Communication-Censored ADMM for Decentralized Consensus Optimization

Yaohua Liu, Wei Xu, Gang Wu, Zhi Tian, and Qing Ling

***Abstract*—In this paper, we devise a communication-efficient decentralized algorithm, named as communication-censored alternating direction method of multipliers (ADMM) (COCA), to solve a convex consensus optimization problem defined over a network. Similar to popular decentralized consensus optimization algorithms such as ADMM, at every iteration of COCA, a node exchanges its local variable with neighbors, and then updates its local variable according to the received neighboring variables and its local cost function. A different feature of COCA is that a node is not allowed to transmit its local variable to neighbors, if this variable is not sufficiently different to the previously transmitted one. The sufficiency of the difference is evaluated by a properly designed censoring function. Though this censoring strategy may slow down the optimization process, it effectively reduces the communication cost. We prove that when the censoring function is properly chosen, COCA converges to an optimal solution of the convex consensus optimization problem. Furthermore, if the local cost functions are strongly convex, COCA has a fast linear convergence rate. Numerical experiments demonstrate that, given a target solution accuracy, COCA is able to significantly reduce the overall communication cost compared to existing algorithms including ADMM, and hence fits for applications where network communication is a bottleneck.**

***Index Terms*—Decentralized network, consensus optimization, communication-censoring strategy, alternating direction method of multipliers (ADMM).**

I. Introduction

THIS paper considers solving a convex consensus optimization problem defined over a bidirectionally connected decentralized network consisting of n nodes, in the form of

$$\tilde{x}^* \in \arg\min_{\tilde{x}} \sum_{i=1}^{n} f_i(\tilde{x}). \tag{1}$$

Here each node i holds a local convex cost function $f_i : \mathcal{R}^p \to \mathcal{R}$ that is kept private, and all the nodes share a common optimization variable $\tilde{x} \in \mathcal{R}^p$. Our aim is to devise a communication-efficient decentralized algorithm such that the nodes can collaboratively find an optimal solution $\tilde{x}^*$ through local computation and limited information exchange among neighbors.

The consensus optimization problem in the form of (1) appears in various applications, such as wireless sensor networks [1], [2], communication networks [4], [5], multi-robot networks [6], [7], smart grids [8], [9], machine learning systems [10], [11], to name a few. Various decentralized algorithms have been proposed to solve this problem in recent years; see the survey paper [12] and an incomplete overview in Section I-A. An ideal decentralized algorithm is expected to reach an optimal solution with minimal communication and computation costs. Nevertheless, the communication-computation tradeoff is essential [12]–[15]. In this paper, we focus on the scenario that computation is relatively cheap, and communication is the major concern. Among existing decentralized algorithms, the alternating direction method of multipliers (ADMM) is especially suitable for this scenario [2], [16], [17]. In this paper, we shall show by numerical experiments that, augmented with a simple communication-censoring strategy which restricts nodes from transmitting "less informative" messages to neighbors, the communication efficiency of ADMM can be significantly improved. The resultant algorithm, termed as communication-censored ADMM (COCA), is able to reach a target solution accuracy with slightly more computation but much less communication compared to the classical ADMM. Rigorous analysis is provided to guide the design of the censoring strategy to guarantee the convergence of COCA.

A. Related Work

A large number of decentralized algorithms have been designed to solve the consensus optimization problem in the form of (1), spurred by their robustness, scalability and potential of privacy preservation in network applications. At every iteration of a typical synchronous decentralized algorithm, there are a communication step and a computation step: a node exchanges its local variable with neighbors and then computes an updated local variable according to the received neighboring variables and its local cost function. According to the complexity of the computation step, the existing decentralized algorithms can be classified as: (i) zeroth-order algorithms where a node is only able to evaluate its local cost function [18], [19]; (ii) first-order

Manuscript received April 1, 2018; revised September 30, 2018 and March 12, 2019; accepted March 13, 2019. Date of publication March 25, 2019; date of current version April 12, 2019. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Gonzalo Mateos. The work of Z. Tian was supported by the National Science Foundation (NSF) under Grants CCF-1527396 and IIS-1741338. The work of Q. Ling was supported by China NSF under Grant 61573331 and Anhui NSF under Grant 1608085QF130. This paper was presented in part at the Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, October, 2018. *(Corresponding author: Qing Ling.)*

Y. Liu, W. Xu, and G. Wu are with the Department of Automation, University of Science and Technology of China, Hefei 230027, China (e-mail: vcifer@mail.ustc.edu.cn; shuaiwei@mail.ustc.edu.cn; wug@ustc.edu.cn).

Z. Tian is with the Department of Electrical and Computer Engineering, George Mason University, Fairfax, VA 22030 USA (e-mail: ztian1@gmu.edu).

Q. Ling is with the School of Data and Computer Science, Guangdong Province Key Laboratory of Computational Science, Sun Yat-Sen University, Guangdong 510006, China (e-mail: lingqing556@mail.sysu.edu.cn).

Digital Object Identifier 10.1109/TSP.2019.2907258

algorithms where a node can utilize its local gradient during the optimization process, such as gradient descent method [20], diffusion method [21], exact first-order algorithm [22], and linearized ADMM [23], [24]; (iii) second-order algorithms where a node can compute or approximately compute its local Hessian, such as network Newton method [25], quasi-Newton method [26], exact second-order method [27], and quadratically approximated ADMM [28]; (iv) "higher-order" algorithms where at every iteration a node needs to solve an optimization problem whose complexity is determined by the local cost function, such as dual decomposition method [1] and ADMM [2], [3]. Note that this categorization is not strict; for example, applying the successive convex approximation technique yields a class of algorithms ranging from the first order to a higher order [29]. This brief survey is also far from complete; for example, it does not include asynchronous algorithms that are important to heterogeneous networks [30]. For a more comprehensive recent survey on decentralized algorithms, readers are referred to [12].

At the expense of higher computation cost per iteration, higher-order algorithms often enjoy faster convergence, which leads to saving in the communication cost. Particularly, ADMM has shown fast convergence in both practice and theory [16], and is hence especially suitable for applications where computation is affordable but communication is expensive. A natural question arises: *Is it possible to further improve the communication efficiency of ADMM, without causing too much computation overhead?* Our answer is *Yes*. The key idea is to embed a simple yet powerful communication-censoring strategy to ADMM. A node is not allowed to transmit its local variable to neighbors, if this variable is not sufficiently different from the previously transmitted one. The sufficiency of the difference is evaluated by a censoring function. Through properly choosing the censoring function, the resultant algorithm, termed as communication-censored ADMM (COCA), is able to converge to an optimal solution of (1). The convergence rate of COCA is almost as fast as that of ADMM such that the increment of computation cost is minimal. Meanwhile, communication is significantly reduced by avoiding transmissions of less informative messages.

To reduce the communication cost of a decentralized algorithm, one approach is to quantize messages so as to transmit less bits. The quantized ADMM is developed following this idea [31]. An extreme is only to transmit one bit at every time, such as the one-bit gradient descent method [32]. However, these algorithms cannot guarantee exact convergence to the consensual optimal solution, and the consensus error is caused by the quantization error [31], [32]. Another approach is to avoid transmissions of "less informative" messages. Given an underlying communication graph, the weighted ADMM deletes some of the links prior to the optimization process so as to reduce the number of message transmissions at every iteration [33]. Our work is also along this line. In contrast, COCA adaptively determines whether a message is informative during the optimization process, different from the weighted ADMM that determines whether a node is informative in advance. Other efforts to reduce the communication cost include the random-walk ADMM that randomly activates a succession of nodes and incrementally updates the optimization variable [34], and the block-iterative method that updates and communicates only a block of a high-dimensional optimization variable at every iteration [35]. In comparison, COCA is a deterministic algorithm, in which every time every node updates the entire local variable.

Related to the communication-censoring strategy in COCA, data-adaptive computation-censoring is a powerful tool to reduce the computation cost of big data processing over networks [36]. When computation, other than communication, is the bottleneck of the network, a node can skip a complicated update when the innovation from the data is not sufficient. COCA is at the other side of the coin, as it concerns more on communication than computation. The concept of communication-censoring is also related to event-triggered control, which is used to reduce the number of actuator updates [37] or message transmissions [38], [39] over *continuous-time* networks. The work in [40], [41] and [42] combines the idea of event-triggered control with *discrete-time* decentralized consensus optimization. However, the event-triggered dual averaging algorithm in [40] and the sub-gradient algorithm in [41] require diminishing step sizes to guarantee exact convergence to an optimal solution, which leads to relatively slow convergence rates. On the other hand, we shall show that the event-triggered zero-gradient-sum algorithm in [42] is indeed a communication-censored version of the dual decomposition method in Section II-D. Because ADMM is much faster than the dual decomposition method, COCA is also much more communication-efficient and computation-efficient than the event-triggered zero-gradient-sum algorithm, as validated by the numerical experiments in Section IV.

B. Our Contributions and Paper Organization

Section II introduces COCA, a novel communication-censored ADMM, to improve the communication efficiency of the classical ADMM, while incurring minimal computation overhead. The key ingredient of COCA is a communication-censoring strategy, which prohibits a node from transmitting its local variable to neighbors, if this variable is not sufficiently different from the previously transmitted one. The sufficiency of the difference is evaluated by a censoring function (Section II-B). When the communication-censoring strategy is absent, COCA degenerates to the classical ADMM (Section II-C). We also show that the state-of-the-art event-triggered zero-gradient-sum algorithm is indeed a communication-censored version of the dual decomposition method, analogy to the connection between COCA and ADMM (Section II-D). In Section III, We prove that when the censoring function is properly chosen, COCA converges to an optimal solution of the convex consensus optimization problem (Theorem 1). Further, if the local cost functions are strongly convex, COCA has a fast linear convergence rate (Theorem 2). The analysis provides guidelines for tuning the parameters of COCA, including the step size and the censoring function. It also characterizes how the convergence rate of COCA is affected by the properties of the cost functions and the communication graph. Section IV presents numerical experiments to demonstrate the communication efficiency of COCA. Section V concludes the paper.

II. Algorithm Development

In this section, we devise COCA, the communication-censored ADMM that improves the communication efficiency of the classical ADMM in decentralized consensus optimization. We also compare COCA with the classical ADMM and the event-triggered zero-gradient-sum algorithm to demonstrate their connections.

A. Network and Communication Models

Network Model: Throughout the paper, we consider a bidirectionally connected network consisting of n nodes and r edges ($2r$ directed arcs). The underlying undirected communication graph is denoted as $\mathcal{G} = \{\mathcal{V}, \mathcal{A}\}$, where $\mathcal{V}$ is the set of nodes with cardinality $|\mathcal{V}| = n$ and $\mathcal{A}$ is the set of directed arcs with cardinality $|\mathcal{A}| = 2r$. Two nodes i and j are called as neighbors if the arc $(i, j) \in \mathcal{A}$ and, by the symmetry of the network, $(j, i) \in \mathcal{A}$. The set of node i's neighbors is denoted as $\mathcal{N}_i$ with cardinality $d_{ii} = |\mathcal{N}_i|$.

Communication Model: Like the classical ADMM, COCA is synchronous. Every iteration consists of two stages: the communication stage where every node exchanges its local variable with neighbors and the computation stage where every node updates its local variable according to the received neighboring variables and its local cost function. In the communication stage, transmissions of messages are in a broadcast mode. When node i is allowed to communicate, it broadcasts the local variable, which is a p-dimensional vector, to all the neighbors, and the resultant communication cost is 1. The transmissions of COCA could also be implemented easily in the unicast mode. When node i is allowed to communicate, it sends the local variable to all the neighbors one by one. Therefore, the resultant communication cost is $|\mathcal{N}_i|$, the number of node i's neighbors. Unlike the classical ADMM, in COCA, a node does not necessarily broadcast at every iteration. It only transmits its local variable to neighbors if this variable is sufficiently different to the previously transmitted one. The sufficiency of the difference is evaluated by a censoring function, which is designed below.

B. COCA: Communication-Censored ADMM

At time k of COCA, every node i keeps $3 + d_{ii}$ local variables, where d_{ii} is its degree. The first is a primal variable $x_i^k \in \mathcal{R}^p$, a copy of the optimization variable $\tilde{x}$. The second is a dual variable $\lambda_i^k \in \mathcal{R}^p$. Node i also keeps a state variable $\hat{x}_i^k$ that records its latest broadcast primal variable up to time k. Likewise, for every neighbor j, node i keeps a state variable $\hat{x}_j^k$ that records its latest received primal variable from j up to time k. The storage requirement is the same as that of ADMM. Similar to ADMM, COCA only needs to transmit the primal variables x_i^k. The dual variables λ_i^k and the state variables $\hat{x}_i^k$ are kept local. Note that node i and its neighbors $j \in \mathcal{N}_i$ maintain an identical state variable $\hat{x}_i^k$.

