1) - U^*(k)\|_G^2 \\ & - c \|Z(k) - Z(k-1)\|_F^2 - \frac{1}{c} \|\Theta(k) - \Theta(k-1)\|_F^2 \\ & + \left(\frac{c\eta_1}{4} - m_f\right) \|X(k) - X^*(k)\|_F^2 \\ & + \left(\frac{c\eta_2}{2} + c\delta\right) \|Z(k) - Z^*(k)\|_F^2 \\ & + \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right) \|\Theta(k) - \Theta^*(k)\|_F^2 + s\varsigma^2. \quad (46) \end{aligned}$$

Step 3: Note that in analyzing the classical dynamic ADMM [23], a key inequality also appears, in the similar form of (46). However, therein, the last four terms at the right-hand side of (46) are absent. In this step, we proceed to manipulate (46) through establishing upper bounds for $\|Z(k) - Z^*(k)\|_F^2$ and $\|\Theta(k) - \Theta^*(k)\|_F^2$.

- 1) For $\|Z(k) - Z^*(k)\|_F^2$, from $Z(k) := (1/2)M_+^T X(k)$ and $Z^*(k) := (1/2)M_+^T X^*(k)$, we can bound $\|Z(k) - Z^*(k)\|_F^2$ with $\|X(k) - X^*(k)\|_F^2$ as

$$\begin{aligned} \|Z(k) - Z^*(k)\|_F^2 &= \frac{1}{4} \|M_+^T (X(k) - X^*(k))\|_F^2 \\ &\leq \frac{\sigma_{\max}^2(M_+)}{4} \|X(k) - X^*(k)\|_F^2. \quad (47) \end{aligned}$$

- 2) For $\|\Theta(k) - \Theta^*(k)\|_F^2$, recall (38) to write

$$\begin{aligned} & \|M_- (\Theta(k) - \Theta^*(k))\|_F^2 \\ &= \left\| \nabla f^k(X(k)) - \nabla f^k(X^*(k)) + cM_+(Z(k) - Z(k-1)) \right. \\ & \quad \left. - \frac{c}{2} M_- M_-^T E(k) - \frac{c}{2} M_+ M_+^T E(k-1) \right\|_F^2. \quad (48) \end{aligned}$$

From the inequality $\|A+B\|_F^2 \leq \mu\|A\|_F^2 + \mu/(\mu-1)\|B\|_F^2$ that holds for any matrices A and B of the same size and for all $\mu > 1$, we know that (48) has an upper bound of

$$\begin{aligned} & \|M_- (\Theta(k) - \Theta^*(k))\|_F^2 \\ & \leq 2 \left(\mu \|\nabla f^k(X(k)) - \nabla f^k(X^*(k))\|_F^2 \right. \\ & \quad \left. + \frac{\mu}{\mu-1} \|cM_+(Z(k) - Z(k-1))\|_F^2 \right) \\ & \quad + 2 \left(2 \left\| \frac{c}{2} M_- M_-^T E(k) \right\|_F^2 \right. \\ & \quad \left. + 2 \left\| \frac{c}{2} M_+ M_+^T E(k-1) \right\|_F^2 \right) \quad (49) \end{aligned}$$

with all $\mu > 1$. Furthermore, by Assumption 1, f^k has Lipschitz continuous gradients with constant M_f , which yields

$$\|\nabla f^k(X(k)) - \nabla f^k(X^*(k))\|_F \leq M_f \|X(k) - X^*(k)\|_F. \quad (50)$$

Using (50), $\|E(k)\|_F \leq \varsigma$ and $\|E(k-1)\|_F \leq \varsigma$ such that we can bound (49) as

$$\begin{aligned} & \|M_-(\Theta(k) - \Theta^*(k))\|_F^2 \\ & \leq 2\left(\mu \|\nabla f^k(X(k)) - \nabla f^k(X^*(k))\|_F^2 \right. \\ & \quad \left. + \frac{\mu}{\mu-1} \|cM_+(Z(k) - Z(k-1))\|_F^2\right) \\ & \quad + \|cM_-M_-^T E(k)\|_F^2 + \|cM_+M_+^T E(k-1)\|_F^2 \\ & \leq 2\mu M_f^2 \|X(k) - X^*(k)\|_F^2 \\ & \quad + \frac{2\mu c^2 \sigma_{\max}^2(M_+)}{\mu-1} \|Z(k) - Z(k-1)\|_F^2 \\ & \quad + c^2 \varsigma^2 (\sigma_{\max}^4(M_+) + \sigma_{\max}^4(M_-)). \end{aligned} \quad (51)$$