A key feature of COCA is that a node i is not allowed to transmit its local variable x_i^k to neighbors, if x_i^k is not sufficiently different from the previously transmitted $\hat{x}_i^{k-1}$, namely, its latest state variable. Define the difference as

$$\xi_i^k = \hat{x}_i^{k-1} - x_i^k. \tag{2}$$

Algorithm 1: COCA Run by Node i.

```
Require: Initialize local variables to $x_i^0 = 0$, $\lambda_i^0 = 0$,
    $\hat{x}_i^0 = 0$, and $\hat{x}_j^0 = 0$ for all $j \in \mathcal{N}_i$.
for iterations $k = 1, 2, \ldots$ do
    Compute local primal variable $x_i^k$ by
        $x_i^k = \arg\min_{x_i} f_i(x_i) + \left\langle x_i, \lambda_i^{k-1} - c \sum_{j \in \mathcal{N}_i} (\hat{x}_i^{k-1} + \hat{x}_j^{k-1}) \right\rangle + c d_{ii} \|x_i\|^2.$
    Compute $\xi_i^k = \hat{x}_i^{k-1} - x_i^k$.
    If $H_i(k, \xi_i^k) \geq 0$, transmit $x_i^k$ to neighbors and let
        $\hat{x}_i^k = x_i^k$; else do not transmit and let $\hat{x}_i^k = \hat{x}_i^{k-1}$.
    If receive $x_j^k$ from any neighbor $j$, let $\hat{x}_j^k = x_j^k$; else
        let $\hat{x}_j^k = \hat{x}_j^{k-1}$.
    Update local dual variable $\lambda_i^k$ as
        $\lambda_i^k = \lambda_i^{k-1} + c \sum_{j \in \mathcal{N}_i} (\hat{x}_i^k - \hat{x}_j^k).$
end for
```

The sufficiency of the difference is evaluated by a censoring function $H_i(k, \xi_i^k) = \|\xi_i^k\| - \tau^k$, where $\{\tau^k\}$ is a non-increasing non-negative sequence. A typical choice for the censoring function is

$$H_i(k, \xi_i^k) = \|\xi_i^k\| - \alpha \rho^k, \tag{3}$$

where $\rho \in (0, 1)$ and $\alpha > 0$ are constants. Node i is allowed to transmit its primal variable x_i^k to neighbors, if and only if

$$H_i(k, \xi_i^k) \geq 0.$$

COCA run by node i is outlined in Algorithm 1, which is devised by incorporating the censoring strategy into the classical ADMM and elaborated in Section II-C. At time 0, node i initializes its local variables to $x_i^0 = 0$, $\lambda_i^0 = 0$, $\hat{x}_i^0 = 0$, and $\hat{x}_j^0 = 0$ for all $j \in \mathcal{N}_i$. For all subsequent times k, node i first computes its local primal variable x_i^k by solving

$$x_i^k = \arg\min_{x_i} f_i(x_i) + \left\langle x_i, \lambda_i^{k-1} - c \sum_{j \in \mathcal{N}_i} (\hat{x}_i^{k-1} + \hat{x}_j^{k-1}) \right\rangle + c d_{ii} \|x_i\|^2, \tag{4}$$

where $c > 0$ is the step size of COCA. To solve (4), node i needs its local dual variable λ_i^{k-1}, the state variable $\hat{x}_i^{k-1}$ of itself and the state variables $\hat{x}_j^{k-1}$ of its neighbors which are already known, as well as the local cost function f_i. Then, node i calculates ξ_i^k, the difference between its current local primal variable x_i^k and the previously transmitted one $\hat{x}_i^{k-1}$ by (2), followed by evaluating $H_i(k, \xi_i^k)$. If $H_i(k, \xi_i^k) \geq 0$, then node i transmits x_i^k to neighbors and lets $\hat{x}_i^k = x_i^k$; otherwise, node i does not transmit and lets $\hat{x}_i^k = \hat{x}_i^{k-1}$. If node i receives x_j^k from any neighbor j, then lets $\hat{x}_j^k = x_j^k$; else, lets $\hat{x}_j^k = \hat{x}_j^{k-1}$. This way, the state variable of any node is identical to all the neighbors. Finally, the local dual variable λ_i^k is updated by

$$\lambda_i^k = \lambda_i^{k-1} + c \sum_{j\in\mathcal{N}_i} (\hat{x}_i^k - \hat{x}_j^k), \tag{5}$$

where c is the same positive step size as that in (4).

Next, we show that COCA is essentially a communication-censored variant of the classical ADMM.

C. Connection With the Classical ADMM

At time k, the primal and dual updates of node i of the classical ADMM [2], [16] are

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

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

The updates of COCA in (4) and (5) use the state variables $\hat{x}_i^{k-1}$ and $\hat{x}_i^k$, while those of the classical ADMM in (6) and (7) use the primal variables x_i^{k-1} and x_i^k. If we set $H_i(k, \xi_i^k) = 0$ (namely, no communication censoring), then COCA degenerates to the classical ADMM. However, it is the communication-censoring strategy that makes COCA more communication-efficient than the classical ADMM.

Intuitively, if the difference between a current local primal variable x_i^k and a previously transmitted one $\hat{x}_i^{k-1}$ is small, then using either one does not make much difference to the optimization process. Therefore, it is not necessary to transmit x_i^k to node i's neighbors and the communication cost is reduced. Nevertheless, the significance of the difference must be carefully evaluated; otherwise, the accumulated error may eventually bias the optimization process. For example, if we set the censoring function to $H_i(k, \xi_i^k) = \|\xi_i^k\| - \alpha\rho^k$ as in (3), then the significance of the difference is evaluated by a geometrically decaying threshold. Note that it does not mean more frequent communications when k is large, since the local primal variables might have been very close to an optimal solution at time k such that $\|\xi_i^k\|$ is also small. Choosing larger α and ρ leads to less communications per iteration. On the other hand, with $\alpha = 0$ or $\rho = 0$, COCA is the same as the classical ADMM.

D. Connection With the Event-Triggered Zero-Gradient-Sum Algorithm

The event-triggered zero-gradient-sum algorithm proposed in [42] combines the idea of event-triggered control, which is tightly related to our communication-censoring strategy, with discrete-time decentralized consensus optimization. At time k, node i runs

$$\begin{cases} x_i^k = \arg\min_{x_i} f_i(x_i) - \Big\langle x_i, c\sum_{j\in\mathcal{N}_i} w_{ij}(\hat{x}_j^{k-1} - \hat{x}_i^{k-1}) \\ \qquad + \nabla f_i(x_i^{k-1})\Big\rangle, \qquad k = 1, 2, \ldots, \\ x_i^0 = x_i^*, \qquad k = 0, \end{cases}$$

where $c > 0$ is the step size, $x_i^* := \arg\min_{\tilde{x}} f_i(\tilde{x})$ is the minimizer of the local cost function, and w_{ij} is the (i, j)-th entry of the adjacency matrix $W \in \mathcal{R}^{n\times n}$ of the communication graph such that $w_{ij} = 1$ if i and j are neighbors or $w_{ij} = 0$ otherwise. The definitions of the local state variables $\hat{x}_i^k$ are the same as those in COCA.

Below we show that the event-triggered zero-gradient-sum algorithm is indeed a communication-censored version of the dual decomposition method, analogous to that COCA is a communication-censored version of ADMM. Because ADMM is much faster than dual decomposition, COCA is also much more efficient than the event-triggered zero-gradient-sum algorithm in terms of both communication and computation. We begin by showing how the uncensored algorithms, ADMM and dual decomposition, solve (1). For clarity, we present the algorithms in matrix forms. Collect all variables x_i, λ_i and $\hat{x}_i$ in matrices

$$X \triangleq \begin{pmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{pmatrix}, \Lambda \triangleq \begin{pmatrix} \lambda_1^T \\ \lambda_2^T \\ \vdots \\ \lambda_n^T \end{pmatrix}, \hat{X} \triangleq \begin{pmatrix} \hat{x}_1^T \\ \hat{x}_2^T \\ \vdots \\ \hat{x}_n^T \end{pmatrix} \in \mathcal{R}^{n\times p}.$$

Define an aggregate cost function $f(X) := \sum_{i=1}^n f_i(x_i)$. The diagonal degree matrix of the communication graph is $D \in \mathcal{R}^{n\times n}$, whose i-th diagonal element is d_{ii}, the degree of node i. Also define the unsigned incidence matrix $M_+ \in \mathcal{R}^{n\times 2r}$ and the signed incidence matrix $M_- \in \mathcal{R}^{n\times 2r}$. If an arc l goes from i to j, then the (i, l)-th and (j, l)-th entries of M_+ are both 1, while the (i, l)-th entry of M_- is 1 and the (j, l)-th entry of M_- is -1. Recalling the definition of the adjacency matrix W, from [43] we have that

$$D + W = \frac{1}{2}M_+M_+^T, D - W = \frac{1}{2}M_-M_-^T.$$

When the underlying communication graph is connected, the unconstrained consensus optimization problem (1) is equivalent to the following constrained form

$$\min_{X,Z} \ f(X) := \sum_{i=1}^n f_i(x_i),$$
$$\text{s.t.} \ \frac{1}{2}\begin{pmatrix} M_+^T + M_-^T \\ M_+^T - M_-^T \end{pmatrix} X = \begin{pmatrix} I_{2r} \\ I_{2r} \end{pmatrix} Z, \tag{8}$$

where $I_{2r} \in \mathcal{R}^{2r\times 2r}$ is an identity matrix and $Z \in \mathcal{R}^{2r\times p}$ is an auxiliary variable. The consensus constraint in (8) enforces all local variables x_i to be equal [16]. The classical ADMM minimizes the augmented Lagrangian of (8) with respect to X and Z in an alternating direction manner, followed by updating the dual variable Λ associated with the consensus constraint [16]. Eventually the auxiliary variable Z is eliminated, yielding

$$X^k = \arg\min_X f(X) + \langle X, \Lambda^{k-1} - c(D+W)X^{k-1}\rangle + \langle X, cDX\rangle,$$
$$\Lambda^k = \Lambda^{k-1} + c(D-W)X^k.$$

which exactly matches the node-wise updates (6) and (7). Correspondingly, the matrix form of COCA is

$$X^k = \arg\min_X f(X) + \langle X, \Lambda^{k-1} - c(D+W)\hat{X}^{k-1}\rangle + \langle X, cDX\rangle, \quad (9)$$

$$\Lambda^k = \Lambda^{k-1} + c(D-W)\hat{X}^k. \quad (10)$$

The dual decomposition method operates on a different equivalent reformulation of (1), in the form of

$$\min_X \; f(X) := \sum_{i=1}^n f_i(x_i), \quad \text{s.t. } M_-^T X = 0, \quad (11)$$

where the consensus constraint also enforces all local variables x_i to be equal [1]. Given the Lagrangian of (11) defined by $L(X,\beta) := f(X) + \langle \beta, M_-^T X\rangle$ where $\beta \in \mathcal{R}^{2r\times p}$ is the dual variable, at time k, the dual decomposition method updates

$$X^k = \arg\min_X L(X, \beta^{k-1}), \quad (12)$$

$$\beta^k = \beta^{k-1} + \frac{c}{2} M_-^T X^k. \quad (13)$$

Write the optimality condition of (12) at two consecutive times $k-1$ and k as

$$\nabla f(X^k) + M_-\beta^{k-1} = 0, \quad (14)$$

$$\nabla f(X^{k-1}) + M_-\beta^{k-2} = 0. \quad (15)$$

Here we assume f to be smooth. When f is non-smooth, all the subsequent derivations still hold true by replacing ∇f with one of its subgradients. Subtracting (14) by (15) and plugging in (13), we have

$$\nabla f(X^k) = \nabla f(X^{k-1}) - \frac{c}{2} M_- M_-^T X^{k-1} = \nabla f(X^{k-1}) + c(W-D)X^{k-1}.$$

Observing the event-triggered zero-gradient-sum algorithm in (8), we can find that its matrix form is

$$\nabla f(X^k) = \nabla f(X^{k-1}) + c(W-D)\hat{X}^{k-1},$$

which is a communication-censored version of the dual decomposition method. Initializing $x_i^0 = x_i^*$ guarantees that (15) is valid for $k=1$, because under this initialization $\nabla f(X^0) = 0$ and lies in the column space of M_-.