By initializing the dual variable $\Lambda(0)$ in the column space of $M_-M_-^T$, we can guarantee the existence of $\Theta(0) \in \mathcal{R}^{2r \times p}$ in the column space of M_-^T , i.e., $\Lambda(0) = M_- \Theta(0)$. Then, $\Theta(k)$ lies in the column space of M_-^T for all the times k due to the dual update (18). Next, combining with the finite optimal dual variable $\Theta^*(k)$ lying in the column space of M_-^T according to [23], we lower bound the left-hand side of (51) as

$$\|M_-(\Theta(k) - \Theta^*(k))\|_F^2 \geq \tilde{\sigma}_{\min}^2(M_-) \|\Theta(k) - \Theta^*(k)\|_F^2 \quad (52)$$

where $\tilde{\sigma}_{\min}(M_-)$ is the lower bound of the nonzero singular values of M_- . Given the inequalities (51) and (52), we have

$$\begin{aligned} \|\Theta(k) - \Theta^*(k)\|_F^2 & \leq \frac{2\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)} \|X(k) - X^*(k)\|_F^2 \\ & \quad + \frac{2\mu c^2 \sigma_{\max}^2(M_+)}{(\mu-1)\tilde{\sigma}_{\min}^2(M_-)} \|Z(k) - Z(k-1)\|_F^2 \\ & \quad + \frac{c^2 \varsigma^2}{\tilde{\sigma}_{\min}^2(M_-)} (\sigma_{\max}^4(M_+) + \sigma_{\max}^4(M_-)). \end{aligned} \quad (53)$$

Substituting (47) and (53) into (46), we have

$$\begin{aligned} & \left(m_f - \frac{c\eta_1}{4} - \left(\frac{c\eta_2}{2} + c\delta\right) \frac{\sigma_{\max}^2(M_+)}{4} \right. \\ & \quad \left. - \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right) \frac{2\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)}\right) \|X(k) - X^*(k)\|_F^2 \\ & \quad + \left(c - \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right) \frac{2\mu c^2 \sigma_{\max}^2(M_+)}{(\mu-1)\tilde{\sigma}_{\min}^2(M_-)}\right) \|Z(k) - Z(k-1)\|_F^2 \\ & \quad + \frac{1}{c} \|\Theta(k) - \Theta(k-1)\|_F^2 \\ & \leq \|U(k-1) - U^*(k)\|_G^2 - (1+\delta) \|U(k) - U^*(k)\|_G^2 \\ & \quad + \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right) \frac{c^2 \varsigma^2}{\tilde{\sigma}_{\min}^2(M_-)} (\sigma_{\max}^4(M_+) + \sigma_{\max}^4(M_-)) + s\varsigma^2 \end{aligned} \quad (54)$$

in which we recall that η_1, η_2 , and η_3 are arbitrary positive constant, μ is an arbitrary constant larger than 1, and δ is an arbitrary constant.

Step 4: In this step, we complete the proof by discarding the left-hand side of (54). To this end, the coefficients of the left-hand side terms must be nonnegative. Fixing η_1, η_2, η_3 ,

and μ, δ must satisfy

$$\begin{aligned} \delta & \leq \min \left\{ \frac{(\mu-1)\tilde{\sigma}_{\min}^2(M_-)}{2\mu\sigma_{\max}^2(M_+)} - \frac{c\eta_3}{2} \right. \\ & \quad \times \left(\frac{c\sigma_{\max}^2(M_+)}{4} + \frac{2\mu M_f^2}{c\tilde{\sigma}_{\min}^2(M_-)} \right)^{-1} \\ & \quad \times \left(m_f - \frac{c\eta_1}{4} - \frac{c\eta_2\sigma_{\max}^2(M_+)}{8} - \frac{\eta_3\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)} \right) \Big\}. \end{aligned} \quad (55)$$