Therefore, we conclude that COCA and the event-triggered zero-gradient-sum algorithm operate on two different equivalent reformulations of (1) (namely, (8) and (11), respectively), and are communication-censored versions of two different algorithms (namely, ADMM and dual decomposition, respectively). Empirically, dual decomposition is much slower than ADMM, and is sensitive to the choice of the step size c. These benefits of ADMM are inherited by COCA, which is also much more communication-efficient and computation-efficient than the event-triggered zero-gradient-sum algorithm, as we shall demonstrate in the numerical experiments in Section IV.

III. Convergence and Linear Rate of Convergence

In this section, we prove that when the censoring function is properly chosen, COCA converges to an optimal solution of the convex consensus optimization problem (1). Further, if the local cost functions are strongly convex, COCA converges at a linear rate.

Before stating the main results, we make several assumptions. Assumptions 1 and 2 are basic ones.

Assumption 1 (Network connectivity): The communication graph $\mathcal{G} = \{\mathcal{V}, \mathcal{A}\}$ is bidirectionally connected.

Assumption 2 (Solution existence): There exists an optimal solution set to (1), denoted by $\mathcal{X}^*$, which has at least one finite element.

With Assumption 3, we can prove convergence of COCA to an optimal solution of (1).

Assumption 3 (Convexity): The local cost functions f_i are convex.

With Assumption 4, COCA converges to the optimal solution of (1) at a linear rate.

Assumption 4 (Strong convexity and Lipschitz continuous gradients): The local cost functions f_i are strongly convex with constants $m_{f_i} > 0$. Given any $\tilde{x}, \tilde{y} \in \mathcal{R}^p$, $\langle \nabla f_i(\tilde{x}) - \nabla f_i(\tilde{y}), \tilde{x} - \tilde{y}\rangle \geq m_{f_i}\|\tilde{x} - \tilde{y}\|_2^2$ for any i. The minimum strong convexity constant is $m_f := \min_i m_{f_i}$. The gradients of the local cost functions are Lipschitz continuous with constants $M_{f_i} > 0$. Given any $\tilde{x}, \tilde{y} \in \mathcal{R}^p$, $\|\nabla f_i(\tilde{x}) - \nabla f_i(\tilde{y})\|_2 \leq M_{f_i}\|\tilde{x} - \tilde{y}\|_2$ for any i. The maximum Lipschitz constant is $M_f := \max_i M_{f_i}$.

Theorem 1: Initialize the dual variable Λ^0 in the column space of M_-, choose any positive step size $c > 0$, and set $\{\tau^k\}$ as a non-increasing non-negative summable sequence such that $\sum_{k=0}^\infty \tau^k < \infty$. Then under Assumptions 1–3, COCA converges to an optimal solution of (1).

Proof: See Appendix A. ■

Theorem 1 asserts that COCA converges to an optimal solution of (1) under mild conditions. The step size c is an arbitrary positive constant. Initialization of the dual variable Λ^0 in the column space of M_- is necessary for the convergence, and can be easily reached by setting $\Lambda^0 = 0$. For the communication-censoring strategy, it is sufficient to guarantee convergence as long as $\{\tau^k\}$ is a non-increasing non-negative summable sequence – for example, $\tau^k = 1/k^2$. Recall that node i is allowed to transmit if and only if $\|\xi_i^k\| \geq \tau^k$ and $\|\xi_i^k\|$ denotes the difference between the current local variable x_i^k and the previously transmitted one $\hat{x}_i^{k-1}$. Thus, we expect that $\|\xi_i^k\|$ decays as fast as τ^k. If not, the communication-censoring strategy shall enforce transmitting the local variable so as to reduce the value of $\|\xi_i^k\|$.

Theorem 2: Initialize the dual variable Λ^0 in the column space of M_-, set $\tau^k = \alpha\rho^k$ with $\alpha > 0$ and $\rho \in (0,1)$, and choose positive step size c such that

$$0 < c < \min\left\{\frac{4m_f}{\eta_1}, \frac{(\mu-1)\tilde{\sigma}_{\min}^2(M_-)}{\mu\eta_3\sigma_{\max}^2(M_+)}, \left(\frac{\eta_1}{4} + \frac{\eta_2\sigma_{\max}^2(M_+)}{8}\right)^{-1}\left(m_f - \frac{\eta_3\mu M_f^2}{2\tilde{\sigma}_{\min}^2(M_-)}\right)\right\}, \quad (16)$$

where $\eta_1 > 0, \eta_2 > 0, \eta_3 > 0$ and $\mu > 1$ are arbitrary constants, m_f is the minimum strong convexity constant of the local cost functions, M_f is the maximum Lipschitz constant of the local gradients, $\sigma_{\max}(M_+)$ is the maximum singular value of the unsigned incidence matrix M_+, and $\tilde{\sigma}_{\min}(M_-)$ is the minimum non-zero singular value of the signed incidence matrix M_-. Then under Assumptions 1–4, COCA converges to the optimal solution of (1) at a linear rate.

Proof: See Appendix B. ■

According to Theorem 2, to show linear convergence of COCA to the optimal solution of (1), we need all the local cost functions to be strongly convex and have Lipschitz continuous gradients, which is common in convex analysis. The condition that Λ^0 stays in the column space of M_- can be satisfied by setting $\Lambda^0 = 0$ as in Theorem 1. For linear convergence, the step size c can be arbitrarily large by properly setting the constants η_1, η_2, η_3 and μ. However, the step size c shall influence the constant of linear convergence rate, as shown in the proof.

Not surprisingly, to guarantee linear convergence, we expect that $\|\xi_i^k\|$ decays as fast as $\tau^k = \alpha\rho^k$ so that the "state error" decays at a linear rate. When the value of $\|\xi_i^k\|$ is larger than the linearly decaying threshold, the communication-censoring strategy allows transmitting the local variable. The resultant constant of linear convergence rate, which is not explicitly shown in Theorem 2 but appears in the proof, is dependent on but not faster than ρ. Besides, the rate is also determined by the step size c, the properties of the cost functions (parameterized by m_f and M_f), as well as the properties of the communication graph (parameterized by $\tilde{\sigma}_{\min}(M_-)$ and $\sigma_{\max}(M_+)$).

Remark 1: When the communication-censoring strategy is absent, COCA degenerates to the classical ADMM, thus its convergence and rate of convergence are the same to those of the classical ADMM. However, the communication-censoring strategy and the resultant error caused by "outdated" information make the theoretical analysis of COCA substantially more challenging than that of the classical ADMM, as we have pointed out in the proof.

Remark 2: Note that the theoretical convergence rate of COCA is no faster than that of the classical ADMM due to the communication-censoring strategy, which, nevertheless, effectively reduces the communication cost per iteration. Therefore, given a target solution accuracy, COCA is able to significantly reduce the overall communication cost compared to the classical ADMM, and hence fits for applications where network communication is a bottleneck, as demonstrated in the numerical experiments in Section IV.

IV. Numerical Experiments

In this section, we conduct numerical experiments to validate the effectiveness of COCA in reducing the overall communication cost. We compare it with four existing algorithms: (i) the classical ADMM without communication censoring [2], [16]; (ii) the random-walk ADMM (RndWalk ADMM) [34]; (iii) the distributed ADMM (D-ADMM) with node coloring [3]; (iv) the event-triggered zero-gradient-sum (ET-ZGS) algorithm that is a communication-censored version of the dual decomposition method [42]. We consider three decentralized consensus optimization problems: (i) least squares; (ii) logistic regression; (iii) geometric median. The local cost functions are smooth in least squares and logistic regression, but is non-smooth in geometric median. The subproblems in the three algorithms have explicit solutions in least squares and geometric median, but require iterative solvers in logistic regression. The accuracy of the local primal variables is defined by $\|X^k - X^*\|_F^2 / \|X^0 - X^*\|_F^2$, where we stack all local primal variables x_i^k in a matrix $X^k \in \mathcal{R}^{n\times p}$ and n optimal solutions x^* in a matrix $X^* \in \mathcal{R}^{n\times p}$.

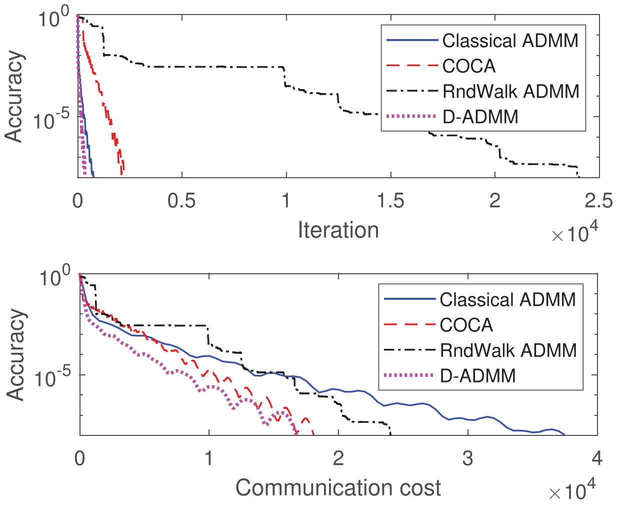

Fig. 1. Performance of COCA and the classical ADMM, random-walk ADMM and distributed ADMM over the line network for decentralized least squares.

A. Decentralized Least Squares

In the decentralized least squares problem, each node i has a local cost function $f_i(\tilde{x}) = (1/2)\|A_{(i)}\tilde{x} - y_{(i)}\|_2^2$, where $A_{(i)} \in \mathcal{R}^{p\times p}$ and $y_{(i)} \in \mathcal{R}^p$ are private. To minimize $f(\tilde{x}) := \sum_{i=1}^n f_i(\tilde{x})$ with COCA, the primal update of node i at time k is

$$x_i^k = (A_{(i)}^T A_{(i)} + 2cd_{ii}I_p)^{-1} \left(A_{(i)}^T y_{(i)} - \lambda_i^{k-1} + c\sum_{j\in\mathcal{N}_i}(\hat{x}_i^{k-1} + \hat{x}_j^{k-1}) \right),$$

where $I_p \in \mathcal{R}^{p\times p}$ is an identity matrix. Note that node i can compute $(A_{(i)}^T A_{(i)} + 2cd_{ii}I_p)^{-1}$ in advance to avoid computing the inverse at every time. In the experiments, entries $A_{(i)}$ and $y_{(i)}$ follow the i.i.d. uniform distribution within [0,10]. We set the size of the network as $n = 50$ and the dimension of the local variables as $p = 3$.

We compare the performance of COCA with the classical ADMM, random-walk ADMM and distributed ADMM over four network topologies: line, star, random and complete, as shown in Figs. 1–4. In the random network, 10% of all possible bi-directional edges are randomly chosen to be connected. The accuracies are compared with respect to the number of iterations and the communication cost, which is defined as the number of messages broadcast by all the nodes up to the current time. The step size c is tuned to the best for the classical ADMM, and COCA uses the same step size. The censoring function of COCA is $H_i(k, \xi_i^k) = \|\xi_i^k\| - \tau^k$, where $\tau^k = \alpha\rho^k$. The parameters α and ρ are also tuned to the best. Similarly, we use the best

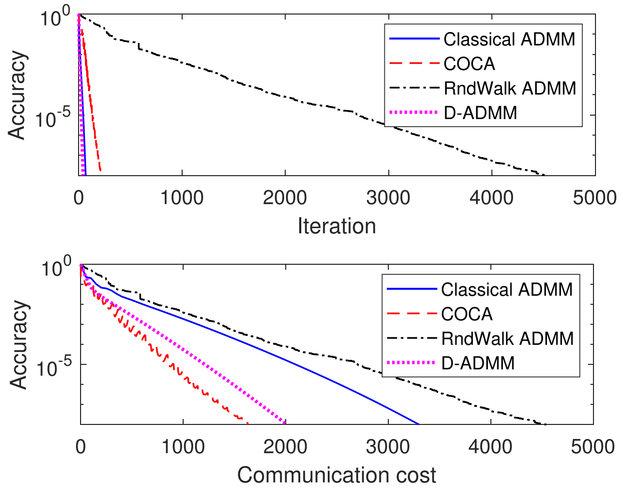

Fig. 2. Performance of COCA and the classical ADMM, random-walk ADMM and distributed ADMM over the star network for decentralized least squares.