Also, $\delta > 0$ is required in the later analysis. To this end, the step size c should satisfy the condition

$$\begin{aligned} c & < \min \left\{ \frac{(\mu-1)\tilde{\sigma}_{\min}^2(M_-)}{\mu\eta_3\sigma_{\max}^2(M_+)}, \left(\frac{\eta_1}{4} + \frac{\eta_2\sigma_{\max}^2(M_+)}{8} \right)^{-1} \right. \\ & \quad \times \left(m_f - \frac{\eta_3\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)} \right) \Big\}. \end{aligned} \quad (56)$$

Thus, discarding the terms at the left-hand side of (54), it holds

$$(1+\delta) \|U(k) - U^*(k)\|_G^2 - \phi^2 \varsigma^2 \leq \|U(k-1) - U^*(k)\|_G^2 \quad (57)$$

where

$$\phi := \sqrt{\left(\frac{\eta_3}{2} + \frac{\delta}{c}\right) \frac{c^2}{\tilde{\sigma}_{\min}^2(M_-)} (\sigma_{\max}^4(M_+) + \sigma_{\max}^4(M_-)) + s}.$$

From (57), we have (25). This completes the proof. ■

APPENDIX B PROOF OF LEMMA 3

Proof: According to the triangle inequality $\|U(k-1) - U^*(k)\|_G - \|U(k-1) - U^*(k-1)\|_G \leq \|U^*(k) - U^*(k-1)\|_G$ and the definition of $\|U\|_G$, we have

$$\begin{aligned} & \|U(k-1) - U^*(k)\|_G - \|U(k-1) - U^*(k-1)\|_G \\ & \leq \sqrt{c\|Z^*(k) - Z^*(k-1)\|_F^2 + \frac{1}{c}\|\Theta^*(k) - \Theta^*(k-1)\|_F^2} \\ & \leq \sqrt{c}\|Z^*(k) - Z^*(k-1)\|_F + \frac{1}{\sqrt{c}}\|\Theta^*(k) - \Theta^*(k-1)\|_F. \end{aligned} \quad (58)$$

Recalling the KKT condition $Z^*(k) = (1/2)M_+^T X^*(k)$ given by (22), for any $k \geq 1$, we have

$$\begin{aligned} \|Z^*(k) - Z^*(k-1)\|_F & \leq \frac{\sigma_{\max}(M_+)}{2} \|X^*(k) - X^*(k-1)\|_F \\ & \leq \frac{\sigma_{\max}(M_+)\epsilon_1}{2}. \end{aligned} \quad (59)$$

Since we initialize that $\Lambda(0)$ is in the column space of $M_-M_-^T$, according to $\Lambda(k) = M_- \Theta(k)$ for any $k \geq 0$, we know that $\Theta(k)$ always stays in the column space of M_-^T . Therefore, for any $k \geq 1$, we have

$$\begin{aligned} & \|M_-(\Theta^*(k) - \Theta^*(k-1))\|_F^2 \\ & \geq (\tilde{\sigma}_{\min}(M_-))^2 \|\Theta^*(k) - \Theta^*(k-1)\|_F^2 \end{aligned}$$

which further leads to

$$\begin{aligned}
& \|\Theta^*(k) - \Theta^*(k-1)\|_F \\
& \leq \frac{1}{\tilde{\sigma}_{\min}(M_-)} \|M_- (\Theta^*(k) - \Theta^*(k-1))\|_F \\
& \stackrel{(c)}{\leq} \frac{1}{\tilde{\sigma}_{\min}(M_-)} \|\nabla f^k(X^*(k)) - \nabla f^{k-1}(X^*(k-1))\|_F \\
& \leq \frac{\epsilon_2}{\tilde{\sigma}_{\min}(M_-)}. \tag{60}
\end{aligned}$$

The inequality (c) is from the KKT condition $M_- \Theta^*(k) = -\nabla f^k(X^*(k))$ given by (20).

Combining (58)–(60) yields (30). This completes the proof. ■

APPENDIX C

PROOF OF THEOREM 1

Proof: Combining (25) and (30), we have

$$\begin{aligned}
& \sqrt{1+\delta} \|U(k) - U^*(k)\|_G \\
& \leq \|U(k-1) - U^*(k-1)\|_G + g + \phi\varsigma. \tag{61}
\end{aligned}$$

Expanding (61) from time 0 to time k yields

$$\begin{aligned}
& \|U(k) - U^*(k)\|_G \\
& \leq \sqrt{1+\delta}^{-1} (\|U(k-1) - U^*(k-1)\|_G + g + \phi\varsigma) \\
& \leq \sqrt{1+\delta}^{-k} \|U(0) - U^*(0)\|_G + \sum_{k'=0}^{k-1} \sqrt{1+\delta}^{k'-k} (g + \phi\varsigma) \\
& \leq \sqrt{1+\delta}^{-k} \|U(0) - U^*(0)\|_G + \frac{1 - \sqrt{1+\delta}^{-k}}{1 - \sqrt{1+\delta}^{-1}} (g + \phi\varsigma). \tag{62}
\end{aligned}$$

Taking limit superior for the two sides of (62) yields

$$\limsup_{k \rightarrow \infty} \|U(k) - U^*(k)\|_G \leq \frac{g + \phi\varsigma}{1 - \sqrt{1+\delta}^{-1}}. \tag{63}$$

Now, throwing away several terms in (44), we are able to obtain

$$\begin{aligned}
& \left(m_f - \frac{c\eta_1}{4}\right) \|X(k) - X^*(k)\|_F^2 \\
& \leq c \|Z(k-1) - Z^*(k)\|_F^2 + \frac{1}{c} \|\Theta(k-1) - \Theta^*(k)\|_F^2 \\
& \quad + \frac{c\eta_2}{2} \|Z(k) - Z^*(k)\|_F^2 + \frac{\eta_3}{2} \|\Theta(k) - \Theta^*(k)\|_F^2 + s\varsigma^2 \\
& \leq \|U(k-1) - U^*(k)\|_G^2 \\
& \quad + \max\left\{\frac{\eta_2}{2}, \frac{c\eta_3}{2}\right\} \|U(k) - U^*(k)\|_G^2 + s\varsigma^2. \tag{64}
\end{aligned}$$

Thus, we have

$$\begin{aligned}
& \sqrt{m_f - \frac{c\eta_1}{4}} \|X(k) - X^*(k)\|_F \\
& \leq \|U(k-1) - U^*(k)\|_G \\
& \quad + \max\left\{\sqrt{\frac{\eta_2}{2}}, \sqrt{\frac{c\eta_3}{2}}\right\} \|U(k) - U^*(k)\|_G + \sqrt{s\varsigma} \\
& \stackrel{(d)}{\leq} \|U(k-1) - U^*(k-1)\|_G + g \\
& \quad + \max\left\{\sqrt{\frac{\eta_2}{2}}, \sqrt{\frac{c\eta_3}{2}}\right\} \|U(k) - U^*(k)\|_G + \sqrt{s\varsigma} \tag{65}
\end{aligned}$$

where (d) uses $\|U(k-1) - U^*(k)\|_G \leq \|U(k-1) - U^*(k-1)\|_G + g$ in (30). Therefore, when

$$c < \frac{4m_f}{\eta_1}$$

using (63), we obtain (33). This completes the proof. ■

APPENDIX D

PROOF OF LEMMA 4

Proof: According to the definition of the difference, we have

$$\begin{aligned}
\|H(k)\|_F &= \|X(k) - \hat{X}(k-1)\|_F \\
&= \|X(k) - X(k-1) - E(k-1)\|_F \\
&\leq \|X(k) - X(k-1)\|_F + \|E(k)\|_F. \tag{66}
\end{aligned}$$

Therein, the second term $\|E(k)\|_F$ at the right-hand side of (66) is upper bounded by $\varsigma := \max\{\sqrt{n}a, \sqrt{np}\frac{\Delta}{2}\}$ according to Lemma 1.