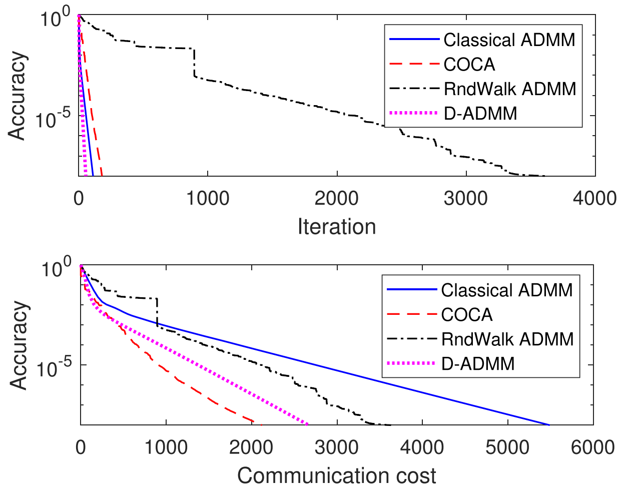

Fig. 3. Performance of COCA and the classical ADMM, random-walk ADMM and distributed ADMM over the random network for decentralized least squares.

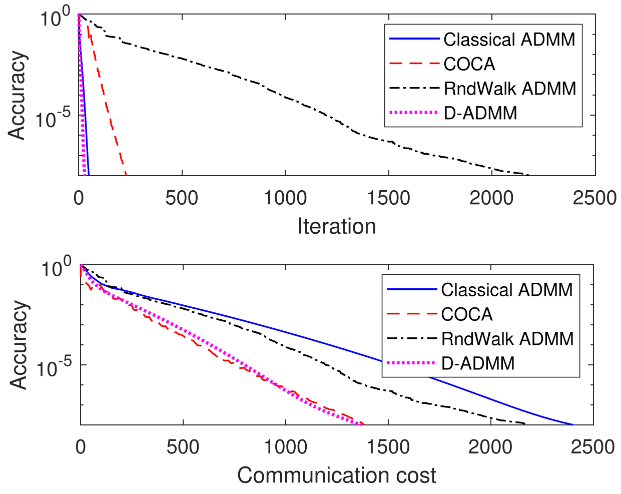

Fig. 4. Performance of COCA and the classical ADMM, random-walk ADMM and distributed ADMM over the complete network for decentralized least squares.

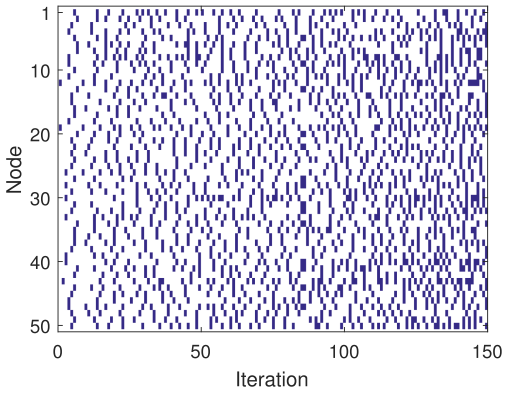

Fig. 5. Censoring pattern of COCA over the random network for decentralized least squares. The x-axis is number of iterations, and the y-axis is node index. A blue dot refers to that the node broadcasts at the time.

parameters for the random-walk ADMM and the distributed ADMM. In all the networks, COCA is slower than the classical ADMM in terms of the number of iterations, but faster in terms of the communication cost. Given a target accuracy of 10^{-8}, the savings in the communication costs are significant: $\sim 1/2$ savings in the line network, $\sim 1/2$ savings in the star network, $\sim 2/3$ savings in the random network, and $\sim 1/3$ savings in the complete network. In the complete network, information fusion is efficient such that "less informative" messages are less frequent and the advantage of COCA over the classical ADMM is less significant, but we can still observe improvement on the communication efficiency. The random-work ADMM randomly activates a succession of nodes, and every iteration amounts to activation of one node. Regarding the communication cost, the random-walk ADMM performs between COCA and classical ADMM in the line, random and complete networks. It performs worse than the other two in the star network. Our conjecture is that, in the star network, the center and the edge nodes are activated in an alternating manner, which makes the center node updates too frequently. Compared with D-ADMM, COCA has better communication efficiency in the random and star networks, but incurs comparable yet slightly higher communication cost in the line and complete networks. Note that D-ADMM is faster than the classical ADMM since it updates the primal variables in an ordered Gauss-Seidel fashion. Whenever a node receives a new message from its neighbor, it utilizes this latest information to update its primal variable. In comparison, the classical ADMM and COCA use Jacobi updates, with the primal variables being calculated using the messages received from the last iteration. In addition, the distributed ADMM requires to color the nodes into several groups, while the classical ADMM and COCA do not need this preprocessing step.

To see how the censoring strategy influences the communications of the nodes, we take the random network as an example to show the censoring pattern, as depicted in Fig. 5. The x-axis is the number of iterations, and the y-axis is the node index. A blue dot refers to that the node broadcasts at the time. Observe that the nodes have similar communication costs eventually. Meanwhile, the frequency of communication censoring does not change too much throughout the optimization process.

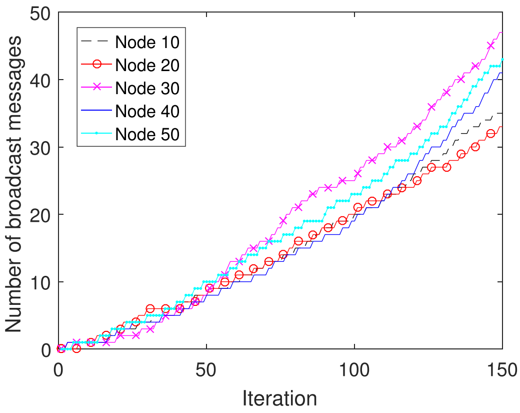

Fig. 6. Numbers of cumulative broadcast messages of some nodes (indexed by 10, 20, 30, 40 and 50) versus number of iterations in the random network for decentralized least squares.

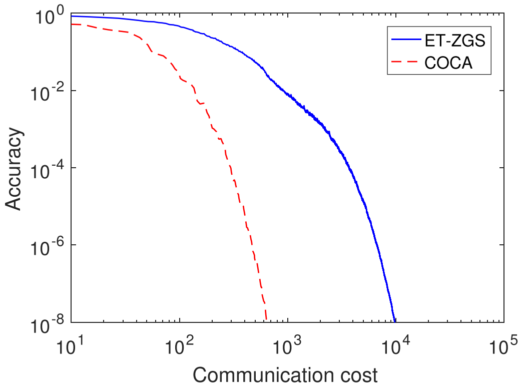

Fig. 7. Performance of COCA and ET-ZGS over the random network for decentralized least squares.

We further illustrate this phenomenon in Fig. 6, which shows how the cumulative numbers of broadcast messages of nodes $10, 20, 30, 40, 50$ evolve with the number of iterations. On average, each of these nodes broadcasts $0.22 \sim 0.32$ message at every time.

We proceed to compare COCA with ET-ZGS over the random network. The parameters are tuned to the best: for COCA, $c = 0.1$, $\alpha = 1$ and $\rho = 0.85$; for ET-ZGS, $c = 0.002$, $\alpha = 1$ and $\rho = 0.995$. As shown in Fig. 7, COCA is much faster than ET-ZGS. To reach a target accuracy of 10^{-8}, COCA needs ~ 700 broadcast messages, while ET-ZGS needs ~ 11000. This is reasonable since COCA and ET-ZGS are communication-censored versions of ADMM and dual decomposition, respectively, and empirically ADMM is faster than dual decomposition. Meanwhile, ADMM is observed to be more computationally stable than dual decomposition. This advantage is also inherited by COCA such that COCA allows for a larger step size, while ET-ZGS must use a smaller step size to guarantee convergence.

B. Decentralized Logistic Regression

In the decentralized logistic regression problem, each node i has a local cost function

$$f_i(\tilde{x}) = \frac{1}{l_i}\sum_{l=1}^{l_i} \ln\left(1 + \exp(-y_{(i)l} q_{(i)l}^T \tilde{x})\right),$$

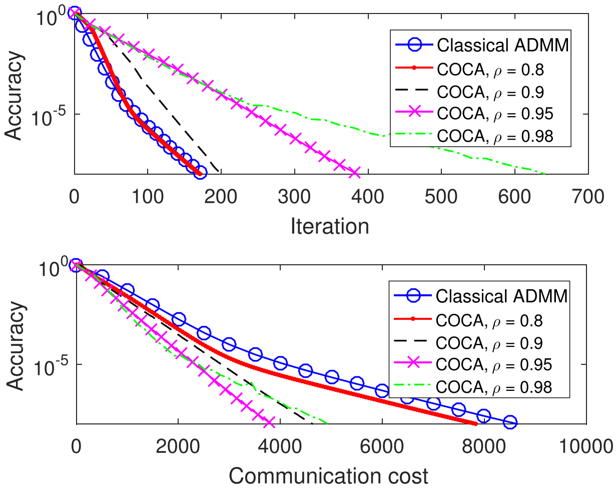

Fig. 8. Performance of COCA and the classical ADMM over the random network for decentralized logistic regression.

where $q_{(i)l}$ is the l-th column of a matrix $Q_{(i)} \in \mathcal{R}^{p \times l_i}$, $y_{(i)l} \in \{-1, +1\}$ is the l-the element of a binary vector $y_{(i)}$, and l_i is the number of samples held by node i. The primal updates of COCA in solving the decentralized logistic regression problem have no explicit solutions. Therefore, at every time we let each node to solve its subproblem by running a gradient descent subroutine, which terminates when the ℓ_2 norm of the gradient is less than 10^{-10}.

In the experiments, the network is random, with $n = 50$ nodes. The dimension of local variables is $p = 3$. The numbers of samples are $l_i = 10$ for all node i. Entries of $Q_{(i)}$ follow the i.i.d. uniform distribution within [0,10]. Entries of $y_{(i)}$ are i.i.d. and uniformly randomly chosen from $\{-1, +1\}$. For the censoring function $H_i(k, \xi_i^k) = \|\xi_i^k\| - \tau^k$ with $\tau^k = \alpha\rho^k$, we set $\alpha = 10$ and vary $\rho = 0.8, 0.9, 0.95$ to evaluate the impact of ρ on the performance of COCA. The step size is tuned to $c = 0.01$ that is the best for the classical ADMM, and COCA uses the same step size.

As depicted in Fig. 8, COCA with $\rho = 0.8$ performs similarly with the classical ADMM in terms of the number of iterations but slightly reduces the communication cost for a given target accuracy. This makes sense because smaller ρ means less communication censoring. When ρ increases, the number of iterations increases but the communication cost decreases. COCA with $\rho = 0.95$ saves $\sim 1/2$ of the communication costs comparing to the classical ADMM. When $\rho = 0.98$, which is very close to 1, communication censoring is too often such that many more iterations are necessary for reaching the target accuracy, and thus the overall communication cost increases. We can see that COCA achieves a favorable communication-computation tradeoff through tuning the censoring function.

C. Decentralized Geometric Median

In the last set of numerical experiments, we consider the decentralized geometric median problem, where each node i has a non-smooth local cost function $\|x_i - y_{(i)}\|_2$ and $y_{(i)} \in \mathcal{R}^p$ is private. We shall show that COCA works well for this non-smooth case, as shown in the theoretical analysis of Theorem 1.

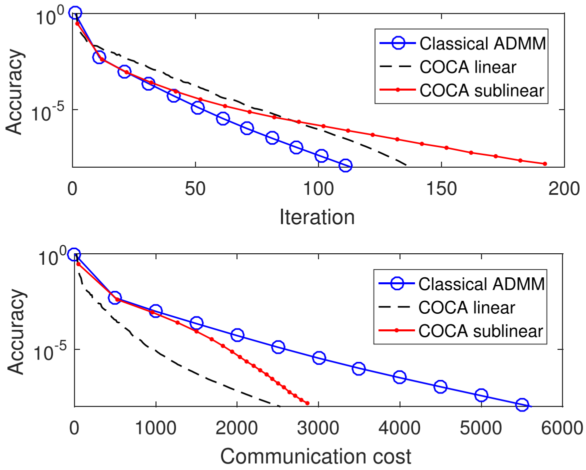

Fig. 9. Performance of COCA and the classical ADMM over the random network for decentralized geometric median.

In COCA, the primal update has an explicit form

$$x_i^k = y_{(i)} - \frac{\epsilon_i^k}{\|\epsilon_i^k\|}\left(\|\epsilon_i^k\| - 1\right)_+,$$

where $\epsilon_i^k := 2cd_{ii}y_{(i)} + \lambda_i^{k-1} - c\sum_{j\in\mathcal{N}_i}(\hat{x}_i^{k-1} + \hat{x}_j^{k-1})$ and $(\cdot)_+$ denotes element-wise non-negative projection.