For the first term $\|X(k) - X(k-1)\|_F$, it holds

$$\begin{aligned}
\|X(k) - X(k-1)\|_F &\leq \|X(k) - X^*(k)\|_F \\
&\quad + \|X(k-1) - X^*(k-1)\|_F \\
&\quad + \|X^*(k) - X^*(k-1)\|_F \tag{67}
\end{aligned}$$

where $\|X^*(k) - X^*(k-1)\|_F \leq \epsilon_1$ if Assumption 2 holds. From (65), we know

$$\begin{aligned}
& \|X(k) - X^*(k)\|_F \\
& \leq \left(\sqrt{m_f - \frac{c\eta_1}{4}}\right)^{-1} \\
& \quad \left(\|U(k-1) - U^*(k-1)\|_G \right. \\
& \quad \left. + \max\left\{\sqrt{\frac{\eta_2}{2}}, \sqrt{\frac{c\eta_3}{2}}\right\} \|U(k) - U^*(k)\|_G + g + \sqrt{s\varsigma}\right). \tag{68}
\end{aligned}$$

Recalling the upper bound of $\|U(k) - U^*(k)\|_G$ given by (62), we have

$$\begin{aligned}
& \|U(k) - U^*(k)\|_G \\
& \leq \sqrt{1+\delta}^{-k} \|U(0) - U^*(0)\|_G + \frac{1 - \sqrt{1+\delta}^{-k}}{1 - \sqrt{1+\delta}^{-1}} (g + \phi\varsigma) \\
& \leq \|U(0) - U^*(0)\|_G + \frac{1}{1 - \sqrt{1+\delta}^{-1}} (g + \phi\varsigma). \tag{69}
\end{aligned}$$

This upper bound also holds for $\|U(k-1) - U^*(k-1)\|_G$. Therefore, (68) becomes

$$\begin{aligned}
& \|X(k) - X^*(k)\|_F \\
& \leq \left(\sqrt{m_f - \frac{c\eta_1}{4}}\right)^{-1} \\
& \quad \times \left(\left(\max\left\{\sqrt{\frac{\eta_2}{2}}, \sqrt{\frac{c\eta_3}{2}}\right\} + 1 \right) \right. \\
& \quad \left. \times \left(\|U(0) - U^*(0)\|_G + \frac{g + \phi\varsigma}{1 - \sqrt{1+\delta}^{-1}} \right) + g + \sqrt{s\varsigma} \right). \tag{70}
\end{aligned}$$

This upper bound also holds for $\|X(k-1) - X^*(k-1)\|_F$.

Combining (66), (67), and (70), for any time $k > 0$, we have

$$\begin{aligned} \|H(k)\|_F &\leq 2\left(\sqrt{m_f - \frac{c\eta_1}{4}}\right)^{-1} \\ &\times \left(\left(\max\left\{ \sqrt{\frac{\eta_2}{2}}, \sqrt{\frac{c\eta_3}{2}} \right\} + 1 \right) \right. \\ &\times \left(\|U(0) - U^*(0)\|_G + \frac{g + \phi\varsigma}{1 - \sqrt{1 + \delta}^{-1}} \right) \\ &\left. + g + \sqrt{\delta}\varsigma \right) + \epsilon_1 + \varsigma \end{aligned} \quad (71)$$

which completes the proof. ■

REFERENCES

- [1] Y. Liu, K. Yuan, G. Wu, Z. Tian, and Q. Ling, "Decentralized dynamic ADMM with quantized and censored communications," in *Proc. Asilo-Mar*, 2019, pp. 1496–1500.
- [2] R. Graham and J. Cortes, "Adaptive information collection by robotic sensor networks for spatial estimation," *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1404–1419, Jun. 2012.
- [3] J.-H. Hours and C. N. Jones, "A parametric nonconvex decomposition algorithm for real-time and distributed NMPC," *IEEE Trans. Autom. Control*, vol. 61, no. 2, pp. 287–302, Feb. 2016.
- [4] H. J. Liu, W. Shi, and H. Zhu, "Decentralized dynamic optimization for power network voltage control," *IEEE Trans. Signal Inf. Process. Over Netw.*, vol. 3, no. 3, pp. 568–579, Sep. 2017.
- [5] Y. Zhang, E. Dall'Anese, and M. Hong, "Dynamic ADMM for real-time optimal power flow," in *Proc. GlobalSIP*, Nov. 2017, pp. 1085–1089.
- [6] C. Enyioha, S. Magnusson, K. Heal, N. Li, C. Fischione, and V. Tarokh, "On variability of renewable energy and online power allocation," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 451–462, Jan. 2018.
- [7] A. L. Dontchev, M. I. Krastanov, R. T. Rockafellar, and V. M. Veliov, "An euler-Newton continuation method for tracking solution trajectories of parametric variational inequalities," *SIAM J. Control Optim.*, vol. 51, no. 3, pp. 1823–1840, Jan. 2013.
- [8] A. Cherukuri and J. Cortés, "Initialization-free distributed coordination for economic dispatch under varying loads and generator commitment," *Automatica*, vol. 74, pp. 183–193, Dec. 2016.
- [9] X. Lian, C. Zhang, H. Zhang, C. Hsieh, W. Zhang, and J. Liu, "Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent," in *Proc. NIPS*, 2017, pp. 5330–5340.
- [10] H. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. AISTATS*, 2017, pp. 1273–1282.
- [11] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.
- [12] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," *SIAM J. Optim.*, vol. 26, no. 3, pp. 1835–1854, 2016.
- [13] A. Sayed, "Adaptation, learning, and optimization over networks," *Found. Trends Mach. Learn.*, vol. 7, pp. 311–801, Jul. 2014.
- [14] S.-Y. Tu and A. H. Sayed, "Distributed decision-making over adaptive networks," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1054–1069, Mar. 2014.
- [15] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Trans. Autom. Control*, vol. 57, no. 3, pp. 592–606, Mar. 2012.
- [16] K. Tsianos and M. Rabbat, "Distributed dual averaging for convex optimization under communication delays," in *Proc. ACC*, 2012, pp. 1067–1072.
- [17] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, "DQM: Decentralized quadratically approximated alternating direction method of multipliers," *IEEE Trans. Signal Process.*, vol. 64, no. 19, pp. 5158–5173, Oct. 2016.
- [18] A. Mokhtari, Q. Ling, and A. Ribeiro, "Network Newton distributed optimization methods," *IEEE Trans. Signal Process.*, vol. 65, no. 1, pp. 146–161, Jan. 2017.
- [19] I. Schizas, A. Ribeiro, and G. 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.
- [20] G. Mateos, J. A. Bazerque, and G. B. Giannakis, "Distributed sparse linear regression," *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5262–5276, Oct. 2010.
- [21] 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.
- [22] L. Majzoubi, F. Lahouti, and V. Shah-Mansouri, "Analysis of distributed ADMM algorithm for consensus optimization in presence of node error," *IEEE Trans. Signal Process.*, vol. 67, no. 7, pp. 1774–1784, Apr. 2019.
- [23] Q. Ling and A. Ribeiro, "Decentralized dynamic optimization through the alternating direction method of multipliers," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1185–1197, Mar. 2014.
- [24] M. Maros and J. Jalden, "ADMM for distributed dynamic beamforming," *IEEE Trans. Signal Inf. Process. Over Netw.*, vol. 4, no. 2, pp. 220–235, Mar. 2018.
- [25] R. L. G. Cavalcante and S. Stanczak, "A distributed subgradient method for dynamic convex optimization problems under noisy information exchange," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 2, pp. 243–256, Apr. 2013.
- [26] A. Simonetto, A. Mokhtari, A. Koppel, G. Leus, and A. Ribeiro, "A class of prediction-correction methods for time-varying convex optimization," *IEEE Trans. Signal Process.*, vol. 64, no. 17, pp. 4576–4591, Sep. 2016.
- [27] M. Fazlyab, S. Paternain, V. M. Preciado, and A. Ribeiro, "Prediction-correction interior-point method for time-varying convex optimization," *IEEE Trans. Autom. Control*, vol. 63, no. 7, pp. 1973–1986, Jul. 2018.
- [28] A. Simonetto, A. Koppel, A. Mokhtari, G. Leus, and A. Ribeiro, "Decentralized prediction-correction methods for networked time-varying convex optimization," *IEEE Trans. Autom. Control*, vol. 62, no. 11, pp. 5724–5738, Nov. 2017.
- [29] A. Simonetto and E. Dall'Anese, "Prediction-correction algorithms for time-varying constrained optimization," *IEEE Trans. Signal Process.*, vol. 65, no. 20, pp. 5481–5494, Oct. 2017.
- [30] S. Rahili and W. Ren, "Distributed continuous-time convex optimization with time-varying cost functions," *IEEE Trans. Autom. Control*, vol. 62, no. 4, pp. 1590–1605, Apr. 2017.
- [31] C. Sun, M. Ye, and G. Hu, "Distributed time-varying quadratic optimization for multiple agents under undirected graphs," *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3687–3694, Jul. 2017.
- [32] M. Fazlyab, C. Nowzari, G. J. Pappas, A. Ribeiro, and V. Preciado, "Self-triggered time-varying convex optimization," in *Proc. CDC*, 2016, pp. 3090–3097.
- [33] M. G. Rabbat and R. D. Nowak, "Quantized incremental algorithms for distributed optimization," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 798–808, Apr. 2005.
- [34] S. Magnússon, C. Enyioha, N. Li, C. Fischione, and V. Tarokh, "Convergence of limited communication gradient methods," *IEEE Trans. Autom. Control*, vol. 63, no. 5, pp. 1356–1371, May 2018.
- [35] Y. Pu, M. N. Zeilinger, and C. N. Jones, "Quantization design for distributed optimization," *IEEE Trans. Autom. Control*, vol. 62, no. 5, pp. 2107–2120, May 2017.
- [36] S. Zhu, M. Hong, and B. Chen, "Quantized consensus ADMM for multi-agent distributed optimization," in *Proc. ICASSP*, 2016, pp. 4134–4138.
- [37] P. Yi and Y. Hong, "Quantized subgradient algorithm and data-rate analysis for distributed optimization," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 4, pp. 380–392, Dec. 2014.
- [38] A. Reisizadeh, A. Mokhtari, H. Hassani, and R. Pedarsani, "Quantized decentralized consensus optimization," in *Proc. CDC*, 2018, pp. 5838–5843.
- [39] T. T. Doan, S. T. Maguluri, and J. Romberg, "Fast convergence rates of distributed subgradient methods with adaptive quantization," 2018, *arXiv:1810.13245*. [Online]. Available: <http://arxiv.org/abs/1810.13245>
- [40] H. Tang, S. Gan, C. Zhang, T. Zhang, and J. Liu, "Communication compression for decentralized training," 2018, *arXiv:1803.06443*. [Online]. Available: <http://arxiv.org/abs/1803.06443>
- [41] S. Magnússon, H. Shokri-Ghadikolaei, and N. Li, "On maintaining linear convergence of distributed learning and optimization under limited communication," 2019, *arXiv:1902.11163*. [Online]. Available: <http://arxiv.org/abs/1902.11163>
- [42] K. Tsianos, S. Lawlor, J. Yu, and M. Rabbat, "Networked optimization with adaptive communication," in *Proc. GlobalSIP*, 2013, pp. 579–582.
- [43] W. Chen and W. Ren, "Event-triggered zero-gradient-sum distributed consensus optimization over directed networks," *Automatica*, vol. 65, pp. 90–97, Mar. 2016.