In the experiments, the network is random, with $n = 50$ nodes. The dimension of local variables is $p = 3$. Entries of $y_{(i)}$ follow the i.i.d. uniform distribution within [0,10]. For the censoring function $H_i(k, \xi_i^k) = \|\xi_i^k\| - \tau^k$, we consider two cases: (i) linear where $\tau^k = \alpha\rho^k$; (ii) sublinear where $\tau^k = \gamma/k^3$. In COCA, the parameters are chosen as $\alpha = 1$, $\rho = 0.85$ and $\gamma = 0.05$. The step size is tuned to $c = 0.3$ that is the best for the classical ADMM, and COCA uses the same step size.

Fig. 9 compares the accuracy of the classical ADMM, COCA with a linear censoring function and COCA with a sublinear censoring function. Similar to the smooth case, COCA is slower than the classical ADMM in terms of the number of iterations, but faster in terms of the communication cost. Indeed, the two COCA approaches save $\sim 1/2$ of the communication cost compared to the classical ADMM, given a target accuracy of 10^{-8}. Comparing the two COCA approaches, the one with the linear censoring function performs better than the one with the sublinear censoring function; the former is shown to be more efficient in both computation and communication in this case. An interesting observation is that, at the beginning stage, COCA with the sublinear censoring function performs similarly to the classical ADMM. This is because the sublinear τ^k is relatively small compared to the difference between the local variable and the previously transmitted one in the beginning, such that communication censoring rarely happens. When the optimization process evolves, τ^k is relatively large comparing to the difference, such that the communication-censoring strategy takes effect and less communications are allowed.

V. Conclusion

In this paper we propose COCA, a communication-censored version of the celebrated ADMM, to solve the decentralized consensus optimization problem. COCA fits for applications where computation is relatively cheap, while communication is a bottleneck. With a slight increase of the computation cost, COCA is able to significantly reduce the overall communication cost compared to the classical ADMM. The key to reaching this favorable communication-computation tradeoff is a communication-censoring strategy, which prevents a node from transmitting its local variable to neighbors, if this variable is "less informative". We theoretically establish convergence and linear rate of convergence for COCA, and validate the communication efficiency of COCA with numerical experiments. One of our future research directions is to apply COCA to practical systems, such as target tracking in a drone network.

Appendix A
Proof of Theorem 1

Proof: In the proof, we assume f to be smooth. When f is non-smooth, all the subsequent derivations still hold true by replacing ∇f with one of its subgradients.

The proof is based on the matrix-form reformulation of (1) given by (8) and the matrix form of COCA given by (9) and (10). From [16], [23], the KKT conditions of (8) are

$$\nabla f(X^*) + M_-\beta^* = 0, \tag{17}$$

$$M_-^T X^* = 0, \tag{18}$$

$$\frac{1}{2}M_+^T X^* = Z^*. \tag{19}$$

Here $X^* \in \mathcal{R}^{n\times p}$ and $Z^* \in \mathcal{R}^{2r\times p}$ are optimal primal variables, and every row of X^* is identical to $(\tilde{x}^*)^T$; $\beta^* \in \mathcal{R}^{2r\times p}$ is an optimal dual variable.

Observe that (9) and (10) only involve variables X, $\hat{X}$ and Λ. Below we define $E = \hat{X} - X$. To facilitate the analysis, from (9) and (10), we construct two new variables $Z \in \mathcal{R}^{2r\times p}$ and $\beta \in \mathcal{R}^{2r\times p}$ which correspond to Z^* and β^*, respectively. For this purpose, we use $D + W = (1/2)M_+M_+^T$, $D - W = (1/2)M_-M_-^T$, $2D = (1/2)M_+M_+^T + (1/2)M_-M_-^T$ and $\hat{X} = X + E$ to rewrite (9) and (10) as

$$\nabla f(X^k) + \frac{c}{2}(M_+M_+^T + M_-M_-^T)X^k + \Lambda^{k-1} - \frac{c}{2}M_+M_+^T X^{k-1} - \frac{c}{2}M_+M_+^T E^{k-1} = 0, \tag{20}$$

$$\Lambda^k - \Lambda^{k-1} - \frac{c}{2}M_-M_-^T X^k - \frac{c}{2}M_-M_-^T E^k = 0. \tag{21}$$

Since Λ^0 is initialized in the column space of M_-, there exists $\beta^0 \in \mathcal{R}^{2r\times p}$ such that $\Lambda^0 = M_-\beta^0$. According to (21), Λ^k always stays in the column space of M_-, and hence we can write $\Lambda^k = M_-\beta^k$ for any $k \geq 0$, where $\beta^k \in \mathcal{R}^{2r\times p}$ satisfies

$$\beta^k - \beta^{k-1} - \frac{c}{2}M_-^T X^k - \frac{c}{2}M_-^T E^k = 0. \tag{22}$$

Using (22) and $\Lambda^{k-1} = M_-\beta^{k-1}$ to eliminate Λ^{k-1} in (20), as well as defining $Z = (1/2)M_+^T X$, we have

$$\nabla f(X^k) + M_-\beta^k + cM_+(Z^k - Z^{k-1}) - \frac{c}{2}M_-M_-^T E^k - \frac{c}{2}M_+M_+^T E^{k-1} = 0. \tag{23}$$

With (22) and (23), we are ready to analyze the convergence properties of COCA.

Step 1: Subtracting (23) by the KKT condition (17) yields

$$\begin{aligned}\nabla f(X^k) - \nabla f(X^*) &= \frac{c}{2}M_-M_-^TE^k + \frac{c}{2}M_+M_+^TE^{k-1}\\ &\quad - M_-(\beta^k - \beta^*) - cM_+(Z^k - Z^{k-1}). \qquad (24)\end{aligned}$$

Multiplying both sides of (24) by $X^k - X^*$ and noticing $\langle \nabla f(X^k) - \nabla f(X^*), X^k - X^* \rangle \geq 0$ that comes from the convexity of f in Assumption 3, we have

$$\begin{aligned}&\frac{c}{2}\langle M_-^TE^k, M_-^T(X^k - X^*)\rangle + c\langle M_+^TE^{k-1}, Z^k - Z^*\rangle\\ &\geq \langle \beta^k - \beta^*, M_-^T(X^k - X^*)\rangle + 2c\langle Z^k - Z^{k-1}, Z^k - Z^*\rangle. \qquad (25)\end{aligned}$$

Therein, we use the equation $Z^k - Z^* = (1/2)M_+^T(X^k - X^*)$. According to the KKT condition (18), $M_-^TX^* = 0$ such that $M_-^T(X^k - X^*) = M_-^TX^k$, which can be written as $(2/c)(\beta^k - \beta^{k-1}) - M_-^TE^k$ from (22). Further rewrite (25) to

$$\begin{aligned}&\langle M_-^TE^k, \beta^k - \beta^{k-1}\rangle + \langle M_-^TE^k, \beta^k - \beta^*\rangle\\ &+ c\langle M_+^TE^{k-1}, Z^k - Z^*\rangle - \frac{c}{2}\|M_-^TE^k\|_F^2\\ &\geq \frac{2}{c}\langle \beta^k - \beta^{k-1}, \beta^k - \beta^*\rangle + 2c\langle Z^k - Z^{k-1}, Z^k - Z^*\rangle\\ &= \frac{1}{c}(\|\beta^k - \beta^{k-1}\|_F^2 + \|\beta^k - \beta^*\|_F^2 - \|\beta^{k-1} - \beta^*\|_F^2)\\ &+ c(\|Z^k - Z^{k-1}\|_F^2 + \|Z^k - Z^*\|_F^2 - \|Z^{k-1} - Z^*\|_F^2). \qquad (26)\end{aligned}$$

Step 2: When the error terms E^k and E^{k-1} are 0, the broadcast messages are the same as the local primal variables, and COCA degenerates to the classical ADMM. In this case, the left-hand side of (26) is 0, which implies convergence of the algorithm; see for reference [8]. It is the error caused by the communication-censoring strategy that makes the convergence analysis of COCA challenging. Below we proceed to handle the error by finding a proper upper bound for the left-hand side of (26). Observe that for two matrices A and B, $\langle A, B\rangle \leq (\eta/2)\|A\|_F\|B\|_F^2 + (1/2\eta)\|A\|_F$ for all $\eta > 0$, as well as $\|AB\|_F \leq \sigma_{\max}(A)\|B\|_F$ where $\sigma_{\max}(A)$ denotes the maximum singular value of the matrix A. By these two inequalities, the first three terms in the left-hand side of (26) are bounded by

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

where η_1, η_2 and η_3 are arbitrary positive constants. Therefore, (26) can be rewritten to

$$\begin{aligned}&\frac{\sigma_{\max}(M_-)}{2\eta_1}\|E^k\|_F + \frac{\sigma_{\max}(M_-)}{2\eta_2}\|E^k\|_F + \frac{c\sigma_{\max}(M_+)}{2\eta_3}\|E^{k-1}\|_F\\ &+ c\|Z^{k-1} - Z^*\|_F^2 + \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2\\ &- \left(\frac{1}{c} - \frac{\eta_2\sigma_{\max}(M_-)\|E^k\|_F}{2}\right)\|\beta^k - \beta^*\|_F^2\\ &- \left(c - \frac{c\eta_3\sigma_{\max}(M_+)\|E^{k-1}\|_F}{2}\right)\|Z^k - Z^*\|_F^2\\ &\geq \left(\frac{1}{c} - \frac{\eta_1\sigma_{\max}(M_-)\|E^k\|_F}{2}\right)\|\beta^k - \beta^{k-1}\|_F^2\\ &+ c\|Z^k - Z^{k-1}\|_F^2 + \frac{c}{2}\|M_-^TE^k\|_F^2. \qquad (27)\end{aligned}$$

Step 3: Now we characterize the upper bounds of $\|E^k\|_F$ and $\|E^{k-1}\|_F$. Recall that the i-th row of E^k is $(\hat{x}_i^k - x_i^k)^T$. According to the communication-censoring strategy, $\hat{x}_i^k = x_i^k$ if $\|\hat{x}_i^{k-1} - x_i^k\| \geq \tau^k$ and $\hat{x}_i^k = \hat{x}_i^{k-1}$ if $\|\hat{x}_i^{k-1} - x_i^k\| < \tau^k$. In both cases, $\|\hat{x}_i^k - x_i^k\| < \tau^k$. Therefore, $\|E^k\|_F < \sqrt{n}\tau^k$. Because $\{\tau^k\}$ is a non-increasing non-negative sequence, $\|E^k\|_F < \sqrt{n}\tau^{k-1}$. Meanwhile, we also have $\|E^{k-1}\|_F < \sqrt{n}\tau^{k-1}$. Thus, replacing $\|E^k\|_F$ and $\|E^{k-1}\|_F$ by their upper bounds and throwing away the non-negative term $(c/2)\|M_-^TE^k\|_F^2$, we have from (27) that

$$\begin{aligned}&\left(\frac{\sigma_{\max}(M_-)}{2\eta_1} + \frac{\sigma_{\max}(M_-)}{2\eta_2} + \frac{c\sigma_{\max}(M_+)}{2\eta_3}\right)\sqrt{n}\tau^{k-1}\\ &+ \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2 + c\|Z^{k-1} - Z^*\|_F^2\\ &- \left(\frac{1}{c} - \frac{\eta_2\sigma_{\max}(M_-)\sqrt{n}\tau^{k-1}}{2}\right)\|\beta^k - \beta^*\|_F^2\\ &- \left(c - \frac{c\eta_3\sigma_{\max}(M_+)\sqrt{n}\tau^{k-1}}{2}\right)\|Z^k - Z^*\|_F^2\\ &\geq \left(\frac{1}{c} - \frac{\eta_1\sigma_{\max}(M_-)\|E^k\|_F}{2}\right)\|\beta^k - \beta^{k-1}\|_F^2\\ &+ c\|Z^k - Z^{k-1}\|_F^2. \qquad (28)\end{aligned}$$