- [44] Q. Lu and H. Li, "Event-triggered discrete-time distributed consensus optimization over time-varying graphs," *Complexity*, vol. 2017, May 2017, Art. no. 5385708.
- [45] Y. Liu, W. Xu, G. Wu, Z. Tian, and Q. Ling, "Communication-censored ADMM for decentralized consensus optimization," *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2565–2579, May 2019.
- [46] W. Li, Y. Liu, Z. Tian, and Q. Ling, "Communication-censored linearized ADMM for decentralized consensus optimization," *IEEE Trans. Signal Inf. Process. Over Netw.*, vol. 6, no. 1, pp. 18–34, Dec. 2020.
- [47] D. Dimarogonas, E. Frazzoli, and K. Johansson, "Distributed event-triggered control for multi-agent systems," *IEEE Trans. Autom. Control*, vol. 57, no. 5, pp. 1291–1297, May 2012.
- [48] E. Garcia, Y. Cao, H. Yu, P. Antsaklis, and D. Casbeer, "Decentralised event-triggered cooperative control with limited communication," *Int. J. Control*, vol. 86, no. 9, pp. 1479–1488, Sep. 2013.
- [49] C. Nowzari and J. Cortés, "Distributed event-triggered coordination for average consensus on weight-balanced digraphs," *Automatica*, vol. 68, pp. 237–244, Jun. 2016.
- [50] S. Liu, L. Xie, and D. E. Quevedo, "Event-triggered quantized communication-based distributed convex optimization," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 1, pp. 167–178, Mar. 2018.
- [51] B. Cheng and Z. Li, "Designing fully distributed adaptive event-triggered controllers for networked linear systems with matched uncertainties," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 12, pp. 3645–3655, Dec. 2019.
- [52] Z. Xu, C. Li, and Y. Han, "Impulsive consensus of nonlinear multi-agent systems via edge event-triggered control," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 6, pp. 1995–2004, Jun. 2020.
- [53] N. Singh, D. Data, J. George, and S. Diggavi, "SPARQ-SGD: Event-triggered and compressed communication in decentralized stochastic optimization," 2019, *arXiv:1910.14280*. [Online]. Available: <http://arxiv.org/abs/1910.14280>
- [54] F. Chung, *Spectral Graph Theory*. Providence, RI, USA: AMS, 1997.
- [55] B. Widrow and I. Kollar, *Quantization Noise*. Cambridge, U.K.: Cambridge Univ. Press, 2008.
- [56] H.-F. Xu, Q. Ling, and A. Ribeiro, "Online learning over a decentralized network through ADMM," *J. Oper. Res. Soc. China*, vol. 3, no. 4, pp. 537–562, Dec. 2015.
- [57] S. Lee, A. Nedic, and M. Raginsky, "Stochastic dual averaging for decentralized online optimization on time-varying communication graphs," *IEEE Trans. Autom. Control*, vol. 62, no. 12, pp. 6407–6414, Dec. 2017.