Set the constants η_1, η_2 and η_3 in (28) as

$$\eta_1 = \frac{1}{c\sqrt{n}\tau^0\sigma_{\max}(M_-)},$$

$$\eta_2 = \frac{1}{c\sqrt{n}\tau^0\sigma_{\max}(M_-)},$$

$$\eta_3 = \frac{1}{\sqrt{n}\tau^0\sigma_{\max}(M_+)}.$$

Therefore, we have

$$\frac{1}{c} - \frac{\eta_2\sigma_{\max}(M_-)\sqrt{n}\tau^{k-1}}{2} = \frac{1}{c}\left(1 - \frac{\tau^{k-1}}{2\tau^0}\right),$$

$$c - \frac{c\eta_3\sigma_{\max}(M_+)\sqrt{n}\tau^{k-1}}{2} = c\left(1 - \frac{\tau^{k-1}}{2\tau^0}\right),$$

$$\frac{1}{c} - \frac{\eta_1\sigma_{\max}(M_-)\|E^k\|_F}{2} \geq \frac{1}{2c},$$

where we use the fact that $\|E^k\|_F \leq \sqrt{n}\tau^k \leq \sqrt{n}\tau^0$ for all k. Thus, (28) becomes

$$\theta\tau^{k-1} + \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2 - \frac{1}{c}\left(1 - \frac{\tau^{k-1}}{2\tau^0}\right)\|\beta^k - \beta^*\|_F^2$$
$$+ c\|Z^{k-1} - Z^*\|_F^2 - c\left(1 - \frac{\tau^{k-1}}{2\tau^0}\right)\|Z^k - Z^*\|_F^2$$
$$\geq \frac{1}{2c}\|\beta^k - \beta^{k-1}\|_F^2 + \frac{c}{2}\|Z^k - Z^{k-1}\|_F^2, \tag{29}$$

where

$$\theta := \left(\frac{\sigma_{\max}(M_-)}{2\eta_1} + \frac{\sigma_{\max}(M_-)}{2\eta_2} + \frac{c\sigma_{\max}(M_+)}{2\eta_3}\right)\sqrt{n}$$
$$= cn\tau_0\left(\sigma_{\max}^2(M_-) + \frac{\sigma_{\max}^2(M_+)}{2}\right).$$

Define $\Delta^k = (1/c)\|\beta^k - \beta^*\|_F^2 + c\|Z^k - Z^*\|_F^2$ and rewrite (29) to

$$\theta\tau^{k-1} + \Delta^{k-1} - \left(1 - \frac{\tau^{k-1}}{2\tau^0}\right)\Delta^k$$
$$\geq \frac{1}{2c}\|\beta^k - \beta^{k-1}\|_F^2 + \frac{c}{2}\|Z^k - Z^{k-1}\|_F^2. \tag{30}$$

Step 4: An immediate observation from (30) is that its right-hand side is non-negative, leading to

$$\theta\tau^{k-1} + \Delta^{k-1} - \left(1 - \frac{\tau^{k-1}}{2\tau^0}\right)\Delta^k \geq 0. \tag{31}$$

We use this fact to show that Δ^k has a finite upper bound. Since $\{\tau^k\}$ is a non-increasing non-negative sequence, $1 - \tau^{k-1}/(2\tau^0) \in [1/2, 1]$. Expanding (31) from time k to time 0 yields

$$\Delta^k \leq \left(1 - \frac{\tau^{k-1}}{2\tau^0}\right)^{-1}(\Delta^{k-1} + \theta\tau^{k-1})$$
$$\leq \left(1 - \frac{\tau^{k-1}}{2\tau^0}\right)^{-1}\left(\left(1 - \frac{\tau^{k-2}}{2\tau^0}\right)^{-1}(\Delta^{k-2} + \theta\tau^{k-2}) + \theta\tau^{k-1}\right)$$
$$\cdots$$
$$\leq \Delta^0 \prod_{k'=0}^{k-1}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1} + \theta\sum_{k''=0}^{k-1}\prod_{k'=k''}^{k-1}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1}\tau^{k''}$$
$$\leq \Delta^0 \prod_{k'=0}^{k-1}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1} + \theta\prod_{k'=0}^{k-1}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1}\sum_{k''=0}^{k-1}\tau^{k''}$$
$$\leq \prod_{k'=0}^{\infty}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1}\left(\Delta^0 + \theta\sum_{k''=0}^{\infty}\tau^{k''}\right). \tag{32}$$

Since τ^k is a summable sequence, $\sum_{k''=0}^{\infty}\tau^{k''}$ is finite. It remains to show that $\prod_{k'=0}^{\infty}(1 - \tau^{k'}/(2\tau_0))^{-1}$ is also finite. Consider its logarithm

$$\log\prod_{k'=0}^{\infty}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1} = \sum_{k'=0}^{\infty}\log\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1}$$
$$\leq \sum_{k'=0}^{\infty}\log(1 + \frac{\tau^{k'}}{\tau_0}) \leq \sum_{k'=0}^{\infty}\frac{\tau^{k'}}{\tau_0} < \infty, \tag{33}$$

where the first inequality holds because $1 - \tau^{k'}/(2\tau^0) \in [1/2, 1]$. Thus, we conclude that Δ^k has a finite upper bound, denoted as $\bar{\Delta}$.

Step 5: Now we begin to prove the main result. Summing up (30) from $k = 0$ to $k = \infty$ yields

$$\sum_{k=0}^{\infty}\left(\frac{1}{2c}\|\beta^k - \beta^{k-1}\|_F^2 + \frac{c}{2}\|Z^k - Z^{k-1}\|_F^2\right)$$
$$\leq \Delta^0 + \sum_{k=0}^{\infty}\frac{\tau^{k-1}}{2\tau^0}\Delta^k + \theta\sum_{k=0}^{\infty}\tau^{k-1}$$
$$\leq \Delta^0 + \bar{\Delta}\sum_{k=0}^{\infty}\frac{\tau^{k-1}}{2\tau^0} + \theta\sum_{k=0}^{\infty}\tau^{k-1} < \infty. \tag{34}$$

Thus, we conclude that $\lim_{k\to\infty}(\beta^k - \beta^{k-1}) = 0$ and $\lim_{k\to\infty}(Z^k - Z^{k-1}) = 0$. Also observe that $\lim_{k\to\infty} E^k = 0$ due to $\|E^k\|_F < \sqrt{n}\tau^k$, which we have shown before. Following these limiting properties, when $k \to \infty$, (23) leads to

$$\lim_{k\to\infty} \nabla f(X^k) + M_-\beta^k = 0, \tag{35}$$

and (22) leads to

$$\lim_{k\to\infty} M_-^T X^k = 0. \tag{36}$$

Since $Z^k = (1/2)M_+^T X^k$ by definition, we have

$$\lim_{k\to\infty}\left(\frac{1}{2}M_+^T X^k - Z^k\right) = 0. \tag{37}$$

Comparing (35), (36) and (37) with (17), (18) and (19), we conclude that the triple (X^k, Z^k, β^k) satisfies the KKT conditions of (8) when k goes to infinity.

It remains to show that $\{(X^k, Z^k, \beta^k)\}$ converges when k goes to infinity. As shown in Step 4, Δ^k has a finite upper bound $\bar{\Delta}$, implying that $\{(Z^k, \beta^k)\}$ is bounded. Thus, there exists a subsequence $\{(Z^{k_t}, \beta^{k_t})\}$ which converges to a cluster point (Z^∞, β^∞) of $\{(Z^k, \beta^k)\}$ and (Z^∞, β^∞) is an optimal solution of (8). Since $(Z^{k_t}, \beta^{k_t}) \to (Z^\infty, \beta^\infty)$, $\sum_{k''=0}^{\infty}\tau^{k''} < \infty$ and $\prod_{k'=0}^{\infty}(1 - \tau^{k'}/(2\tau_0))^{-1} < \infty$, for any constant $\zeta > 0$ there exists an integer $\bar{t}$ such that

$$\frac{1}{c}\|\beta^{k_{\bar{t}}} - \beta^\infty\|_F^2 + c\|Z^{k_{\bar{t}}} - Z^\infty\|_F^2 < \frac{\zeta}{4}, \tag{38}$$

$$\frac{\zeta}{4\theta}\left(\prod_{k'=0}^{\infty}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1}\right)^{-1} > \sum_{k''=k_{\bar{t}}}^{\infty}\tau^{k''}, \tag{39}$$

$$\prod_{k'=k_{\bar{t}}}^{\infty}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1} < 2. \tag{40}$$

Since (Z^∞, β^∞) is an optimal solution of (8), we follow the derivation of (32) to obtain for any $k > k_{\bar{t}}$ that

$$\frac{1}{c}\|\beta^k - \beta^\infty\|_F^2 + c\|Z^k - Z^\infty\|_F^2$$
$$\leq \left(\frac{1}{c}\|\beta^{k_{\bar{t}}} - \beta^\infty\|_F^2 + c\|Z^{k_{\bar{t}}} - Z^\infty\|_F^2\right)\prod_{k'=k_{\bar{t}}}^{\infty}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1}$$
$$+ \theta\prod_{k'=k_{\bar{t}}}^{\infty}\left(1 - \frac{\tau^{k'}}{2\tau_0}\right)^{-1}\sum_{k''=k_{\bar{t}}}^{k-1}\tau^{k''} < \zeta. \tag{41}$$

Therefore, $(Z^k, \beta^k) \to (Z^\infty, \beta^\infty)$. Further, by (36) and (37), $M_-^T X^k \to 0$ and $(1/2)M_+^T X^k \to Z^\infty$. With the property $D = (1/4)M_+ M_+^T + (1/4)M_- M_-^T$, we have $DX^k \to (1/2) M_+ Z^\infty$, implying $X^k \to (1/2)D^{-1}M_+ Z^\infty \triangleq X^\infty$, which is an optimal primal solution of (8). Hence, we conclude that $\{(X^k, Z^k, \beta^k)\}$ converges to an optimal solution $(X^\infty, Z^\infty, \beta^\infty)$ of (8) when k goes to infinity and complete the proof. ■

Appendix B
Proof of Theorem 2

Proof: Not surprisingly, the proof follows the linear convergence rate analysis of the classical ADMM in [16]. However, the error caused by the communication-censoring strategy complicates the proof.

Step 1: We begin from (24) which has been given in the proof of Theorem 1. For clarity, rewrite it here as

$$\nabla f(X^k) - \nabla f(X^*) = \frac{c}{2} M_- M_-^T E^k + \frac{c}{2} M_+ M_+^T E^{k-1} - M_-(\beta^k - \beta^*) - cM_+(Z^k - Z^{k-1}). \quad (42)$$

Similar to how we obtain (25), multiplying both sides of (42) with $X^k - X^*$ yields

$$\begin{aligned}
&\langle \nabla f(X^k) - \nabla f(X^*), X^k - X^* \rangle \\
&= -\langle \beta^k - \beta^*, M_-^T(X^k - X^*)\rangle - 2c\langle Z^k - Z^{k-1}, Z^k - Z^*\rangle \\
&+ \frac{c}{2}\langle M_-^T E^k, M_-^T(X^k - X^*)\rangle + c\langle M_+^T E^{k-1}, Z^k - Z^*\rangle \\
&= -\frac{2}{c}\langle \beta^k - \beta^*, \beta^k - \beta^{k-1}\rangle + \langle \beta^k - \beta^*, M_-^T E^k\rangle \\
&- 2c\langle Z^k - Z^{k-1}, Z^k - Z^*\rangle \\
&+ \frac{c}{2}\langle M_-^T E^k, M_-^T(X^k - X^*)\rangle + c\langle M_+^T E^{k-1}, Z^k - Z^*\rangle.
\end{aligned} \quad (43)$$

In the last equality, we use the dual update $\beta^k - \beta^{k-1} - (c/2) M_-^T X^k - (c/2)M_-^T E^k = 0$ given by (22) and the KKT condition $M_-^T X^* = 0$ given by (18) to split the term $-\langle \beta^k - \beta^*, M_-^T(X^k - X^*)\rangle$.

By Assumption 4, f is strongly convex with constant m_f such that

$$\langle \nabla f(X^k) - \nabla f(X^*), X^k - X^* \rangle \geq m_f \|X^k - X^*\|_F^2, \quad (44)$$

which gives a lower bound for the left-hand side of (43). Below we derive an upper bound for the right-hand side of (43). It is known that

$$\begin{aligned}
&-\frac{2}{c}\langle \beta^k - \beta^*, \beta^k - \beta^{k-1}\rangle - 2c\langle Z^k - Z^{k-1}, Z^k - Z^*\rangle \\
&= \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2 - \frac{1}{c}\|\beta^k - \beta^{k-1}\|_F^2 - \frac{1}{c}\|\beta^k - \beta^*\|_F^2 \\
&+ c\|Z^{k-1} - Z^*\|_F^2 - c\|Z^k - Z^{k-1}\|_F^2 - c\|Z^k - Z^*\|_F^2.
\end{aligned} \quad (45)$$

For the rest three terms, observe that for two matrices A and B, $\langle A, B\rangle \leq (\eta/2)\|A\|_F^2 + (1/2\eta)\|B\|_F^2$ for all $\eta > 0$, as well as $\|AB\|_F \leq \sigma_{\max}(A)\|B\|_F$ where $\sigma_{\max}(A)$ denotes the maximum singular value of the matrix A. With these inequalities, we obtain

$$\begin{aligned}
&\frac{c}{2}\langle M_-^T E^k, M_-^T(X^k - X^*)\rangle + c\langle M_+^T E^{k-1}, Z^k - Z^*\rangle \\
&+ \langle \beta^k - \beta^*, M_-^T E^k\rangle \\
&\leq \frac{c\eta_1}{4}\|X^k - X^*\|_F^2 + \frac{c\sigma_{\max}^4(M_-)}{4\eta_1}\|E^k\|_F^2 \\
&+ \frac{c\eta_2}{2}\|Z^k - Z^*\|_F^2 + \frac{c\sigma_{\max}^2(M_+)}{2\eta_2}\|E^{k-1}\|_F^2 \\
&+ \frac{\eta_3}{2}\|\beta^k - \beta^*\|_F^2 + \frac{\sigma_{\max}^2(M_-)}{2\eta_3}\|E^k\|_F^2 \\
&\leq \frac{c\eta_1}{4}\|X^k - X^*\|_F^2 + \frac{c\eta_2}{2}\|Z^k - Z^*\|_F^2 \\
&+ \frac{\eta_3}{2}\|\beta^k - \beta^*\|_F^2 + s\|E^{k-1}\|_F^2,
\end{aligned} \quad (46)$$

where

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

and η_1, η_2 and η_3 are positive constants. Note that to derive the last inequality of (46), we use the fact that $\|E^k\|_F \leq \|E^{k-1}\|_F$ as shown in the proof of Theorem 1. Also note that in the analysis of the classical ADMM, the error term is absent and the left-hand side of (46) is 0, which significantly simplifies the proof.

Substituting (44)–(46) into (43) yields

$$\begin{aligned}
&c\|Z^k - Z^*\|_F^2 + \frac{1}{c}\|\beta^k - \beta^*\|_F^2 \\
&\leq c\|Z^{k-1} - Z^*\|_F^2 + \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2 \\
&- c\|Z^k - Z^{k-1}\|_F^2 - \frac{1}{c}\|\beta^k - \beta^{k-1}\|_F^2 \\
&+ \left(\frac{c\eta_1}{4} - m_f\right)\|X^k - X^*\|_F^2 + \frac{c\eta_2}{2}\|Z^k - Z^*\|_F^2 \\
&+ \frac{\eta_3}{2}\|\beta^k - \beta^*\|_F^2 + s\|E^{k-1}\|_F^2,
\end{aligned} \quad (47)$$

or equivalently

$$\begin{aligned}
&(1+\delta)c\|Z^k - Z^*\|_F^2 + (1+\delta)\frac{1}{c}\|\beta^k - \beta^*\|_F^2 \\
&\leq c\|Z^{k-1} - Z^*\|_F^2 + \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2 \\
&- c\|Z^k - Z^{k-1}\|_F^2 - \frac{1}{c}\|\beta^k - \beta^{k-1}\|_F^2 \\
&+ \left(\frac{c\eta_1}{4} - m_f\right)\|X^k - X^*\|_F^2 + \left(\frac{c\eta_2}{2} + c\delta\right)\|Z^k - Z^*\|_F^2 \\
&+ \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right)\|\beta^k - \beta^*\|_F^2 + s\|E^{k-1}\|_F^2,
\end{aligned} \quad (48)$$

for any constant δ.

Step 2: We again emphasize that in the analysis of the classical ADMM, (48) is also a key inequality, with η_1, η_2, η_3 and s being all zero. In that case, the linear convergence rate is an immediate result. But for COCA, we still need to manipulate (48). To do so, we establish upper bounds for $\|Z^k - Z^*\|_F^2$ and $\|\beta^k - \beta^*\|_F^2$ as follows.

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

$$\begin{aligned}\|Z^k - Z^*\|_F^2 &= \frac{1}{4}\|M_+^T(X^k - X^*)\|_F^2\\ &\le \frac{\sigma_{\max}^2(M_+)}{4}\|X^k - X^*\|_F^2. \end{aligned} \tag{49}$$

For $\|\beta^k - \beta^*\|_F^2$, recall (42) to write

$$\begin{aligned}&\|M_-(\beta^k - \beta^*)\|_F^2 = \|\nabla f(X^k) - \nabla f(X^*)\\ &+ cM_+(Z^k - Z^{k-1}) - \frac{c}{2}M_-M_-^T E^k - \frac{c}{2}M_+M_+^T E^{k-1}\|_F^2. \end{aligned} \tag{50}$$

Applying $\|A + B\|_F^2 \le \mu\|A\|_F^2 + \mu/(\mu - 1)\|B\|_F^2$ for all $\mu > 1$, we obtain an upper bound of (50) as

$$\begin{aligned}\|M_-(\beta^k - \beta^*)\|_F^2 \le\ & \gamma_1(\mu\|\nabla f(X^k) - \nabla f(X^*)\|_F^2\\ &+ \frac{\mu}{\mu - 1}\|cM_+(Z^k - Z^{k-1})\|_F^2)\\ &+ \frac{\gamma_1}{\gamma_1 - 1}\left(\gamma_2\left\|\frac{c}{2}M_-M_-^T E^k\right\|_F^2\right.\\ &\left.+ \frac{\gamma_2}{\gamma_2 - 1}\left\|\frac{c}{2}M_+M_+^T E^{k-1}\right\|_F^2\right), \end{aligned} \tag{51}$$

for all $\mu > 1$, $\gamma_1 > 1$ and $\gamma_2 > 1$. With particular note, in the analysis of the classical ADMM, the bound corresponding to (51) is $\|M_-(\beta^k - \beta^*)\|_F^2 \le \mu\|\nabla f(X^k) - \nabla f(X^*)\|_F^2 + \mu/(\mu - 1)\|cM_+(Z^k - Z^{k-1})\|_F^2$ because of the absence of the error terms. For simplicity, below we choose $\gamma_1 = \gamma_2 = 2$ and keep μ as it is. By Assumption 4, f has Lipschitz continuous gradients with constant M_f such that

$$\|\nabla f(X^k) - \nabla f(X^*)\|_F \le M_f\|X^k - X^*\|_F. \tag{52}$$

Using (52) and $\|E^k\|_F \le \|E^{k-1}\|_F < \sqrt{n}\tau^{k-1} = \sqrt{n}\alpha\rho^{k-1}$ to further bound (51) as

$$\begin{aligned}&\|M_-(\beta^k - \beta^*)\|_F^2\\ &\le 2\mu\|\nabla f(X^k) - \nabla f(X^*)\|_F^2 + \frac{2\mu}{\mu - 1}\|cM_+(Z^k - Z^{k-1})\|_F^2\\ &+ \|cM_-M_-^T E^k\|_F^2 + \|cM_+M_+^T E^{k-1}\|_F^2\\ &\le 2\mu M_f^2\|X^k - X^*\|_F^2 + \frac{2\mu c^2\sigma_{\max}^2(M_+)}{\mu - 1}\|Z^k - Z^{k-1}\|_F^2\\ &+ \frac{nc^2\alpha^2}{\rho^2}\left(\sigma_{\max}^4(M_+) + \sigma_{\max}^4(M_-)\right)\rho^{2k}. \end{aligned} \tag{53}$$

Since the dual variable Λ^0 is initialized in the column space of M_-, there exists $\beta^0 \in \mathcal{R}^{2r\times p}$ staying in the column space of M_-^T such that $\Lambda^0 = M_-\beta^0$. Then, by the dual update (22), every β^k is in the column space of M_-^T. Meanwhile, there must exist a finite optimal dual variable β^* in the column space of M_-^T as shown by [23]. Thus, the left-hand side of (53) is lower-bounded by

$$\|M_-(\beta^k - \beta^*)\|_F^2 \ge \tilde{\sigma}_{\min}^2(M_-)\|\beta^k - \beta^*\|_F^2, \tag{54}$$

where $\tilde{\sigma}_{\min}(M_-)$ is the minimum non-zero singular value of M_-. Combining (53) and (54) yields the upper bound of $\|\beta^k - \beta^*\|_F^2$ as

$$\begin{aligned}\|\beta^k - \beta^*\|_F^2 \le\ & \frac{2\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)}\|X^k - X^*\|_F^2\\ &+ \frac{2\mu c^2\sigma_{\max}^2(M_+)}{(\mu - 1)\tilde{\sigma}_{\min}^2(M_-)}\|Z^k - Z^{k-1}\|_F^2\\ &+ \frac{nc^2\alpha^2}{\rho^2\tilde{\sigma}_{\min}^2(M_-)}(\sigma_{\max}^4(M_+) + \sigma_{\max}^4(M_-))\rho^{2k}. \end{aligned} \tag{55}$$

Substituting (55) into (48), using $\|E^{k-1}\|_F \le \sqrt{n}\alpha\rho^{k-1}$ and reorganizing terms, we have

$$\begin{aligned}&\left(c - \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right)\frac{2\mu c^2\sigma_{\max}^2(M_+)}{(\mu - 1)\tilde{\sigma}_{\min}^2(M_-)}\right)\|Z^k - Z^{k-1}\|_F^2\\ &+ \frac{1}{c}\|\beta^k - \beta^{k-1}\|_F^2\\ &\le c\|Z^{k-1} - Z^*\|_F^2 + \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2\\ &- (1 + \delta)c\|Z^k - Z^*\|_F^2 - (1 + \delta)\frac{1}{c}\|\beta^k - \beta^*\|_F^2\\ &+ \left(\frac{c\eta_1}{4} + \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right)\frac{\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)} - m_f\right)\|X^k - X^*\|_F^2\\ &+ \left(\frac{c\eta_2}{2} + c\delta\right)\|Z^k - Z^*\|_F^2 + \left(\left(\frac{\eta_3}{2} + \frac{\delta}{c}\right)\frac{nc^2\alpha^2}{\rho^2\tilde{\sigma}_{\min}^2(M_-)}\right.\\ &\left.(\sigma_{\max}^4(M_+) + \sigma_{\max}^4(M_-)) + \frac{ns\alpha^2}{\rho^2}\right)\rho^{2k}. \end{aligned} \tag{56}$$

Further substituting (49) into (56) yields

$$\begin{aligned}&\left(m_f - \frac{c\eta_1}{4} - \left(\frac{c\eta_2}{2} + c\delta\right)\frac{\sigma_{\max}^2(M_+)}{4}\right.\\ &\left.- \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right)\frac{\mu M_f^2}{\tilde{\sigma}_{\min}^2(M_-)}\right)\|X^k - X^*\|_F^2\\ &+ \left(c - \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right)\frac{2\mu c^2\sigma_{\max}^2(M_+)}{(\mu - 1)\tilde{\sigma}_{\min}^2(M_-)}\right)\|Z^k - Z^{k-1}\|_F^2\\ &+ \frac{1}{c}\|\beta^k - \beta^{k-1}\|_F^2\\ &\le c\|Z^{k-1} - Z^*\|_F^2 + \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2\\ &- (1 + \delta)c\|Z^k - Z^*\|_F^2 - (1 + \delta)\frac{1}{c}\|\beta^k - \beta^*\|_F^2\\ &+ \left(\left(\frac{\eta_3}{2} + \frac{\delta}{c}\right)\frac{nc^2\alpha^2}{\rho^2\tilde{\sigma}_{\min}^2(M_-)}\right.\\ &\left.(\sigma_{\max}^4(M_+) + \sigma_{\max}^4(M_-)) + \frac{ns\alpha^2}{\rho^2}\right)\rho^{2k}, \end{aligned} \tag{57}$$

in which we recall that η_1, η_2 and η_3 are any positive constant, μ is any constant larger than 1, and δ is any constant. Fixing η_1, η_2, η_3 and μ, we choose a particular δ such that the coefficients in the left-hand side of (57) are non-negative. To this end, δ must satisfy

$$\begin{aligned}&\delta \le \min\left\{ \frac{(\mu-1)\tilde{\sigma}^2_{\min}(M_-)}{2\mu\sigma^2_{\max}(M_+)} - \frac{c\eta_3}{2}, \right.\\ &\left(\frac{c\sigma^2_{\max}(M_+)}{4} + \frac{\mu M_f^2}{c\tilde{\sigma}^2_{\min}(M_-)}\right)^{-1}\\ &\left.\left(m_f - \frac{c\eta_1}{4} - \frac{c\eta_2\sigma^2_{\max}(M_+)}{8} - \frac{\eta_3\mu M_f^2}{2\tilde{\sigma}^2_{\min}(M_-)}\right)\right\}. \qquad (58)\end{aligned}$$