Yaohua Liu received the B.E. degree in automation from the South China University of Technology, Guangzhou, China, in 2015, and the Ph.D. degree in control theory and control engineering from the University of Science and Technology of China, Hefei, China, in 2020.

Since August 2020, she has been with the School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing, China. Her current research interest includes decentralized network optimization and its applications.



Gang Wu received the B.E. degree in automatic control and the M.S. degree in control theory and application from the University of Science and Technology of China, Hefei, China, in 1986 and 1989, respectively.

Since 1991, he has been with the Department of Automation, University of Science and Technology of China, where he is currently a Professor. His research interest focuses on electric vehicle control systems.



Zhi Tian (Fellow, IEEE) received the B.E. degree in electrical engineering from the University of Science and Technology of China, Hefei, China, in 1994, and the M.S. and Ph.D. degrees from George Mason University, Fairfax, VA, USA, in 1998 and 2000, respectively.

From 2000 to 2014, she was on the faculty of Michigan Technological University, Houghton, MI, USA. From 2011 to 2014, she was the Program Director of the US National Science Foundation. Since 2015, she has been a Professor with the

Electrical and Computer Engineering Department, George Mason University. Her general interests lie in the areas of signal processing, wireless communications, and estimation and detection theory. Her current research focuses on distributed learning and inference, as well as massive multiple-input-multiple-output (MIMO) systems for 5G wireless.

Dr. Tian received the CAREER Award from the U.S. National Science Foundation in 2003. She served as an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the IEEE TRANSACTIONS ON SIGNAL PROCESSING. She is a Distinguished Lecturer of the IEEE Vehicular Technology Society and the IEEE Communications Society.



Qing Ling (Senior Member, IEEE) received the B.E. degree in automation and the Ph.D. degree in control theory and control engineering from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively.

He was a Post-Doctoral Research Fellow with the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI, USA, from 2006 to 2009, and an Associate Professor with the Department of Automation, University of Science and Technology of China, from 2009 to

2017. He is currently a Professor with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China. His current research interest includes decentralized network optimization and its applications.

Dr. Ling received the 2017 IEEE Signal Processing Society Young Author Best Paper Award as a Supervisor. He is also an Associate Editor of the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT and a Senior Area Editor of the IEEE SIGNAL PROCESSING LETTERS.