In the later analysis, we also need $\delta > 0$. This is attainable as long as the COCA step size c satisfies

$$\begin{aligned}&c < \min\left\{ \frac{(\mu-1)\tilde{\sigma}^2_{\min}(M_-)}{\mu\eta_3\sigma^2_{\max}(M_+)}, \right.\\ &\left.\left(\frac{\eta_1}{4} + \frac{\eta_2\sigma^2_{\max}(M_+)}{8}\right)^{-1}\left(m_f - \frac{\eta_3\mu M_f^2}{2\tilde{\sigma}^2_{\min}(M_-)}\right)\right\}. \qquad (59)\end{aligned}$$

Thus, throwing away the left-hand side terms of (57), we have

$$\begin{aligned}&c\|Z^{k-1} - Z^*\|_F^2 + \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2 - (1+\delta)c\|Z^k - Z^*\|_F^2\\ &- (1+\delta)\frac{1}{c}\|\beta^k - \beta^*\|_F^2 + \phi\rho^{2k} \ge 0, \qquad (60)\end{aligned}$$

where we introduce a positive constant

$$\begin{aligned}\phi :=& \frac{ns\alpha^2}{\rho^2}\\ &+ \left(\frac{\eta_3}{2} + \frac{\delta}{c}\right)\frac{nc^2\alpha^2}{\rho^2\tilde{\sigma}^2_{\min}(M_-)}\left(\sigma^4_{\max}(M_+) + \sigma^4_{\max}(M_-)\right).\end{aligned}$$

Step 3: Expanding (60) from time 0 to time k yields

$$\begin{aligned}&c\|Z^k - Z^*\|_F^2 + \frac{1}{c}\|\beta^k - \beta^*\|_F^2\\ &\le (1+\delta)^{-k}\left(c\|Z^0 - Z^*\|_F^2 + \frac{1}{c}\|\beta^0 - \beta^*\|_F^2\right)\\ &+ \phi\sum_{k'=0}^{k-1}\rho^{2k'}(1+\delta)^{-(k-k')}. \qquad (61)\end{aligned}$$

Denote $\varepsilon_1 = \min\{(1+\delta)^{-1}, \rho^2\}$ and $\varepsilon_2 = \max\{(1+\delta)^{-1}, \rho^2\}$. Because $\delta > 0$ and $\rho \in (0,1)$, we have $\varepsilon_2 \in (0,1)$ and $(1+\varepsilon_2)/2 \in (\varepsilon_2, 1)$. Thus, we rewrite (61) as

$$\begin{aligned}&c\|Z^k - Z^*\|_F^2 + \frac{1}{c}\|\beta^k - \beta^*\|_F^2\\ &\le \varepsilon_2^k\left(c\|Z^0 - Z^*\|_F^2 + \frac{1}{c}\|\beta^0 - \beta^*\|_F^2\right) + \phi\sum_{k'=0}^{k-1}\varepsilon_1^{k'}\varepsilon_2^{k-k'}\\ &\le \left(\frac{1+\varepsilon_2}{2}\right)^k\left(c\|Z^0 - Z^*\|_F^2 + \frac{1}{c}\|\beta^0 - \beta^*\|_F^2\right)\\ &+ \phi\sum_{k'=0}^{k-1}\varepsilon_1^{k'}\left(\frac{1+\varepsilon_2}{2}\right)^{k-k'} \qquad (62)\\ &\le \left(\frac{1+\varepsilon_2}{2}\right)^k\left(c\|Z^0 - Z^*\|_F^2 + \frac{1}{c}\|\beta^0 - \beta^*\|_F^2\right)\\ &+ \phi\left(\frac{1+\varepsilon_2}{2}\right)^k\sum_{k'=0}^{k-1}\left(\frac{2\varepsilon_1}{1+\varepsilon_2}\right)^{k'}\\ &\le \left(\frac{1+\varepsilon_2}{2}\right)^k\\ &\left(c\|Z^0 - Z^*\|_F^2 + \frac{1}{c}\|\beta^0 - \beta^*\|_F^2 + \phi\left(1 - \frac{2\varepsilon_1}{1+\varepsilon_2}\right)^{-1}\right).\end{aligned}$$

Hence (62) implies that $\{(Z^k, \beta^k)\}$ converges to the optimal solution (Z^*, β^*) of (8) when k goes to infinity at a Q-linear rate $(1+\varepsilon_2)/2$. To show the convergence of $\{X^k\}$ to X^*, revisiting (47) and throwing away several terms, we have

$$\begin{aligned}&\left(m_f - \frac{c\eta_1}{4}\right)\|X^k - X^*\|_F^2\\ &\le c\|Z^{k-1} - Z^*\|_F^2 + \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2\\ &+ \frac{c\eta_2}{2}\|Z^k - Z^*\|_F^2 + \frac{\eta_3}{2}\|\beta^k - \beta^*\|_F^2 + s\|E^k\|_F^2\\ &\le c\|Z^{k-1} - Z^*\|_F^2 + \frac{1}{c}\|\beta^{k-1} - \beta^*\|_F^2\\ &+ \frac{c\eta_2}{2}\|Z^k - Z^*\|_F^2 + \frac{\eta_3}{2}\|\beta^k - \beta^*\|_F^2 + \frac{ns\alpha^2}{\rho^2}\rho^{2k}. \qquad (63)\end{aligned}$$

Hence, we conclude that when

$$c < \frac{4m_f}{\eta_1}, \qquad (64)$$

$\{X^k\}$ converges to the optimal solution X^* of (8) when k goes to infinity at a R-linear rate $\max\{(1+\varepsilon_2)/2, \rho^2\}$ and complete the proof. ■

References

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

[2] 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.

[3] J. Mota, J. Xavier, P. Aguiar, and M. Puschel, “D-ADMM: A communication-efficient distributed algorithm for separable optimization,” *IEEE Trans. Signal Process.*, vol. 61, no. 10, pp. 2718–2723, May 2013.

[4] F. Zeng, C. Li, and Z. Tian, “Distributed compressive spectrum sensing in cooperative multi-hop wideband cognitive networks,” *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 1, pp. 37–48, Feb. 2011.

[5] G. Giannakis, Q. Ling, G. Mateos, I. Schizas, and H. Zhu, “Decentralized learning for wireless communications and networking,” *Splitting Methods Commun. Imag., Sci. Eng.*, pp. 461–497, 2016.

[6] F. Bullo, J. Cortes, and S. Martinez, *Distributed Control of Robotic Networks*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

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

[8] G. Giannakis, V. Kekatos, N. Gatsis, S. Kim, H. Zhu, and B. Wollenberg, "Monitoring and optimization for power grids: A signal processing perspective," *IEEE Signal Process. Mag.*, vol. 30, no. 5, pp. 107–128, Sep. 2013.

[9] H. Liu, W. Shi, and H. Zhu, "Distributed voltage control in distribution networks: Online and robust implementations," *IEEE Trans. Smart Grid*, vol. 9, no. 6, pp. 6106–6117, Nov. 2018.

[10] A. Mokhtari, "Efficient methods for large-scale empirical risk minimization," Ph.D. dissertation, Dept. Elect. Syst. Eng., University of Pennsylvania, Philadelphia, PA, USA, 2017.

[11] 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. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5330–5340.

[12] A. Nedic, A. Olshevsky, and M. Rabbat, "Network topology and communication-computation tradeoffs in decentralized optimization," in *Proc. IEEE*, vol. 106, no. 5 May 2018, pp. 953–976.

[13] A. Berahas, R. Bollapragada, N. Keskar, and E. Wei, "Balancing communication and computation in distributed optimization," *IEEE Trans. Autom. Control*, 2017, arXiv: 1709.02999.

[14] G. Lan, S. Lee, and Y. Zhou, "Communication-efficient algorithms for decentralized and stochastic optimization," *Math. Programming*, 2017, arXiv: 1701.03961.

[15] K. Tsianos, S. Lawlor, and M. Rabbat, "Communication/computation tradeoffs in consensus-based distributed optimization," in *Proc. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1943–1951.

[16] 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.

[17] A. Makhdoumi and A. Ozdaglar, "Convergence rate of distributed ADMM over networks," *IEEE Trans. Autom. Control*, vol. 62, no. 10, pp. 5082–5095, Oct. 2017.

[18] D. Yuan and D. Ho, "Randomized gradient-free method for multiagent optimization over time-varying networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1342–1347, Jun. 2015.

[19] D. Hajinezhad, M. Hong, and A. Garcia, "Zeroth order nonconvex multi-agent optimization over networks," *IEEE Trans. Autom. Control*, 2017, arXiv: 1710.09997.

[20] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multiagent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.

[21] A. Sayed, "Adaptation, learning, and optimization over networks," *Found. Trends Mach. Learn.*, vol. 7, pp. 311–801, 2014.

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

[23] Q. Ling, W. Shi, G. Wu, and A. Ribeiro, "DLM: Decentralized linearized alternating direction method of multipliers," *IEEE Trans. Signal Process.*, vol. 63, no. 15, pp. 4051–4064, Aug. 2015.

[24] T. Chang, M. Hong, and X. Wang, "Multi-agent distributed optimization via inexact consensus ADMM," *IEEE Trans. Signal Process.*, vol. 63, no. 2, pp. 482–497, Jan. 2015.

[25] 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.

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

[27] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, "A decentralized second-order method with exact linear convergence rate for consensus optimization," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 4, pp. 507–522, Dec. 2016.

[28] 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.

[29] P. D. Lorenzo and G. Scutari, "NEXT: In-network nonconvex optimization," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 2. pp. 120–136, Jun. 2016.

[30] T. Wu, K. Yuan, Q. Ling, W. Yin, and A. Sayed, "Decentralized consensus optimization with asynchrony and delays," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 4, no. 2, pp. 293–307, Jun. 2018.

[31] S. Zhu, M. Hong, and B. Chen, "Quantized consensus ADMM for multi-agent distributed optimization," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2016, pp. 4134–4138.

[32] J. Zhang, K. You, and T. Basar, "Distributed discrete-time optimization in multi-agent networks using only sign of relative state," *IEEE Trans. Automat. Control*, 2017, arXiv: 1709.08360.

[33] Q. Ling, Y. Liu, W. Shi, and Z. Tian, "Weighted ADMM for fast decentralized network optimization," *IEEE Trans. Signal Process.*, vol. 64, no. 22, pp. 5930–5942, Nov. 2016.

[34] W. Yin, X. Mao, K. Yuan, Y. Gu, and A. H. Sayed, "A communication-efficient random-walk algorithm for decentralized optimization," 2018, arXiv: 1804.06568.

[35] I. Notarnicola, Y. Sun, G. Scutari, and G. Notarstefano, "Distributed big-data optimization via block-iterative convexification and averaging," in *Proc. IEEE 56th Annu. Conf. Decis. Control*, 2017, pp. 2281–2288.

[36] Z. Wang, Z. Yu, Q. Ling, D. Berberidis, and G. Giannakis, "Decentralized RLS with data-adaptive censoring for regressions over large-scale networks," *IEEE Trans. Signal Process.*, vol. 66, no. 6, pp. 1634–1648, Mar. 2018.

[37] 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.

[38] 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, 2013.

[39] C. Nowzari and J. Cortes, "Distributed event-triggered coordination for average consensus on weight-balanced digraphs," *Automatica*, vol. 68, pp. 237–244, 2016.

[40] K. Tsianos, S. Lawlor, J. Yu, and M. Rabbat, "Networked optimization with adaptive communication," in *Proc. IEEE Global Conf. Signal Inf. Process.*, 2013, pp. 579–582.

[41] Q. Lu and H. Li, "Event-triggered discrete-time distributed consensus optimization over time-varying graphs," *Complexity*, vol. 2017, 2017, Art. no. 5385708.

[42] W. Chen and W. Ren, "Event-triggered zero-gradient-sum distributed consensus optimization over directed networks," *Automatica*, vol. 65, pp. 90–97, 2016.

[43] F. Chung, *Spectral Graph Theory*. Providence, RI, USA: American Mathematical Society, 1997.